An Introduction to Probabalistic Programming with Markov Chain Monte Carlo (MCMC)

Daniel Polson

12/15/2022

1 Background

Markov Chain Monte Carlo (MCMC) is a class of algorithms for sampling from a probability distribution. The idea behind MCMC is to construct a Markov chain that has the desired distribution as its equilibrium distribution, and then to sample from the chain after it has reached equilibrium. This allows us to draw samples from a distribution without having to directly compute the distribution. The methods focussed on in this paper include the Metropolis-Hastings Algorithm [Has70], and the Gibbs Sampler [Stu84]. These are long-used variations of MCMC, and continue to be in use to this day. The paper below requires some prior knowledge of measure theory and probability theory to prove the usefulness of these methods, and follows up with some basic applications.

2 Metropolis-Hastings Algorithm

2.1 Theory

Definition 1. *Let $(E, \mathcal{E})$ be a measurable space. A transition interval from $(E, \mathcal{E})$ into $(E, \mathcal{E})$ is considered a Markov Kernel whenever $K(x, E) = 1$ for all x.*

Definition 2. *A Markov Kernel $K(x, A)$ is called reversible with respect to π if for all bounded measurable functions f from $X \times X$ to $\mathbb{R}$,*

$$\int \int_{X \times X} f(x, y) \, \pi(dx) \, K(x, dy) = \int \int_{X \times X} f(y, x) \, \pi(dx) \, K(x, dy).$$

Definition 3. *The probability measure π is called invariant for a function P whenever $\pi = \pi P$, and π is called the stationary distribution of P.*

Before defining the Metropolis-Hastings update, we require some prior notation. Let g be the unnormalized density function with respect to a positive measure μ, and let $q(x, \cdot)$ be a normalized density function with respect to μ. Define:

$$r(x, y) = \frac{g(y) \, q(y, x)}{g(x) \, q(x, y)}$$

And,

$$a(x,y) = \min(1, r(x,y))$$

Theorem 1. *The Metropolis-Hastings update, which is given as P_{mh}below:*

$$P_{mh}(x,A) = \left[1 - \int_A q(x,y)\, a(x,y)\, \mu(dy)\right] \mathbf{1}_A(x) + \int_A q(x,y)\, a(x,y)\, \mu(dy)$$

is a Markov kernel.

Proof.

$$\begin{aligned} P_{mh}(x,E) &= \left[1 - \int_E q(x,y)\, a(x,y)\, \mu(dy)\right] \mathbf{1}_E(x) + \int_E q(x,y)\, a(x,y)\, \mu(dy) \\ &= 1 - \int_E q(x,y)\, a(x,y)\, \mu(dy) + \int_E q(x,y)\, a(x,y)\, \mu(dy) \\ &= 1 \quad \square \end{aligned}$$

Theorem 2. *The Metropolis-Hastings update, which is given as P_{mh} below:*

$$P_{mh}(x,A) = \left[1 - \int_A q(x,y)\, a(x,y)\, \mu(dy)\right] \mathbf{1}_A(x) + \int_A q(x,y)\, a(x,y)\, \mu(dy)$$

is reversible with respect to the distribution with unnormalized density function g.

Proof. Let π be the measure with density function g with respect to μ. Then, we have:

$$\int\int f(x,y)\, \pi(dx)\, P_{mh}(x,dy) = \int\int f(x,y)\, g(x)\, P_{mh}(x,dy)\, \mu(dx)$$

Using the definition of P_{mh}, we can update this integral as follows:

$$\begin{aligned} \int\int f(x,y)\, \pi(dx)\, P_{mh}(x,dy) = &\int\int f(x,y)\, g(x) \left(1 - \int_A q(x,y)\, a(x,y)\, \mu(dy)\right) \mathbf{1}_{dy}(x)\, \mu(dx) \\ &+ \int\int f(x,y)\, g(x)\, q(x,y)\, a(x,y)\, \mu(dx)\, \mu(dy) \end{aligned} \tag{1}$$

However, the presence of $\mathbf{1}_{dy}(x)$ means the second integral in the sum on the right is equal to 0 wherever $x \neq y$. Thus, we can rewrite this term as:

$$\int f(x,x)\, g(x) \left(1 - \int_A q(x,y)\, a(x,y)\, \mu(dy)\right) \mu(dx)$$

But, now the variable we are integrating over becomes arbitrary, so we can rewrite this term once again as:

$$\int f(y,y)\, g(y) \left(1 - \int_A q(y,x)\, a(y,x)\, \mu(dx)\right) \mu(dy)$$

Now, consider the second term in the sum on the right-hand side of (1), first considering the case that $r(x,y) \geq 1$, so that $a(x,y) = 1$ and $a(y,x) = r(y,x)$.

$$\begin{aligned}\int\int f(x,y)\, g(x)\, q(x,y)\, a(x,y)\, \mu(dx)\, \mu(dy) &= \int\int f(x,y)\, g(x)\, q(x,y)\, \mu(dx)\, \mu(dy) \\ &= \int\int f(x,y)\, g(y)\, q(y,x)\, \frac{g(x)\, q(x,y)}{g(y)\, q(y,x)} \mu(dx)\, \mu(dy) \\ &= \int\int f(x,y)\, g(y)\, q(y,x)\, a(y,x)\, \mu(dx)\, \mu(dy)\end{aligned}$$

The last equality implies that the order of x, y are irrelevant in this case, so we have:

$$\int\int f(x,y)\, g(x)\, q(x,y)\, a(x,y)\, \mu(dx)\, \mu(dy) = \int\int f(y,x)\, g(y)\, q(y,x)\, r(y,x)\, \mu(dx)\, \mu(dy)$$

Therefore, for this case, we find that:

$$\begin{aligned}\int\int f(x,y)\, \pi(dx)\, P_{mh}(x,dy) &= \int\int f(y,x)\, g(y)\, q(y,x)\, r(y,x)\, \mu(dx)\, \mu(dy) \\ &\quad + \int f(y,y)\, g(y) \left(1 - \int_A q(y,x)\, a(y,x)\, \mu(dx)\right) \mu(dy)\end{aligned}$$

Reversing the steps we took in (1) on the right-hand side of the above equation,

$$\int\int f(x,y)\, \pi(dx)\, P_{mh}(x,dy) = \int\int f(y,x)\, \pi(dx)\, P_{mh}(x,dy)$$

Thus, the Kernel P_{mh} is reversible with respect to the distribution with unnormalized density function g in this case. The case where $r(x,y) < 1$ is left as an exercise for the reader. □

Now that we have determined the reversibility of P_{mh} with respect to the distribution with unnormalized density function g, we find that:

$$\begin{aligned}\int \pi(dx)\, P_{mh}(x,A) &= \int\int \mathbf{1}_A(y)\, \pi(dx)\, P_{mh}(x,dy) \\ &= \int\int \mathbf{1}_A(x)\, \pi(dx)\, P_{mh}(x,dy) \\ &= \int \mathbf{1}_A(x)\, \pi(dx) \left[\int P_{mh}(x,dy)\right] \\ &= \int \mathbf{1}_A(x)\, \pi(dx) \\ &= \pi(A)\end{aligned}$$

Where the second equality follows from reversibility, and the second last equality follows from Definition 2. This implies that $\pi = \pi P_{mh}$, meaning that by Definition Y, π is invariant for P_{mh}, and π is the stationary distribution of P_{mh}.

2.2 Application

The above work means that in practice, generating samples using the Metropolis Hastings update will be equivalent to drawing samples from any distribution which we know the unnormalized density function for. The typical usage for this algorithm is to generate samples from a posterior distribution, since the usage of Bayes' Theorem often results in an unnormalized conditional density function.

Bayes' Theorem is difficult to prove in the context of measure theory, but Schervish [Sch95] does an excellent job. The basic idea provided is that given a random vector X and parameter vector θ, with conditional density function $f_{X|\Theta}(x|\theta)$, and a prior distribution for Θ with density given by $f_\Theta(\theta)$,

$$f_{\Theta|X}(\theta|x) = \frac{f_{X|\Theta}(x|\theta)_\Theta f(\theta)}{\int f_{X|\Theta}(x|\theta)_\Theta f(\theta)}$$

Often, we are unable to calculate the denominator of the above function, and we are left with an unnormalized density function instead of $f_{\Theta|X}(\theta|x)$. In this case, we need to use the Metropolis Hastings update as follows:

1. Let $g_{\Theta|X}(\theta|x) = \alpha f_{\Theta|X}(\theta|x)$ be an unnormalized version of the desired conditional density.
2. Assume that $\theta^{(k)}$ is the current state of the sequence. Generate θ_0 from a distribution which is easy to generate for any possible states of $\theta^{(k)}$. That is, let $\theta_0 \sim q\left(\cdot|\theta^{(k)}\right)$.
3. Take $U \sim \text{UNIF}(0,1)$.
4. $\theta^{(k+1)}$ is then defined as:

$$\theta^{(k+1)} = \begin{cases} \theta_0 & U \leq g_{\Theta|X}(\theta_0|x)\, q\left(\theta^{(k)}|\theta_0\right) / g_{\Theta|X}\left(\theta^{(k)}|x\right) q\left(\theta_0|\theta^{(k)}\right) \\ \theta^{(k)} & U > g_{\Theta|X}(\theta_0|x)\, q\left(\theta^{(k)}|\theta_0\right) / g_{\Theta|X}\left(\theta^{(k)}|x\right) q\left(\theta_0|\theta^{(k)}\right) \end{cases}$$

Example 1. *Suppose that $Y_1, \ldots, Y_n \sim POI(\lambda)$, and assume that the prior distribution for θ is given by $\theta \sim GAMMA(\alpha, \beta)$. Then, by definition,*

$$f_{Y|\Lambda}(\mathbf{y}|\lambda) = \prod_{i=1}^{n} \frac{e^{-\lambda}\lambda^{y_i}}{y_i!} = \frac{e^{-n\lambda}\lambda^{\sum_i y_i}}{\prod_i y_i!}$$

And,

$$f_\Lambda(\lambda) = \frac{\beta^\alpha \lambda^{\alpha-1} e^{-\lambda\beta}}{\Gamma(\alpha)}.$$

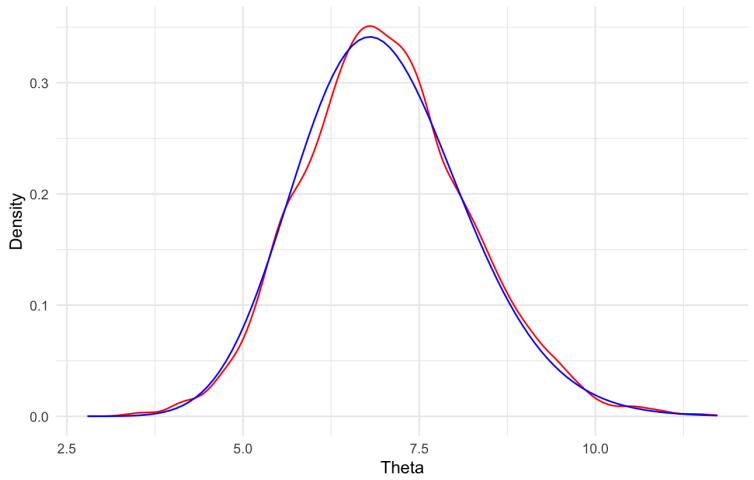

Figure 1: Simulated Posterior Distribution from Metropolis-Hastings (Red Line) versus True Posterior Distribution (Blue Line)

This gives the unnormalized density function:

$$g_{\Lambda|\mathbf{Y}}(\lambda|\mathbf{y}) = exp(-\lambda(n+\beta))\,\lambda^{\sum_i y_i+\alpha-1}$$

Applying the Metropolis Hastings algorithm in the steps described above, and repeating 10,000 times, we obtain the sample distribution as shown (red line) in Figure 1. Upon completing more algebra, we find the normalized density function is that of a GAMMA $(\sum_i y_i + \alpha, n+\beta)$ *distribution. The density function for this distribution (with some parameters given) is shown (blue line) in figure 1.*

We can see from Figure 1 how our sample distribution approaches the desired posterior distribution over large-scale simulation. This means that we are able to sample from many unknown posterior distributions, despite not being able to solve their normalizing constant.

3 Gibbs Sampler

The Gibbs Sampler is a particular case of the Metropolis-Hastings Algorithm. It applies the algorithm in partial, fully conditional steps. Often, we have access to conditional distributions, and limited access to joint distributions for a vector of parameters. For our purposes, we will consider the case where $\mathbf{\Theta} = (\Theta_1, \Theta_2)'$, but it can be extended for more variables.

The steps for the Gibbs Sampler are as follows:

1. Set θ_2 to any estimate
2. Select $\theta_1^* \sim f(\theta_1^*|\theta_2, x)$
3. Select $\theta_2^* \sim f(\theta_2^*|\theta_1^*, x)$
4. (θ_1^*, θ_2^*) is a draw from the joint distribution. Repeat from step 2.

For this process, the transition kernel can be considered:

$$K(\theta_1, \theta_2|\theta_1^*, \theta_2^*) = f_{\Theta_1|\Theta_2}(\theta_1^*|\theta_2) f_{\Theta_2|\Theta_1}(\theta_2^*|\theta_1^*)$$

Where $f_{\Theta_1|\Theta_2}$ is the conditional density of $\Theta_1|\Theta_2$, and $f_{\Theta_2|\Theta_1}$ is the conditional density of $\Theta_2|\Theta_1$.

Theorem 3. *The joint distribution of Θ_1 and Θ_2, $f_{\Theta_1,\Theta_2}(\theta_1, \theta_2)$ is the stationary distribution of K.*

Proof:

$$\begin{aligned}
(K f_{\Theta_1,\Theta_2})(\theta_1^*, \theta_2^*) &= \int\int f_{\Theta_1,\Theta_2}(\theta_1, \theta_2) K(\theta_1, \theta_2|\theta_1^*, \theta_2^*) d\theta_1 d\theta_2 \\
&= \int\int f_{\Theta_1,\Theta_2}(\theta_1, \theta_2) f_{\Theta_1|\Theta_2}(\theta_1^*|\theta_2) f_{\Theta_2|\Theta_1}(\theta_2^*|\theta_1^*) d\theta_1 d\theta_2 \\
&= \int\int f_{\Theta_1,\Theta_2}(\theta_1, \theta_2) \frac{f_{\Theta_1,\Theta_2}(\theta_1^*, \theta_2)}{f_{\Theta_2}(\theta_2)} \frac{f_{\Theta_1,\Theta_2}(\theta_1^*, \theta_2^*)}{f_{\Theta_1}(\theta_1^*)} d\theta_1 d\theta_2 \\
&= \int f_{\Theta_2}(\theta_2) \frac{f_{\Theta_1,\Theta_2}(\theta_1^*, \theta_2)}{f_{\Theta_2}(\theta_2)} \frac{f_{\Theta_1,\Theta_2}(\theta_1^*, \theta_2^*)}{f_{\Theta_1}(\theta_1^*)} d\theta_2 \\
&= \int f_{\Theta_1,\Theta_2}(\theta_1^*, \theta_2) \frac{f_{\Theta_1,\Theta_2}(\theta_1^*, \theta_2^*)}{f_{\Theta_1}(\theta_1^*)} d\theta_2 \\
&= f_{\Theta_1}(\theta_1^*) \frac{f_{\Theta_1,\Theta_2}(\theta_1^*, \theta_2^*)}{f_{\Theta_1}(\theta_1^*)} \\
&= f_{\Theta_1,\Theta_2}(\theta_1^*, \theta_2^*) \qquad \square
\end{aligned}$$

Therefore, we can use the Gibbs Sampler method to sample from any joint distribution given conditional distributions.

Example 2. *Suppose that we want to draw samples from the multivariate normal distribution below:*

$$(\theta_1, \theta_2)' \sim N\left(\begin{pmatrix} 2 \\ 3 \end{pmatrix}, \begin{pmatrix} 7 & 2 \\ 2 & 10 \end{pmatrix}\right)$$

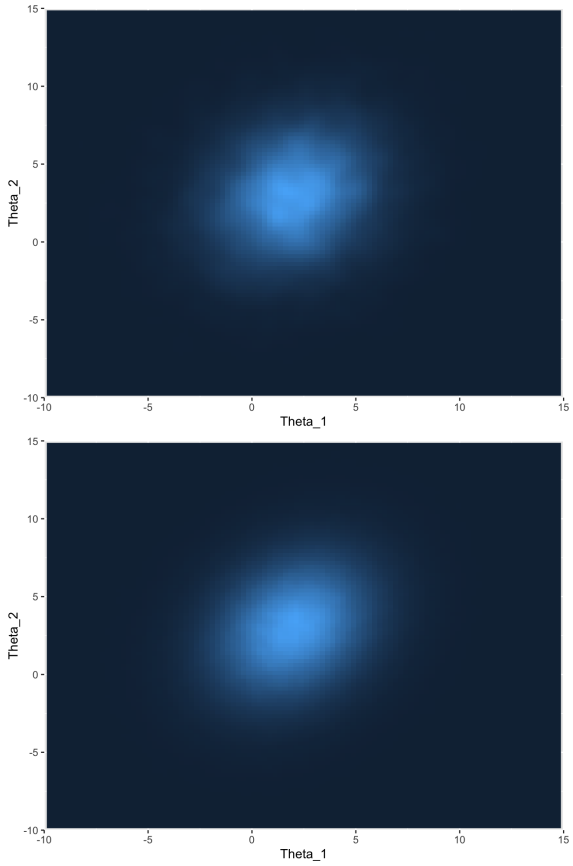

Figure 2: Heatmap of the Simulated Mulitvariate Normal Density from Example 3 (Top) versus True Multivariate Normal Density (Bottom).

The conditionals from this density function are given by:

$$\theta_1|\theta_2 \sim N\left(2 + \frac{1}{5}\left(\theta_2 - 3\right), \frac{33}{5}\right), \text{ and}$$

$$\theta_2|\theta_1 \sim N\left(3 + \frac{1}{5}\left(\theta_1 - 2\right), \frac{66}{7}\right)$$

Using the steps we developed for performing a Gibbs Sampler, we started with $\theta_2 = 3$*, and repeated steps 2-4 11,000 times, retaining the last 10,000 iterations. The resultant simulated density is shown in Figure 2.*

4 Research Directions

At the current time, the field of machine learning is using Markov Chain Monte Carlo to assist the field of AI. It allows researchers to draw samples from density functions which are impossible to solve for. Current research involves finding efficient proposal kernels to improve computational speed of various algorithms. Ziheng Yang and Carlos E. Rodríguez [Zih13] propose a unique class of Bactrian kernels which claim to be 50% more efficient than the usual Gaussian proposal kernel. The idea behind the Bactrian kernels is to create a mixture distribution from multiple component distributions to increase efficiency. A selection of these distributions are shown in Figure 3.

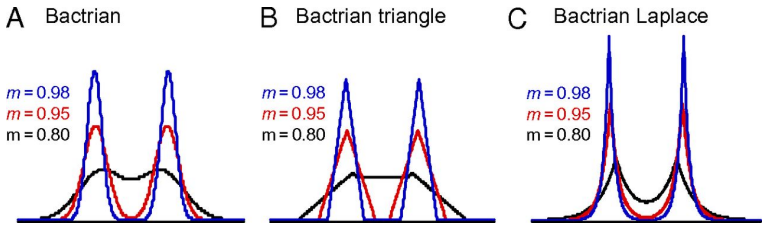

Figure 3: Some Bactrian Proposal Distributions given by Yang and Rodríguez (2013)

Another more recent development in Markov Chain Monte Carlo comes from Matthew Hoffman, Alexey Radul, and Pavel Sountsov [Mat21] in their use of no-U-turn sampler (NUTS) which eliminates the simulation length requirement parameter which can limit the capabilities of other simulations discussed when working with high-dimensional distributions. Instead of the random walk idea used by the Metropolis-Hastings algorithm, NUTS uses a recursive algorithm which builds a set of likely candidate points, and aborts when it starts doubling back on itself. With machine learning heading in a direction of such high-dimensional models, simulations which cut down their simulation time are extremely important.

With Markov Chain Monte Carlo being so prevalent in modern statistics, it is important to understand both the theory and applications before going into the field.

5 Questions (+ Answers)

1. Show that the Kernel P_{mh} from Theorem 2 is reversible with respect to the distribution with unnormalized density function g in the case where $r(x,y) < 1$.

   Solution. By our assumption, it must be that $a(y,x) = 1$ and $a(x,y) = r(x,y)$. Now, we can take:

$$\begin{aligned}
\int\int f(x,y)\, g(x)\, q(x,y)\, a(x,y)\, \mu(dx)\, \mu(dy) &= \int\int f(x,y)\, g(x)\, q(x,y)\, r(x,y)\, \mu(dx)\, \mu(dy) \\
&= \int\int f(x,y)\, g(x)\, q(x,y)\, \frac{g(y)\, q(y,x)}{g(x)\, q(x,y)} \mu(dx)\, \mu(dy) \\
&= \int\int f(x,y)\, g(y)\, q(y,x)\, \mu(dx)\, \mu(dy) \\
&= \int\int f(x,y)\, g(y)\, q(y,x)\, a(y,x)\, \mu(dx)\, \mu(dy)
\end{aligned}$$

   The last equality implies that the order of x, y are irrelevant in this case, so we can continue as we did in Theorem 2, and obtain the same proof.

2. Reproduce the chart given in Example 1 for $\alpha = \beta = 2$, $\mathbf{y} = (10, 10, 13)'$, for $11,000$ iterations where the first $1,000$ iterations are removed.

   Solution. See attached code.

References

[BS86] I. Beichl and F. Sullivan. "The Metropolis algorithm". In: *Computing in Science and Engineering* 2:1 (1986), pp. 65–69.

[CP 04] G. C. C.P. Robert. *Monte Carlo Statistical Methods*. Springer New York, NY, 2004.

[Fre97] I. Freely. "A small paper". In: *The journal of small papers* -1 (1997). to appear.

[Gil21] A. S. Gilles Barthe Joost-Pieter Katoen. *Foundations of Probabilistic Programming*. Cambridge University Press, 2021.

[Has70] W. K. Hastings. "Monte Carlo sampling methods using Markov chains and their applications". In: *Biometrika* 57:1 (1970), pp. 97–109.

[Ken] T. Kennedy. "Chapter 8, Markov Chain Monte Carlo". Insightful notes with no citations.

[Lee92] M. E. Lee J. Bain. *Introduction to Probability and Mathematical Statistics*. Brooks/Cole, 1992.

[Mat21] P. S. Matthew Hoffman Alexey Radul. "An Adaptive-MCMC Scheme for Setting Trajectory Lengths in Hamiltonian Monte Carlo". In: *Proceedings of Machine Learning Research* 130 (2021).

[Nic53] e. a. Nicholas Metropolis. "Equation of State Calculations by Fast Computing Machines". In: *The Journal of Chemical Physics* 21 (1953).

[Sch95] M. J. Schervish. *Theory of Statistics*. Springer New York, NY, 1995.

[Stu84] D. G. Stuart Geman. "Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images". In: *TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE* 6 (1984).

[Zih13] C. E. R. Ziheng Yang. "Searching for efficient Markov chain Monte Carlo proposal kernels". In: *PNAS* 110:48 (2013).