

Gibbs sampling

Dr. Jarad Niemi

Iowa State University

March 29, 2016

Outline

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

Gibbs sampling

Suppose our target distribution is $p(\theta|y)$ with $\theta = (\theta_1, \theta_2)$ and we can sample from $p(\theta_1|\theta_2, y)$ and $p(\theta_2|\theta_1, y)$. Beginning with an initial value (θ_1^0, θ_2^0) , the **Gibbs sampler** is

1. sample $\theta_1^t \sim p(\theta_1|\theta_2^{t-1}, y)$ and then
2. sample $\theta_2^t \sim p(\theta_2|\theta_1^t, y)$.

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

$$\theta \sim N_2(0, \Sigma) \quad \Sigma = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}.$$

Then

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

are the conditional distributions.

Assuming initial value (θ_1^0, θ_2^0) , the Gibbs sampler proceeds as follows:

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


Normal model

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

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

Note: this is NOT the conjugate prior.

The full posterior we are interested in is

$$p(\mu, \sigma^2 | y) \propto (\sigma^2)^{-n/2} \exp\left(-\frac{1}{2\sigma^2} \left(\sum_{i=1}^n (y_i - \mu)^2\right)\right) \exp\left(-\frac{1}{2C} (\mu - m)^2\right) \\ \times (\sigma^2)^{-(\nu/2+1)} \exp\left(-\frac{\nu s^2}{2\sigma^2}\right)$$

To run the Gibbs sampler, we need to derive

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

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

Recall

$$p(\mu, \sigma^2|y) \propto (\sigma^2)^{-n/2} \exp\left(-\frac{1}{2\sigma^2} \sum_{i=1}^n (y_i - \mu)^2\right) \exp\left(-\frac{1}{2C}(\mu - m)^2\right) \\ \times (\sigma^2)^{-(\nu/2+1)} \exp\left(-\frac{\nu s^2}{2\sigma^2}\right)$$

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

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

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

$$m' = C' \left(\frac{\bar{y}}{\sigma^2/n} + \frac{m}{C} \right) \\ C' = \left(\frac{1}{\sigma^2/n} + \frac{1}{C} \right)^{-1}$$

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

Recall

$$p(\mu, \sigma^2|y) \propto (\sigma^2)^{-n/2} \exp\left(-\frac{1}{2\sigma^2} \sum_{i=1}^n (y_i - \mu)^2\right) \exp\left(-\frac{1}{2C} (\mu - m)^2\right) \\ \times (\sigma^2)^{-(\nu/2+1)} \exp\left(-\frac{\nu s^2}{2\sigma^2}\right)$$

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

$$p(\sigma^2|\mu, y) \propto p(\mu, \sigma^2|y) \\ \propto (\sigma^2)^{-([\nu+n]/2+1)} \exp\left(-\frac{1}{2\sigma^2} [\nu s^2 + \sum_{i=1}^n (y_i - \mu)^2]\right)$$

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

$$\begin{aligned} \nu' &= \nu + n \\ \nu'(s')^2 &= \nu s^2 + \sum_{i=1}^n (y_i - \mu)^2 \end{aligned}$$

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
```

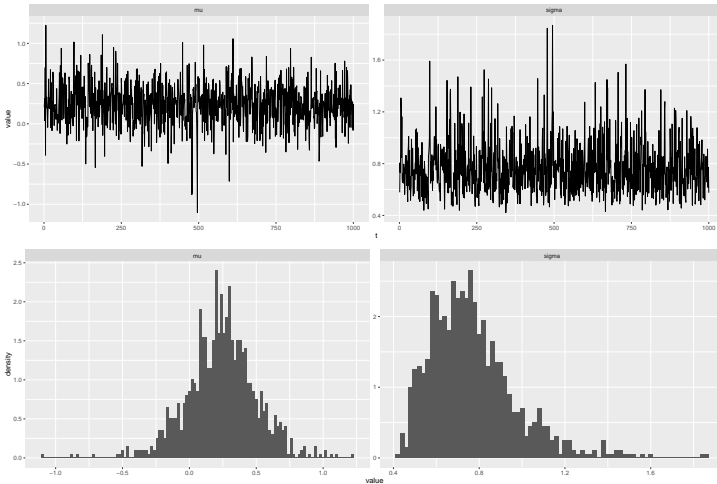
R code for Gibbs sampler

```
# 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 $\theta = (\theta_1, \dots, \theta_K)$, then the following Gibbs sampler can be constructed

$$\theta_1^t \sim p(\theta_1 | \theta_2^{t-1}, \dots, \theta_K^{t-1}, y)$$

$$\theta_2^t \sim p(\theta_2 | \theta_1^t, \theta_3^{t-1}, \dots, \theta_K^{t-1}, y)$$

$$\vdots$$

$$\theta_k^t \sim p(\theta_k | \theta_1^t, \dots, \theta_{k-1}^t, \theta_{k+1}^{t-1}, \dots, \theta_K^{t-1}, y)$$

$$\vdots$$

$$\theta_K^t \sim p(\theta_K | \theta_1^t, \dots, \theta_{K-1}^t, y)$$

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

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

for $i = 1, \dots, I, j = 1, \dots, n_i$, $n = \sum_{i=1}^I n_i$ 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

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

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

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
```


Hierarchical normal model

Let

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

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

$$p(\eta, \tau^2, \sigma) \propto Ca^+(\tau^2; 0, b_\tau) Ca^+(\sigma^2; 0, b_\sigma).$$

The full conditionals are exactly the same except

$$p(\sigma^2 | \mu, \eta, \tau^2, y) \propto IG(n/2, \sum_{i=1}^I \sum_{j=1}^{n_i} (y_{ij} - \mu_i)^2 / 2) Ca^+(0, b_\sigma)$$

$$p(\tau^2 | \mu, \eta, \sigma^2, y) \propto IG(I/2, \sum_{i=1}^I (\mu_i - \eta)^2 / 2) Ca^+(0, b_\tau)$$

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

Hierarchical normal model

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

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

Hierarchical binomial model

Let

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

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

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

For shorthand, we often use $\theta_i | \dots$ where \dots indicates *everything else*.

Full conditional for θ_i

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

The full conditional for θ_i is

$$\begin{aligned} p(\theta_i | \dots) &\propto p(y_i | \theta_i) p(\theta_i | \alpha, \beta) p(\alpha, \beta) \\ &\propto [\prod_{i=1}^n p(y_i | \theta_i)] [\prod_{i=1}^n p(\theta_i | \alpha, \beta)] \\ &\propto \prod_{j=1}^n \theta_j^{y_j} (1 - \theta_j)^{n_j - y_j} \theta_j^{\alpha - 1} (1 - \theta_j)^{\beta - 1} \\ &\propto \theta_i^{\alpha + y_i - 1} (1 - \theta_i)^{\beta + n_i - y_i - 1} \end{aligned}$$

Thus $\theta_i | \dots \sim \text{Be}(\alpha + y_i, \beta + n_i - y_i)$.

Full conditional for α and β

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

The full conditional for α is

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

which is not a known density.

The full conditional for β is

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

which is not a known density.

Full conditional functions

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

```
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("Ch05a-dawkins.csv")
```

```
Warning in file(file, "rt"): cannot open file 'Ch05a-dawkins.csv': No such file or directory
```

```
Error in file(file, "rt"): cannot open the connection
```

```
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))
```

```
Warning in rbeta(length(y), alpha + y, beta + n - y): NAs produced
```

```
Warning in rnorm(1, alpha, tune$alpha): NAs produced
```

```
Error in if (alpha < 0) return(-Inf): missing value where TRUE/FALSE needed
```

```
Error in eval(expr, envir, enclos): object 'r' not found
```

```
Error in empty(.data): object 'r_subset' not found
```

```
Error in ggplot(r_subset, aes(x = iteration, y = value)): object 'r_subset' not found
```

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 $\theta = (\theta_1, \dots, \theta_n)$ 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**.

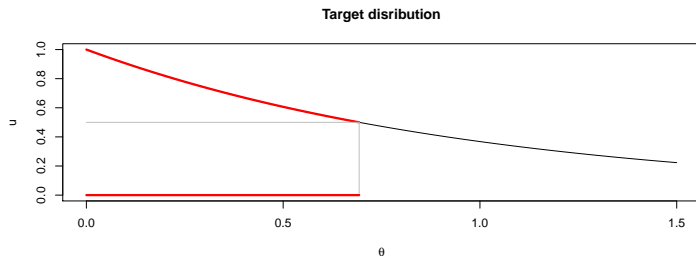
Slice sampling performs the following Gibbs sampler:

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 $\theta|y \sim \text{Exp}(1)$, then

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



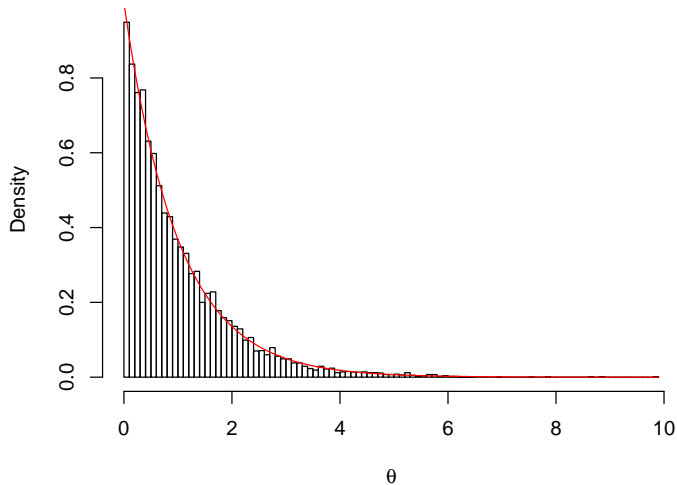
Slice sampling in R

```
# 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 approximation to $\text{Exp}(1)$ distribution



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.