Lecture Notes for STAT546: Computational Statistics

—Lecture 5: Monte Carlo

Faming Liang

Purdue University

August 28, 2024

Slice sampler:

Suppose that one is interested in sampling from a density $f(x)$, $x \in \mathcal{X}$. Recall that sampling $x \sim f(x)$ is equivalent to sampling uniformly from the area under the graph $f(x)$:

$$\mathcal{A} = \{(x, u) : 0 \leq u \leq f(x)\},$$

which is the basis of the acceptance-rejection algorithm described in §**??**. To achieve this goal, one can augment the target distribution by an auxiliary variable U, which, conditional on x, is uniformly distributed on the interval $[0, f(x)]$. Therefore, the joint density function of (X, U) is

$$f(x, u) = f(x)f(u|x) \propto 1_{(x,u)\in\mathcal{A}},$$

which can be sampled using the Gibbs sampler as follows:

Slice Sampler

- Draw $u_{t+1} \sim \text{Uniform}[0, f(x_t)]$.
- Draw x_{t+1} uniformly from the region $\{x : f(x) \geq u_{t+1}\}$.

This sampler is called the slice sampler by Higdon (1998), which potentially can be more efficient than the simple MH algorithm for the multimodal distributions, due to the free between-mode-transitions within a slice (as illustrated by Figure 1).

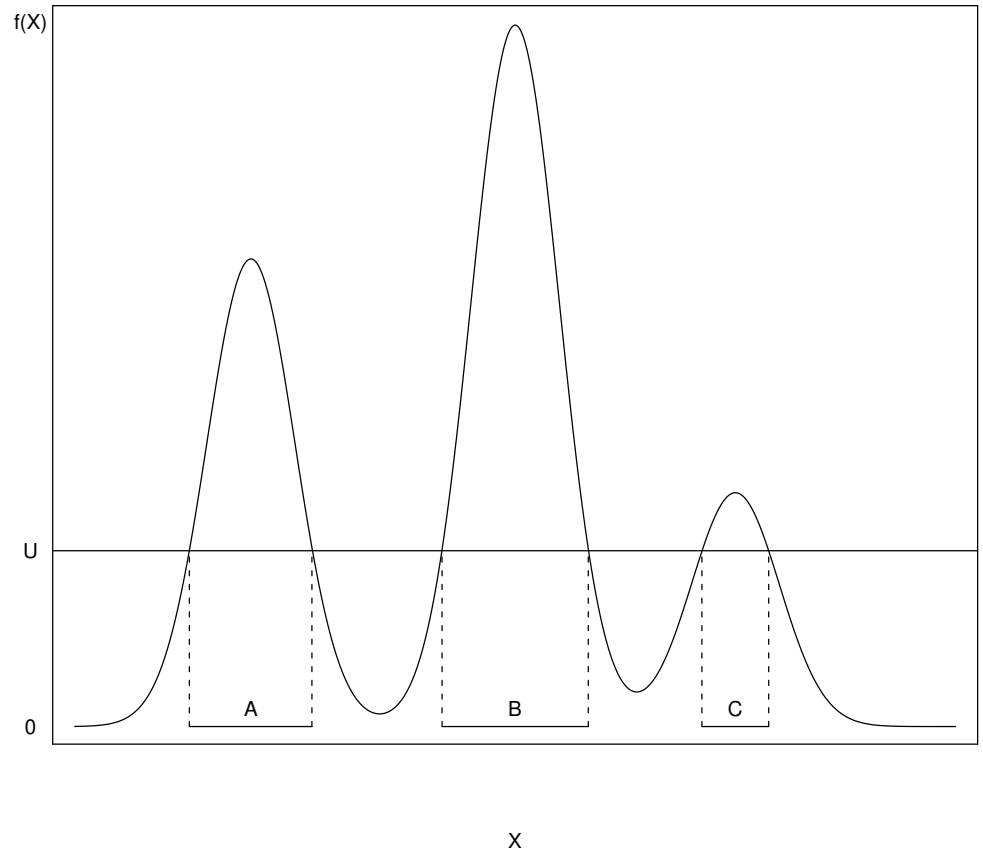

Figure 1: The slice sampler for a multimodal distribution: Draw $X|U$ uniformly from the sets labeled by A, B and C.

Edwards and Sokal (1988) noted that when $f(x)$ can be decomposed into a product of k distribution functions, i.e., $f(x) \propto f_1(x) \times f_2(x) \times \cdots \times f_k(x)$, the slice sampler can be easily implemented. To sample from such a distribution, Edwards and Sokal introduced k auxiliary variables, $U_1, \ldots, U_k$, and proposed the following algorithm:

- Draw $u_{t+1}^{(i)} \sim \text{Uniform}[0, f_i(x_t)]$, $i = 1, \ldots, k$.
- Draw x_{t+1} uniformly from the region $\mathcal{S} = \cap_{i=1}^{k} \{x : f_i(x) \geq u_{t+1}^{(i)}\}$.

This algorithm is a generalization of the Swendsen-Wang algorithm (Swendsen and Wang, 1987).

Consider a 2-D Ising model with the Boltzmann density

$$f(\boldsymbol{x}) \propto \exp\left\{ K \sum_{i \sim j} x_i x_j \right\}, \tag{1}$$

where the spins $x_i = \pm 1$, K is the inverse temperature, and $i \sim j$ represents the nearest neighbors on the lattice.

When the temperature is high, this model can be easily simulated using the Gibbs sampler (Geman and Geman, 1984): Iteratively reset the value of each spin according to the conditional distribution

$$P(x_i = 1 | x_j, j \in n(i)) = \frac{1}{1 + \exp\{-2K \sum_{j \in n(i)}\}}, \qquad (2)$$
$$P(x_i = -1 | x_j, j \in n(i)) = 1 - P(x_i = 1 | x_j, j \in n(i)),$$

where $n(i)$ denotes the set of neighbors of spin i.
However, the Gibbs sampler slows down rapidly when the temperature is approaching or below the critical temperature. This is the so-called "critical slowing down".

Swendsen-Wang algorithm

As in slice sampling, the density (1) is rewritten in the form

$$f(\boldsymbol{x}) \propto \prod_{i \sim j} \exp\{K(1 + x_i x_j)\} = \prod_{i \sim j} \exp\{\beta I(x_i = x_j)\}, \qquad (3)$$

where $\beta = 2K$ and $I(\cdot)$ is the indicator function, as $1 + x_i x_j$ is either 0 or 2. If we introduce auxiliary variables $\boldsymbol{u} = (u_{i \sim j})$, where each component $u_{i \sim j}$, conditional on x_i and x_j, is uniformly distributed on $[0, \exp\{\beta I(x_i = x_j)\}]$, then

$$f(\boldsymbol{x}, \boldsymbol{u}) \propto \prod_{i \sim j} I(0 \leq u_{i \sim j} \leq \exp\{\beta I(x_i = x_j)\}).$$

In Swendsen and Wang (1987), $u_{i \sim j}$ is called a "bond variable", which can be viewed as a variable physically sitting on the edge between spin i and spin j.

If $u_{i\sim j} > 1$, then $\exp\{\beta I(x_i = x_j) > 1$ and there must be $x_i = x_j$. Otherwise, there is no constraint on x_i and x_j. Let $b_{i\sim j}$ be an indicator variable for the constraint. If x_i and x_j are constrained to be equal, we set $b_{i\sim j} = 1$ and 0 otherwise. Note that for any two "like-spin" (i.e., the two spins have the same values) neighbors, they are bonded with probability $1 - \exp(-\beta)$. Based on the configurations of $\boldsymbol{u}$, we can "cluster" the spins according to whether they are connected via a "mutual bond" (i.e., $b_{i\sim j} = 1$). Then all the spins within the same cluster will have identical values, and flipping all the spins in a cluster simultaneously will not change the equilibrium of $f(\boldsymbol{x}, \boldsymbol{u})$.

The Swendsen-Wang Algorithm

- *Update bond values*: Check all "like-spin" neighbors, and set $b_{i\sim j} = 1$ with probability $1 - \exp(-\beta)$.
- *Update spin values*: Cluster spins by connecting neighboring sites with a mutual bond, and then flip each cluster with probability 0.5.

For the Ising model, the introduction of the auxiliary variable $\boldsymbol{u}$ has the dependence between neighboring spins partially decoupled, and the resulting sampler can thus converge substantially faster than the single site updating algorithm. As demonstrated by Swendsen and Wang (1987), this algorithm can eliminate much of the critical slowing down.

spatial autologistic model

Spatial models, e.g., the autologistic model, the Potts model, and the autonormal model (Besag, 1974), have been used in modeling of many scientific problems. A major problem with these models is that the normalizing constant is intractable. Suppose we have a dataset $\boldsymbol{X}$ generated from a statistical model with the likelihood function

$$f(\boldsymbol{x}|\theta) = \frac{1}{Z(\theta)} \exp\{-U(\boldsymbol{x}, \theta)\}, \quad \boldsymbol{x} \in \mathcal{X}, \quad \theta \in \Theta, \tag{4}$$

where θ is the parameter, and $Z(\theta)$ is the normalizing constant which depends on θ and is not available in closed form. Let $f(\theta)$ denote the prior density of θ. The posterior distribution of θ given $\boldsymbol{x}$ is then given by

$$f(\theta|\boldsymbol{x}) \propto \frac{1}{Z(\theta)} \exp\{-U(\boldsymbol{x}, \theta)\} f(\theta). \tag{5}$$

Let $\boldsymbol{y}$ denote the auxiliary variable, which shares the same state space with $\boldsymbol{x}$. Let

$$f(\theta, \boldsymbol{y}|\boldsymbol{x}) = f(\boldsymbol{x}|\theta)f(\theta)f(\boldsymbol{y}|\theta, \boldsymbol{x}), \tag{6}$$

denote the joint distribution of θ and $\boldsymbol{y}$ condional on $\boldsymbol{x}$, where $f(\boldsymbol{y}|\theta, \boldsymbol{x})$ is the distribution of the auxiliary variable $\boldsymbol{y}$.

To simulate from (6) using the MH algorithm, one can use the proposal distribution

$$q(\theta', \boldsymbol{y}'|\theta, \boldsymbol{y}) = q(\theta'|\theta, \boldsymbol{y})q(\boldsymbol{y}'|\theta'), \tag{7}$$

which corresponds to the usual change on the parameter vector $\theta \to \theta'$, followed by an exact sampling (Propp and Wilson, 1996) step of drawing $\boldsymbol{y}'$ from $q(\cdot|\theta')$.

If $q(\boldsymbol{y}'|\theta')$ is set as $f(\boldsymbol{y}'|\theta)$, then the MH ratio can be written as

$$r(\theta, \boldsymbol{y}, \theta', \boldsymbol{y}'|\boldsymbol{x}) = \frac{f(\boldsymbol{x}|\theta')f(\theta')f(\boldsymbol{y}'|\theta', \boldsymbol{x})q(\theta|\theta', \boldsymbol{y}')f(\boldsymbol{y}|\theta)}{f(\boldsymbol{x}|\theta)f(\theta)f(\boldsymbol{y}|\theta, \boldsymbol{x})q(\theta'|\theta, \boldsymbol{y})f(\boldsymbol{y}'|\theta')}, \quad (8)$$

where the unknown normalizing constant $Z(\theta)$ can be canceled. To ease computation, Møller *et al.* further suggested to set the proposal distributions $q(\theta'|\theta, \boldsymbol{y}) = q(\theta'|\theta)$ and $q(\theta|\theta', \boldsymbol{y}') = q(\theta|\theta')$, and to set the auxiliary distributions

$$f(\boldsymbol{y}|\theta, \boldsymbol{x}) = f(\boldsymbol{y}|\widehat{\theta}), \quad f(\boldsymbol{y}'|\theta', \boldsymbol{x}) = f(\boldsymbol{y}'|\widehat{\theta}), \quad (9)$$

where $\widehat{\theta}$ denotes an estimate of θ, for example, which can be obtained by maximizing a pseudo-likelihood function.

The Møller Algorithm

- Generate θ' from the proposal distribution $q(\theta'|\theta_t)$.
- Generate an exact sample $\boldsymbol{y}'$ from the distribution $f(\boldsymbol{y}|\theta')$.
- Accept $(\theta', \boldsymbol{y}')$ with probability $\min(1, r)$, where

$$r = \frac{f(\boldsymbol{x}|\theta')f(\theta')f(\boldsymbol{y}'|\widehat{\theta})q(\theta_t|\theta')f(\boldsymbol{y}|\theta_t)}{f(\boldsymbol{x}|\theta_t)f(\theta_t)f(\boldsymbol{y}|\widehat{\theta})q(\theta'|\theta_t)f(\boldsymbol{y}'|\theta')}.$$

If it is accepted, set $(\theta_{t+1}, \boldsymbol{y}_{t+1}) = (\theta', \boldsymbol{y}')$. Otherwise, set $(\theta_{t+1}, \boldsymbol{y}_{t+1}) = (\theta_t, \boldsymbol{y}_t)$.

The exchange algorithm

- Propose $\theta' \sim q(\theta'|\theta, \boldsymbol{x})$.
- Generate an auxiliary variable $\boldsymbol{y} \sim f(\boldsymbol{y}|\theta')$ using an exact sampler.
- Accept θ' with probability $\min\{1, r(\theta, \theta', \boldsymbol{y}|\boldsymbol{x})\}$, where

$$r(\theta, \theta', \boldsymbol{y}|\boldsymbol{x}) = \frac{\pi(\theta')f(\boldsymbol{x}|\theta')f(\boldsymbol{y}|\theta)q(\theta|\theta')}{\pi(\theta)f(\boldsymbol{x}|\theta)f(\boldsymbol{y}|\theta')q(\theta'|\theta)}. \tag{10}$$

The exchange algorithm can be viewed as an auxiliary variable MCMC algorithm with the proposal distribution being augmented, for which the proposal distribution can be written as

$$T(\theta \to (\theta', \boldsymbol{y})) = q(\theta'|\theta) f(\boldsymbol{y}|\theta'), \quad T(\theta' \to (\theta, \boldsymbol{y})) = q(\theta|\theta') f(\boldsymbol{y}|\theta).$$

This simply validates the algorithm, following the arguments for auxiliary variable Markov chains made in previous lectures.

Monte Carlo Metropolis-Hastings Algorithm (Liang and Jin, 2010)

Let θ_t denote the current draw of θ by the algorithm. Let $y_1^{(t)}, \ldots, y_m^{(t)}$ denote the auxiliary samples simulated from the distribution $f(y|\theta_t)$. The MCMH algorithm works by iterating between the following steps:

1. Draw ϑ from some proposal distribution $Q(\theta_t, \vartheta)$.
2. Estimate the normalizing constant ratio $R(\theta_t, \vartheta) = \kappa(\vartheta)/\kappa(\theta_t)$ by

$$\widehat{R}_m(\theta_t, \boldsymbol{y}_t, \vartheta) = \frac{1}{m} \sum_{i=1}^{m} \frac{g(y_i^{(t)}, \vartheta)}{g(y_i^{(t)}, \theta_t)},$$

where $g(y, \theta) = \exp\{-U(y, \theta)\}$, and $\boldsymbol{y}_t = (y_1^{(t)}, \ldots, y_m^{(t)})$ denotes the collection of the auxiliary samples.

3. Calculate the Monte Carlo MH ratio

$$\tilde{r}_m(\theta_t, \boldsymbol{y}_t, \vartheta) = \frac{1}{\widehat{R}_m(\theta_t, \boldsymbol{y}_t, \vartheta)} \frac{g(x, \vartheta) f(\vartheta)}{g(x, \theta_t) f(\theta_t)} \frac{Q(\vartheta, \theta_t)}{Q(\theta_t, \vartheta)},$$

where $f(\theta)$ denotes the prior distribution imposed on θ.

4. Set $\theta_{t+1} = \vartheta$ with probability $\tilde{\alpha}(\theta_t, \boldsymbol{y}_t, \vartheta) = \min\{1, \tilde{r}_m(\theta_t, \boldsymbol{y}_t, \vartheta)\}$, and set $\theta_{t+1} = \theta_t$ with the remaining probability.
5. If the proposal is rejected in step 4, set $\boldsymbol{y}_{t+1} = \boldsymbol{y}_t$. Otherwise, draw samples $\boldsymbol{y}_{t+1} = (y_1^{(t+1)}, \ldots, y_m^{(t+1)})$ from $f(y|\theta_{t+1})$ using either a MCMC algorithm or an automated rejection sampling algorithm.

Since the algorithm involves a Monte Carlo step to estimate the unknown normalizing constant ratio, it is termed as "Monte Carlo MH". Clearly, the samples $\{(\theta_t, \boldsymbol{y}_t)\}$ forms a Markov chain, for which the transition kernel can be written as

$$\begin{aligned}\tilde{P}_m(\theta, \boldsymbol{y}; d\vartheta, d\boldsymbol{z}) &= \tilde{\alpha}(\theta, \boldsymbol{y}, \vartheta) Q(\theta, d\vartheta) f_\vartheta^m(d\boldsymbol{z}) \\ &+ \delta_{\theta,\boldsymbol{y}}(d\vartheta, d\boldsymbol{z}) \big[1 - \int_{\Theta \times \mathbb{Y}} \tilde{\alpha}(\theta, \boldsymbol{y}, \vartheta) Q(\theta, d\vartheta) f_\theta^m(d\boldsymbol{y})\big],\end{aligned} \tag{11}$$

where $f_\theta^m(\boldsymbol{y}) = f(y_1, \ldots, y_m|\theta)$ denotes the joint density of $y_1, \ldots, y_m$.

However, since here we are mainly interested in the marginal law of θ, in what follows we will consider only the marginal transition kernel

$$\tilde{P}_m(\theta, d\vartheta) = \int_{\mathbb{Y}} \tilde{\alpha}(\theta, \boldsymbol{y}, \vartheta) Q(\theta, d\vartheta) f_\theta^m(d\boldsymbol{y}) + \delta_\theta(d\vartheta)\Big[1 - \int_{\Theta\times\mathbb{Y}} \tilde{\alpha}_m(\theta, \boldsymbol{y}, \vartheta) Q(\theta, d\vartheta) f_\theta^m(d\boldsymbol{y})\Big], \tag{12}$$

showing that $\tilde{P}_m(\theta, d\vartheta)$ will converge to the posterior distribution $f(\theta|x)$ when m is large and the number of iterations goes to infinity.

Monte Carlo MH Algorithm II

1. Draw ϑ from some proposal distribution $Q(\theta_t, \vartheta)$.
2. Draw auxiliary samples $\boldsymbol{y}_t = (y_1^{(t)}, \ldots, y_m^{(t)})$ from $f(y|\theta_t)$ using a MCMC algorithm.
3. Estimate the normalizing constant ratio $R(\theta_t, \vartheta) = \kappa(\vartheta)/\kappa(\theta_t)$ by

$$\widehat{R}_m(\theta_t, \boldsymbol{y}_t, \vartheta) = \frac{1}{m} \sum_{i=1}^{m} \frac{g(y_i^{(t)}, \vartheta)}{g(y_i^{(t)}, \theta_t)}.$$

4. Calculate the Monte Carlo MH ratio

$$\tilde{r}_m(\theta_t, \boldsymbol{y}_t, \vartheta) = \frac{1}{\widehat{R}_m(\theta_t, \boldsymbol{y}_t, \vartheta)} \frac{g(x, \vartheta) f(\vartheta)}{g(x, \theta_t) f(\theta_t)} \frac{Q(\vartheta, \theta_t)}{Q(\theta_t, \vartheta)}.$$

5. Set $\theta_{t+1} = \vartheta$ with probability $\tilde{\alpha}(\theta_t, \boldsymbol{y}_t, \vartheta) = \min\{1, \tilde{r}_m(\theta_t, \boldsymbol{y}_t, \vartheta)\}$ and set $\theta_{t+1} = \theta_t$ with the remaining probability.

Convergence of the MCMH sampler

(A_1) Assume that P defines an irreducible and aperiodic Markov chain such that $\pi P = \pi$, and for any $\theta_0 \in \Theta$, $\lim_{k\to\infty} \|P^k(\theta_0, \cdot) - \pi(\cdot)\| = 0$.

(A_2) For any $(\theta, \vartheta) \in \Theta \times \Theta$,

$$\gamma_m(\theta, \boldsymbol{u}, \vartheta) > 0, \quad f_\theta^m(\cdot) - a.s.$$

(A_3) For any $\theta \in \Theta$ and any $\epsilon > 0$,

$$\lim_{m\to\infty} Q\left(\theta, f_\theta^m(\lambda_m(\theta, \boldsymbol{u}, \vartheta) > \epsilon)\right) = 0,$$

where

$$Q\left(\theta, f_\theta^m(\lambda_m(\theta, \boldsymbol{u}, \vartheta) > \epsilon)\right) = \int_{\{(\vartheta, \boldsymbol{u}):\lambda_m(\theta, \boldsymbol{u}, \vartheta) > \epsilon\}} f_\theta^m(d\boldsymbol{u}) Q(\theta, d\vartheta).$$

Theorem 1.

Assume (A_1), (A_2) and (A_3) hold. For any $\varepsilon \in (0, 1]$, there exist $M \in \mathbb{N}$ and $K \in \mathbb{N}$ such that for any $m > M$ and $k > K$

$$\|\tilde{P}_m^k(\theta_0, \cdot) - \pi(\cdot)\| \leq \varepsilon,$$

where $\pi(\cdot)$ denotes the posterior density of θ.

Table 1: Computational results for the U.S. cancer mortality data (Liang and Jin, 2010).

Algorithm	Setting	$\widehat{\alpha}$	$\widehat{\beta}$	CPU
MCMH I	$m = 20$	-0.3018 (1.2×10^{-3})	0.1232 (6.0×10^{-4})	11
	$m = 50$	-0.3018 (1.1×10^{-3})	0.1230 (5.3×10^{-4})	24
	$m = 100$	-0.3028 (6.7×10^{-4})	0.1225 (3.8×10^{-4})	46
MCMH II	$m = 20$	-0.3028 (1.2×10^{-3})	0.1226 (5.9×10^{-4})	26
	$m = 50$	-0.3019 (1.0×10^{-3})	0.1228 (5.3×10^{-4})	63
	$m = 100$	-0.3016 (8.2×10^{-4})	0.1231 (3.8×10^{-4})	129
Exchange	—	-0.3015 (4.3×10^{-4})	0.1229 (2.3×10^{-4})	33