# Minicourse on:

# Markov Chain Monte Carlo: Simulation Techniques in Statistics

Eric Slud, Statistics Program

**Lecture 1:** Metropolis-Hastings Algorithm, plus background in Simulation and Markov Chains.

**Lecture 2:** The 'Gibbs Sampler', via motivation from Metropolis-Hastings.

In both lectures, there will be computational illustrations: in Lecture 1 and beginning of Lecture 2, an extended example involving simulation of uniform random points in convex regions defined by linear constraints. In Lecture 2, further examples of statistical interest.

# 1 Introduction to the Idea of Simulation

(A) Building-block for Simulation is existence of algorithmically generated *pseudo-random* numbers $U_1$, $U_2$, ..., behaving as though independent identically distributed (*iid*) Uniform$(0, 1)$.

(B) Objective is usually to evaluate an integral

$$\int g(x)f(x)d\mu(x) \;=\; E(g(X))$$

where $X$ is a random variable or vector with values in possibly high-dimensional $\mathbf{R}^d$, where $X$ has density $f$ (known at least implicitly) with respect to $\sigma$-finite measure $\mu$ (usually Lebesgue or a counting measure).

(C) If we were able to find an explicit, easily codable function $h(U_1, U_2, \ldots)$ of Uniform$(0, 1)$ variates with values in $\mathbf{R}^d$ and probability distribution the same as $X$, then we would evaluate the desired integral as

$$N^{-1} \sum_{j=1}^{N} g(h(U_{j1}, U_{j2}, \ldots))$$

for large $N$ by the Law of Large Numbers, where $U_{jk}$ is a doubly indexed array of *iid* Uniform variables.

*It is allowed for $h$ to depend on unboundedly many $U$ variables, as long as the number of such variables required is a r.v. with finite expectation.*

(D) We are interested in examples of $X$ with probability distributions, exhibiting either hierarchically structured dependence for statistical applications or dependence defined by geometric constraints. We restrict attention to the latter in this first lecture: the objective is to simulate uniformly distributed random points $X$ in a region $D \subset \mathbf{R}^d$.

## Accept-Reject Algorithms

Suppose that it is easy to simulate $X$ uniformly distributed in a region $B \supset D$, with $k$ fixed and $h(U_1, \ldots, U_k)$ uniformly distributed in $B$. Then the rule

$$X \equiv h(U_{n1}, \ldots, U_{nk}) \quad \text{for}$$

$$n = \inf\{i \geq 1 : \quad h(U_{i1}, \ldots, U_{ik}) \in D\}$$

based on doubly indexed *iid* array $\{U_{ij}\}$, is uniformly distributed in $D$. Here $n$ is random, with distribution Geometric($p$), $p = vol(D)/vol(B)$).

**Example.** $B = \{(x_1, \ldots, x_d) : \ x_i \geq 0, \ \Sigma_{i=1}^d \ x_i \leq 1\}$, and for fixed $\mathbf{a} \in (\mathbf{R}^+)^d$, $b > 0$,

$$D \ = \ \{\mathbf{x} \in B : \quad \mathbf{x} \cdot \mathbf{a} \leq b\}$$

This is a simplified example: more generally, the region $D$ is defined by linear constraints within $\mathbf{R}^d$, $d$ large.

## 2 Metropolis-Hastings Algorithm

Suppose that you want to generate a random vector $X \in \mathbf{R}^d$ with density $\pi$ (for definiteness, with respect to Lebesgue measure.) The Metropolis-Hastings algorithm generates as a function of pseudo-random variates $U_1, U_2, \ldots,,$ a discrete-time random sequence $X_0, X_1, \ldots, X_t, \ldots \in \mathbf{R}^d$ which has a unique stationary, or long-term equilibrium distribution such that the probability law of $X_t$ converges for large $t$ to the probability measure with density $\pi$. For large $t$, $X_t$ is a random vector with approximate density $\pi$, and even better, for large M,N,

$$E(g(X)) \quad \approx \quad N^{-1} \sum_{t=M+1}^{M+N} g(X_t)$$

The algorithm has three ingredients:

(1) A *Proposal Markov Chain* expressed by a transition kernel $q(x, y)$ (regarded as conditional density of landing in one transition-step at $y$ starting from $x$, from which it must be easy to simulate random vectors of density $q(x, \cdot)$ for each choice of $x$.

(2) The *Accept-Reject Rule* which says that if $X_k$ has been generated previously, and $Y_k \sim q(X_k, \cdot)$ is simulated using new (and therefore independent)

pseudo-random variates $U_i$ , then

$X_{k+1} = Y_k$ with prob. $\min(1, \frac{\pi(Y_k)q(Y_k, X_k)}{\pi(X_k)q(X_k, Y_k)})$

and $= X_k$ with the remaining probability.

(3) A decision rule for stopping: typically $M$ (initial point for ergodic averaging) is taken much larger than $N$ (the number of iterates in the average).

**Example** (continued): Suppose $q$ corresponds to the 'independence chain', $q(x, y) \equiv (1/vol(B))\, I_{[y \in B]}$ with $B \subset \mathbf{R}^d$ the unit simplex. The simulation of $Y_t$ values is Uniform in B, which is easy to do: starting with $U_1, \ldots, U_d$ *iid* Uniform$(0, 1)$, define the coordinates of each $Y \in \mathbf{R}^d$ by taking successive differences among 0 and the sorted-increasing values $U_{(1)} < \cdots < U_{(d)}$. This is a small exercise in Jacobian change of variable: the joint density of $U_{(1)}, \ldots, U_{(d)}$ is $d!$ on the set of sorted-increasing $d$-tuples in $(0, 1)^d$, and the density of the resulting $Y$ vectors on $B$ is also $d!$

Leting $X_0$ be an arbitrary element of $B$, and recall $D \subset B$: the Metropolis-Hastings algorithm successively defines, for $t \geq 0$:

$$X_{t+1} = \begin{cases} X_t & \text{if} \quad X_t \in D,\, Y_t \notin D \\ Y_t & \text{if} \quad Y_t \in D \end{cases}$$

Thus, in the example, since $q$ has always the same constant value whenever it is nonzero, and the same is true for the desired density $\pi(\mathbf{x}) = I_{[\mathbf{x}\in D]}/vol(D)$, Metropolis-Hastings is precisely the Accept-Reject algorithm !

# Computations in Example

Let us fix a vector $\mathbf{a}$ at random, in the case d=10.

```
0.513 0.944 0.960 0.116 0.032 0.944
0.691 0.489 0.020 0.710
```

We try two examples of choices for $b$, recalling that the set $D$ of interest is

$$D = \{\mathbf{x} \in B : \quad \mathbf{x} \cdot \mathbf{a} \leq b\}$$

First, with $b = \mathbf{a} \cdot \mathbf{1}/10 = .542$: note that a random element $\mathbf{X}$ of $B$ has expectation $\frac{1}{11}\mathbf{1}$, so it is not too surprising that the fraction $vol(D)/vol(B) > 0.5$. In fact, this ratio is around 0.66, since of 10000 randomly generated uniformly distributed elements $Y_t \in B$, 6641 were found to satisfy the criterion $\mathbf{a} \cdot Y_t \leq b$. This high proportion means that Accept-Reject would likely be the best way to generate random points in $D$. (Generating and testing the 10000 points took 5 seconds on my home PC.)

Next, fix $b = \mathbf{a} \cdot \mathbf{1}/20 = .271$. Now, of 10000 randomly generated points in $B$, only 207 points fell in $D = \{x \in B : \mathbf{a} \cdot \mathbf{x} \leq b\}$. This fraction $vol(D)/vol(B) \approx .02$ is small enough that perhaps Accept-Reject can be improved. The situation only becomes worse if $D$ is defined by more linear constraints in higher-dimensional simplices !

It is obvious that the coordinate values in $D$ tend to be smaller than those in $B$. One way to visualize this is to make pictures (histograms, or smoothed versions called "density estimates") of random variable values like $x_1 + \cdots + x_6$, as plotted in Fig. 1.

Figure 1: Pair of density estimates (smoothed histograms) for the partial sum $X_1 + \cdots + X_6$ of the first 6 coordinate entries of each random vector from a block of 1000 random 10-vectors $\mathbf{X}$ generated (a) in the unit simplex (sold line), and (b) in the unit simplex further restricted by a single linear constraint $\mathbf{X} \cdot \mathbf{a} \leq \mathbf{1} \cdot \mathbf{a}/2$ (dashed line).

**Example**, continued. Now consider a Metropolis-Hastings algorithm with a non-independent Proposal Chain defined by kernel $q(x, y)$. The goal is to devise an easy-to-implement transition mechanism which with positive probability (in a bounded number of steps) carries any point in B to the neighborhood of any point in D.

Here is a construction, depending on two positive parameters $\alpha, \beta$. Starting from $\mathbf{x}$, define

$$\gamma(\mathbf{x}) \;=\; (1+\beta)\,\min(\,\mathbf{x}\cdot\mathbf{1},\;\mathbf{x}\cdot\mathbf{a}/b)$$

Then define $\mathbf{y}$ through its coordinates:

$$y_i \;=\; \exp(Z_i)\,(x_i/\gamma(\mathbf{x}))$$

where the r.v.'s $Z_i,\ 1 \le i \le d,$ are $iid$ $\mathcal{N}(0,\alpha)$.

The Metropolis-Hastings algorithm starts with arbitrary $X_0 \in B$. At $k$'th stage, with $X_k$ given, calculate $\gamma(\mathbf{x})$ as above, and define $Y_k$ as above, with a new and independent batch of $Z_i$ r.v.'s. Then $X_{k+1}$ is $Y_k$ if $X_k \notin D$, $Y_k \in D$, and is $X_k$ if $Y_k \notin D$. But if both $X_k,\, Y_k \in D,$ then $X_{k+1} = Y_k$ with probability

$$\min(1,\ \exp\,(\frac{1}{\alpha}\,(log(\gamma(X_k)) - log(\gamma(Y_k)))\cdot$$

$$(\sum_{i=1}^{d}\ log(Y_{k,i}/X_{k,i}) - \ \frac{d}{2}\,(log(\gamma(X_k)) + log(\gamma(Y_k))))$$

# 3 Discrete-time Markov Chains Discrete or Continuous States

Under tractable conditions of *irreducibility* and *ergodicity*, the Markov chain defined by the Metropolis-Hastings algorithm has a unique stationary distribution to which it converges, perhaps rapidly, as time $t$ gets large.

Each of these notions is well-known in Markov chains with discrete time-parameter. We recall definitions from the discrete case and give parallel definitions and results for continuous-state cases. Basic references: Karlin & Taylor (1975) for discrete state and Robert & Casella (1999) for continuous.

Generally, a sequence of r.v.'s $X_t$, $t = 0, 1, 2, \ldots$ taking values in the same state-space $\mathbf{R}^d$ is a **Markov Chain** if for all Borel sets $A \subset \mathbf{R}^d$, $t \in \mathbf{Z}^+$,

$$P(X_t \in A \,|\, X_s,\ s < t) \ = \ P(X_t \in A \,|\, X_{t-1})$$

and such one-step *transition probabilities* are specified by a *Markov kernel* $q(x, y)$, as $\int_A q(x, y) d\mu(y)$. If all $X_t$ takes values in the same countable set $S$, then with $\mu$ counting measure on $S$, the kernel $q$ is called a (possibly $\infty \times \infty$) *stochastic matrix*.

In discrete states, say the chain defined by $q$ is **irreducible** if $\forall x,\, y \in S,\ \exists\ \{z_i\}_{i=1}^m \subset S\ :$

$$x = z_1,\, z_m = y: \quad q(z_i, z_{i+1}) > 0,\ 1 \le i < m$$

With continuous states and a measure with density $f$, say the chain is **f-irreducible** if for all $A$ with positive $f$ measure, and all $x,\ \exists\, m: \quad P(X_m \in A \,|\, X_0 = x) > 0.$

There is a Theorem (Robert & Casella Thm 6.2.5) saying that if the Metropolis-Hastings chain is $\pi$-irreducible and has nonzero probabilities of 'rejecting' (ie of $X_{t+1} = X_t$) then for every initial distribution for $X_0$, the distribution of $X_t$ converges in Total Variation to the distribution with density $\pi$. In some problems one can say more (*geometric ergodicity*): that the convergence is exponentially fast in $t$.

## 4 Reversibility & Convergence

A key property of the Metropolis-Hastings chain is **reversibility**. This chain has transition kernel $M(x, y)$ (weighted combination of $q(x, y)$ and point-mass $\delta_x(y)$ ) which satisfies the *detailed balance* relation

$$M(y, x)\pi(y) \;=\; M(x, y)\pi(x)$$

(says the chain can be run backwards in time by the same probabilistic transition mechanism. Integrating this relation over $y$ (using $M(x, \mathbf{R}^d) = 1$) yields

$$\int \pi(y)M(y, x)dy \;=\; \pi(x)$$

and says that $\pi$ is **invariant** or **stationary**. Conditions on mutual accessibility, like *pi*-irreducibility, lead to uniqueness for the invariant distribution.

**Remarks about Metropolis-Hastings algorithms**.

(0) The Metropolis-Hastings steps can be implemented even if $pi$ has an unknown normalizing constant (because the constant cancels out of the accept-probability ratios).

(1) If $q$ were symmetric (the original suggestion), transition steps move to higher $\pi$ density regions automtically, to lower density regions only with some probability.

(2) Billera & Diaconis (2001) characterize this algorithm (in the case of finite-support $\mathbf{X}$) within a class of Markov chains with stationary density $\pi$ as the closest to the Markov chain with kernel $q$.

(3) The choice of $q$ makes a huge difference to the successful convergence of the algorithm.

(4) The choice of stopping-criterion is still not well understood: Jones and Hobert (2001) following Meyn & Tweedie (1993) and others show how to find computable theoretical bounds for rates of *geometric ergodicity*, but these may not accurately reflect algorithms' success in practice.

**Further comments on the Metropolis chain behavior in the example.**

The displayed pictures show that the proposal chain transitions are too active, preventing the blocks of 1000 successive generated values $X_t$ from settling down rapdidly as we would want them to.

Figure 2: Density estimates (smoothed histograms) for the partial sum $X_1 + \cdots + X_6$ of the first 6 coordinate entries of each random vector from ech of six successive blocks of 1000 random 10-vectors $\mathbf{X}$ generated by the Metropolis-Hastings algorithm described on previous slides (with proposal chain multiplying individual coordinates of the 10-vectors by independent lognormal variables.) Parameters $\alpha, \beta$ in the proposal kernel $q$ were $\alpha = .04$, $\beta = .05$. The chain was "pre-iterated" or 'burned-in' for 1500 transition steps before creating and plotting density estimates from successive blocks of 1000 iterates.

Figure 3: Density estimates (smoothed histograms) for the partial sum $X_1 + \cdots + X_6$ of the first 6 coordinate entries of each random vector from each of five successive blocks of 1000 random 10-vectors $\mathbf{X}$ generated by the Metropolis-Hastings algorithm described on previous slides. The proposal chain, including parameters, were exactly the same as in Figure 2, and the blocks of iterates from which density estimates are plotted in this Figure are a continuation, after a gap of 1000 'wasted' iterates, of the Metropolis-Hastings realization plotted in Figure 2.

Figure 4: Density estimates (smoothed histograms) for the partial sum $X_1 + \cdots + X_6$ of the first 6 coordinate entries of each random vector from each of siz successive blocks of 1000 random 10-vectors $\mathbf{X}$ generated by the Metropolis-Hastings algorithm described on previous slides (with proposal chain multiplying individual coordinates of the 10-vectors by independent lognormal variables.) Parameters $\alpha, \beta$ in the proposal kernel $q$ were $\alpha = .1$, $\beta = e^{.05} - 1$. The chain was "pre-iterated" or 'burned-in' for 6000 transition steps before creating and plotting density estimates from successive blocks of 1000 iterates.

Figure 5: Density estimates (smoothed histograms) for the partial sum $X_1 + \cdots + X_6$ of the first 6 coordinate entries of each random vector from each of siz successive blocks of 1000 random 10-vectors $\mathbf{X}$ generated by the Metropolis-Hastings algorithm described on previous slides (with proposal chain multiplying individual coordinates of the 10-vectors by independent lognormal variables.) Parameters $\alpha, \beta$ in the proposal kernel $q$ were $\alpha = .02$, $\beta = e^{.01} - 1$. The chain was "pre-iterated" or 'burned-in' for 10000 transition steps before creating and plotting density estimates from successive blocks of 1000 iterates.

# Another Proposal Chain

An approach which turns out to work better in the extended Example is to implement Metropolis-Hastings by changing only one coordinate at a time. This violates the stationary-Markov transition kernel unless we combine $d$ successive steps for each of the coordinates in turn, but the proposal transitions $q_i$ applying to the $i$'th coordinate are easier to describe individually.

For $q_i$, starting from $\mathbf{X} \in D$, we simply replace $X_i$ by the conditional law for the i'th coordinate given the other coordinates, for a random point of $D$. In the example, we have

$$0 \le x_i \le \min(1 - \sum_{j:j\neq i} x_j,\ (b - \sum_{j:j\neq i} a_j x_j)/a_j)$$

Replacing $x_i$ by a uniformly distributed value between 0 and the displayed upper bound, gives a value $y_i$ such that

$$(x_1, \ldots, x_{i-1}, y_i, x_{i+1}, \ldots, x_d) \in D$$

a point which would be accepted by a M-H step with probability 1. An iteration based on this step, with $i$ successively ranging (in random order) over $\{1, \ldots, d\}$, is our first example of a Gibbs Sampler MCMC scheme, and we will see how it works at the beginning of the next Lecture.

# References

Billera, L. and Diaconis, P. (2001) Geometric interpretation of the Metropolis-Hastings Algorithm, *Statist. Sci.* **16** 335-339.

Jones, G. and Hobert, J. (2001) Honest exploration of intractable probability distributions via Markov Chain Monte Carlo *Statist. Sci.* **16** 312-334.

Karlin, S. and Taylor, H. (1975) **A First Course in Stochastic Processes**, 2nd ed. Academic Press.

Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A., and Teller, E. (1953) *Jour. Chem. Phys.*

Meyn, S. and Tweedie, R. (1993) **Markov Chains and Stochastic Stability**. Springer-Verlag.

Robert, C. and Casella, G. (1999) **Monte Carlo Statistical Methods**. Springer-Verlag.