Gibbs sampling

Dr. Jarad Niemi

Iowa State University

March 30, 2017

Outline

- Two-component Gibbs sampler
  - Full conditional distribution
- K-component Gibbs sampler
  - Blocked Gibbs sampler
- Metropolis-within-Gibbs
- Slice sampler
  - Latent variable augmentation

Two component Gibbs sampler

Suppose our target distribution is $p(\theta|y)$ with $\theta = (\theta_1, \theta_2)$ and we can sample from $p\left(\theta_1|\theta_2, y\right)$ and $p\left(\theta_2|\theta_1, y\right)$. Beginning with an initial value $\left(\theta_1^0, \theta_2^0\right)$, an iteration of the Gibbs sampler involves

1. Sampling $\theta_1^{(t)} \sim p\left(\theta_1|\theta_2^{(t-1)}, y\right)$.
2. Sampling $\theta_2^{(t)} \sim p\left(\theta_2|\theta_1^{(t)}, y\right)$.

By the Law of Large Numbers, $(\theta_1^{(t)}, \theta_2^{(t)})$ converges to samples from $p(\theta|y)$.

Thus in order to run a Gibbs sampler, we need to derive the full conditional for θ_1 and θ_2, i.e. the distribution for θ_1 and θ_2 conditional on everything else.

Bivariate normal example

Let our target be

$$\theta \sim N_2(0, \Sigma) \qquad \Sigma = \left[\begin{array}{cc} 1 & \rho \\ \rho & 1 \end{array}\right].$$

Then

$$\begin{array}{rl} \theta_1|\theta_2 & \sim N\left(\rho\theta_2, [1-\rho^2]\right) \\ \theta_2|\theta_1 & \sim N\left(\rho\theta_1, [1-\rho^2]\right) \end{array}$$

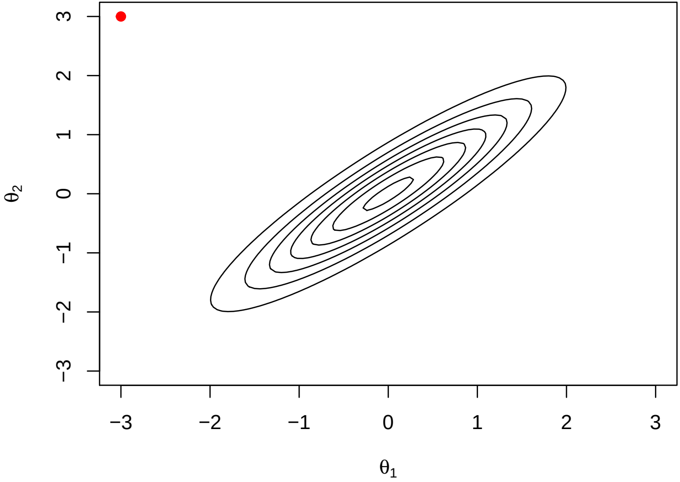
are the conditional distributions.

Assuming initial value $\left(\theta_1^0, \theta_2^0\right)$, the Gibbs sampler proceeds as follows:

Iteration	Sample θ_1	Sample θ_2
1	$\theta_1^{(1)} \sim N\left(\rho\theta_2^0, [1-\rho^2]\right)$	$\theta_2^{(1)} \sim N\left(\rho\theta_1^{(1)}, [1-\rho^2]\right)$
	$\vdots$	
t	$\theta_1^{(t)} \sim N\left(\rho\theta_2^{(t-1)}, [1-\rho^2]\right)$	$\theta_2^{(t)} \sim N\left(\rho\theta_1^{(t)}, [1-\rho^2]\right)$
	$\vdots$	

R code for bivariate normal Gibbs sampler

```
gibbs_bivariate_normal = function(x0, n_points, rho) {
  x = matrix(x0, nrow=n_points, ncol=2, byrow=TRUE)
  v = sqrt(1-rho^2)
  for (i in 2:n_points) {
    x[i,1] = rnorm(1, rho*x[i-1,2], v)
    x[i,2] = rnorm(1, rho*x[i  ,1], v)
  }
  return(x)
}

x = gibbs_bivariate_normal(c(-3,3), n<-20, rho=rho<-0.9)
```


3
2
1
0
−1
−2
−3
θ_2
−3
−2
−1
0
1
2
3
θ_1

Normal model

Suppose $Y_i \stackrel{ind}{\sim} N(\mu, \sigma^2)$ and we assume the prior

$$\mu \sim N(m, C) \qquad \text{and} \qquad \sigma^2 \sim \text{Inv-}\chi^2(v, s^2).$$

Note: this is NOT the conjugate prior.

The full posterior we are interested in is

$$\begin{array}{rl} p(\mu, \sigma^2|y) \propto & (\sigma^2)^{-n/2} \exp\left(-\frac{1}{2\sigma^2}\left(\sum_{i=1}^n (y_i - \mu)^2\right) \exp\left(-\frac{1}{2C}(\mu - m)^2\right)\right. \\ & \times(\sigma^2)^{-(v/2+1)} \exp\left(-\frac{vs^2}{2\sigma^2}\right) \end{array}$$

To run the Gibbs sampler, we need to derive

- $\mu|\sigma^2, y$ and
- $\sigma^2|\mu, y$

Derive $\mu|\sigma^2, y$.

Recall

$$\begin{array}{rl} p(\mu,\sigma^2|y) \propto & (\sigma^2)^{-n/2} \exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^n (y_i-\mu)^2\right) \exp\left(-\frac{1}{2C}(\mu-m)^2\right) \\ & \times(\sigma^2)^{-(v/2+1)} \exp\left(-\frac{vs^2}{2\sigma^2}\right) \end{array}$$

Now find $\mu|\sigma^2, y$:

$$\begin{array}{rl} p(\mu|\sigma^2,y) & \propto p(\mu,\sigma^2|y) \\ & \propto \exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^n (y_i-\mu)^2\right) \exp\left(-\frac{1}{2C}(\mu-m)^2\right) \\ & \propto \exp\left(-\frac{1}{2}\left[\left(\frac{1}{\sigma^2/n}+\frac{1}{C}\right)\mu^2 - 2\mu\left(\frac{\overline{y}}{\sigma^2/n}+\frac{m}{C}\right)\right]\right) \end{array}$$

thus $\mu|\sigma^2, y \sim N(m', C')$ where

$$\begin{array}{rl} m' & = C'\left(\frac{\overline{y}}{\sigma^2/n}+\frac{m}{C}\right) \\ C' & = \left(\frac{1}{\sigma^2/n}+\frac{1}{C}\right)^{-1} \end{array}$$

Derive $\sigma^2|\mu, y$.

Recall

$$\begin{array}{rl} p(\mu,\sigma^2|y) \propto & (\sigma^2)^{-n/2} \exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^n (y_i-\mu)^2\right) \exp\left(-\frac{1}{2C}(\mu-m)^2\right) \\ & \times (\sigma^2)^{-(v/2+1)} \exp\left(-\frac{vs^2}{2\sigma^2}\right) \end{array}$$

Now find $\sigma^2|\mu, y$:

$$\begin{array}{rl} p(\sigma^2|\mu,y) & \propto p(\mu,\sigma^2|y) \\ & \propto (\sigma^2)^{-([v+n]/2+1)} \exp\left(-\frac{1}{2\sigma^2}\left[vs^2+\sum_{i=1}^n (y_i-\mu)^2\right]\right) \end{array}$$

and thus $\sigma^2|\mu, y \sim \text{Inv-}\chi^2(v', (s')^2)$ where

$$\begin{array}{rl} v' & = v+n \\ v'(s')^2 & = vs^2 + \sum_{i=1}^n (y_i-\mu)^2 \end{array}$$

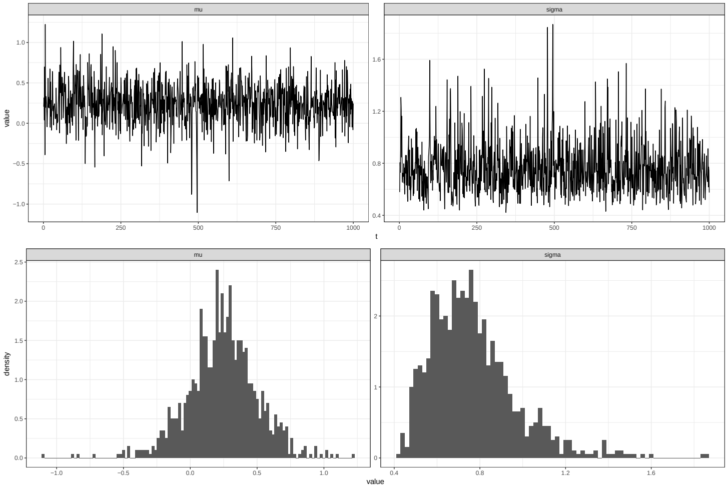
R code for Gibbs sampler

```
# Data and prior
y = rnorm(10)
m = 0; C = 10
v = 1; s = 1

# Initial values
mu = 0
sigma2 = 1

# Save structures
n_iter = 1000
mu_keep = rep(NA, n_iter)
sigma_keep = rep(NA, n_iter)

# Pre-calculate
n = length(y)
sum_y = sum(y)
vp = v+n
vs2 = v*s^2
```

R code for Gibbs sampler

```
# Gibbs sampler
for (i in 1:n_iter) {
  # Sample mu
  Cp = 1/(n/sigma2+1/C)
  mp = Cp*(sum_y/sigma2+m/C)
  mu = rnorm(1, mp, sqrt(Cp))

  # Sample sigma
  vpsp2 = vs2 + sum((y-mu)^2)
  sigma2 = 1/rgamma(1, vp/2, vpsp2/2)

  # Save iterations
  mu_keep[i] = mu
  sigma_keep[i] = sqrt(sigma2)
}
```

Posteriors



K-component Gibbs sampler

Suppose $\theta = (\theta_1, \ldots, \theta_K)$, then an iteration of a K-component Gibbs sampler is

$$
\begin{aligned}
\theta_1^{(t)} &\sim p\left(\theta_1 | \theta_2^{(t-1)}, \ldots, \theta_K^{(t-1)}, y\right) \\
\theta_2^{(t)} &\sim p\left(\theta_2 | \theta_1^{(t)}, \theta_3^{(t-1)}, \ldots, \theta_K^{(t-1)}, y\right) \\
&\vdots \\
\theta_k^{(t)} &\sim p\left(\theta_k | \theta_1^{(t)} \ldots, \theta_{k-1}^{(t)}, \theta_{k+1}^{(t-1)}, \ldots, \theta_K^{(t-1)}, y\right) \\
&\vdots \\
\theta_K^{(t)} &\sim p\left(\theta_K | \theta_1^{(t)}, \ldots, \theta_{K-1}^{(t)}, y\right)
\end{aligned}
$$

By the Law of Large Numbers, $(\theta_1^{(t)}, \theta_2^{(t)})$ converges to samples from $p(\theta|y)$.

The distributions above are called the full conditional distributions. If some of the θ_k are vectors, then this is called a block Gibbs sampler.

Hierarchical normal model

Let

$$Y_{ij} \stackrel{ind}{\sim} N(\mu_i, \sigma^2), \qquad \mu_i \stackrel{ind}{\sim} N(\eta, \tau^2)$$

for $i = 1, \ldots, \mathrm{I}$, $j = 1, \ldots, n_i$, $n = \sum_{i=1}^{\mathrm{I}} n_i$ and prior

$$p(\eta, \tau^2, \sigma) \propto IG(\tau^2; a_\tau, b_\tau) IG(\sigma^2; a_\sigma, b_\sigma).$$

The full conditionals are

$$\begin{array}{rl}
p(\mu|\eta, \sigma^2, \tau^2, y) & = \prod_{i=1}^{n} p(\mu_i|\eta, \sigma^2, \tau^2, y_i) \\
p(\mu_i|\eta, \sigma^2, \tau^2, y_i) & = N\left(\left[\frac{1}{\sigma^2/n_i} + \frac{1}{\tau^2}\right] \left[\frac{\overline{y}_i}{\sigma^2/n_i} + \frac{\eta}{\tau^2}\right], \left[\frac{1}{\sigma^2/n_i} + \frac{1}{\tau^2}\right]^{-1} \right) \\
p(\eta|\mu, \sigma^2, \tau^2, y) & = N\left(\overline{\mu}, \tau^2/\mathrm{I}\right) \\
p(\sigma^2|\mu, \eta, \tau^2, y) & = IG(a_\sigma + n/2, b_\sigma + \sum_{i=1}^{\mathrm{I}} \sum_{j=1}^{n_j} (y_{ij} - \mu_i)^2/2) \\
p(\tau^2|\mu, \eta, \sigma^2, y) & = IG(a_\tau + \mathrm{I}/2, b_\tau + \sum_{i=1}^{\mathrm{I}} (\mu_i - \eta)^2/2)
\end{array}$$

where $n_i \overline{y}_i = \sum_{j=1}^{n_i} y_{ij}$ and $\mathrm{I}\,\overline{\mu} = \sum_{i=1}^{\mathrm{I}} \mu_i$.

Metropolis-within-Gibbs

We have discussed two Markov chain approaches to sample from a target distribution:

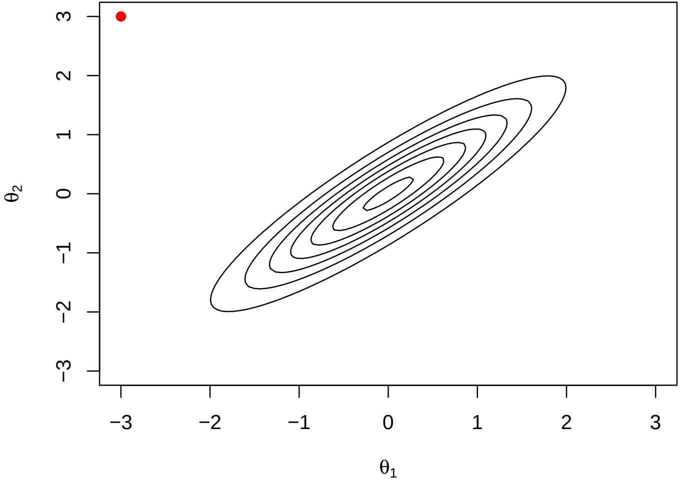
- Metropolis-Hastings algorithm
- Gibbs sampling

Gibbs sampling assumed we can sample from $p(\theta_k|\theta_{-k}, y)$ for all k, but what if we cannot sample from all of these full conditional distributions? For those $p(\theta_k|\theta_{-k})$ that cannot be sampled directly, a single iteration of the Metropolis-Hastings algorithm can be substituted.

Bivariate normal with $\rho = 0.9$

Reconsider the bivariate normal example substituting a Metropolis step in place of a Gibbs step:

```
gibbs_and_metropolis = function(x0, n_points, rho) {
  x = matrix(x0, nrow=n_points, ncol=2, byrow=TRUE)
  v = sqrt(1-rho^2)
  for (i in 2:n_points) {
    x[i,1] = rnorm(1, rho*x[i-1,2], v)

    # Now do a random-walk Metropolis step
    x_prop = rnorm(1, x[i-1,2], 2.4*v) # optimal proposal variance
    logr   = dnorm(x_prop,   rho*x[i,1], v, log=TRUE) -
             dnorm(x[i-1,2], rho*x[i,1], v, log=TRUE)
    x[i,2] = ifelse(log(runif(1))<logr, x_prop, x[i-1,2])
  }
  return(x)
}

x = gibbs_and_metropolis(c(-3,3), n, rho)
length(unique(x[,2]))/length(x[,2]) # acceptance rate


[1] 0.5
```



3
2
1
0
−1
−2
−3
θ_2
−3
−2
−1
0
1
2
3
θ_1

Hierarchical normal model

Let

$$Y_{ij} \stackrel{ind}{\sim} N(\mu_i, \sigma^2), \qquad \mu_i \stackrel{ind}{\sim} N(\eta, \tau^2)$$

for $i = 1, \ldots, \mathrm{I}$, $j = 1, \ldots, n_i$, $n = \sum_{i=1}^{\mathrm{I}} n_i$ and prior

$$p(\eta, \tau, \sigma) \propto Ca^+(\tau; 0, b_\tau) Ca^+(\sigma; 0, b_\sigma).$$

The full conditionals are exactly the same except

$$\begin{aligned} p(\sigma|\mu, \eta, \tau^2, y) &\propto IG(\sigma^2; n/2, \textstyle\sum_{i=1}^{\mathrm{I}} \sum_{j=1}^{n_j} (y_{ij} - \mu_i)^2/2) Ca^+(\sigma; 0, b_\sigma) \\ p(\tau^2|\mu, \eta, \sigma^2, y) &\propto IG(\tau^2; \mathrm{I}/2, \textstyle\sum_{i=1}^{\mathrm{I}} (\mu_i - \eta)^2/2) Ca^+(\tau; 0, b_\tau) \end{aligned}$$

where $n_i \overline{y}_i = \sum_{j=1}^{n_i} y_{ij}$ and $\mathrm{I}\,\overline{\mu} = \sum_{i=1}^{\mathrm{I}} \mu_i$.

Hierarchical normal model

To sample from $p(\tau|y) \propto IG(\tau^2; a, b) Ca^+(0, b_\tau)$ (or equivalently $p(\sigma|y)$), we have a variety of possibilities. Here are three:

1. Rejection sampling with $(\tau^*)^2 \sim IG(a, b)$ and thus $M^*_{opt} = Ca^+(0; 0, b_\tau)$ and the acceptance probability is $Ca^+(\tau^*; 0, b_\tau)/M^*_{opt}$.
2. Independence Metropolis-Hastings with $(\tau^*)^2 \sim IG(a, b)$ and thus the acceptance probability is $Ca^+(\tau^*; 0, b_\tau)/Ca^+(\tau^{(t)}; 0, b_\tau)$.
3. Random-walk Metroplis-Hastings with $\tau^* \sim g(\cdot|\tau^{(t)})$ and acceptance probability is $q(\tau^*|y)/q(\tau^{(t)}|y)$.

Hierarchical binomial model

Let

$$\begin{aligned} Y_i &\stackrel{ind}{\sim} Bin(n_i, \theta_i) \\ \theta_i &\stackrel{iid}{\sim} Be(\alpha, \beta) \\ p(\alpha, \beta) &\propto (\alpha+\beta)^{-5/2} \end{aligned}$$

We will use a dependson to sample from $\theta_1, \ldots, \theta_n, \alpha, \beta$, so we need to derive the following conditional distributions:

- $\theta_i|\theta_1, \ldots, \theta_{i-1}, \theta_{i+1}, \ldots, \theta_n, \alpha, \beta, y$
- $\alpha|\theta_1, \ldots, \theta_n, \beta, y$
- $\beta|\theta_1, \ldots, \theta_n, \alpha, y$

For shorthand, I often use $\theta_i| \ldots$ where "$\ldots$" indicates *everything else*.

Full conditional for θ_i

$$Y_i \stackrel{ind}{\sim} Bin(n_i, \theta_i), \quad \theta_i \stackrel{iid}{\sim} Be(\alpha, \beta), \quad p(\alpha, \beta) \propto (\alpha + \beta)^{-5/2}$$

The full conditional for θ_i is

$$\begin{array}{rl} p(\theta_i | \ldots) & \propto p(y|\theta)p(\theta|\alpha,\beta)p(\alpha,\beta) \\ & \propto \left[\prod_{i=1}^n p(y_i|\theta_i)\right] \left[\prod_{i=1}^n p(\theta_i|\alpha,\beta)\right] \\ & \propto \prod_{j=1}^n \theta_j^{y_j}(1-\theta_j)^{n_j-y_j}\theta_j^{\alpha-1}(1-\theta_j)^{\beta-1} \\ & \propto \theta_i^{\alpha+y_i-1}(1-\theta_i)^{\beta+n_i-y_i-1} \end{array}$$

Thus $\theta_i | \ldots \sim Be(\alpha + y_i, \beta + n_i - y_i)$.

Full conditional for α and β

$$Y_i \stackrel{ind}{\sim} Bin(n_i, \theta_i), \quad \theta_i \stackrel{iid}{\sim} Be(\alpha, \beta), \quad p(\alpha, \beta) \propto (\alpha + \beta)^{-5/2}$$

The full conditional for α is

$$\begin{array}{rl} p(\alpha | \ldots) & \propto p(y|\theta) p(\theta|\alpha, \beta) p(\alpha, \beta) \\ & \propto \left[\prod_{i=1}^n p(\theta_i | \alpha, \beta)\right] p(\alpha, \beta) \\ & \propto \frac{\left(\prod_{i=1}^n \theta_i\right)^{\alpha - 1}}{Beta(\alpha, \beta)^n} (\alpha + \beta)^{-5/2} \end{array}$$

which is not a known density.

The full conditional for β is

$$\begin{array}{rl} p(\beta | \ldots) & \propto p(y|\theta) p(\theta|\alpha, \beta) p(\alpha, \beta) \\ & \propto \left[\prod_{i=1}^n p(\theta_i | \alpha, \beta)\right] p(\alpha, \beta) \\ & \propto \frac{\left(\prod_{i=1}^n [1 - \theta_i]\right)^{\beta - 1}}{Beta(\alpha, \beta)^n} (\alpha + \beta)^{-5/2} \end{array}$$

which is not a known density.

Full conditional functions

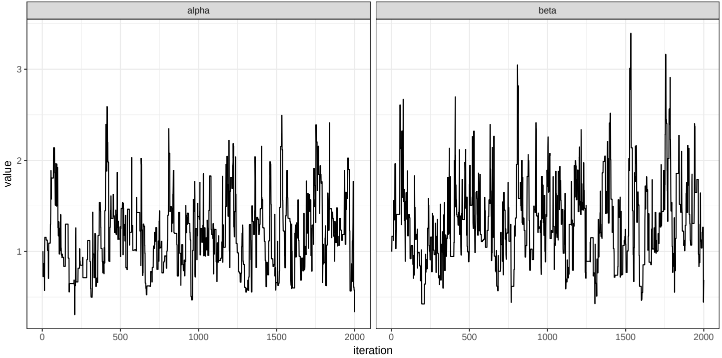
$$\begin{aligned}
\log p(\alpha|\ldots) &\propto (\alpha-1)\sum_{i=1}^n \log\left(\quad\theta_i\right) + n\log(Beta(\alpha,\beta)) - 5/2\log(\alpha+\beta) \\
\log p(\beta|\ldots) &\propto (\beta-1)\sum_{i=1}^n \log\left(1-\theta_i\right) + n\log(Beta(\alpha,\beta)) - 5/2\log(\alpha+\beta)
\end{aligned}$$

```
log_fc_alpha = function(theta, alpha, beta) {
  if (alpha<0) return(-Inf)
  n = length(theta)
  (alpha-1)*sum(log(theta))-n*lbeta(alpha,beta)-5/2*(alpha+beta)
}

log_fc_beta = function(theta, alpha, beta) {
  if (beta<0) return(-Inf)
  n = length(theta)
  (beta-1)*sum(log(1-theta))-n*lbeta(alpha,beta)-5/2*(alpha+beta)
}
```

```
mcmc = function(n_sims, dat, inits, tune) {
  n_groups = nrow(dat)
  alpha = inits$alpha
  beta  = inits$beta

  # Recording structure
  theta_keep = matrix(NA, nrow=n_sims, ncol=n_groups)
  alpha_keep = rep(alpha, n_sims)
  beta_keep  = rep(beta , n_sims)

  for (i in 1:n_sims) {
    # Sample thetas
    theta = with(dat, rbeta(length(y), alpha+y, beta+n-y))

    # Sample alpha
    alpha_prop = rnorm(1, alpha, tune$alpha)
    logr = log_fc_alpha(theta, alpha_prop, beta)-log_fc_alpha(theta, alpha, beta)
    alpha = ifelse(log(runif(1))<logr, alpha_prop, alpha)

    # Sample beta
    beta_prop = rnorm(1, beta, tune$beta)
    logr = log_fc_beta(theta, alpha, beta_prop)-log_fc_beta(theta, alpha, beta)
    beta = ifelse(log(runif(1))<logr, beta_prop, beta)

    # Record parameter values
    theta_keep[i,] = theta
    alpha_keep[i]  = alpha
    beta_keep[ i]  = beta
  }

  return(data.frame(iteration=1:n_sims,
                    parameter=rep(c("alpha","beta",paste("theta[",1:n_groups,"]",sep="")),each=n_sims),
                    value=c(alpha_keep,beta_keep,as.numeric(theta_keep))))
}
```

```
d = read.csv("../Ch05/Ch05a-dawkins.csv")
dat=data.frame(y=d$made, n=d$attempt)
inits = list(alpha=1, beta=1)

# Run the MCMC
r = mcmc(2000, dat=dat, inits=inits, tune=list(alpha=1,beta=1))
```

```
  parameter acceptance_rate
1     alpha           0.261
2      beta           0.306
```



Block Gibbs sampler

It appears that the Gibbs sampler we have constructed iteratively samples from

1. $\theta_1 \sim p(\theta_1|\theta_{-1}, \alpha, \beta, y)$
2. $\vdots$
3. $\theta_n \sim p(\theta_n|\theta_{-n}, \alpha, \beta, y)$
4. $\alpha \sim p(\alpha|\theta, \beta, y)$
5. $\beta \sim p(\beta|\theta, \alpha, y)$

where $\theta = (\theta_1, \ldots, \theta_n)$ and θ_{-i} is θ without element i.

But notice that

$$p(\theta|\alpha, \beta, y) = \prod_{i=1}^{n} p(\theta_i|\alpha, \beta, y_i)$$

and thus the θ_i are conditionally independent. Thus, we actually ran the following block Gibbs sampler:

1. $\theta \sim p(\theta|\alpha, \beta, y)$
2. $\alpha \sim p(\alpha|\theta, \beta, y)$
3. $\beta \sim p(\beta|\theta, \alpha, y)$

Slice sampling

Suppose the target distribution is $p(\theta|y)$ with scalar θ. Then,

$$p(\theta|y) = \int_0^{p(\theta|y)} du$$

Thus, $p(\theta|y)$ can be thought of as the marginal distribution of

$$(\theta, U) \sim \text{Unif}\{(\theta, u) : 0 < u < p(\theta|y)\}$$

where u is an auxiliary variable.

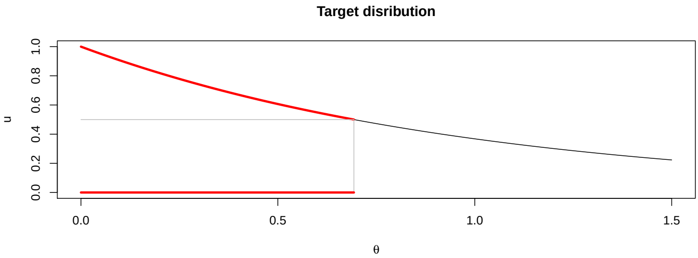
Slice sampling performs the following Gibbs sampler:

1. $u^{(t)}|\theta^{(t-1)}, y \sim \text{Unif}\{u : 0 < u < p(\theta^{(t-1)}|y)\}$ and
2. $\theta^{(t)}|u^{(t)}, y \sim \text{Unif}\{\theta : u^{(t)} < p(\theta|y)\}$.

Slice sampler for exponential distribution

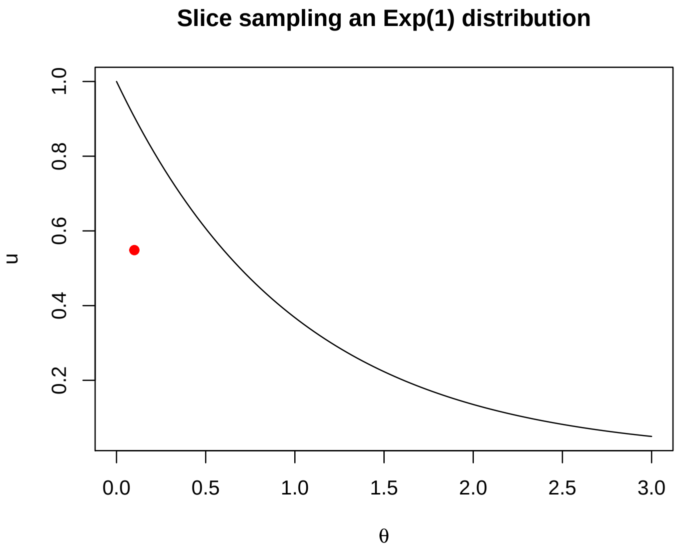
Consider the target $\theta|y \sim Exp(1)$,then

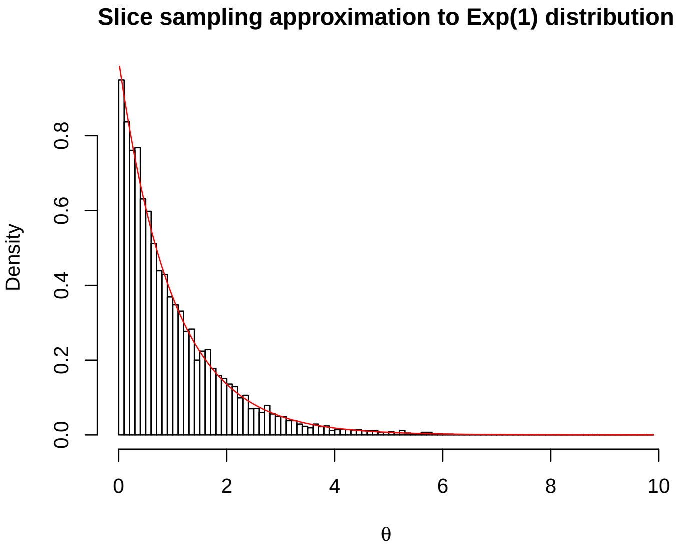
$$\{\theta : u < p(\theta|y)\} = (0, -\log(u)).$$



Slice sampling in R

```r
# Slice sampler
slice = function(n,init_theta,target,A) {
  u = theta = rep(NA,n)
  theta[1] = init_theta
  u[1] = runif(1,0,target(theta[1])) # This never actually gets used

  for (i in 2:n) {
    u[i] = runif(1,0,target(theta[i-1]))
    endpoints = A(u[i],theta[i-1]) # The second argument is used in the second example
    theta[i] = runif(1, endpoints[1],endpoints[2])
  }
  return(list(theta=theta,u=u))
}
# Exponential example
set.seed(6)
A = function(u,theta=NA) c(0,-log(u))
res = slice(10, 0.1, dexp, A)
```


Slice sampling an Exp(1) distribution
u
1.0
0.8
0.6
0.4
0.2
0.0
0.5
1.0
1.5
2.0
2.5
3.0
θ


Slice sampling approximation to Exp(1) distribution
Density
0.0
0.2
0.4
0.6
0.8
0
2
4
6
8
10
θ

Summary

- Gibbs sampling breaks down a hard problem of sampling from a high dimensional distribution to a set of easier problems, i.e. sampling from low dimensional full conditional distributions.
- If the low dimensional distributions have an unknown form, then alternative methods can be used, e.g. (adaptive) rejection sampling, Metropolis-Hastings, etc.
- A Gibbs sampler can always be constructed by introducing an auxiliary variable that horizontally slices the target density.