Biostats 285: Advanced Bayesian Computing **Spring 2021**

Lecture 2: April 13

Lecturer: Andrew Holbrook *Scribes: Nicholas Marco*

Note: *LaTeX template courtesy of UC Berkeley EECS dept.*

2.1 Introduction

2.1.1 Metropolis-Hastings Algorithm

Let the target distribution have density π. Let $\mathbf{X}_n$ be the n^{th} sample from our Markov chain. We will denote the proposed value for our $(n+1)^{th}$ sample as $\mathbf{Y}_{n+1}$. Let $q(\mathbf{X}_n, \mathbf{Y}_{n+1})$ be the proposal distribution of $\mathbf{Y}_{n+1}$. Thus we have the probability accepting the transition from $\mathbf{X}_n$ to $\mathbf{Y}_{n+1}$ is:

$$\alpha(\mathbf{X}_n, \mathbf{Y}_{n+1}) = \begin{cases} \min\left\{\frac{\pi(\mathbf{Y}_{n+1})}{\pi(\mathbf{X}_n)}\frac{q(\mathbf{Y}_{n+1},\mathbf{X}_n)}{q(\mathbf{X}_n,\mathbf{Y}_{n+1})}, 1\right\}, & \pi(\mathbf{X}_n)q(\mathbf{X}_n, \mathbf{Y}_{n+1}) > 0 \\ 1, & \pi(\mathbf{X}_n)q(\mathbf{X}_n, \mathbf{Y}_{n+1}) = 0 \end{cases} \tag{2.1}$$

If the proposed value is accepted, then $\mathbf{X}_{n+1} = \mathbf{Y}_{n+1}$, otherwise $\mathbf{X}_{n+1} = \mathbf{X}_n$.

2.1.2 Random-Walk Metropolis Algorithm (RWM)

In cases where π cannot be sampled from, people often use the *symmetric random-walk Metropolis algorithm* (RWM). Using RWM, the proposal distribution becomes $\mathbf{Y}_{n+1} = \mathbf{X}_n + \mathbf{Z}_{n+1}$, where $\mathbf{Z}_{n+1}$ are i.i.d. from a fixed symmetric distribution (i.e. $\mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I}_d)$).

2.1.3 Comparing Markov Chains

The performance of the RWM algorithm greatly depends on the scale parameter used in fixed symmetric distribution (for example σ^2 when $\mathbf{Z}_{n+1} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I}_d)$). The choice of the scale parameter (σ^2) becomes important because too small of a scale parameter would lead to slow exploration of the parameter space, and too large of a scale parameter would lead to a low acceptance rate.

When considering two Markov chains, P_1 and P_2, it is important to be able to compare the two chains in order to say which one is "better". To do that, we may use one of the following definitions:

1. P_1 *converges faster than* P_2 if $sup_A|P_1^N(x, A) - \pi(A)| \leq sup_A|P_2^N(x, A) - \pi(A)|$ for all n and x.
2. P_1 *has smaller variance than* P_2 if $\text{Var}(\frac{1}{n}\sum_{i=1}^n g(X_i))$ is smaller when $\{X_i\}$ follows $P_!$ than when it follows P_2 (assuming $\{X_n\}$ is in stationarity).

3. Similar to 2, if a Markov chain $\{X_n\}$ is in stationarity, then for large n, we have $\text{Var}(\frac{1}{n}\sum_{i=1}^{n} g(X_i)) \approx \frac{1}{n}\text{Var}_\pi(g)\tau_g$, where $\tau_g = \sum_{k=-\infty}^{\infty} \text{Corr}(g(X_0), g(X_k)) = 1+2\sum_{i=1}^{\infty} \text{Corr}(g(X_0), g(X_i))$ is the integrated correlation time. Thus, we say that P_1 *has smaller asymptotic variance than* P_2 if τ_g is smaller under P_1 than P_2.

4. Assuming $\{X_n\}$ is in stationarity, we say that P_1 *mixes faster than* P_2 if $\mathbf{E}[(X_n - X_{n-1})^2]$ is larger under P_1 than under P_2. We can estimate $\mathbf{E}[(X_n - X_{n-1})^2]$ by $\frac{1}{n-B}\sum_{i=B}^{n}(X_i - X_{i-1})^2$, where B are the number of samples in the burn-in period.

2.2 Optimal Scaling of Random-Walk Metropolis (RWM)

2.2.1 Optimal Scaling Parameter and Acceptance Rate

Consider RWM on $\mathbb{R}^d$, where the target densities have the form:

$$\pi(x_1, x_2, \ldots, x_d) = f(x_1)f(x_2)\ldots f(x_d) \tag{2.2}$$

Consider a RWM algorithm where the proposals are of the form $\mathbf{Y}_{n+1} = \mathbf{X}_n + \mathbf{Z}_{n+1}$, where $\mathbf{Z}_i \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I}_d)$. Let $\sigma = \ell/\sqrt{d}$ for some $\ell > 0$. Roberts et al. proved that maximizing $h(\ell)$ with respect to ℓ would give you the optimal σ. Roberts et al. proved that if we maximize the speed function ($h(\ell)$) with respect to ℓ, then we will be able to find the optimal σ. The speed function takes the form of:

$$h(\ell) = 2\ell^2\Phi\left(-\frac{\sqrt{I}\ell}{2}\right),$$

where Φ is the cdf of a standard normal random variable and I is a constant depending on f. The optimal value of ℓ is found to be $2.38/\sqrt{I}$. Asymptotically, as $d \to \infty$, Roberts et all proved that the optimal acceptance rate is 0.234. Numerical studies have shown that for $d \geq 5$ that the acceptance rate should be between 0.1 and 0.6 (figure 5 graphically shows this). For $d = 1$, the optimal acceptance rate should be 0.44.

Rosenthal and Roberts expanded on this concept by considering target distributions of the form:

$$\pi(\mathbf{x}) = \prod_{i=1}^{d} C_i f(C_i x_i). \tag{2.3}$$

Roberts and Rosenthal concluded that the larger $\mathbf{E}(C_i^2)/(\mathbf{E}(C_i)^2)$, the slower the mixing will be (the target distribution is considered more *inhomogeneous* the larger $\mathbf{E}(C_i^2)/(\mathbf{E}(C_i)^2)$). Consider a case where the target distribution is $\mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$ and $\mathbf{Z}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_p)$. Note that this is equivalent to letting $\mathbf{Z}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_p)$ and the target distribution being $\mathcal{N}(\mathbf{0}, \mathbf{\Sigma}\mathbf{\Sigma}_p^{-1})$. This corresponds to the case where $C_i = \sqrt{\lambda_i}$ where $\{\lambda_i\}_{i=1}^d$ are eigenvalues of the matrix $\mathbf{\Sigma}\mathbf{\Sigma}_p^{-1}$. Thus for large d, we have

$$b = \frac{\mathbf{E}(C_i^2)}{(\mathbf{E}(C_i))^2} \approx \frac{\frac{1}{d}\sum_{j=1}^{d}\lambda_j}{\left(\frac{1}{d}\sum_{j=1}^{d}\sqrt{\lambda_j}\right)^2} = d\frac{\sum_{j=1}^{d}\lambda_j}{\left(\sum_{j=1}^{d}\sqrt{\lambda_j}\right)^2}$$

Since we know that $b = \frac{\mathbf{E}(C_i^2)}{(\mathbf{E}(C_i))^2} \geq 1$, we can see that if $\lambda_j = c$ for all j, we have

$$\frac{\mathbf{E}(C_i^2)}{(\mathbf{E}(C_i))^2} \approx \frac{d(dc)}{(d\sqrt{c})^2} = 1$$

Thus we have minimized the expression above, and can conclude that the the mixing will be optimal when $\mathbf{\Sigma}_p = k\mathbf{\Sigma}$. If not, the chain will lead to additional slow-down by a factor of b. Using the results from Roberts (1997), we know that the optimal $\mathbf{\Sigma}_p$ is:

$$\mathbf{\Sigma}_p = [(2.38)^2/d]\mathbf{\Sigma}.$$

2.2.2 Metropolis-Adjusted Langevin algorithm (MALA)

Metropolis-Adjusted Langevin algorithm (MALA) is an algorithm similar to RWM, except we have that $\mathbf{Y}_{n+1} = \mathbf{X}_n + \mathbf{Z}_{n+1}$, where

$$\mathbf{Z}_i \sim \mathcal{N}(\frac{\sigma^2}{2}\Delta log(\pi(\mathbf{X}_n)), \sigma^2\mathbf{I}_d).$$

Using the information from the gradient of the target distribution allows us to take larger steps while also having larger acceptance rates when compared to RWM with the same scale parameter. Therefore, we are able to achieve faster convergence with MALA than RWM, however it may be expensive to calculate the gradient at each step. Roberts and Rosenthal proved that if the target distribution has the same form as equation 2.2, that the optimal acceptance rate is 0.574 (higher than the 0.234 under RWM).

2.3 Adaptive MCMC

From the last section, we know that for $d \geq 5$, we have that an acceptance rate of 0.234 seems to be optimal. However, it is not clear how we should pick the scaling parameter and $\mathbf{\Sigma}_p$. Suppose we consider algorithms which will optimize the mixing themselves. Let $\{P_\gamma\}_{\gamma\in\mathcal{Y}}$ be a family of Markov chain Kernels, each having the same stationary distribution π. Let Γ_n be the chosen kernel at choice at the n^{th} iteration (for RWM, this can correspond to letting $\mathbf{Z}_n \sim \mathcal{N}(\mathbf{0}, \sigma_\gamma^2\mathbf{\Sigma}_\gamma)$). Thus we have that

$$P(\mathbf{X}_{n+1} \in A|\mathbf{X}_n = \mathbf{x}, \Gamma_n = \gamma, \mathbf{X}_{n-1}, \ldots, \mathbf{X}_0, \Gamma_{n-1}, \ldots, \Gamma_0) = P_\gamma(\mathbf{x}, A).$$

In practice the pairs process $\{(\mathbf{X}_n, \Gamma_n)\}_{n=0}^\infty$ is Markovian. Roberts and Rosenthal (2005) proved the that an adaptive sampling scheme will asymptotically converge to the target distribution and that the WLLN holds assuming the following:

1. *Diminishing Adaptation*

$$\lim_{n\to\infty} sup_{x\in\mathcal{X}}||P_{\Gamma_{n+1}}(x,.) - P_{\Gamma_n}(x,.)|| = 0 \text{ in probability}$$

2. *Containment*

$$\{\inf\{n \geq 1 : ||P_{\Gamma_i}^n(x,.) - \pi(.)|| \leq \epsilon\}\}_{i=1}^\infty \text{ is bounded in probability, } \epsilon > 0$$

Containment holds for most reasonable adaptive schemes and is therefore largely ignored. The Diminishing Adaptation condition says that as $n \to \infty$, the amount of adaptation goes to zero (i.e. if $p(n)$ be the probability that the algorithm adapts at the n^{th} iteration, then $p(n) \to 0$ as $n \to \infty$).

2.3.1 Adaptive Metropolis

The Adaptive Metropolis (AM) algorithm takes advantage of the fact that when we have a normal target distribution and use the RWM algorithm, that the optimal $\mathbf{Z}_i$ has covariance of the form $(2.38)^2/d$ times

the target covariance matrix, $\mathbf{\Sigma}$. However, since $\mathbf{\Sigma}$ is not know, we will use an empirical estimate of the covariance matrix using $\mathbf{X}_0, \ldots, \mathbf{X}_n$. Thus if we let

$$\mathbf{\Sigma}_n = \frac{1}{n}\left(\sum_{i=0}^{n} \mathbf{X}_i \mathbf{X}_i' - (n+1)\bar{\mathbf{X}}_n \bar{\mathbf{X}}_n'\right),$$

our AM algorithm will be given by

$$\mathbf{Y}_{n+1} \sim \mathcal{N}(\mathbf{X}_n, [(2.38)^2/d]\mathbf{\Sigma}_n).$$

To ensure that the covariance matrix is positive definite, we can add $\epsilon \mathbf{I}_d$ to $\mathbf{\Sigma}_n$ for some $\epsilon > 0$. Another popular proposal to ensure the the covariance matrix is p.d. is to use the following proposal distribution:

$$(1-\beta)\mathcal{N}(\mathbf{X}_n, [(2.38)^2/d]\mathbf{\Sigma}_n) + \beta\mathcal{N}(\mathbf{X}_n, \mathbf{\Sigma}_0),$$

for some $0 < \beta < 1$ and some p.d. $\mathbf{\Sigma}_0$. Since empirical estimates change at the n^{th} iteration only by $\mathcal{O}(1/n)$, we know that this satisfies the Diminishing Adaptation condition.

2.3.2 Adaptive Metropolis-Within-Gibbs

n alternative to the full-dimensional Metropolis algorithm is the "Metropolis-within-Gibbs" algorithm in which each random variable is updated one at a time using a one-dimensional Metropolis algorithm. In this sampling scheme, we will be updating the i^{th} coordinate using the proposal increment distribution $\mathcal{N}(0, e^{2ls_i})$, where ls_i is the log of the standard deviation of increment. From the optimal scaling of RMW section, we know that the acceptance rate should be close to 0.44, however this can be difficult to do manually. Roberts and Rosenthal (2006) proposed the following adaptive algorithm to help find the optimal ls_i:

1. Start with initial estimate of ls_i
2. Run the Metropolis-within-Gibbs algorithm for the batch size (i.e. 50 iterations)
3. For the n^{th} batch, if the acceptance rate is greater than 0.44 for the i^{th} coordinate, set $ls_i = ls_i + \delta(n)$, for some $\delta(n) > 0$. If the acceptance rate is less than 0.44 for the i^{th} coordinate, set $ls_i = ls_i - \delta(n)$.
4. Repeat steps (2) and (3) until convergence of the Markov chain.

In order to satisfy Diminishing Adaptation, we require that $\delta(n) \to \infty$ as $n \to \infty$. To ensure this, we can use $\delta(n) = \min\{0.01, n^{-1/2}\}$. In simulations, this adaptive sampling scheme seems to have better mixing compared to fixing ls_i, especially in high dimensions.