MATH 598: TOPICS IN STATISTICS

MCMC FOR HIERARCHICAL REGRESSION MODELS

Simple linear regression: We consider linear regression data from m individuals, where it is considered that the simple linear regression parameters (β_0, β_1) for each individual are different, thereby necessitating a hierarchical model: if $i = 1, \ldots, m$ indexes individuals in the study, and $j = 1, \ldots, n$ indexes the observations made on each individual, then

- STAGE 1: **Observed data:**

$$Y_{ij} \sim Normal(\beta_{i0} + \beta_{i1} x_{ij}, \sigma_i^2) \qquad i = 1, \ldots, m, j = 1, \ldots, n \text{ independently}$$

- STAGE 2: **Population model:** for $i = 1, \ldots, m$ independently, assume that

$$(\beta_{i0}, \beta_{i1}) \sim Normal_2(\mu, \Sigma) \qquad\qquad \sigma_i^2 \sim InverseGamma(a/2, b/2).$$

  where $\mu \in \mathbb{R}^2$ and Σ is a 2×2 positive definite matrix, and $a, b > 0$ are fixed scalars. Notice that this is not an identical specification to that used in the non-hierarchical version of the model.

- STAGE 3: **Prior model:** the prior model in this case is the **Normal-Inverse Wishart** model

$$\Sigma \sim InverseWishart(\nu, \Psi) \qquad\qquad \mu|\Sigma \sim Normal_2(\eta, \Sigma/\lambda)$$

  where $\eta \in \mathbb{R}^2$, $\nu > 1$ is a scalar and Ψ is a 2×2 positive definite matrix.

This prior is the standard conjugate prior for multivariate Normal problems. The Inverse Wishart prior is a conjugate prior for $d \times d$ covariance matrices. It has two hyperparameters:

- ν (a degrees of freedom parameter) which is a real-valued scalar such that $\nu > d - 1$; as ν increases;
- Ψ is a positive definite $d \times d$ "precision" matrix.

The expected value of this prior is

$$\frac{1}{\nu - d - 1}\Psi$$

provided $\nu > d + 1$, and the variance is finite if $\nu > d + 3$ – the variance decreases with ν^2.

MCMC: We now formulate a Gibbs sampler strategy:

1. For the **first stage** parameters conditional on the first stage parameters, we have a standard linear model, and the full conditional posteriors exhibit a conditional independent structure for $i = 1, \ldots, m$. However, the analysis is not quite the same as in the non-hierarchical case, and so we again use a Gibbs sampler structure. If $\mathbf{X}_i$ is the $n \times 2$ design matrix for the ith simple linear regression, $\mathbf{y}_i$ is the vector of response data, we have

   - $\beta_i|\sigma_i^2, \mu, \Sigma$: we have that the full conditional posterior is proportional to

$$\exp\left\{-\frac{1}{2\sigma_i^2}(\mathbf{y}_i - \mathbf{X}_i\beta_i)^\top(\mathbf{y}_i - \mathbf{X}_i\beta_i)\right\}\exp\left\{-\frac{1}{2}(\beta_i - \mu)^\top\Sigma^{-1}(\beta_i - \mu)\right\}$$

     and therefore

$$\pi_n(\beta_i|\sigma_i^2, \mu, \Sigma) \equiv Normal_2(\mathbf{m}_{ni}, \mathbf{M}_{ni})$$

     where

$$\mathbf{m}_{ni} = (\sigma_i^{-2}\mathbf{X}_i^\top\mathbf{X}_i + \Sigma^{-1})^{-1}(\sigma_i^{-2}\mathbf{X}_i^\top\mathbf{y}_i + \Sigma^{-1}\mu) \qquad\qquad \mathbf{M}_{ni} = (\sigma_i^{-2}\mathbf{X}_i^\top\mathbf{X}_i + \Sigma^{-1})^{-1}$$

   - $\sigma_i^2|\beta_i, \mu, \Sigma$: we have that the full conditional posterior is proportional to

$$\left(\frac{1}{\sigma_i^2}\right)^{n/2}\exp\left\{-\frac{1}{2\sigma_i^2}(\mathbf{y}_i - \mathbf{X}_i\beta_i)^\top(\mathbf{y}_i - \mathbf{X}_i\beta_i)\right\}\left(\frac{1}{\sigma_i^2}\right)^{a/2+1}\exp\left\{-\frac{b}{2}\right\}$$

     and then by inspection, we see that

$$\pi_n(\sigma_i^2|\beta_i, \mu, \Sigma) \equiv InverseGamma(a_n/2, b_{ni}/2)$$

     and

$$a_n = n + a \qquad b_{ni} = (\mathbf{y}_i - \mathbf{X}_i\beta_i)^\top(\mathbf{y}_i - \mathbf{X}_i\beta_i) + b$$

These two full conditional distributions can be sampled directly using rmvn from the mvnfast library, and the rgamma function (after reciprocation) respectively.

2. For the **second stage** parameters conditional on the first stage parameters, if $\beta_i = (\beta_{i0}, \beta_{i1})$ are regarded as known quantities, then independently for $i = 1, \ldots, n$

$$\beta_i|\mu, \Sigma \sim Normal_d(\mu, \Sigma).$$

We treat the β_i vectors as pseudo-data, and attempt to perform inference for the second stage "population" parameters. Here, the $NIW(\eta, \lambda, \nu, \Psi)$ prior is conjugate to the corresponding second-stage "likelihood", and it follows that the posterior distribution for μ and Σ is $NIW(\mu_n, \lambda_n, \nu_n, \Psi_n)$ where

$$\eta_n = \frac{m\overline{\beta} + \lambda\eta}{m + \lambda} \qquad \lambda_n = m + \lambda \qquad \nu_n = m + \nu$$

and

$$\Psi_n = \sum_{i=1}^{m}(\beta_i - \overline{\beta})(\beta_i - \overline{\beta})^\top + \frac{m\lambda}{m+\lambda}(\overline{\beta} - \eta)(\overline{\beta} - \eta)^\top + \Psi \qquad\qquad \overline{\beta} = \frac{1}{m}\sum_{i=1}^{m}\beta_i.$$

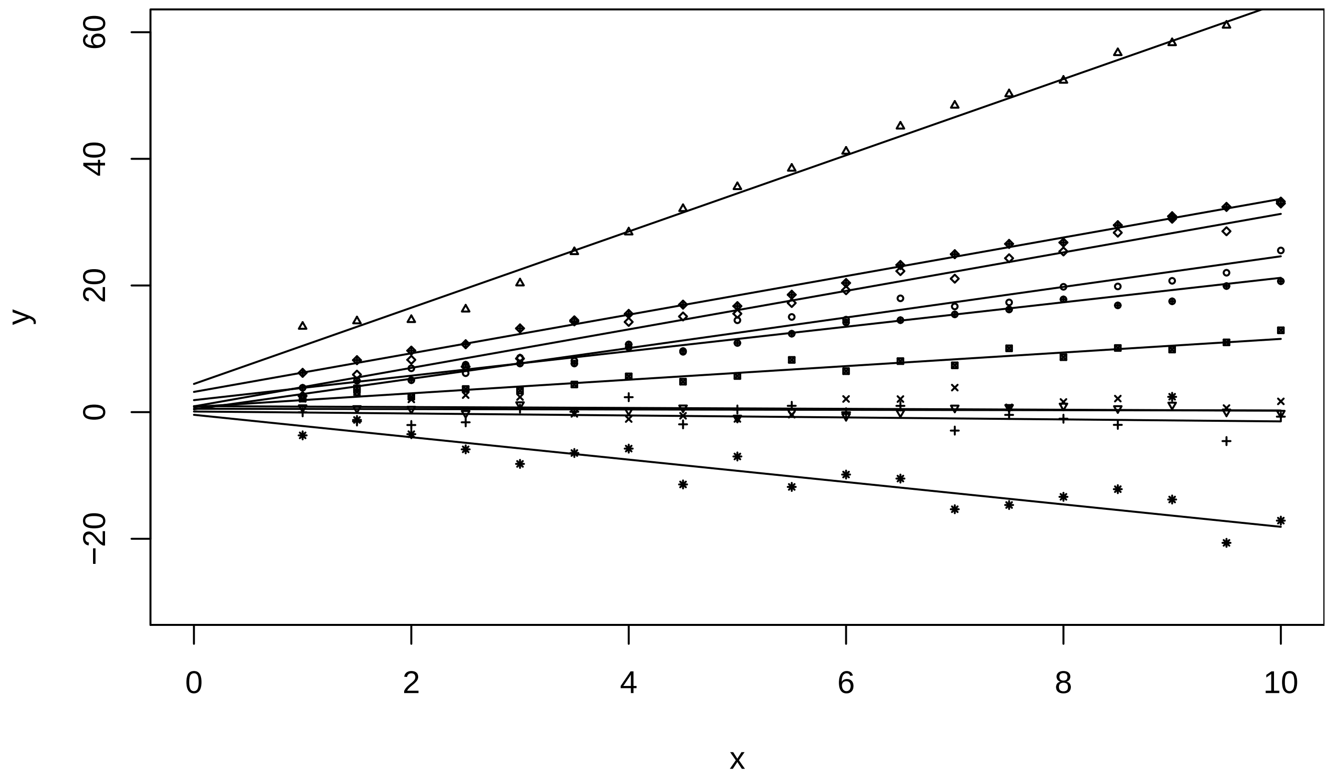
and η, ν and Ψ are hyperparameters. Then we have the results

$$\pi_n(\mu|\Sigma, -) \equiv Normal(\eta_n, \Sigma/\lambda_n) \qquad\qquad \pi_n(\Sigma|-) \equiv InverseWishart(\nu_n, \Psi_n).$$

These two full conditional distributions can be sampled directly using rmvn from the mvnfast library, and the riwish function from the MCMCpack library.

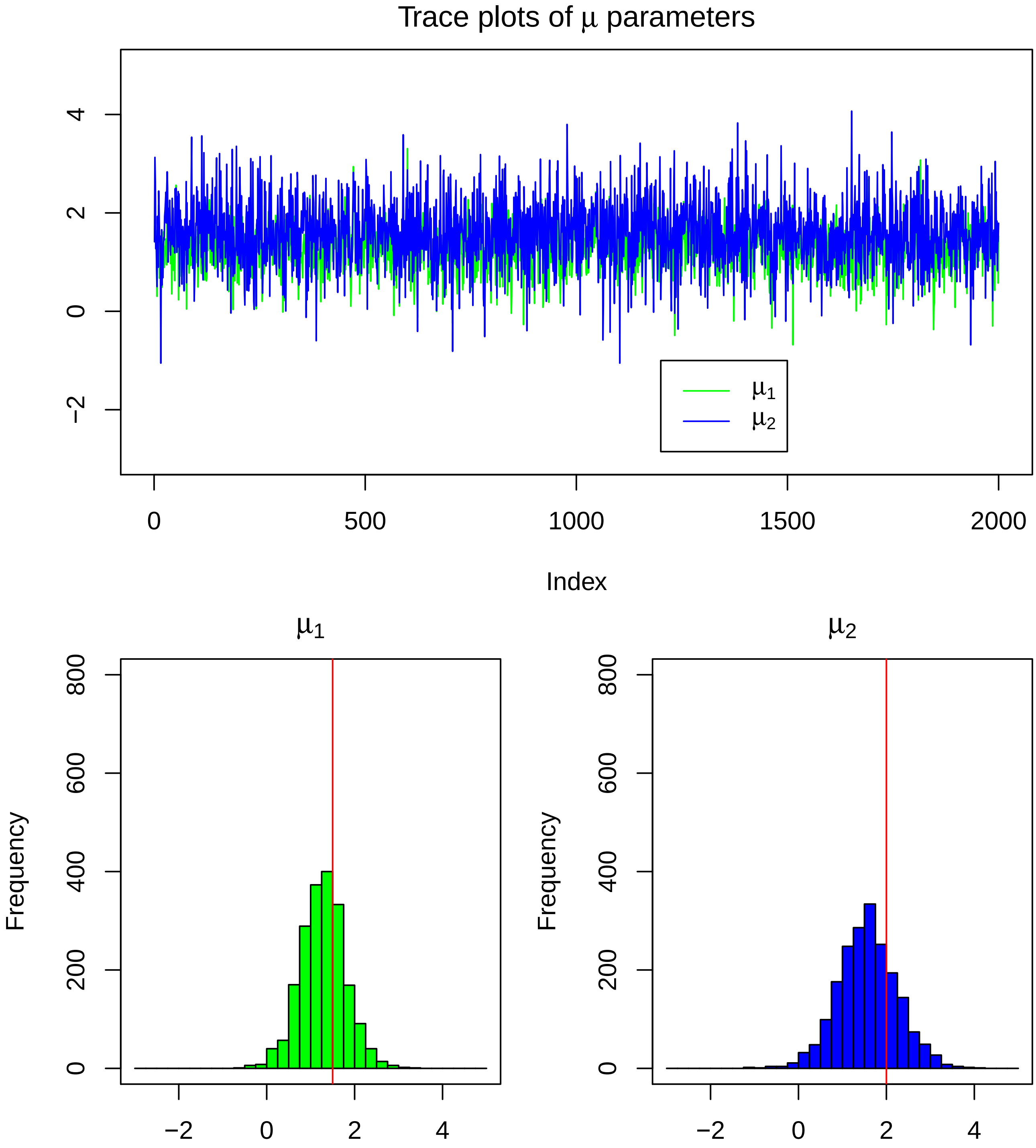
We examine the performance of the algorithm for the following simulated data.

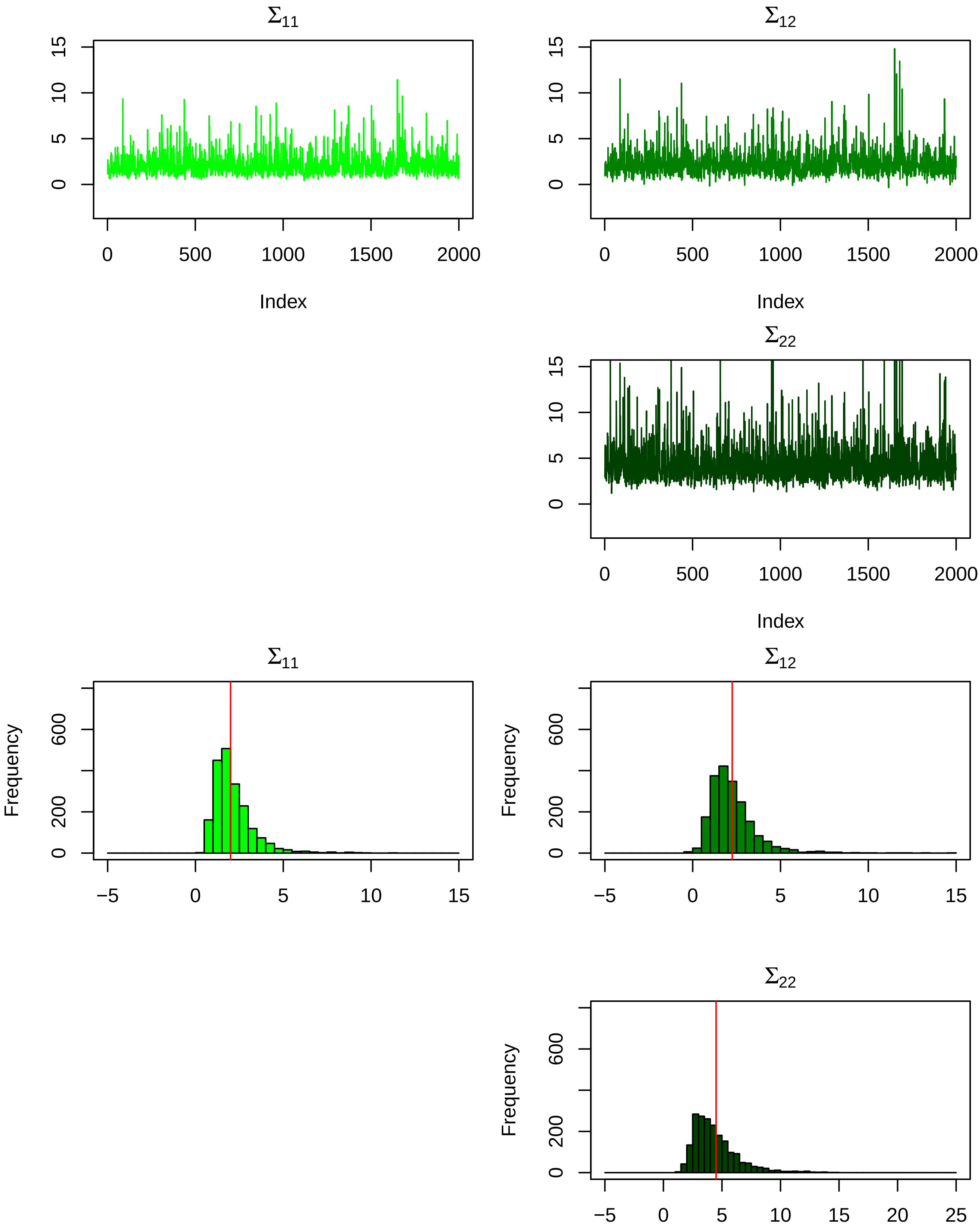
```
set.seed(2300)
#Simulation
library(mvnfast)
m<-10
mu<-c(1.5,2.0)
Sigma<-2*diag(c(1,1.5))%*% matrix(c(1,0.75,0.75,1),2,2)%*%diag(c(1,1.5))
be<-rmvn(m,mu,Sigma)
ysd<-sqrt(1/rgamma(m,2,2))

xv<-seq(0,10,by=0.01)
x<-seq(1,10.0,by=0.5)
n<-length(x)
par(mar=c(4,4,1,0))
plot(xv,xv,type='n',ylim=range(-30,60),xlab='x',ylab='y')
y<-matrix(0,nrow=m,ncol=n)
for(i in 1:m){
        muv<-be[i,1]+be[i,2]*x
        lines(xv,be[i,1]+be[i,2]*xv)
        y[i,]<-muv+rnorm(n)*ysd[i]
        points(x,y[i,],cex=0.4,pch=i)
}
```

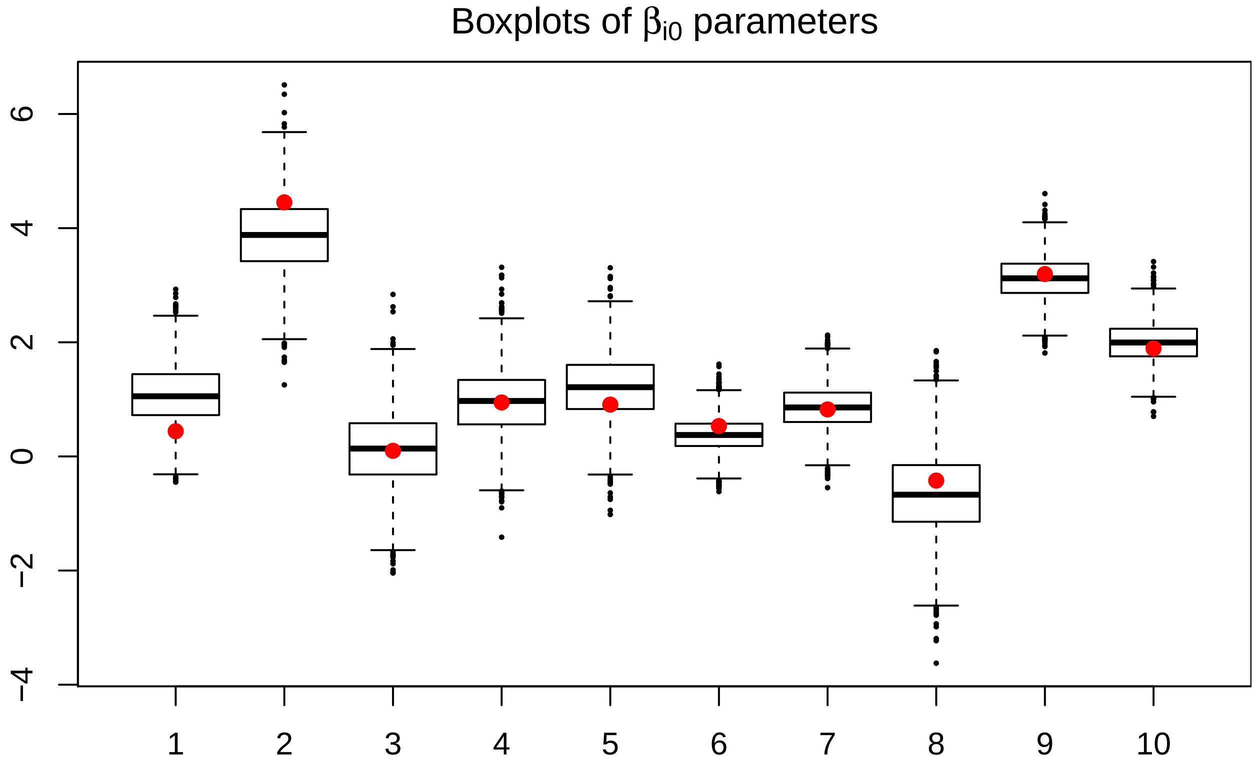


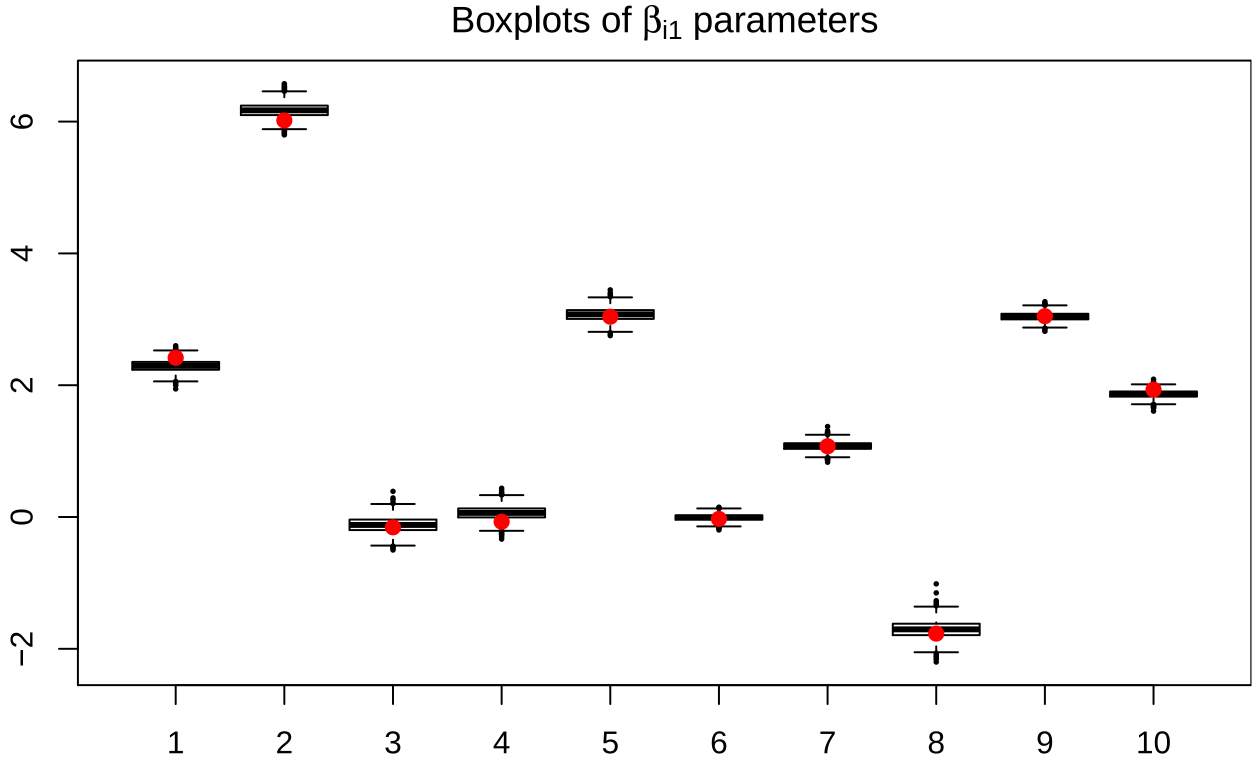
The Gibbs sampler proceeds as follows.

```r
library(MCMCpack)
X<-cbind(1,x)
XTX<-crossprod(X)
tXy<-(t(X) %*% t(y))

a<-b<-1
eta<-c(0,0)
lambda<-0.1
nu<-4
Psi<-nu*matrix(c(1,0,0,1),2,2)

#Starting values
old.beta<-matrix(0,nrow=m,ncol=2)
old.sigsq<-rep(0,m)
for(i in 1:m){
    fit0<-lm(y[i,]~x)
    old.beta[i,]<-coef(fit0)
    old.sigsq[i]<-summary(fit0)$sigma^2
}
old.mu<-apply(old.beta,2,mean)
old.Sigma<-var(old.beta)

nsamp<-2000; nburn<-50; nthin<-1
nits<-nburn+nsamp*nthin
ico<-0

beta.samp<-array(0,c(nsamp,m,2))
sigsq.samp<-matrix(0,nrow=nsamp,ncol=m)
mu.samp<-matrix(0,nrow=nsamp,ncol=2)
Sigma.samp<-array(0,c(nsamp,2,2))
for(iter in 1:nits){
    #Update the betas
    old.Siginv<-solve(old.Sigma)
    ssq<-rep(0,m)
    for(i in 1:m){
        M.ninv<-((XTX/old.sigsq[i])+old.Siginv)
        M.n<-solve(M.ninv)
        m.n<-M.n %*% (old.Siginv %*% old.mu + tXy[,i]/old.sigsq[i])
        old.beta[i,]<-rmvn(1,m.n,M.n)
        ssq[i]<-sum((y[i,]-X%*%old.beta[i,])^2)
    }

    #Update the sigmas
    a.n<-a+n
    b.n<-ssq+b
    old.sigsq<-1/rgamma(m,a.n/2,b.n/2)

    #Update Sigma
    nu.n<-nu+m
    beta.bar<-apply(old.beta,2,mean)
    Psi.n<-Psi+((m*lambda)/(m+lambda))*((beta.bar-eta) %*% t(beta.bar-eta))+var(old.beta)*(m-1)
    old.Sigma<-riwish(nu.n,Psi.n)

    #Update mu given Sigma
    lambda.n<-lambda+m
    eta.n<-(m*beta.bar+lambda*eta)/(m+lambda)
    old.mu<-matrix(rmvn(1,eta.n,old.Sigma/lambda.n),ncol=1)

    if(iter > nburn & iter %% nthin == 0){
        ico<-ico+1
        beta.samp[ico,,]<-old.beta
        sigsq.samp[ico,]<-old.sigsq
        mu.samp[ico,]<-old.mu
        Sigma.samp[ico,,]<-old.Sigma
    }

}
```


Trace plots of μ parameters
4
2
0
−2
0
500
1000
1500
2000
Index
μ_1
μ_2
μ_1
Frequency
800
600
400
200
0
−2
0
2
4
μ_2
Frequency
800
600
400
200
0
−2
0
2
4


Σ_{11}
15
10
5
0
0
500
1000
1500
2000
Index
Σ_{12}
15
10
5
0
0
500
1000
1500
2000
Index
Σ_{22}
15
10
5
0
0
500
1000
1500
2000
Index
Σ_{11}
Frequency
600
200
0
−5
0
5
10
15
Σ_{12}
Frequency
600
200
0
−5
0
5
10
15
Σ_{22}
Frequency
600
200
0
−5
0
5
10
15
20
25


Boxplots of β_{i0} parameters
6
4
2
0
−2
−4
1
2
3
4
5
6
7
8
9
10


Boxplots of β_{i1} parameters
6
4
2
0
−2
1
2
3
4
5
6
7
8
9
10