HW 7: Advanced MCMC

Problem 8.1

One approach to Bayesian variable selection for linear regression models is described in Section 8.2.1 and further examined in Example 8.3. For a Bayesian analysis for the model in Equation (8.20), we might adopt the normal–gamma conjugate class of priors $\beta|m_k \sim N(\alpha_{m_k}, \sigma^2 \mathbf{V}_{m_k})$ and $v\lambda/\sigma^2 \sim \chi^2_v$. Show that the marginal density of $Y|m_k$ is given by

$$\frac{\Gamma((v+n)/2)(v\lambda)^{v/2}}{\pi^{n/2}\Gamma(v/2)|\mathbf{I}+\mathbf{X}_{m_k}\mathbf{V}_{m_k}\mathbf{X}_{m_k}^T|^{-(v+n)/2}} \times \left[\lambda v + (\mathbf{Y}-\mathbf{X}_{m_k}\alpha_{m_k})^T(\mathbf{I}+\mathbf{X}_{m_k}\mathbf{V}_{m_k}\mathbf{X}_{m_k}^T)^{-1}(\mathbf{Y}-\mathbf{X}_{m_k}\alpha_{m_k})\right]^{-(v+n)/2} \tag{1}$$

where $\mathbf{X}_{m_k}$ is the design matrix, α_{m_k} is the mean vector, and $\mathbf{V}_{m_k}$ is the covariance matrix for β_{m_k} for the model m_k.

Since $\mathbf{Y} = \mathbf{X}_{m_k}\beta_{m_k} + \epsilon$, meanwhile $\beta|m_k \sim N(\alpha_{m_k}, \sigma^2\mathbf{V}_{m_k})$ and $\epsilon \sim N(0, \sigma\mathbf{I})$.

$\mathbf{Y}|m_k$ is obviously a sum of two normal random variables, which leads to

$$\Rightarrow Y|m_k \sim N(\mathbf{X}_{m_k}\alpha_{m_k}, \sigma^2(\mathbf{X}_{m_k}\mathbf{V}_{m_k}\mathbf{X}'_{m_k} + \mathbf{I})) \tag{2}$$

And because $\frac{v\lambda}{\sigma^2} \sim \chi^2_v$,

$$\begin{aligned}
\Rightarrow p(\frac{v\lambda}{\sigma^2}) &= \frac{1}{2^{v/2}+\Gamma(\frac{v}{2})}(\frac{v\lambda}{\sigma^2})^{\frac{v}{2}-1}e^{-\frac{v\lambda}{2\sigma^2}} \\
\Rightarrow p(\sigma^2) &= \frac{1}{2^{v/2}+\Gamma(\frac{v}{2})}(\frac{v\lambda}{\sigma^2})^{\frac{v}{2}-1}e^{-\frac{v\lambda}{2\sigma^2}} \cdot \frac{v\lambda}{(\sigma^2)^2} \text{ (Change variable)} \\
&= \frac{(v\lambda)^{v/2}}{2^{v/2}\Gamma(\frac{v}{2})}(\frac{1}{\sigma^2})^{\frac{v}{2}+1}e^{-\frac{v\lambda}{2\sigma^2}}
\end{aligned} \tag{3}$$

Therefore, by inserting equation (3) to equation (2).

$$\begin{aligned}
p(y,\sigma^2|m_k) =& p(y|\sigma^2, m_k)\cdot p(\sigma^2) \\
=& \left(\frac{(v\lambda)^{v/2}}{2^{v/2}\Gamma(\frac{v}{2})}\Big/ (2\pi)^{n/2}|\mathbf{X}_{m_k}\mathbf{V}_{m_k}\mathbf{X}'_{m_k}+\mathbf{I}|^{1/2}\right)\cdot \\
& e^{-\frac{(\mathbf{y}-\mathbf{X}_{m_k}\alpha_{m_k})'(\mathbf{X}_{m_k}\mathbf{V}_{m_k}\mathbf{X}'_{m_k}+\mathbf{I})^{-1}(\mathbf{y}-\mathbf{X}_{m_k}\alpha_{m_k})}{2\sigma^2}} \cdot (\frac{1}{\sigma^2})^{\frac{v+1}{2}} \cdot e^{-\frac{v\lambda}{2\sigma^2}}
\end{aligned} \tag{4}$$

In the end, we can calculate $p(y|m_k)$ by integrating σ^2,

$$
\begin{aligned}
p(y|m_k) &= \int p(y,\sigma^2|m_k)\, d\sigma^2 \\
&= \frac{\Gamma((v+n)/2)(v\lambda)^{v/2}}{\pi^{n/2}\Gamma(v/2)|\mathbf{I}+\mathbf{X}_{m_k}\mathbf{V}_{m_k}\mathbf{X}_{m_k}^T|^{-(v+n)/2}} \\
&\quad \times \left[\lambda v + (\mathbf{Y}-\mathbf{X}_{m_k}\alpha_{m_k})^T(\mathbf{I}+\mathbf{X}_{m_k}\mathbf{V}_{m_k}\mathbf{X}_{m_k}^T)^{-1}(\mathbf{Y}-\mathbf{X}_{m_k}\alpha_{m_k})\right]^{-(v+n)/2}
\end{aligned} \tag{5}
$$

Problem 8.3

Suppose we desire a draw from the marginal distribution of X that is determined by the assumptions that $\theta \sim Beta(\alpha,\beta)$ *and* $X|\theta \sim Bin(n,\theta)$*.*

***a**. Show that* $\theta|X \sim Beta(\alpha+x, \beta+n-x)$*.*

$\because \theta \sim Beta(\alpha,\beta)$ and $X|\theta \sim Bin(n,\theta)$,

i.e. $p(\theta|\alpha,\beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\theta^{\alpha-1}(1-\theta)^{\beta-1}$ and $p(x|\theta) = \binom{n}{x}\theta^x(1-\theta)^{n-x} = \frac{n!}{x!(n-x)!}\theta^x(1-\theta)^{n-x}$.

$\therefore$

$$
\begin{aligned}
p(\theta|x,\alpha,\beta) &\propto p(\theta|\alpha,\beta)\cdot p(x|\theta) \\
&\propto \theta^{x+\alpha-1}(1-\theta)^{n+\beta-x-1}
\end{aligned} \tag{6}
$$

Thus, it is obviously the kernel of Beta distribution, which leads to,

$$
\Rightarrow \theta|X \sim Beta(\alpha+x, \beta+n-x) \tag{7}
$$

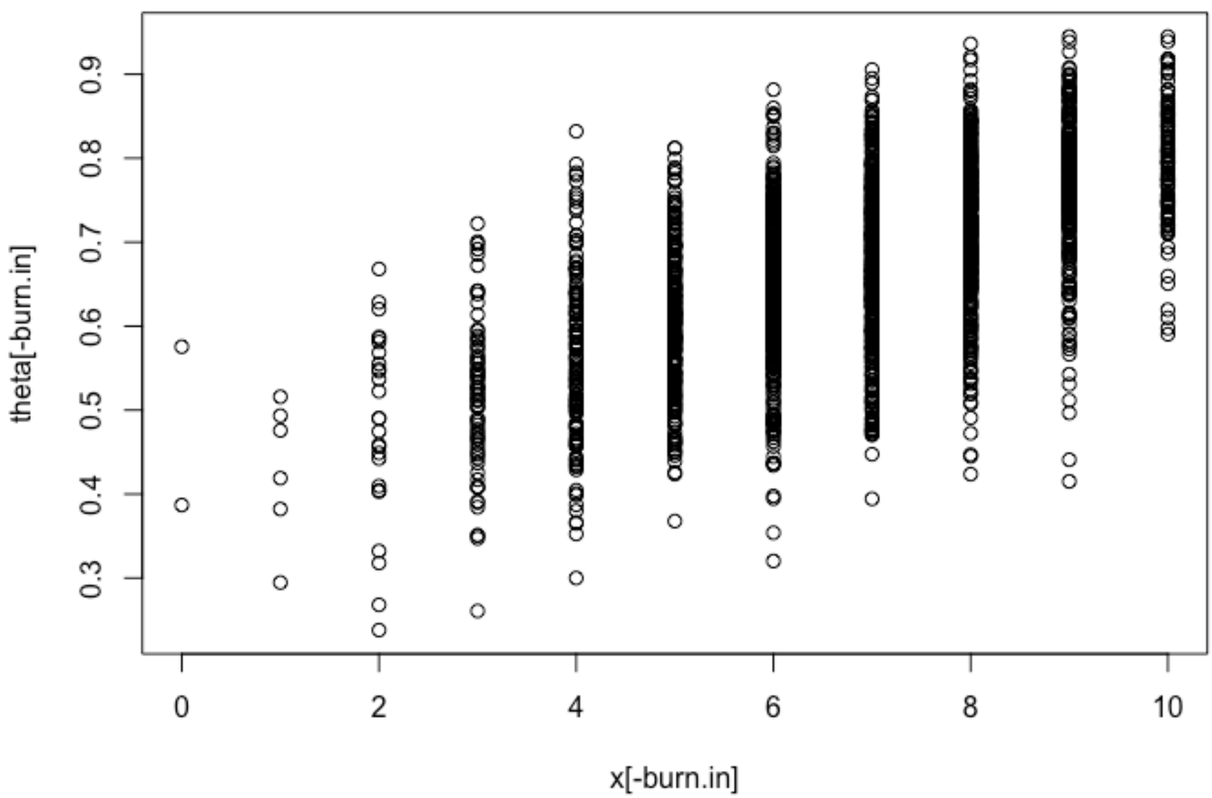
***b**. What is the marginal expected value of* X

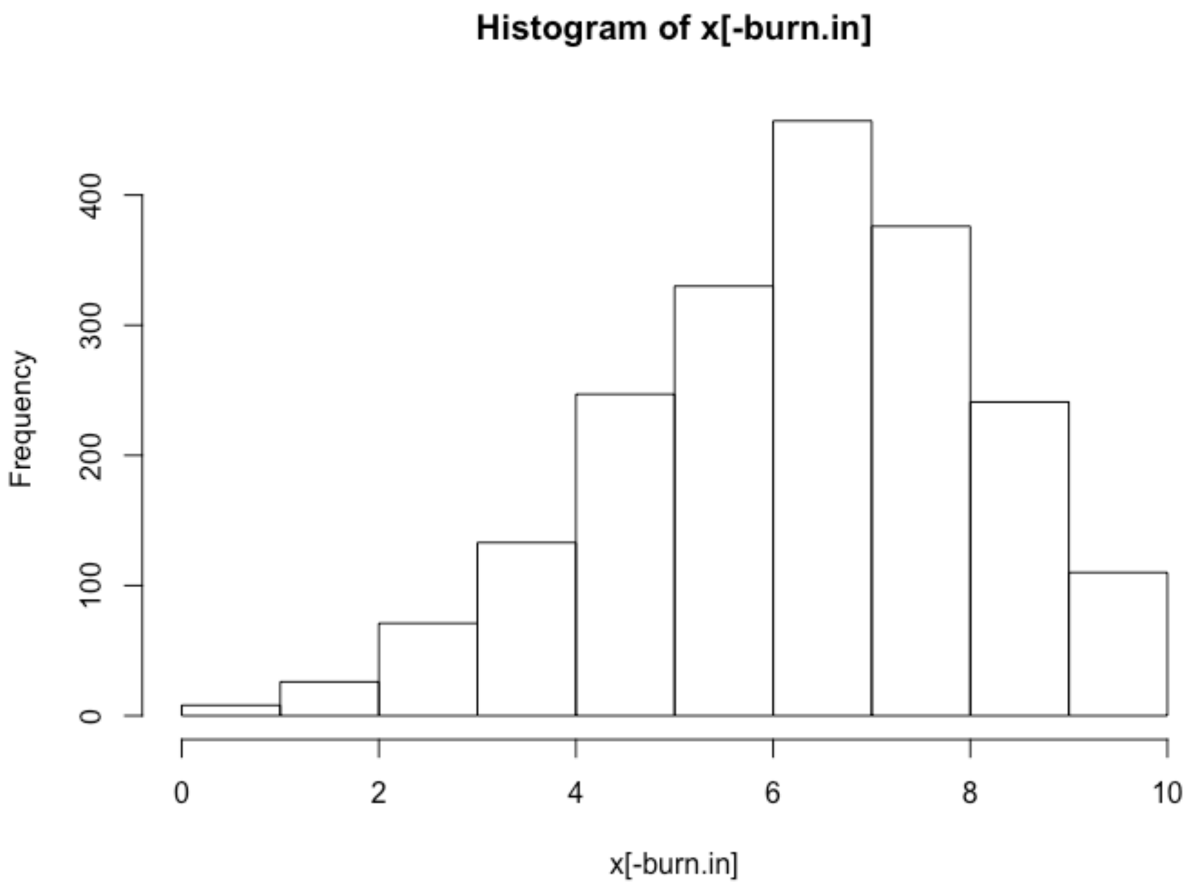
$$
E(X) = E[E(x|\theta)] = E(n\theta) = nE(\theta) = n\cdot\frac{\alpha}{\alpha+\beta} \tag{8}
$$

***c**. Implement a Gibbs sampler to obtain a joint sample of* (θ, X)*, using* $x^{(0)} = 0, \alpha = 10, \beta = 5$*, and* $n = 10$*.*

```r
set.seed(920804)
alpha = 10
beta =5
n = 10

pt.x = function (x){
  return(rbeta(1, alpha+x, n - x + beta))
}
px.t = function (t){
  return(rbinom(1, 10, t))
}

```

```r
iter     = 2000
burn.in = 100
xint     = 0
x0       = xint
x        = rep(0, iter)
theta    = rep(0, iter)

for (i in 1:iter){
  theta[i] = pt.x(x0)
  x[i]     = px.t(theta[i])
  x0       = x[i]
}

plot(x[-burn.in], theta[-burn.in])
hist(x[-burn.in])
```

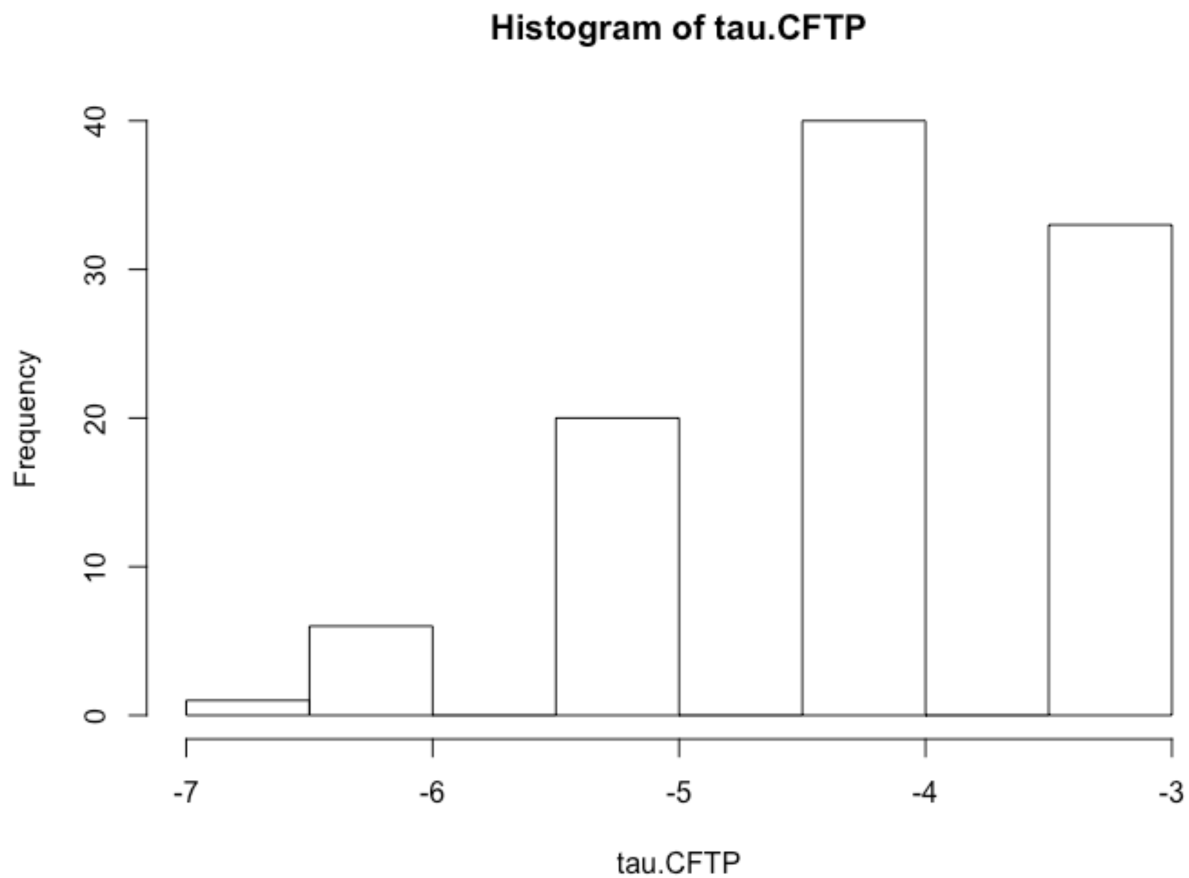


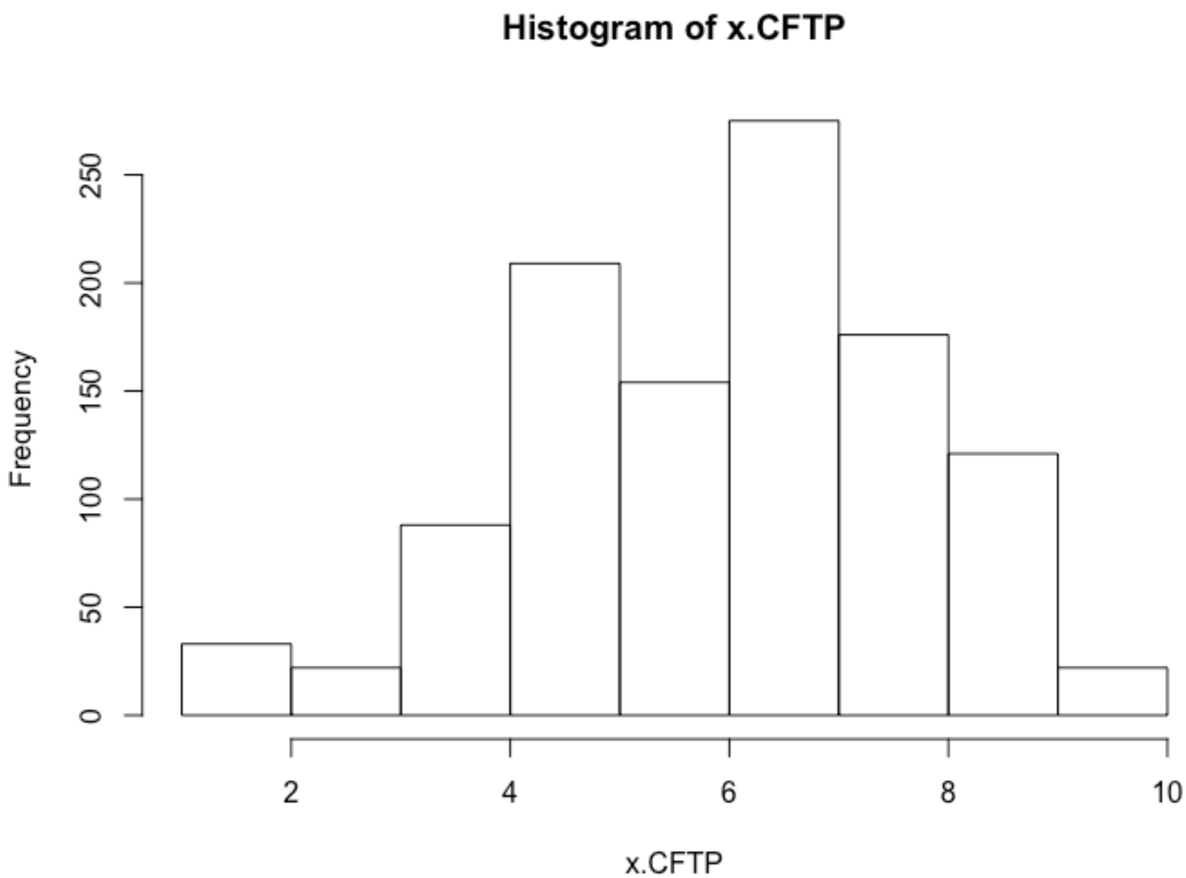


d. *Make a histogram of the 100 realizations of* $X^{(0)}$.

```
set.seed(920804)
alpha = 10
beta  = 5
n     = 10
q.xUV = function(U,V, x){
  res = NULL
  for (t in x){
    theta = qbeta(U, alpha+t, beta +n -t)
    res   = c(res, qbinom(V, n, theta))
  }
  return(res)
}
CFTP = function(){
  U = NULL
  V = NULL
  map2zero = rep(0, n+1)
  tau      = -1
  repeat{
    U0 = runif(1,0,1)
```

```
    V0 = runif(1,0,1)
    U = c(U0, U)
    V = c(V0, V)
    XtauPlus1 = q.xUV(U = U0,V = V0, 0:n)
    if (tau==-1){
      map2zero = XtauPlus1
    }else{
      map2zero = map2zero[XtauPlus1+1]
    }
    if (length(unique(map2zero)) == 1){
      break
    }else{
      tau = tau-1
    }
  }
  return(list(tau = tau, x = map2zero, U= U, V = V))
}
x.CFTP = NULL
tau.CFTP = NULL
for (i in 1:100){
  a        = CFTP()
  x.CFTP   = c(x.CFTP, a$x)
  tau.CFTP = c(tau.CFTP, a$tau)
}
hist(tau.CFTP)
hist(x.CFTP)
```

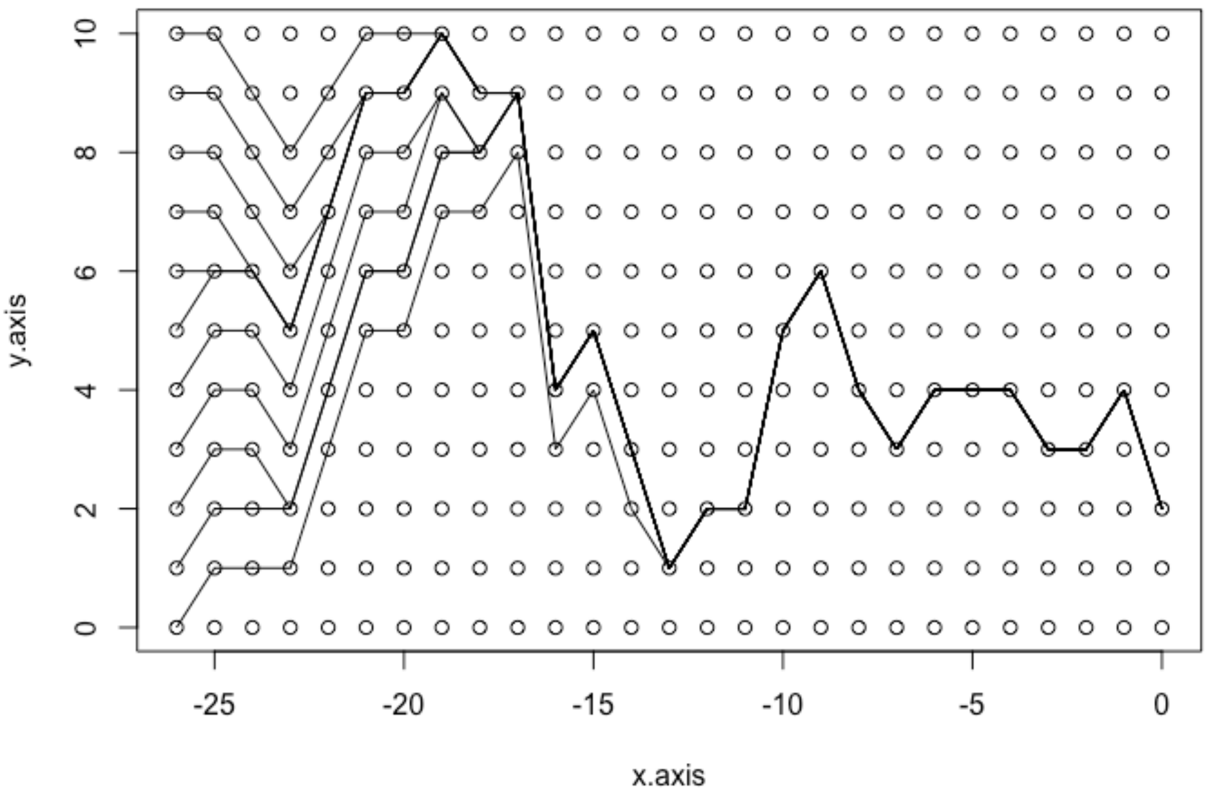

Histogram of tau.CFTP
Frequency
0
10
20
30
40
-7
-6
-5
-4
-3
tau.CFTP



e. *Run the function from part (d) several times for* $\alpha = 1.001, \beta = 1$*, and* $n = 10$*.*

```
set.seed(920804)
alpha = 1.001
beta  = 1
n     = 10
repeat{
  a = CFTP()
  if (a$tau <= -15){
    break
  }
}
x.axis = rep(a$tau:0,11)
y.axis = NULL
for (t in 0:10){
  y.axis = c(y.axis, rep(t, abs(a$tau)+1))
}
plot(x.axis,y.axis)
x.start = 0:n
for(i in 1:abs(a$tau)){
  x.end = q.xUV(U = a$U[i], V =a$V[i], x.start)
```

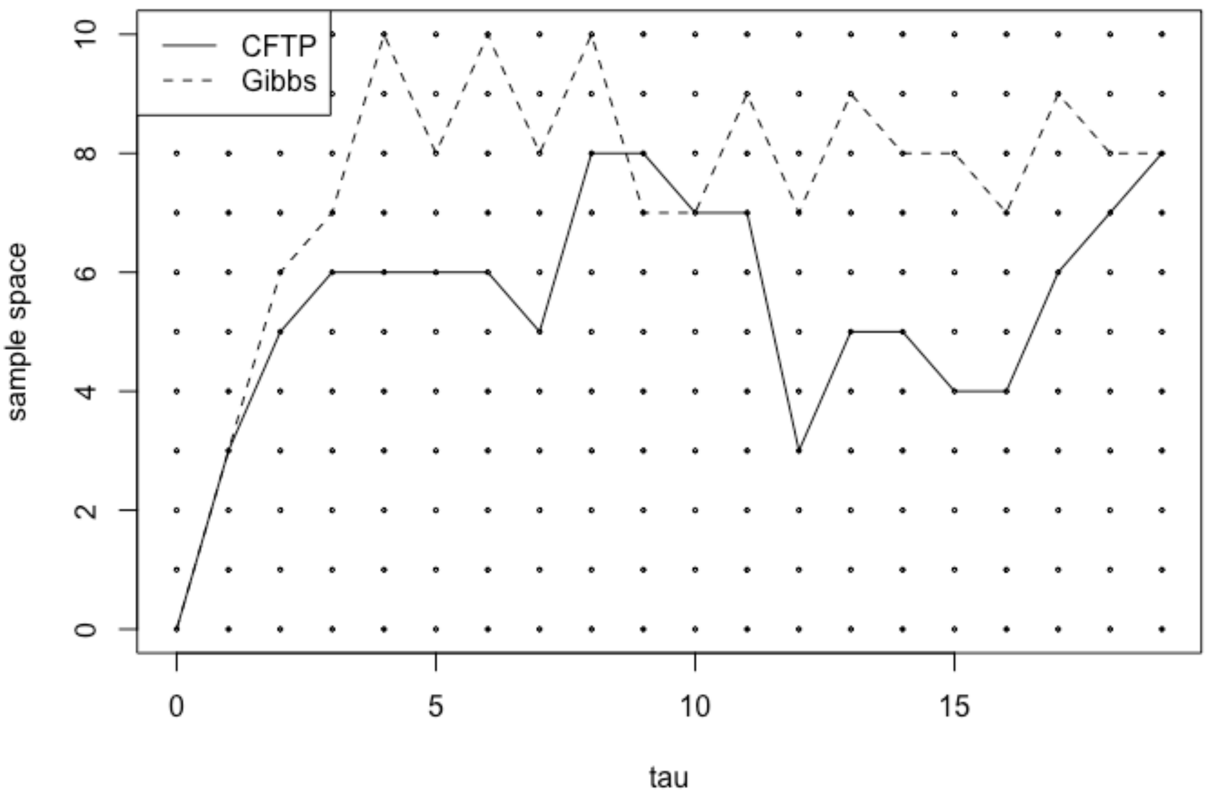
```r
  for (t in 1:(n+1)){
    segments(y0 = x.start[t], y1 = x.end[t], x0=a$tau +i -1, x1 = a$tau+i)
  }
  x.start = x.end
}
```



f. *Run the algorithm from part (d) several times.*

```r
a = 10
b = 5
n = 10
k = 11
set.seed(k)
xx = ss
U  = runif(2)
u  = U[1]
v  = U[2]
for(i in 1:(n+1)) xx[i] = q(ss[i], u, v)

while( length(table(xx)) != 1){
  U = rbind(U, runif(2))
```

```
  path = xx = ss
  for(tau in (dim(U)[1]:1) ){
    u = U[tau, 1]; v = U[tau, 2]
    for(i in 1:length(ss) ) xx[i] = q(xx[i], u, v)
    path = rbind(xx, path)
  }
}

theta = gibbs = perfect = rep(0, 20)
gibbs[1] = perfect[1] = 0
theta[1] = rbeta(1, a+gibbs[1], b+n-gibbs[1])

for(i in 2:20){
  u = runif(1)
  v = runif(1)
  perfect[i] = q(perfect[i-1], u, v)
  gibbs[i] = rbinom(1, n, prob = theta[i-1])
  theta[i] = rbeta(1, a+gibbs[i-1], b+n-gibbs[i-1])
}


plot(rep((0:19), 11),
     rep(rep(0:10), each = 20),
     cex = 0.3, xlab = "tau", ylab = "sample space" )
points(0:19, perfect, "l")
points(0:19, gibbs, lty = 2, "l")
legend("topleft", legend = c("CFTP", "Gibbs"), lty=1:2)
```



CFTP
Gibbs
sample space
tau
0
2
4
6
8
10
0
5
10
15