Lecture Notes for STAT546: Computational Statistics

—Lecture 11: Monte Carlo

Faming Liang

Purdue University

September 25, 2024

Motivation Example

Consider the Markov transition probability matrix

$$P_\theta = \begin{pmatrix} 1-\theta & \theta \\ \theta & 1-\theta \end{pmatrix},$$

on the state space $\mathcal{X} = \{1, 2\}$ for some fixed $\theta \in (0, 1)$. It is easy to see that for any θ, P_θ leaves $\pi = (1/2, 1/2)$ invariant; that is, $\pi P_\theta = \pi$. However, if we let $\theta : \mathcal{X} \to (0, 1)$ be a function of the current state X, i.e., introducing self-adaptation of proposals, then the transition probability matrix becomes

$$\widetilde{P} = \begin{pmatrix} 1-\theta(1) & \theta(1) \\ \theta(2) & 1-\theta(2) \end{pmatrix},$$

which admits $(\theta(2)/(\theta(1)+\theta(2)), \theta(1)/(\theta(1)+\theta(2)))$ as its invariant distribution.

This example implies that introduction of self-adaptation of proposals might not preserve the stationarity of the target distribution.

For the above example, to recover the target distribution π, one can either remove or diminish the dependence of P_θ on X with iterations. This has led to some adaptive MCMC algorithms developed in the literature. These algorithms can be roughly divided into the following four categories:

- *Stochastic approximation-based adaptive algorithms:* The algorithms diminish the adaptation gradually with iterations.
- *Adaptive independent MH algorithms:* The algorithms work with proposals which are adaptive, but do not depend on the current state of the Markov chain. The diminishing adaptation condition may not necessarily hold for them.
- *Regeneration-based adaptive algorithms:* The algorithms are designed based on a basic property of the Markov chain, whose future outputs become independent of the past after each regeneration point.
- *Population-based adaptive algorithms:* The algorithms work on an enlarged state space, which gives us much freedom to design adaptive proposals and to incorporate sophisticated computational techniques into MCMC simulations.

Stochastic Approximation-based Adaptive Algorithms

Haario *et al.* (2001) prescribed an adaptive Metropolis algorithm which learns to build an efficient proposal distribution on the fly; that is, the proposal distribution is updated at each iteration based on the past samples. Under certain settings, it was shown that the proposal distribution will converge to the "optimal" one. Andrieu and Robert (2001) observed that the algorithm of Haario *et al.* (2001) can be viewed as a stochastic approximation algorithm (Robbins and Monro, 1951). Then, under the framework of stochastic approximations, Atchadé and Rosenthal (2005) and Andrieu and Moulines (2006) proved the ergodicity of more general adaptive algorithms. Andrieu and Moulines (2006) also proved a central limit theorem result. The theory on the adaptive MCMC algorithms was further developed by Roberts and Rosenthal (2007) and Yang (2007). They present somewhat simpler conditions, which still ensure ergodicity for specific target distributions.

Ergodicity and Weak Law of Large Numbers

Let $\pi(\cdot)$ be a fixed target distribution defined on a state space $\mathcal{X} \subset \mathbb{R}^d$ with σ-algebra $\mathcal{F}$. Let $\{P_\theta\}_{\theta \in \Theta}$ be a collection of Markov chain transition kernels on $\mathcal{X}$, each of which admits $\pi(\cdot)$ as the stationary distribution, i.e., $(\pi P_\theta)(\cdot) = \pi(\cdot)$. Assuming that P_θ is irreducible and aperiodic, then P_θ is ergodic with respect to $\pi(\cdot)$; that is, $\lim_{n\to\infty} \|P_\theta^n(x,\cdot) - \pi(\cdot)\| = 0$ for all $x \in \mathcal{X}$, where $\|\mu(\cdot) - \nu(\cdot)\| = \sup_{A\in\mathcal{F}} \|\mu(A) - \nu(A)\|$ is the usual total variation distance. So, if θ is kept fixed, then the samples generated by P_θ will eventually converge to $\pi(\cdot)$ in distribution.

However, some transition kernel P_θ may lead to a far less efficient Markov chain than others, and it is hard to know in advance which transition kernel is preferable. To deal with this, adaptive MCMC algorithms allow the transition kernel to be changed at each iteration according to some specific rules. Let Γ_n be a Θ-valued random variable, which specifies the transition kernel to be used at iteration n. Let X_n denote the state of the Markov chain at iteration n. Thus,

$$P_\theta(x, A) = P(X_{n+1} \in A | X_n = x, \Gamma_n = \theta, \mathcal{G}_n), \quad A \in \mathcal{F},$$

where $\mathcal{G}_n = \sigma(X_0, \ldots, X_n, \Gamma_0, \ldots, \Gamma_n)$ is a filtration generated by $\{X_i, \Gamma_i\}_{i \leq n}$.

Define

$$A^n((x,\theta),B) = P(X_n \in B | X_0 = x, \Gamma_0 = \theta), \quad B \in \mathcal{F},$$

which denotes the conditional probabilities of X_n given the initial conditions $X_0 = x$ and $\Gamma_0 = \theta$, and

$$T(x,\theta,n) = \|A^n((x,\theta),\cdot) - \pi(\cdot)\| = \sup_{B \in \mathcal{F}} |A^n((x,\theta),B) - \pi(B)|,$$

which denotes the total variation distance between the distribution of X_n and the target distribution $\pi(\cdot)$.

To ensure ergodicity of the adaptive Markov chain, i.e., $\lim_{n\to\infty} T(x,\theta,n) = 0$ for all $x \in \mathcal{X}$ and $\theta \in \Theta$, Roberts and Rosenthal (2007) prescribes two conditions, namely the *bounded convergence* condition and the *diminishing adaptation* condition. Let

$$M_\epsilon(x,\theta) = \inf\{n \geq 1 : \|P_\theta^n(x,\cdot) - \pi(\cdot)\| \leq \epsilon\},$$

be the convergence time of the kernel P_θ when starting in state $x \in \mathcal{X}$. The *bounded convergence* condition is that for any $\epsilon > 0$, the stochastic process $\{M_\epsilon(X_n, \Gamma_n)\}$ is bounded in probability given the initial values $X_0 = x$ and $\Gamma_0 = \theta$.

Let

$$D_n = \sup_{x \in \mathcal{X}} \| P_{\Gamma_{n+1}}(x, \cdot) - P_{\Gamma_n}(x, \cdot) \|.$$

The diminishing adaptation conditions states that $\lim_{n \to \infty} D_n = 0$, which can be achieved by modifying the parameters by smaller and smaller amounts as in the adaptive Metropolis algorithm of Haario *et al.* (2001), or by doing the adaption with smaller and smaller probability as in the adaptive evolutionary Monet Carlo algorithm of Ren *et al.* (2008). In summary, Roberts and Rosenthal (2007) proved the following theorem:

Theorem 1

For an MCMC algorithm with adaptive proposals, if it satisfies the bounded convergence and diminishing adaptation conditions, then it is ergodic with respect to the stationary distribution $\pi(\cdot)$.

Since the quantity $M_\epsilon(x,\theta)$ is rather abstract, Roberts and Rosenthal (2007) gave one condition, *simultaneous geometrical ergodicity*, which ensures the bounded convergence condition. A family $\{P_\theta\}_{\theta\in\Theta}$ of Markov chain transition kernels is said to be *simultaneously geometrically ergodic* if there exists $C \in \mathcal{F}$, a drift function $V : \mathcal{X} \to [1,\infty)$, $\delta > 0$, $\lambda < 1$ and $b < \infty$, such that $\sup_{x\in C} V(x) = v < \infty$ and the following conditions hold:

(i) (Minorization condition) For each $\theta \in \Theta$, there exists a probability measure $\nu_\theta(\cdot)$ on C with $P_\theta(x,\cdot) \geq \delta\nu_\theta(\cdot)$ for all $x \in C$.

(ii) (Drift condition) $P_\theta V \leq \lambda V + bI(x \in C)$, where $I(\cdot)$ is the indicator function.

This results in the following theorem:

Theorem 2

For an MCMC algorithm with adaptive proposals, if it satisfies the diminishing adaptation condition and the family $\{P_\theta\}_{\theta\in\Theta}$ is simultaneously geometrically ergodic with $E(V(X_0)) < \infty$, then it is ergodic with respect to the stationary distribution $\pi(\cdot)$.

In addition to the ergodicity of the Markov chain, in practice, one often interests in the weak law of large numbers (WLLN), i.e., whether or not the sample path average $(1/n)\sum_{i=1}^{n} h(X_i)$ will converge to the mean $E_\pi h(x) = \int h(x)\pi(x)dx$ for some function $h(x)$. For adaptive MCMC algorithms, this is even more important than the ergodicity to some extent. Under slightly stronger conditions, the *simultaneous uniform ergodicity* and *diminishing adaptation* conditions, Roberts and Rosenthal (2007) showed that $(1/n)\sum_{i=1}^{n} h(X_i)$ will converge to $E_\pi h(x)$, provided that $h(x)$ is bounded. A family $\{P_\theta\}_{\theta\in\Theta}$ of Markov chain transition kernels is said to be *simultaneously uniformly ergodic* if $\sup_{(x,\theta)\in\mathcal{X}\times\Theta} M_\epsilon(x,\theta) < \infty$. In summary, the WLLN for adaptive MCMC algorithms can be stated as follows:

Theorem 3

For an MCMC algorithm with adaptive proposals, if it satisfies the simultaneous uniform ergodicity and diminishing adaptation conditions, then for any starting values $x \in \mathcal{X}$ and $\theta \in \Theta$,

$$\frac{1}{n}\sum_{i=1}^{n} h(X_i) \to E_\pi h(x),$$

provided that $h : \mathcal{X} \to \mathbb{R}$ is a bounded measurable function.

Adaptive Metropolis Algorithm

Let $\pi(x)$ denote the target distribution. Consider a Gaussian random-walk MH algorithm, for which the proposal distribution is $q(x, y) = N(y; x, \Sigma)$, and $N(y; x, \Sigma)$ denotes the density of a multivariate Gaussian with mean x and covariance matrix Σ. It is known that either too small or too large a covariance matrix will lead to a highly correlated Markov chain. Under certain settings, Gelman *et al.* (1996) showed that the "optimal" covariance matrix for the Gaussian random-walk MH algorithm is $(2.38^2/d)\Sigma_\pi$, where d is the dimension of x and Σ_π is the true covariance matrix of the target distribution $\pi(\cdot)$. Haario *et al.* (2001) proposed to "learn Σ_π on the fly"; that is, estimating Σ_π from the empirical distribution of the available Markov chain outputs, and thus adapting the estimate of Σ while the algorithm runs.

- Initialize X_0, μ_0 and Σ_0.
- At iteration $k+1$, given X_k, μ_k and Σ_k
  a. Generate X_{k+1} via the MH kernel $P_{\theta_k}(X_k, \cdot)$, where $\theta_k = (\mu_k, \Sigma_k)$.
  b. Update

$$
\begin{aligned}
\mu_{k+1} &= \mu_k + \gamma_{k+1}(X_{k+1} - \mu_k), \\
\Sigma_{k+1} &= \Sigma_k + \gamma_{k+1}\left[(X_{k+1} - \mu_k)(X_{k+1} - \mu_k)^T - \Sigma_k\right].
\end{aligned}
$$

Let $\lambda\Sigma_k$ denote the covariance matrix used by the adaptive Metropolis algorithm at iteration k, where λ is preset at $2.38^2/d$ as suggested by Gelman *et al.* (1996). As pointed out by Andrieu and Thoms (2008), if $\lambda\Sigma_k$ is either too large in some directions or too small in all directions, the algorithm will have either a very small or a very large acceptance probability, rendering a slow learning of Σ_π due to limited exploration of the sample space $\mathcal{X}$.

To alleviate this difficulty, Andrieu and Thoms (2008) proposed to simultaneously adapt the parameter λ and the covariance matrix Σ in order to coerce the acceptance probability to a preset and sensible value, e.g., 0.234. Roberts and Rosenthal (2001) showed that, for large d, the optimal acceptance rate of the random-walk Metropolis algorithm is 0.234 when the components of $\pi(\cdot)$ are approximately uncorrelated but heterogeneously scaled. Assuming that for any fixed covariance matrix Σ the corresponding expected acceptance rate is a non-increasing function of λ, the following recursion can be used for learning λ according to the Robbins-Monro algorithm (Robbins and Monro, 1951):

$$\log(\lambda_{k+1}) = \log(\lambda_k) + \gamma_{k+1}[\alpha(X_k, X^*) - \alpha^*],$$

where X^* denotes the proposed value, and α^* denotes the targeted acceptance rate, e.g., 0.234.

- Initialize X_0, λ_0, μ_0 and Σ_0.
- At iteration $k+1$, given X_k, λ_k, μ_k and Σ_k
  a. Draw X^* from the proposal distribution $N(X_k, \lambda_k \Sigma_k)$, set $X_{k+1} = X^*$ with probability $\alpha(X_k, X^*)$, and set $X_{k+1} = X_k$ with the remaining probability.
  b. Update

$$\begin{aligned}
\log(\lambda_{k+1}) &= \log(\lambda_k) + \gamma_{k+1}[\alpha(X_k, X^*) - \alpha^*], \\
\mu_{k+1} &= \mu_k + \gamma_{k+1}(X_{k+1} - \mu_k), \\
\Sigma_{k+1} &= \Sigma_k + \gamma_{k+1}\left[(X_{k+1} - \mu_k)(X_{k+1} - \mu_k)^T - \Sigma_k\right].
\end{aligned}$$

The adaption of λ can be very useful in the early stage of the simulation, although it is likely not needed in the long run.

Adaptive independent MH algorithm

Holden *et al.* (2009) describes an independent MH algorithm, for which the proposal is adaptive with (part of) past samples, but avoids the requirement of diminishing adaptation. The algorithm can be described as follows.

Let $q_t(z|\boldsymbol{y}_{t-1})$ denote the proposal used at iteration t, where $\boldsymbol{y}_{t-1}$ denotes the set of past samples used in forming the proposal. Since the basic requirement for the independent MH algorithm is that its proposal is independent of the current state x_t, $\boldsymbol{y}_{t-1}$ can not include x_t as an element. Suppose that z has been generated from the proposal $q_t(z|\boldsymbol{y}_{t-1})$. If it is accepted, then set $x_{t+1} = z$ and append $\boldsymbol{y}_{t-1}$ with x_t. Otherwise, set $x_{t+1} = x_t$ and append $\boldsymbol{y}_{t-1}$ with z. The difference between the traditional independent MH algorithm and the adaptive independent MH algorithm is only that the proposal function $q_t(\cdot)$ may depend on a history vector, which can include all samples that $\pi(x)$ has been evaluated except for the current state of the Markov chain.

- Set $\boldsymbol{y}_0 = \emptyset$, and generate an initial sample x_0 in $\mathcal{X}$.
- For $t = 1, \ldots, n$:
  (a) Generate a state z from the proposal $q_t(z|\boldsymbol{y}_{t-1})$, and calculate the acceptance probability

$$\alpha(z, x_t, \boldsymbol{y}_{t-1}) = \min\left\{1, \frac{\pi(z)q_t(x_t|\boldsymbol{y}_{t-1})}{\pi(x_t)q_t(z|\boldsymbol{y}_{t-1})}\right\}.$$

  (b) If it is accepted, set $x_{t+1} = z$ and $\boldsymbol{y}_t = \boldsymbol{y}_{t-1} \cup \{x_t\}$. Otherwise, set $x_{t+1} = x_t$ and $\boldsymbol{y}_t = \boldsymbol{y}_{t-1} \cup \{z\}$.

Holden *et al.* (2009) showed the following theorem for the algorithm, which implies that the chain never leaves the stationary distribution $\pi(x)$ once it is reached.

Theorem 4

The target distribution $\pi(x)$ *is invariant for the adaptive independent MH algorithm; that is,* $p_t(x_t|\boldsymbol{y}_{t-1}) = \pi(x_t)$ *implies* $p_t(x_{t+1}|\boldsymbol{y}_t) = \pi(x_{t+1})$*, where* $p_t(\cdot|\cdot)$ *denotes the distribution of* x_t *conditional on the past samples.*