Unit 4: Reversible-jump and Pseudo-marginal MCMC

In this unit, we discuss two important and more sophisticated MCMC algorithms of Metropolis–Hastings type (in the sense that an acceptance-rejection step is used to ensure the correct stationary distribution).

4.1 A Motivating Example for Reversible-jump MCMC

We first review a change-point detection problem that was used in [2] to motivate the reversible-jump MCMC algorithm.

Model. Consider a non-homogeneous Poisson point process where the arrival rate is given by an unknown function $\lambda(t) > 0$. This means that the number of events happening during the time period $[t_1, t_2)$ follows a Poisson distribution with rate parameter $\int_{t_1}^{t_2} \lambda(s)\mathrm{d}s$. Let our observed data be the arrival times $T_1 < T_2 < \cdots < T_n$. Assume that $T_n \leq T_{\max}$, where $T_{\max}$ is the duration of the observation period. One major application of this model is the queueing theory, where T_i can be thought of as the arrival time of the i-th customer at a store. Recall that $T_1, T_2 - T_1, T_3 - T_2, \ldots$ are independent, and T_1 has density function $p(T_1 \,|\, \lambda) = \lambda(T_1) \exp\left(-\int_0^{T_1} \lambda(t)\mathrm{d}t\right)$. Hence, we can write down the log-likelihood as

$$\log p(D \,|\, \lambda) = -\int_0^{T_{\max}} \lambda(t)\mathrm{d}t + \sum_{i=1}^{n} \log \lambda(T_i), \tag{1}$$

where $D = (T_1, \ldots, T_n)$ denotes our data.

Our goal is to learn $\lambda(t)$, which we model as a positive step function with break points

$$b_0 = 0 < b_1 < b_2 < \cdots < b_K < T_{\max} = b_{K+1}.$$

On $t \in [b_{i-1}, b_i)$, we have $\lambda(t) = h_{i-1}$ for some constant $h_{i-1} > 0$. So to learn $\lambda(t)$ means to identify the parameter vector $\theta = (b_1, \ldots, b_K, h_0, h_1, \ldots, h_K)$. The length of this vector is $2K + 1$. Denote the likelihood of θ by $p(D \,|\, \theta)$, which can be calculated by (1) with $\lambda(t)$ replaced by the step function corresponding to θ. We want to compute the posterior distribution $p(\theta \,|\, D) \propto p(D \,|\, \theta)p(\theta)$, where $p(\theta)$ denotes the prior.

Proposal Scheme. Consider the following four types of proposal moves:

(a) Change b_k for some $1 \leq k \leq K$.

(b) Change h_k for some $0 \leq k \leq K$.

(c) Add a break point $b^* \in (b_k, b_{k+1})$ for some $0 \leq k \leq K$ and change $\lambda(t)$ on $t \in [b_k, b_{k+1})$.

(d) Remove a break point b_k for some $1 \leq k \leq K$ and change $\lambda(t)$ on $t \in [b_{k-1}, b_{k+1})$.

An MCMC sampler with these four proposals will be able to explore the space of all positive step functions on $[0, T_{\max})$. Moves of type (a) and (b) are straightforward to implement, but moves of type (c) and (d) can be complicated. In [2], the following scheme is used for implementing a type (c) move, where $\theta = (b_1, \ldots, b_K, h_0, \ldots, h_K)$ denotes the current state.

(i) Draw b^* from $[0, T_{\max})$. We have $b^* \in [b_k, b_{k+1})$ for some k.

(ii) Set the new parameter vector to $\theta' = (b'_1, \ldots, b'_{K+1}, h'_0, \ldots, h'_{K+1})$, where

$$b'_i = \begin{cases} b_i, & \text{if } i \le k, \\ b^*, & \text{if } i = k+1, \\ b_{i-1}, & \text{if } i \ge k+2. \end{cases} \qquad h'_i = \begin{cases} h_i, & \text{if } i \le k-1, \\ h_{i-1}, & \text{if } i \ge k+2. \end{cases}$$

Note that h'_k, h'_{k+1} have not been determined yet.

(iii) Draw $u \sim \text{Unif}(0, 1)$.

(iv) Set $h'_{k+1} = h'_k(1-u)/u$.

(v) Find h'_k by solving

$$(b^* - b_k)\log(h'_k) + (b_{k+1} - b^*)\log(h'_{k+1}) = (b_{k+1} - b_k)\log h_k.$$

The main motivation behind this scheme is that we want to propose h'_k, h'_{k+1} such that the new step function does not look too different from the current one on the interval $[b_k, b_{k+1})$; otherwise, the acceptance rate of the resulting sampler tends to be low. Other schemes are of course possible. For example, one can generate $u \sim \text{Unif}(0, 1)$, and set $h'_k = 2h_k u$, $h'_{k+1} = 2h_k(1-u)$ so that the average of h'_k, h'_{k+1} is always equal to h_k. An important observation is that steps (i) to (iv) define a one-to-one differentiable mapping from (b^*, u, h_k) to $(b'_{k+1}, h'_k, h'_{k+1})$ for any $b^* \in [b_k, b_{k+1})$.

Similarly, when we make a type (d) proposal, we reverse this process. If the current state is θ' as generated above and we propose to delete b'_{k+1}, the resulting new state will be θ.

Proposal Densities. Now let's calculate the proposal density of the procedure described by steps (i) to (iv). First, denote the probability of making a type (c) move when the current function has K break points by $q(K, K+1)$, which is allowed to depend on K. Let $q_c(\theta, (b^*, u))$ denote the density of proposing b^*, u in a type (c) proposal. The product $q(K, K+1)q_c(\theta, (b^*, u))$ is the density of proposing a type (c) move with new break point b^* and auxiliary variable u. We know that θ' is determined by θ, b^*, u, so it seems that we can obtain the proposal density $q(\theta, \theta')$ by taking into account the effect of the transformation $u \mapsto (h'_k, h'_{k+1})$. But here we encounter a dimensionality issue: the distribution of (h'_k, h'_{k+1}) is degenerate and does not have a density with respect to the Lebesgue measure on $\mathbb{R}^2$. To circumvent this difficulty, let's match the dimension by pretending that h_k is random, which enables us to apply the change-of-variable formula. This yields

$$q(\theta, \theta') = q(K, K+1)q_c(\theta, (b^*, u))\frac{h_k}{(h'_k + h'_{k+1})^2},$$

where the last term is the Jacobian determinant of the transformation $(h'_k, h'_{k+1}) \mapsto (u, h_k)$.

The proposal density of moving from θ' to θ is

$$q(\theta', \theta) = q(K+1, K) q_{\mathrm{d}}(\theta', k+1)$$

where $q(K+1, K)$ is the probability of making a type (d) proposal when the current function has $K+1$ break points, and $q_{\mathrm{d}}(\theta', k+1)$ denotes the probability of deleting the break point b'_{k+1}. Note that $q(\theta, \theta')$ and $q(\theta', \theta)$ are NOT comparable, since they are densities on spaces of different dimensions. The length of θ is $2K+1$, while that of θ' is $2K+3$.

Acceptance Probability. If we ignore this dimensionality issue and still proceed to calculate the acceptance probability in the usual way, we get

$$\begin{aligned}
\alpha(\theta, \theta') &= \min\left\{1, \frac{p(D \mid \theta')p(\theta')}{p(D \mid \theta)p(\theta)} \frac{q(\theta', \theta)}{q(\theta, \theta')}\right\} \\
&= \min\left\{1, \frac{p(D \mid \theta')p(\theta')}{p(D \mid \theta)p(\theta)} \frac{q(K+1, K) q_{\mathrm{d}}(\theta', k+1)}{q(K, K+1) q_{\mathrm{c}}(\theta, (b^*, u))} \frac{(h'_k + h'_{k+1})^2}{h_k}\right\}.
\end{aligned}$$

This turns out to be correct, and such a dimension-jumping Metropolis–Hastings algorithm is known as reversible-jump MCMC.

4.2 Reversible-jump MCMC

We now formally introduce the reversible-jump algorithm of [2]. Let the target distribution be $\pi(m, \theta) = \pi(m)\pi(\theta \mid m)$, where m can be thought of as the model taking values in a finite space $\mathcal{M}$, and $\theta \in \mathbb{R}^{d(m)}$ can be thought of as the parameter of the model m, where $d(m)$ is the dimension of the parameter of model m. We assume that $\pi(\theta \mid m)$ is the density with respect to the Lebesgue measure on $\mathbb{R}^{d(m)}$.

Suppose that at each (m, θ), we can propose the next state using the following scheme.

(i) Draw m' with probability $q(m, m')$.

(ii) Suppose that $d(m') > d(m)$. Then, we draw $w \in \mathbb{R}^{d(m')-d(m)}$ with density $q_{m \to m'}(\theta, w)$ and set $\theta' = f_{m \to m'}(\theta, w)$, where $f_{m \to m'}$ is a differentiable bijective function.

(iii) Accept θ' with probability $\alpha((\theta, w), \theta')$.

The proposal move from (m', θ') to (m, θ) is implemented by reversing the above calculations. That is, we first draw m with probability $q(m', m)$ and then calculate $(\theta, w) = f^{-1}_{m \to m'}(\theta')$. We accept θ with probability $\alpha(\theta', (\theta, w))$, and the auxiliary variable w is discarded.

Recall that this algorithm is reversible with respect to π if

$$\int_{\theta \in A} \pi(m, \theta) P((m, \theta), (m', B)) \mathrm{d}\theta = \int_{\theta' \in B} \pi(m', \theta') P((m', \theta'), (m, A)) \mathrm{d}\theta \tag{2}$$

for any $A \in \mathcal{B}(\mathbb{R}^{d(m)}), B \in \mathcal{B}(\mathbb{R}^{d(m')})$. We now use (2) to find the expression of α.

Without loss of generality we assume $d(m') > d(m)$. According to the proposal scheme described above,

$$P((m,\theta),(m',B)) = q(m,m')\int_{E_\theta} q_{m\to m'}(\theta,w)\alpha((\theta,w),\theta')\mathrm{d}w$$

where $\theta' = f_{m\to m'}(\theta,w)$ and $E_\theta = \{w\colon f_{m\to m'}(\theta,w)\in B\}$. Hence,

$$\begin{aligned}&\int_{\theta\in A}\pi(m,\theta)P((m,\theta),(m',B))\mathrm{d}\theta\\ =&\int_{(\theta,w)\in E}\pi(m,\theta)q(m,m')q_{m\to m'}(\theta,w)\alpha((\theta,w),\theta')\mathrm{d}\theta\mathrm{d}w,\end{aligned}$$

where

$$E = \{(\theta,w)\colon \theta\in A, f_{m\to m'}(\theta,w)\in B\}.$$

Because $f_{m\to m'}$ is differentiable and bijective, we can perform a change of variable from (θ,w) to $\theta' = f_{m\to m'}(\theta,w)$, i.e.,

$$\mathrm{d}\theta' = \left|\det\left(\frac{\partial\theta'}{\partial(\theta,w)}\right)\right|\mathrm{d}\theta\mathrm{d}w.$$

This yields

$$\begin{aligned}&\int_{\theta\in A}\pi(m,\theta)P((m,\theta),(m',B))\mathrm{d}\theta\\ =&\int_{\theta'\in F}\pi(m,\theta)q(m,m')q_{m\to m'}(\theta,w)\alpha((\theta,w),\theta')\left|\det\left(\frac{\partial\theta'}{\partial(\theta,w)}\right)\right|^{-1}\mathrm{d}\theta',\end{aligned}\tag{3}$$

where $(\theta,w) = f^{-1}_{m\to m'}(\theta')$, and

$$F = \left\{\theta'\colon \theta'\in B,\ f^{-1}_{m\to m'}(\theta')\in A\times\mathbb{R}^{d(m')-d(m)}\right\}.$$

We need to make this equal to the right-hand side of (2), which can be written as

$$\int_{\theta'\in F}\pi(m',\theta')q(m',m)\alpha(\theta',(\theta,w))\mathrm{d}\theta'.\tag{4}$$

Note that no auxiliary variable needs to be sampled when moving back from θ' to θ. Comparing (3) with (4), we see that all we need is

$$\frac{\alpha((\theta,w),\theta')}{\alpha(\theta',(\theta,w))} = \frac{\pi(m',\theta')q(m',m)}{\pi(m,\theta)q(m,m')q_{m\to m'}(\theta,w)}\left|\det\left(\frac{\partial\theta'}{\partial(\theta,w)}\right)\right|.$$

This holds if we set

$$\alpha((\theta,w),\theta') = \min\left\{1,\ \frac{\pi(m',\theta')q(m',m)}{\pi(m,\theta)q(m,m')q_{m\to m'}(\theta,w)}\left|\det\left(\frac{\partial\theta'}{\partial(\theta,w)}\right)\right|\right\},\tag{5}$$

$$\alpha(\theta',(\theta,w)) = \min\left\{1,\ \frac{\pi(m,\theta)q(m,m')q_{m\to m'}(\theta,w)}{\pi(m',\theta')q(m',m)}\left|\det\left(\frac{\partial(\theta,w)}{\partial\theta'}\right)\right|\right\}.\tag{6}$$

This is known as the reversible-jump MCMC algorithm. To summarize:

Theorem 4.1. *The reversible-jump MCMC algorithm has stationary distribution $\pi(m, \theta)$, provided that the acceptance probability is calculated by* (5) *and* (6).

Remark 4.1. The general methodology described in this section does not fully justify the MCMC algorithm considered in Section 4.1 for the change-point detection problem, though that can be justified by essentially the same reasoning. For the change-point problem, we can think of K as the model, and our target posterior distribution is $p(K, \theta \mid D)$ where $\theta = (b_1, \ldots, b_K, h_0, \ldots, h_K)$. When we insert a break point, we sample b^* (location of the new break point) and u (auxiliary variable) and then compute the mapping $(\theta, b^*, u) \mapsto \theta'$. However, this mapping is not bijective, since to move back, we need to know which break point to delete. So it is better to think of the generation of (b^*, u) as a two-step procedure: we first decide which interval $[b_k, b_{k+1})$ contains the new break point, and then sample b^* (restricted to this interval) and u. Once the interval $[b_k, b_{k+1})$ is selected, the mapping $(b^*, u, h_k) \mapsto (b'_{k+1}, h'_k, h'_{k+1})$ becomes bijective, with all other parameters treated as fixed. Similarly, when reducing K, we need to first sample which break point to delete and then perform a deterministic calculation to obtain the new parameter vector.

4.3 Pseudo-marginal MCMC

Essentially, pseudo-marginal MCMC is a generalization of Metropolis–Hastings sampling where we replace $\pi(x)$ with an unbiased estimator of it. Denote such an estimator by $\hat{\pi}(x)$ and its distribution by F_x. Actually, it only needs to be unbiased up to a normalizing constant (which makes the resulting algorithm useful for, e.g., Bayesian posterior calculations); this means that $\mathbb{E}[\hat{\pi}(x)] = \int \hat{\pi} F_x(\mathrm{d}\hat{\pi}) = C\pi(x)$ where $C > 0$ is a constant independent of x

Algorithm 4.1 (Pseudo-marginal MCMC)**.** Initialize the sampler at some $X_0 = x_0$ and estimator $\hat{\pi}(x_0)$. For $t = 1, 2, \ldots,$

(i) Sample Y from the distribution $Q(X_{t-1}, \cdot)$.

(ii) Given $Y = y$, generate $\hat{\pi}(y)$.

(iii) Calculate the acceptance probability

$$\hat{\alpha} = \min\left\{1, \frac{\hat{\pi}(y)q(y, x_{t-1})}{\hat{\pi}(x_{t-1})q(x_{t-1}, y)}\right\}.$$

(iv) With probability $\hat{\alpha}$, set $X_t = y$ and $\hat{\pi}(x_t) = \hat{\pi}(y)$; with probability $1 - \hat{\alpha}$, set $X_t = x_{t-1}$ and $\hat{\pi}(x_t) = \hat{\pi}(x_{t-1})$.

Particular attention should be given to step (iv). No matter whether we accept the proposed state y or stay at the previous state x_{t-1}, we will keep using $\hat{\pi}(y)$ or $\hat{\pi}(x_{t-1})$ in the next iteration. This is crucial to ensuring that the algorithm is invariant with respect to π, which we prove below.

Theorem 4.2. *Under the unbiasedness assumption on $\hat{\pi}(x)$, $(X_t)_{t\geq 0}$ generated from the pseudo-marginal MCMC algorithm has stationary distribution π.*

Proof. Our strategy for proving this result is different from the previous proofs. The main idea is to view the pseudo-marginal MCMC algorithm as a bivariate Markov chain and show that its invariant distribution has $\pi(x)$ as the marginal. This technique is very important and will be often used in later units.

First, let's define $U_x = \hat{\pi}(x)/\pi(x)$, and denote the density of U_x by $r(x, u_x)$. Generating $\hat{\pi}(x)$ is equivalent to generating the random variable U_x, and since $\hat{\pi}(x)$ is unbiased, we have

$$\int u_x r(x, u_x) \mathrm{d}u_x = C,$$

for some fixed constant $C > 0$. (Of course, we cannot observe the value of U_x in practice.) Now we view the pseudo-marginal MCMC algorithm as targeting the joint distribution

$$\bar{\pi}(x, u_x) = C^{-1} u_x r(x, u_x) \pi(x).$$

According steps (i) and (ii), the proposal density from (x, u_x) to (y, u_y) is given by

$$q((x, u_x), (y, u_y)) = q(x, y) r(y, u_y).$$

If we calculate the acceptance probability as in the standard Metropolis–Hastings algorithm, we get

$$\begin{aligned}
\alpha((x, u_x), (y, u_y)) &= \min\left\{1, \frac{\bar{\pi}(y, u_y) q((y, u_y), (x, u_x))}{\bar{\pi}(x, u_x) q((x, u_x), (y, u_y))}\right\} \\
&= \min\left\{1, \frac{u_y r(y, u_y) \pi(y) q(y, x) r(x, u_x)}{u_x r(x, u_x) \pi(x) q(x, y) r(y, u_y)}\right\} \\
&= \min\left\{1, \frac{\hat{\pi}(y) q(y, x)}{\hat{\pi}(x) q(x, y)}\right\},
\end{aligned}$$

which coincides with the expression given in the step (iii). That is, pseudo-marginal MCMC is a standard Metropolis–Hastings algorithm with stationary distribution $\bar{\pi}(x, u_x)$. When running this algorithm, we only collect the samples $X_0, X_1, \ldots$, but their stationary distribution is just the marginal distribution of $\bar{\pi}(x, u)$, which is $\pi(x)$. □

Example 4.1. Here is a typical scenario in Bayesian statistics where pseudo-marginal MCMC can be helpful. Consider a joint posterior distribution $\pi(x, z)$, where x is the parameter of interest (e.g. x can be the model in a model selection problem, and z is the parameter associated with model x). We want to directly sample from the marginal distribution $\pi(x)$, which requires us to compute the integral $\pi(x) = \int \pi(x, z) \mathrm{d}z$ up to a normalizing constant. In some cases, we can use a conjugate prior on z given x and this integral has a closed-form expression; one example is the spike-and-slab variable selection discussed in Unit 3. However, very often $\int \pi(x, z) \mathrm{d}z$ is difficult to compute, in which case the distribution $\pi(x)$

is described as doubly intractable [3]. One simple example is variable selection for logistic regression (instead of linear regression), which was considered in [1]. We can always construct an unbiased estimator for $\int \pi(x,z)\mathrm{d}z$ using importance sampling. Letting $Z_1, Z_2, \ldots, Z_n$ be i.i.d. samples from a distribution with density $g(z)$, we can express our estimator by

$$\hat{\pi}(x) = \frac{1}{n}\sum_{i=1}^{n} \frac{\pi(x, Z_i)}{g(Z_i)}.$$

Then we can use pseudo-marginal MCMC to sample from $\pi(x)$. We can further generalize this method by using dependent samples generated sequentially, which we describe in the exercise below. Application of this sequential importance sampling technique to pseudo-marginal MCMC was studied in [1].

Remark 4.2. Consider the model selection problem discussed for reversible-jump MCMC. Let's write

$$\pi(m,\theta) = C\, p(D \,|\, m,\theta) p(\theta \,|\, m) p(m)$$

where C is the normalizing constant, D is the data, $p(D \,|\, m,\theta)$ is the likelihood of (m,θ), $p(m)$ is the prior probability of the model m, and $p(\theta \,|\, m)$ is the conditional prior density of θ given model m. Reversible-jump MCMC aims at directly sampling from $\pi(m,\theta)$, while pseudo-marginal MCMC aims at the marginal distribution

$$\pi(m) = C\, p(m) \int p(D \,|\, m,\theta) p(\theta \,|\, m) d\theta,$$

where the integral needs to be unbiasedly estimated.

Exercise 4.1. Let $I = \int_{\mathbb{R}} f(z)\mathrm{d}z < \infty$. Let $Z_1, Z_2, \ldots, Z_n$ be generated from a Markov chain; denote the density of Z_1 by $g(z_1)$ and the density of Z_i given $Z_{i-1} = z_{i-1}$ by $g(z_{i-1}, z_i)$. Assume that $p(z), p(z,z') > 0$ everywhere. Define

$$\hat{I} = \frac{1}{n}\left(\frac{f(Z_1)}{g(Z_1)} + \sum_{i=2}^{n} \frac{f(Z_i)}{g(Z_{i-1}, Z_i)}\right).$$

Show that $\hat{I}$ is an unbiased estimator of I.

References

[1] Christophe Andrieu and Gareth O Roberts. The pseudo-marginal approach for efficient Monte Carlo computations. *The Annals of Statistics*, 37(2):697–725, 2009.

[2] Peter J Green. Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, 82(4):711–732, 1995.

[3] Iain Murray, Zoubin Ghahramani, and David JC MacKay. MCMC for doubly-intractable distributions. In *Proceedings of the Twenty-Second Conference on Uncertainty in Artificial Intelligence*, pages 359–366, 2006.