Stat 543
A VERY Brief Introduction to Markov Chain Monte Carlo

The object here is to describe algorithms for generating a sequence of parameter vectors $\boldsymbol{\theta}_1^*, \boldsymbol{\theta}_2^*, \boldsymbol{\theta}_3^*, \ldots$ whose long run empirical (relative frequency) distribution can approximate a (posterior) distribution specified by a density proportional to the product of a likelihood and a prior $L(\boldsymbol{\theta})\, g(\boldsymbol{\theta})$ (for $\boldsymbol{\theta}$ a k-dimensional vector). (Of course, the product $L(\boldsymbol{\theta})\, g(\boldsymbol{\theta})$ could be replaced by a general $h(\boldsymbol{\theta})$ specifying a distribution up to a normalizer in contexts other than Bayes analysis.)

This is intended as an "introductory operational" presentation. It does not address the theoretical questions of **why** or **when** these algorithms work or what versions of them might be best (in terms of quickly producing a set of iterates representative of the target distribution). Neither does it address practical/applied issues of detecting potential problems in MCMC simulations, determining when any "transient"/"start-up" effects have been "washed out" and it makes sense to begin accumulating iterates for subsequent analysis, or deciding how many iterations to use. These all are matters for other courses.

Successive Substitution Sampling (Gibbs Sampling)

Abbreviate (for iterates $\boldsymbol{\theta}_i^*$) for each $j = 1, 2, \ldots, k$

$$\boldsymbol{\theta}_{i,<j}^* = \left(\theta_{i,1}^*, \theta_{i,2}^*, \ldots, \theta_{i,j-1}^*\right) \text{ and } \boldsymbol{\theta}_{i,>j}^* = \left(\theta_{i,j+1}^*, \theta_{i,j+2}^*, \ldots, \theta_{i,k}^*\right)$$

Begin with $\boldsymbol{\theta}_0^*$ (possibly generated from an approximation to $L(\boldsymbol{\theta})\, g(\boldsymbol{\theta})$ or from the prior $g(\boldsymbol{\theta})$). With $\boldsymbol{\theta}_i^*$ in hand, generate $\boldsymbol{\theta}_{i+1}^*$ as follows. For each $j = 1, 2, \ldots, k$, generate $\theta_{i+1,j}^*$ from

$$L\left(\boldsymbol{\theta}_{i+1,<j}^*, \cdot, \boldsymbol{\theta}_{i,>j}^*\right) g\left(\boldsymbol{\theta}_{i+1,<j}^*, \cdot, \boldsymbol{\theta}_{i,>j}^*\right)$$

(that is, hold all entries of $\boldsymbol{\theta}$ except the jth at their current iterate values, and generate a replacement for the current value, $\theta_{i,j}^*$, from the resulting density).

Sometimes one can see by inspection how to do this directly. Sometimes once can use the rejection algorithm to do this. Other times clever new tricks are needed. But when this all can be done, under appropriate conditions this can produce a sequence with the ergodicity property.

Example 1 *Suppose parameters μ_1, μ_2 both have values in $\Re$ and $c \in (0,1)$. Given these parameters suppose further that (X, Y) has joint distribution specified by*

$$X \sim U(0,1)$$

and conditioned on the value of X

$$\begin{array}{ll} Y \sim N(\mu_1, 1) & \text{if } X < c \\ Y \sim N(\mu_2, 1) & \text{if } X \geq c \end{array}$$

The conditional mean of Y *given* $X = x$ *is the step function*

$$E[Y|X = x] = \begin{cases} \mu_1 & \text{if } x < c \\ \mu_2 & \text{if } x \geq c \end{cases}$$

and based on n *iid observations* (X_l, Y_l)*, the likelihood is*

$$L(\mu_1, \mu_2, c) = \left(\frac{1}{\sqrt{2\pi}}\right)^n \exp\left(-\frac{1}{2}\left(\sum_{x_l<c}(y_l - \mu_1)^2 + \sum_{x_l \geq c}(y_l - \mu_2)^2\right)\right) I[0 < c < 1]$$

Consider then the use of a prior distribution of independence specified by

$$\begin{aligned} c &\sim U(0,1) \\ \mu_1 &\sim N\left(0, \gamma^2\right) \\ \mu_2 &\sim N\left(0, \gamma^2\right) \end{aligned}$$

so that

$$g(\mu_1, \mu_2, c) = \left(\frac{1}{2\pi\gamma^2}\right) \exp\left(-\frac{1}{2\gamma^2}\left(\mu_1^2 + \mu_2^2\right)\right) I[0 < c < 1]$$

One "Gibbs" (successive substitution sampling) algorithm is then as follows. With iterate $(\mu_i^*, \mu_i^*, c_i^*)$ *in hand, one may*

1. *update* $\mu_{1,i}^*$ *to* $\mu_{1,i+1}^*$ *sampling from the*

$$N\left(\bar{y}_1\left(\frac{n_1\gamma^2}{n_1\gamma^2+1}\right), \frac{\gamma^2}{n_1\gamma^2+1}\right)$$

distribution, where $\bar{y}_1$ *is the sample mean* y_l *for* x_l*'s that are less than* c_i^* *and* $n_1 = \#[x_l < c_i^*]$*,*

2. *update* $\mu_{2,i}^*$ *to* $\mu_{2,i+1}^*$ *sampling from the*

$$N\left(\bar{y}_2\left(\frac{n_2\gamma^2}{n_2\gamma^2+1}\right), \frac{\gamma^2}{n_2\gamma^2+1}\right)$$

distribution, where $\bar{y}_2$ *is the sample mean* y_l *for* x_l*'s that are at least as large as* c_i^* *and* $n_2 = \#[x_l \geq c_i^*]$*, and*

3. *update* c_i^* *to* c_{i+1}^* *sampling from a density on* $(0,1)$ *that is constant between ordered* x *values* $x_{(l)}$*, with value proportional to*

$$h_0 = \exp\left(-\frac{1}{2}\left(\sum_l \left(y_l - \mu_{2,i+1}^*\right)^2\right)\right)$$

on $\left(0, x_{(1)}\right)$*, with value proportional to*

$$h_m = \exp\left(-\frac{1}{2}\left(\sum_{x_l \leq x_{(m)}}\left(y_l - \mu_{1,i+1}^*\right)^2 + \sum_{x_l \geq x_{(m+1)}}\left(y_l - \mu_{2,i+1}^*\right)^2\right)\right)$$

on $\left(x_{(m)}, x_{(m+1)}\right)$ *for* $1 \leq m \leq n-1$, *and with value proportional to*

$$h_n = \exp\left(-\frac{1}{2}\left(\sum_l \left(y_l - \mu^*_{1,i+1}\right)^2\right)\right)$$

on $\left(x_{(n)}, 1\right)$. *That is, with*

$$H = h_0 \cdot x_{(1)} + \sum_{m=1}^{n-1} h_m \cdot \left(x_{(m+1)} - x_{(m)}\right) + h_n \cdot \left(1 - x_{(n)}\right)$$

update c_i^* *to* c_{i+1}^* *sampling from a density that is* h_0/H *on* $\left(0, x_{(1)}\right)$, h_m/H *on* $\left(x_{(m)}, x_{(m+1)}\right)$ *for* $1 \leq m \leq n-1$, *and* h_n/H *on* $\left(x_{(n)}, 1\right)$.

Example 2 *For* $f(y|\mu)$ *the* $N(\mu, 1)$ *density consider Bayes analysis for a sample* $Y_1, Y_2, \ldots, Y_n$ *iid from the mixture distribution specified by density*

$$(1-\alpha) f(y|\mu_0) + \alpha f(y|\mu_1)$$

for $\alpha \in (0,1)$, *for real* $\mu_0 < \mu_1$. *The likelihood here is nasty, but (in a way related to how one approaches maximization of the likelihood through the EM algorithm) it's possible to apply a kind of "latent" (unobserved) or "auxiliary" variable argument to do MCMC from a posterior.*

That is, consider a model for $\boldsymbol{X} = (W, Y)$ *where*

$$\begin{aligned} W &\sim Ber(\alpha) \\ Y|W &\sim N(\mu_W, 1) \end{aligned}$$

In this model for $\boldsymbol{X}$, *the marginal distribution for* Y *is exactly the mixture distribution of interest. An iid sample* $\boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_n$ *produces likelihood*

$$L_{\boldsymbol{x}}(\mu_0, \mu_1, \alpha) = \prod_{l=1}^{n} \left((1-a) f y_l|\mu_0\right)^{(1-w_l)} \left(\alpha f(y_l|\mu_1)\right)^{w_l}$$

Then with, e.g., independent priors for (μ_0, μ_1) *and* α *with*

$$\alpha \sim U(0,1)$$

and (μ_0, μ_1) *bivariate normal with mean* $\mathbf{0}$ *and covariance matrix* $\gamma^2 \boldsymbol{I}$ ***conditioned on*** $\mu_0 < \mu_1$ *so that the prior pdf is*

$$2I[\mu_0 < \mu_1] \left(\frac{1}{2\pi\gamma^2}\right) \exp\left(-\frac{1}{2\gamma^2}\left(\mu_0^2 + \mu_1^2\right)\right) I[0 < \alpha < 1]$$

an SSS algorithm for

$$\boldsymbol{\theta} = (\alpha, W_1, W_2, \ldots, W_n, \mu_0, \mu_1)$$

(the parameters ***and latent variables****) can be produced fairly easily.*

Suppose that $\boldsymbol{\theta}_0^*$ *is some starting value. With* $\boldsymbol{\theta}_i^* = \left(\alpha_i^*, w_{1,i}^*, w_{2,i}^*, \ldots, w_{n,i}^*, \mu_{0,i}^*, \mu_{1,i}^*\right)$ *in hand, one may*

1. *update* α_i^* *to* α_{i+1}^* *sampling from*

$$Beta\left(\sum_{l=1}^{n} w_{l,i}^* + 1, n - \sum_{l=1}^{n} w_{l,i}^* + 1\right)$$

2. *update each* $w_{l,i}^*$ *to* $w_{l,i+1}^*$ *sampling from*

$$Ber\left(\frac{\alpha_{i+1}^* f\left(y_l|\mu_{1,i}^*\right)}{\left(1-\alpha_{i+1}^*\right) f\left(y_l|\mu_{0,i}^*\right) + \alpha_{i+1}^* f\left(y_l|\mu_{1,i}^*\right)}\right)$$

3. *update* $\mu_{0,i}^*$ *to* $\mu_{0,i+1}^*$ *sampling from a density proportional to*

$$I\left[\mu_0 < \mu_{1,i}^*\right] \exp\left(-\frac{1}{2\gamma^2}\mu_0^2 - \frac{1}{2}\sum_{\substack{l \text{ such that} \\ w_{l,i+1}^*=0}} (y_l - \mu_0)^2\right)$$

which (for $\bar{y}_0$ *the sample mean* y_l *for* $w_{l,i+1}^*$*'s that are* 0 *and* $n_0 = \#\left[w_{l,i+1}^* = 0\right]$*) is a truncated (above at* $\mu_{1,i}^*$*)*

$$N\left(\bar{y}_0\left(\frac{n_0\gamma^2}{n_0\gamma^2+1}\right), \frac{\gamma^2}{n_0\gamma^2+1}\right)$$

density, and

4. *update* $\mu_{1,i}^*$ *to* $\mu_{1,i+1}^*$ *sampling from a density proportional to*

$$I\left[\mu_1 > \mu_{0,i+1}^*\right] \exp\left(-\frac{1}{2\gamma^2}\mu_1^2 - \frac{1}{2}\sum_{\substack{l \text{ such that} \\ w_{l,i+1}^*=1}} (y_l - \mu_1)^2\right)$$

which (for $\bar{y}_1$ *the sample mean* y_l *for* $w_{l,i+1}^*$*'s that are* 1 *and* $n_1 = \#\left[w_{l,i+1}^* = 1\right]$*) is a truncated (below at* $\mu_{0,i+1}^*$*)*

$$N\left(\bar{y}_1\left(\frac{n_{10}\gamma^2}{n_1\gamma^2+1}\right), \frac{\gamma^2}{n_1\gamma^2+1}\right)$$

density.

Upon generating a sequence of realizations $\left(\alpha_i^*, w_{1,i}^*, w_{2,i}^*, \ldots, w_{n,i}^*, \mu_{0,i}^*, \mu_{1,i}^*\right)$*, the sub-vectors* $\left(\alpha_i^*, \mu_{0,i}^*, \mu_{1,i}^*\right)$ *can be used to approximate the posterior for* (α, μ_0, μ_1)*.*

The Metropolis-Hastings Algorithm

An alternative to the Gibbs algorithm (or an alternative to a "sample from a conditional density" step in an SSS algorithm) is the so-called Metropolis-Hastings algorithm. We begin with the "M-H alone" version.

Begin with $\boldsymbol{\theta}_0^*$ (possibly generated from an approximation to $L(\boldsymbol{\theta})\, g(\boldsymbol{\theta})$ or from the prior $g(\boldsymbol{\theta})$). With $\boldsymbol{\theta}_i^*$ in hand, generate $\boldsymbol{\theta}_{i+1}^*$ as follows. For each $\boldsymbol{\theta}$, let $J_{i+1}\left(\boldsymbol{\theta}'|\boldsymbol{\theta}\right)$ specify a distribution for $\boldsymbol{\theta}'$ from which it is possible to sample. Use it to generate a "proposal"/"candidate" replacement for $\boldsymbol{\theta}_i^*$ as

$$\boldsymbol{\theta}_{i+1}^{**} \sim J_{i+1}\left(\cdot|\boldsymbol{\theta}_i^*\right)$$

and accept this proposal based on

$$r_{i+1} = \frac{L\left(\boldsymbol{\theta}_{i+1}^{**}\right) g\left(\boldsymbol{\theta}_{i+1}^{**}\right) / J_{i+1}\left(\boldsymbol{\theta}_{i+1}^{**}|\boldsymbol{\theta}_i^*\right)}{L\left(\boldsymbol{\theta}_i^*\right) g\left(\boldsymbol{\theta}_i^*\right) / J_{i+1}\left(\boldsymbol{\theta}_i^*|\boldsymbol{\theta}_{i+1}^{**}\right)}$$

i.e. with probability $\min(1, r_{i+1})$, otherwise setting $\boldsymbol{\theta}_{i+1}^* = \boldsymbol{\theta}_i^*$. That is, with $Y_{i+1} \sim$Ber$(\min(1, r_{i+1}))$

$$\boldsymbol{\theta}_{i+1}^* = Y_{i+1}\boldsymbol{\theta}_{i+1}^{**} + (1 - Y_{i+1})\,\boldsymbol{\theta}_i^*$$

This is a kind of "adaptive rejection sampling" methodology. One usually chooses $J_{i+1}\left(\boldsymbol{\theta}'|\boldsymbol{\theta}\right)$ to specify $\boldsymbol{\theta}'$ as a small random perturbation of $\boldsymbol{\theta}$.

A very nice special instance of this is one where J_{i+1} is symmetric, i.e. $J_{i+1}\left(\boldsymbol{\theta}'|\boldsymbol{\theta}\right) = J_{i+1}\left(\boldsymbol{\theta}|\boldsymbol{\theta}'\right)$. In this special case the jumping ratio is

$$r_{i+1} = \frac{L\left(\boldsymbol{\theta}_{i+1}^{**}\right) g\left(\boldsymbol{\theta}_{i+1}^{**}\right)}{L\left(\boldsymbol{\theta}_i^*\right) g\left(\boldsymbol{\theta}_i^*\right)}$$

(and one always jumps to the proposal if it takes one up-hill on $L(\boldsymbol{\theta})\, g(\boldsymbol{\theta})$) and this is a so-called "Metropolis" algorithm.

Also (in a very important development) one can use the M-H idea to replace straight Gibbs updates in an SSS algorithm. That is, when updating $\theta_{i,j}^*$ (having updated $\theta_{i,l}^*$ to $\theta_{i+1,l}^*$ for all $l < j$) one specifies $J_{i+1,j}\left(\theta_j'|\theta_j\right)$ (that can actually depend upon not only on θ_j but also on the the current values $\boldsymbol{\theta}_{i+1,<j}^*$ and $\boldsymbol{\theta}_{i,>j}^*$) and generate a proposal

$$\theta_{i+1,j}^{**} \sim J_{i+1,j}\left(\cdot|\theta_{i,j}^*\right)$$

and accept it based on

$$r_{i+1,j} = \frac{L\left(\boldsymbol{\theta}_{i+1,<j}^*, \theta_{i+1,j}^{**}, \boldsymbol{\theta}_{i,>j}^*\right) g\left(\boldsymbol{\theta}_{i+1,<j}^*, \theta_{i+1,j}^{**}, \boldsymbol{\theta}_{i,>j}^*\right) / J_{i+1,j}\left(\theta_{i+1,j}^{**}|\theta_{i,j}^*\right)}{L\left(\boldsymbol{\theta}_{i+1,<j}^*, \theta_{i,j}^*, \boldsymbol{\theta}_{i,>j}^*\right) g\left(\boldsymbol{\theta}_{i+1,<j}^*, \theta_{i,j}^*, \boldsymbol{\theta}_{i,>j}^*\right) / J_{i+1,j}\left(\theta_{i,j}^*|\theta_{i+1,j}^{**}\right)}$$

otherwise setting $\theta_{i+1,j}^* = \theta_{i,j}^*$.

Example 3 *For Example 1, consider replacing the "Gibbs step" for updating c_i^* to c_{i+1}^* in an SSS algorithm with a "M-H step" based on a symmetric jumping kernel. It's not clear how to do this based directly on $c \in (0,1)$. So instead we do the following.*

Defining

$$d = \ln\left(\frac{c}{1-c}\right)$$

we have

$$c = \frac{1}{1+\exp(-d)}$$

The uniform prior for c used before implies that a priori

$$P[d \le t] = P\left[c < \frac{1}{1+\exp(-t)}\right] = \frac{1}{1+\exp(-t)}$$

so that a corresponding prior pdf for d is for $t \in \Re$

$$\frac{d}{dt}\left(\frac{1}{1+\exp(-t)}\right) = \frac{\exp(-t)}{(1+\exp(-t))^2}$$

So one way to replace the Gibbs step with a Metropolis step is to operate on d rather than c and propose

$$d_{i+1}^{**} \sim N\left(d_i^*, \tau^2\right)$$

(where τ^2 is a tuning parameter for the algorithm). Since

$$J_{i+1}(d'|d) = \frac{1}{\sqrt{2\pi\tau^2}}\exp\left(-\frac{1}{2\tau^2}(d-d')^2\right)$$

is symmetric, d_{i+1}^{**} (and the corresponding $c_{i+1}^{**} = 1/\left(1+\exp\left(-d_{i+1}^{**}\right)\right)$) is accepted based on the jumping ratio involving the likelihood times prior

$$\begin{aligned} r_{i+1} &= \frac{L\left(\mu_{1,i+1}^*, \mu_{2,i+1}^*, c_{i+1}^{**}\right) \cdot \left(\frac{\exp\left(-d_{i+1}^{**}\right)}{\left(1+\exp\left(-d_{i+1}^{**}\right)\right)^2}\right)}{L\left(\mu_{1,i+1}^*, \mu_{2,i+1}^*, c_i^*\right) \cdot \left(\frac{\exp(-d_i^*)}{(1+\exp(-d_i^*))^2}\right)} \\ &= \frac{L\left(\mu_{1,i+1}^*, \mu_{2,i+1}^*, c_{i+1}^{**}\right) \cdot c_{i+1}^{**}\left(1-c_{i+1}^{**}\right)}{L\left(\mu_{1,i+1}^*, \mu_{2,i+1}^*, c_i^*\right) \cdot c_i^*\left(1-c_i^*\right)} \end{aligned}$$