Computational Statistics. Chapter 5: MCMC. Solution of exercises

Thierry Denoeux

2024-03-19

```
set.seed(2021)
```

Exercise 1

As the density of ϵ is symmetric, the MH ratio is the ratio of the densities at x^* and $x^{(t-1)}$, i.e., we have

$$R(x^{(t-1)}, x^*) = \frac{f(x^*)}{f(x^{(t-1)})} = \exp(|x^{(t-1)}| - |x^*|).$$

The following function `MH_Laplace` implements the random walk MH algorithm for this problem:

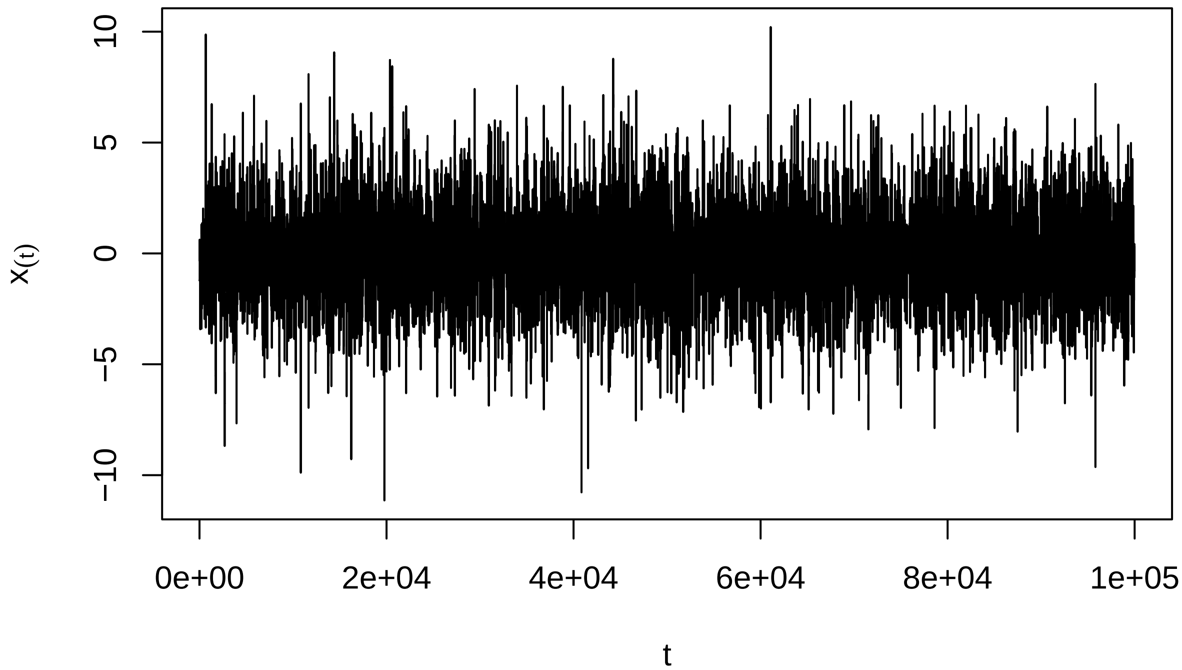
```
MH_Laplace <- function(N,sig){
  x<-vector(N,mode="numeric")
  x[1]<-rnorm(1,mean=0,sd=sig)
  for(t in (2:N)){
    epsilon<-rnorm(1,mean=0,sd=sig)
    xstar<-x[t-1]+ epsilon
    U<-runif(1)
    R<-exp(abs(x[t-1]) - abs(xstar))
    if(U <= R) x[t]<-xstar else x[t]<-x[t-1]
  }
  return(x)
}
```

Let us generate a sample of size 10^5 with $\sigma = 10$:

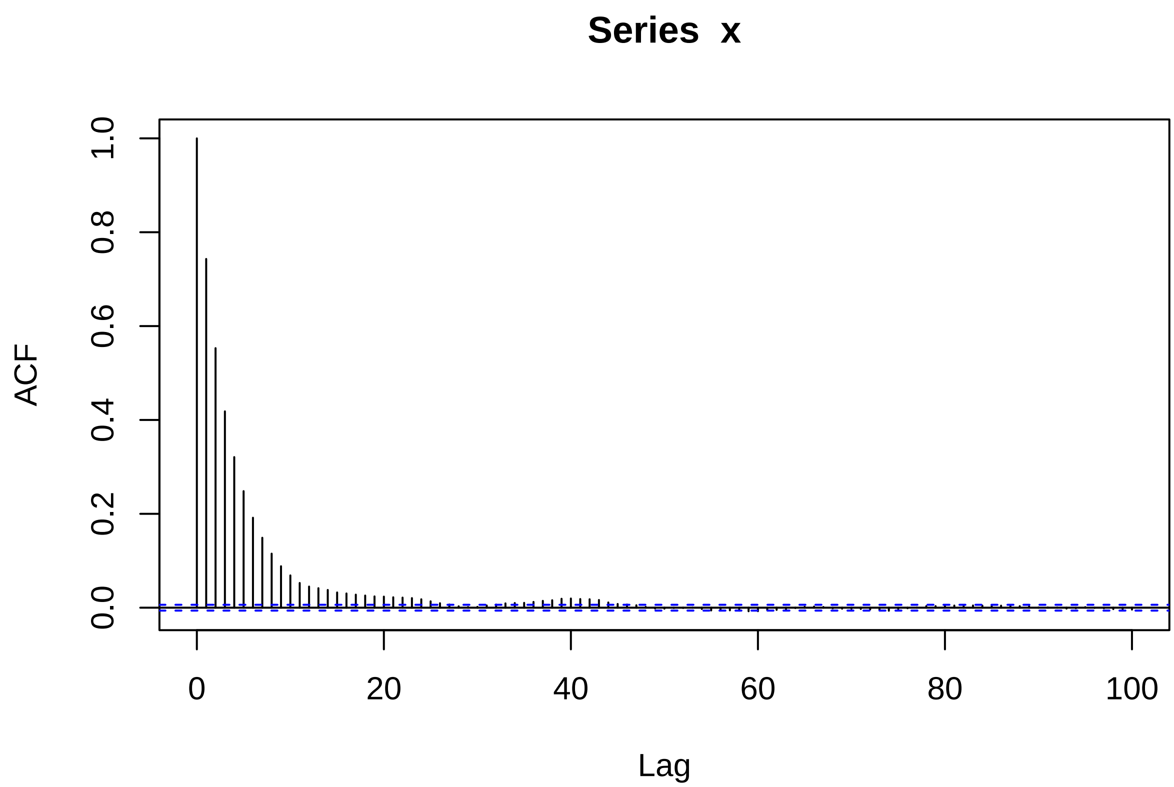
```
x <- MH_Laplace(100000,10)
```

The sample path and correlation plots show good mixing (the chain quickly moves away from its starting value, and the autocorrelation decreases quickly as the lag between iterations increases):

```
plot(x,type="l",xlab='t',ylab=expression(x[(t)]))
```

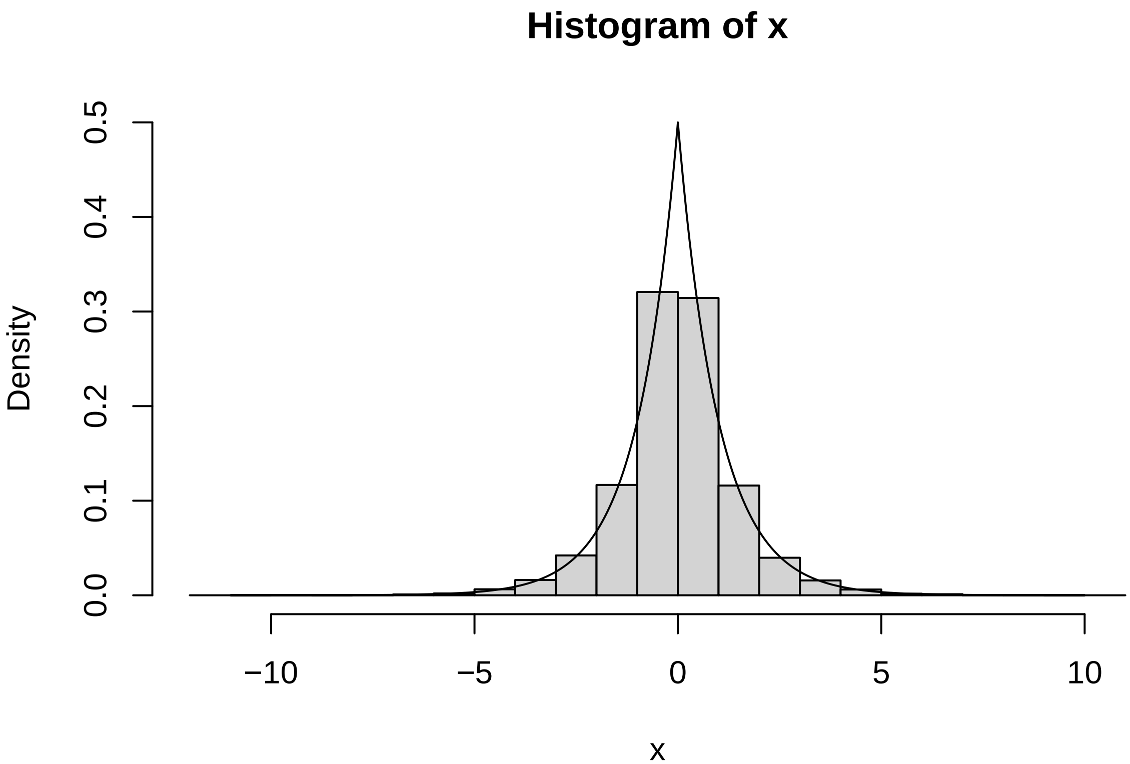



```
acf(x,lag.max=100)
```



Plot of the histogram with the Laplace density:

```
u<-seq(-10,10,0.01)
fu<-0.5*exp(-abs(u))
hist(x,freq=FALSE,ylim=range(fu))
lines(u,0.5*exp(-abs(u)))
```

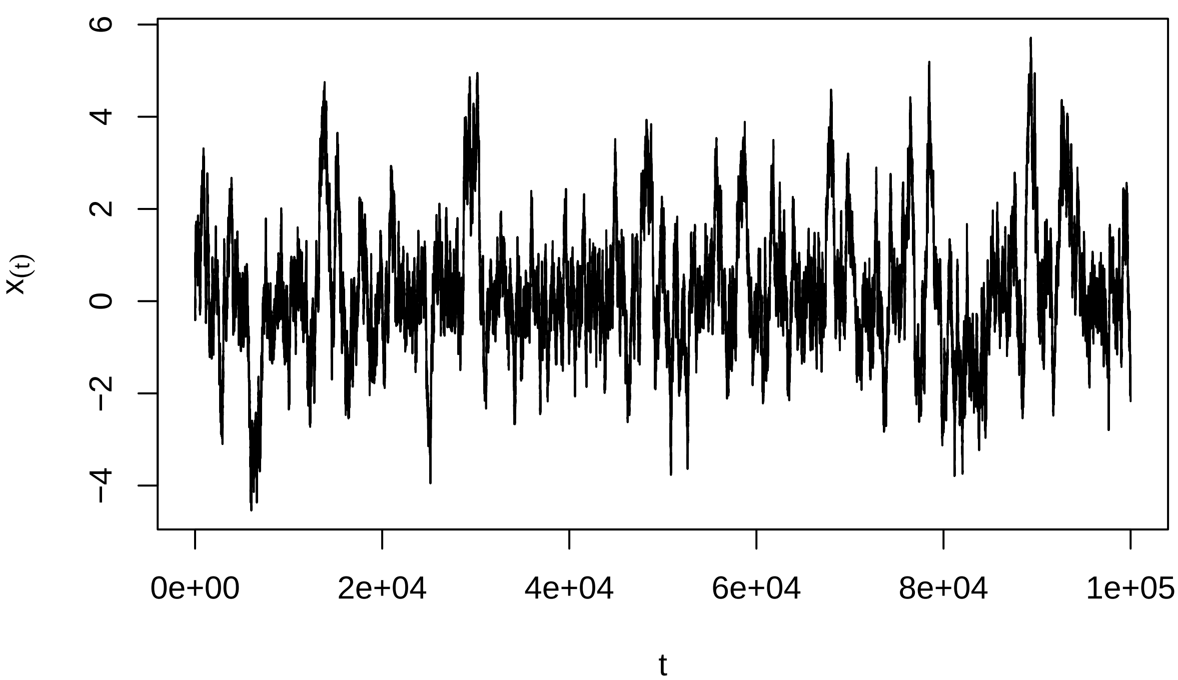


Let us now generate another sample of the same size, this time with $\sigma = 0.1$:

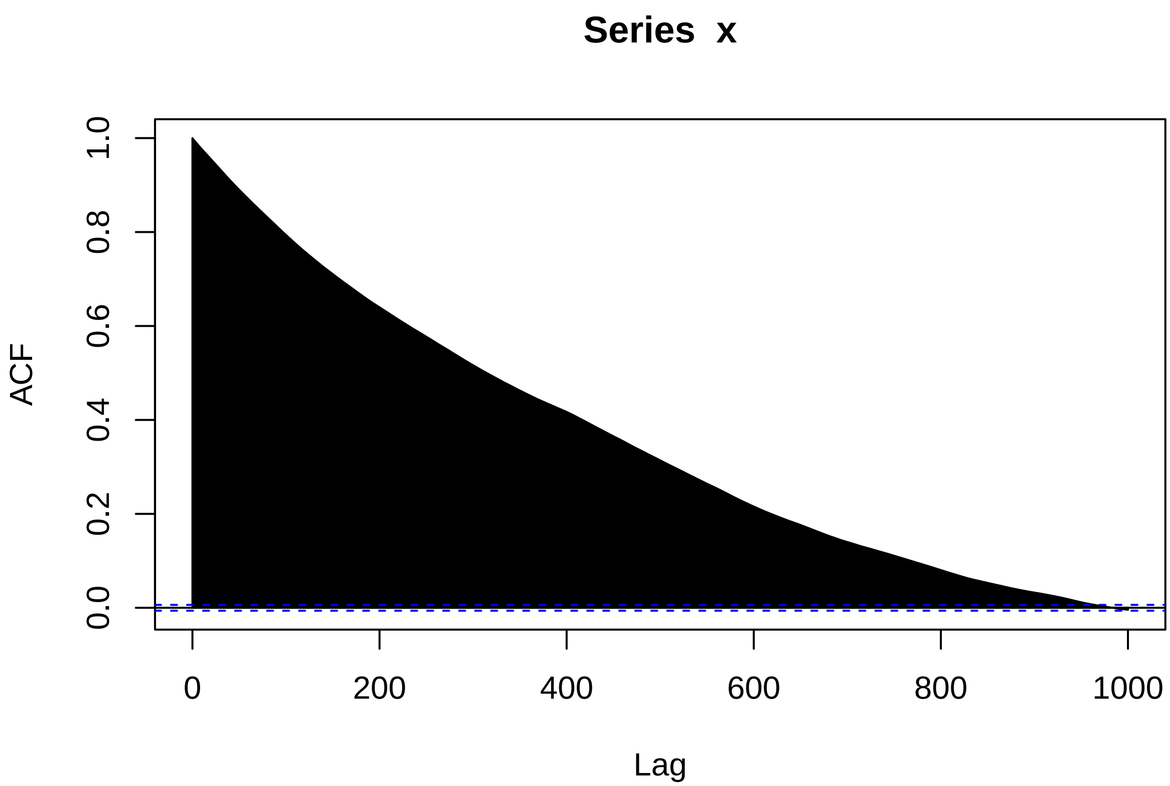
```
x<-MH_Laplace(100000,0.1)
```

This time, the sample path and correlation plots show poor mixing (the chain remains at or near the same value for many iterations, and the autocorrelation decays very slowly):

```
plot(x,type="l",xlab='t',ylab=expression(x[(t)]))
```



```r
acf(x,lag.max=1000)
```



Exercise 2

Question a

The likelihood function is

$$L(\beta; y_1, \ldots, y_n) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}(y_i - \beta x_i)^2\right) = (2\pi)^{-n/2} \exp\left(-\frac{1}{2}\sum_{i=1}^{n}(y_i - \beta x_i)^2\right).$$

The density of the Gamma distribution with shape parameter a and rate b is $f(\beta) \propto \beta^{a-1} \exp(-b\beta) I(\beta > 0)$. Here $a = 2$ and $b = 1$, so $f(\beta) \propto \beta \exp(-\beta) I(\beta > 0)$. Consequently, the posterior density is

$$f(\beta \mid y_1, \ldots, y_n) \propto \beta \exp(-\beta) \exp\left(-\frac{1}{2}\sum_{i=1}^{n}(y_i - \beta x_i)^2\right) I(\beta > 0).$$

Question b

We first write a function that computes the log-likelihood:

```r
loglik <- function(beta,x,y){
    n<- length(x)
    return(-0.5 * sum((y-beta*x)^2) - n/2*log(2*pi))
}
```

We then write a function that generates a MC of size N for a given data set:

```r
gen_MH<-function(x,y,N){
  beta <- vector(N,mode="numeric")
  beta[1] <- rgamma(1,shape=2,rate=1)
  for(t in (2:N)){
    beta_star <- rgamma(1,shape=2,rate=1)
    u <- runif(1)
    logR <- loglik(beta_star,x,y)-loglik(beta[t-1],x,y)
    if( log(u) <= logR ) beta[t] <- beta_star   else beta[t]<- beta[t-1]
  }
  return(beta)
}
```

Question c

Data generation:

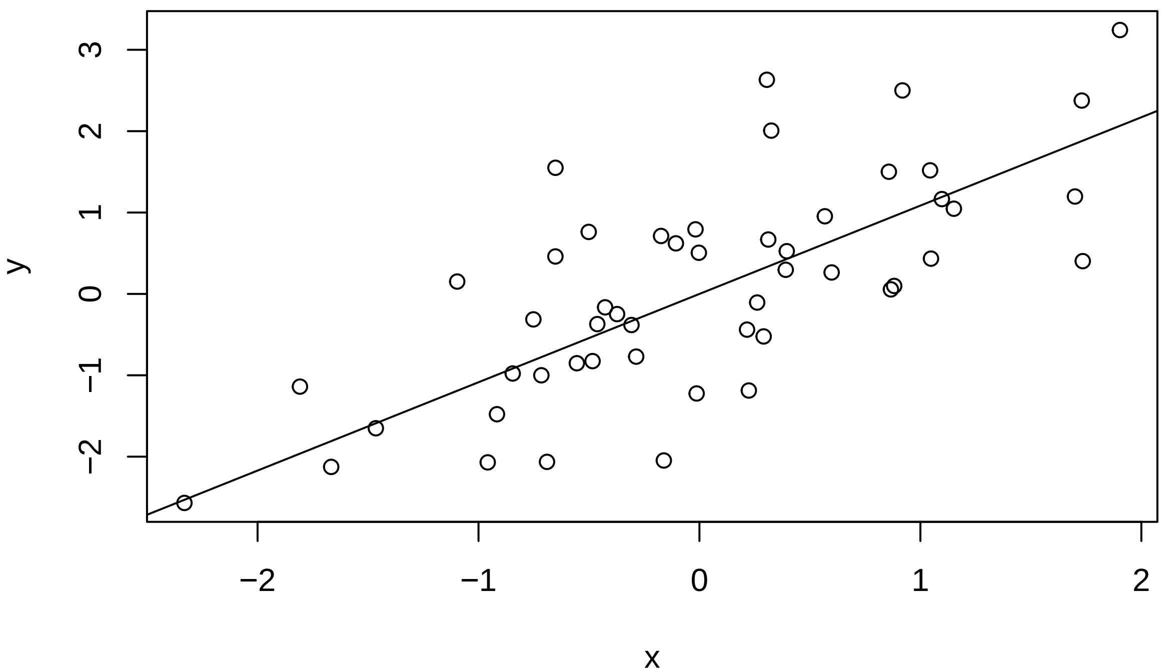
```r
beta0<- rgamma(1,shape=2,rate=1) # Generation of beta
cat(beta0)
```

```
## 1.085444
```

```r
n<-50
x<-rnorm(n)
y<-x*beta0+rnorm(n)
```

Plot of the data:

```r
plot(x,y)
abline(0,beta0)
```



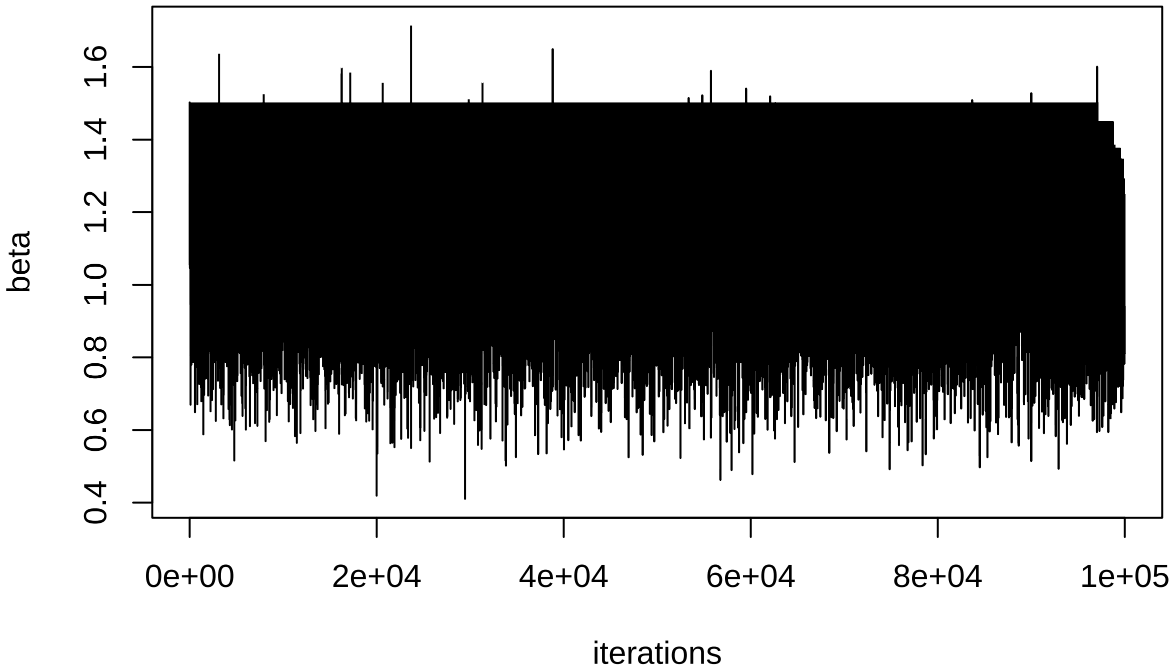
Running the MH algorithm:

```r
N <- 100000
beta <- gen_MH(x,y,N)
```

Question d

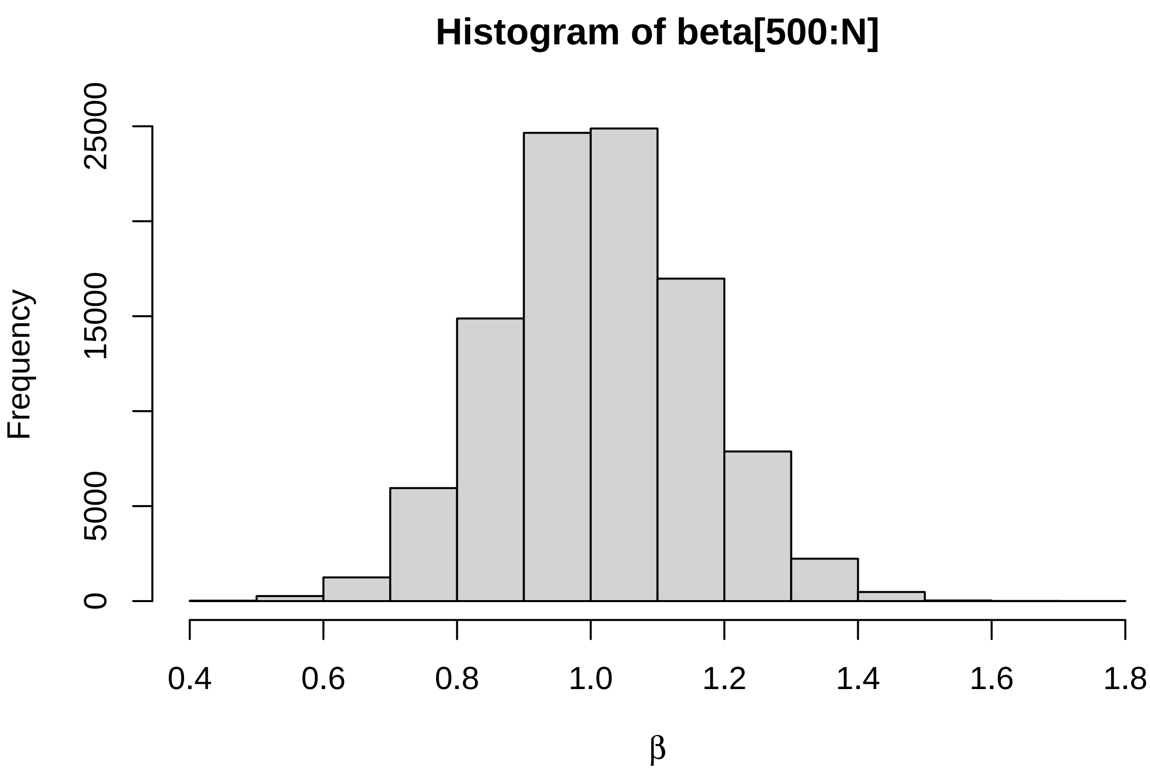
Sample path:

```
plot(beta,type="l",xlab="iterations")
```



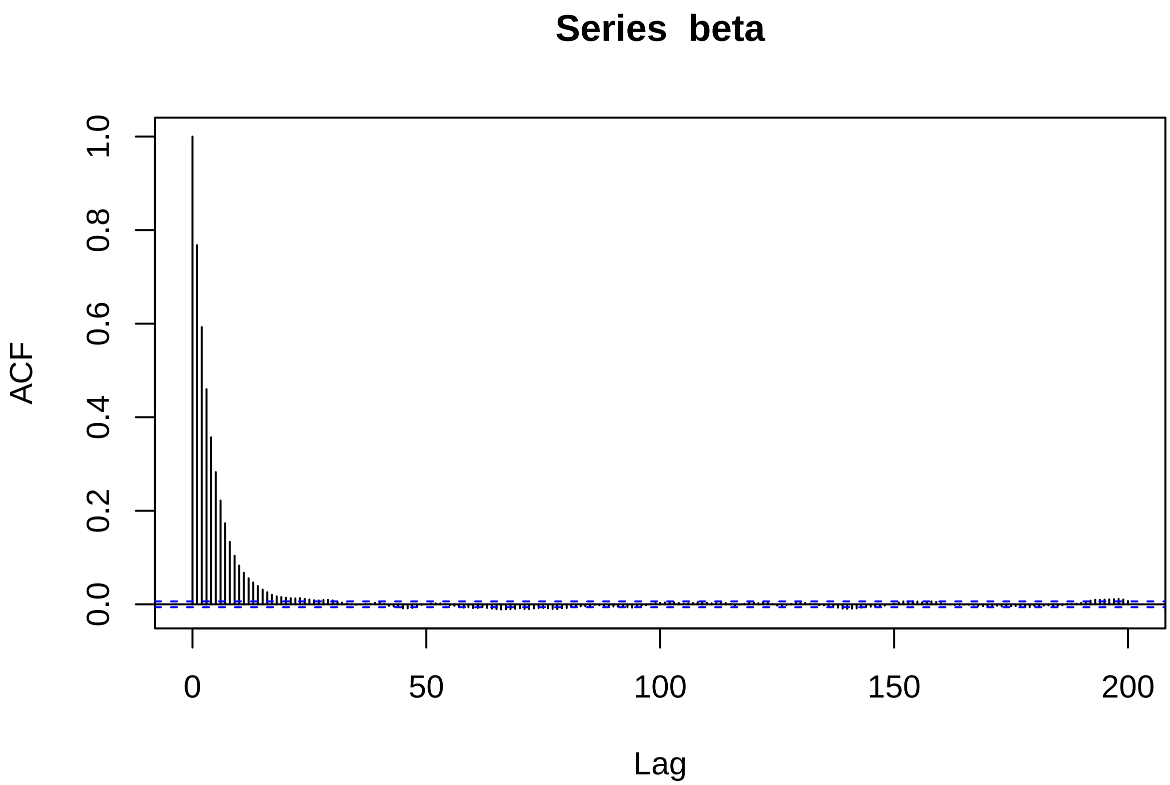
Histogram (leaving out the first 500 values):

```
hist(beta[500:N],xlab=expression(beta))
```



Autocorrelation plot:

```
acf(beta,lag.max=200)
```



Question e

We use the batch means method. We first determine the lag k_0 such that the autocorrelation is small enough to be neglected:

```
ACF<-acf(beta,lag.max=200,plot=FALSE)
k0<-ACF$lag[min(which(abs(ACF$acf)<0.01))]
cat(k0)
```

```
## 26
```

We fix the burn-in period and we compute the number of batches:

```
D<-1000 # burn in
B<-floor((N-D)/k0)
```

We compute the means within each block:

```
Z<-vector(B,mode="numeric")
for(b in (1:B)) Z[b]<-mean(beta[(D+(b-1)*k0+1):(D+b*k0)])
```

The estimated simulation standard error is the standard deviation of the batch means divided by the square root of the number of batches:

```
se <- sd(Z)/sqrt(B)
```

Estimated posterior expectation of β and simulation standard error:

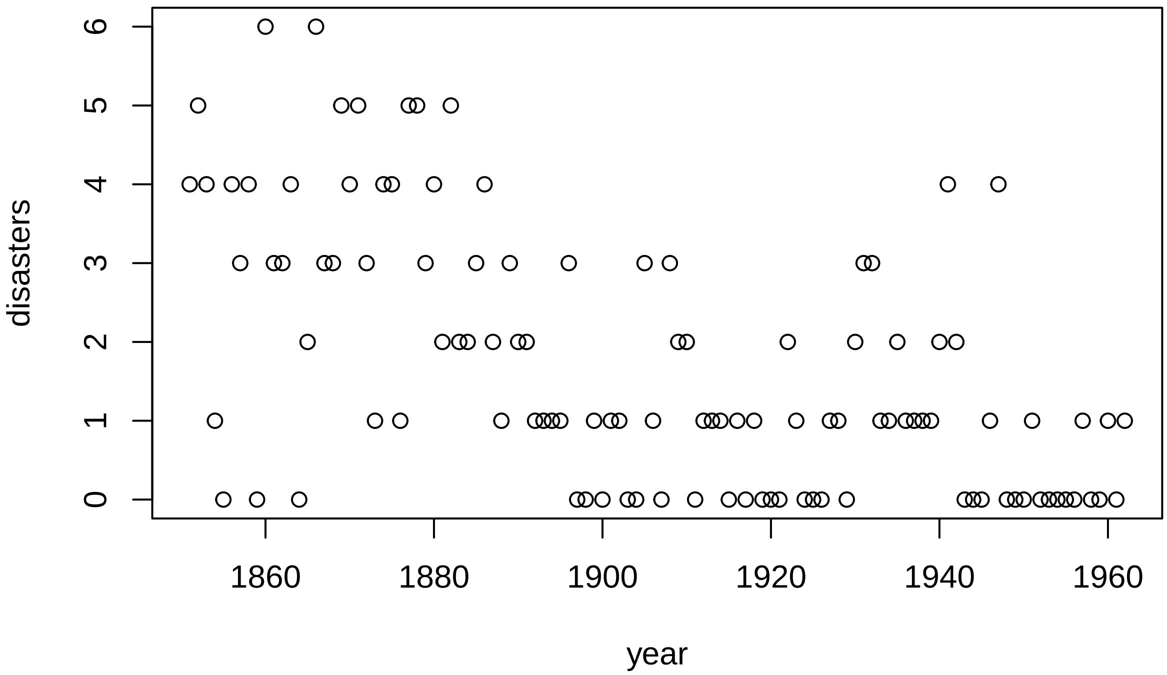
```r
print(c(mean(beta[(D+1):N]),se),3)
```

```
## [1] 1.01312 0.00123
```

Exercise 3

Question a

```r
coal <- read.table("/Users/Thierry/Documents/R/Data/Compstat/coal.dat",header=TRUE)
plot(coal)
```



Question b

The likelihood function is

$$L(\theta_1, \theta_2, k \mid \mathbf{x}) \propto \prod_{i=1}^{k} e^{-\theta_1}\theta_1^{x_i} \prod_{i=k+1}^{n} e^{-\theta_2}\theta_2^{x_i}.$$

We obtain the posterior distribution by multiplying the likelihood and the prior:

$$f(\theta_1, \theta_2, k|\mathbf{x}) \propto \underbrace{\theta_1^{\alpha_{01}-1}e^{-\beta_{01}\theta_1}}_{f(\theta_1)} \underbrace{\theta_2^{\alpha_{02}-1}e^{-\beta_{02}\theta_2}}_{f(\theta_2)} \underbrace{\prod_{i=1}^{k} e^{-\theta_1}\theta_1^{x_i} \prod_{i=k+1}^{n} e^{-\theta_2}\theta_2^{x_i}}_{L(\theta_1,\theta_2,k|\mathbf{x})}.$$

Now,

$$
\begin{aligned}
f(\theta_1 \mid \theta_2, k, \mathbf{x}) &\propto \frac{f(\theta_1, \theta_2, k \mid \mathbf{x})}{f(\theta_2, k)} \\
&\propto L(\theta_1, \theta_2, k \mid \mathbf{x}) f(\theta_1) \\
&\propto \theta_1^{\alpha_{01}-1} e^{-\beta_{01}\theta_1} \prod_{i=1}^{k} e^{-\theta_1} \theta_1^{x_i} \prod_{i=k+1}^{n} e^{-\theta_2} \theta_2^{x_i} \\
&\propto \theta_1^{\alpha_{01}+\sum_{i=1}^{k} x_i - 1} \exp\left(-(\beta_{01}+k)\theta_1\right).
\end{aligned}
$$

Consequently,

$$
f(\theta_1 \mid \theta_2, k, \mathbf{x}) = f(\theta_1 \mid k, \mathbf{x}) \sim G(\alpha_{01} + \sum_{i=1}^{k} x_i, \beta_{01} + k).
$$

Symmetrically, we obtain in the same way

$$
f(\theta_2 \mid \theta_1, k, \mathbf{x}) = f(\theta_2 \mid k, \mathbf{x}) \sim G(\alpha_{02} + \sum_{i=k+1}^{n} x_i, \beta_{02} + k).
$$

We can observe that θ_1 and θ_2 are conditionally independent given k and $\mathbf{x}$.

Finally, the conditional probability mass function of k is

$$
\begin{aligned}
f(k \mid \theta_1, \theta_2, \mathbf{x}) &\propto \frac{f(\theta_1, \theta_2, k \mid \mathbf{x})}{f(\theta_1) f(\theta_2)} \\
&\propto L(\theta_1, \theta_2, k \mid \mathbf{x}) \\
&\propto \exp\left[k(\theta_2 - \theta_1)\right] \left(\frac{\theta_1}{\theta_2}\right)^{\sum_{i=1}^{k} x_i}.
\end{aligned}
$$

Question c

The following function implements the Gibbs algorithm for this problem:

```r
gibbs<-function(x,N,alpha10,beta10,alpha20,beta20){
  n<-length(x)
  # Initialization
  theta1 <- vector(length=N,mode="numeric")
  theta2 <- vector(length=N,mode="numeric")
  k <- vector(length=N,mode="numeric")
  p<-vector(length=n,mode="numeric")
  # First cycle
  # Sampling of k[1] from a uniform distribution
  k[1]<-sample(n,size=1)
  theta1[1]<-rgamma(1,shape=alpha10+sum(x[1:k[1]]),rate=beta10+k[1])
  theta2[1]<-rgamma(1,shape=alpha20+sum(x[(k[1]+1):n]),rate=beta20+n-k[1])
  for(t in (2:N)){
    # Conditional pmf of k
    for (j in (1:n)){
      p[j]<- (theta1[t-1]/theta2[t-1])^sum(x[1:j]) * exp(j*(theta2[t-1]-theta1[t-1]))
    }
    p<-p/sum(p)
    k[t]<- sample(n,size=1,prob=p)
    theta1[t]<-rgamma(1,shape=alpha10+sum(x[1:k[t]]),rate=beta10+k[t])
```

```r
    theta2[t]<-rgamma(1,shape=alpha20+sum(x[(k[t]+1):n]),rate=beta20+n-k[t])
  }
  return(list(k=k,theta1=theta1,theta2=theta2))
}
```

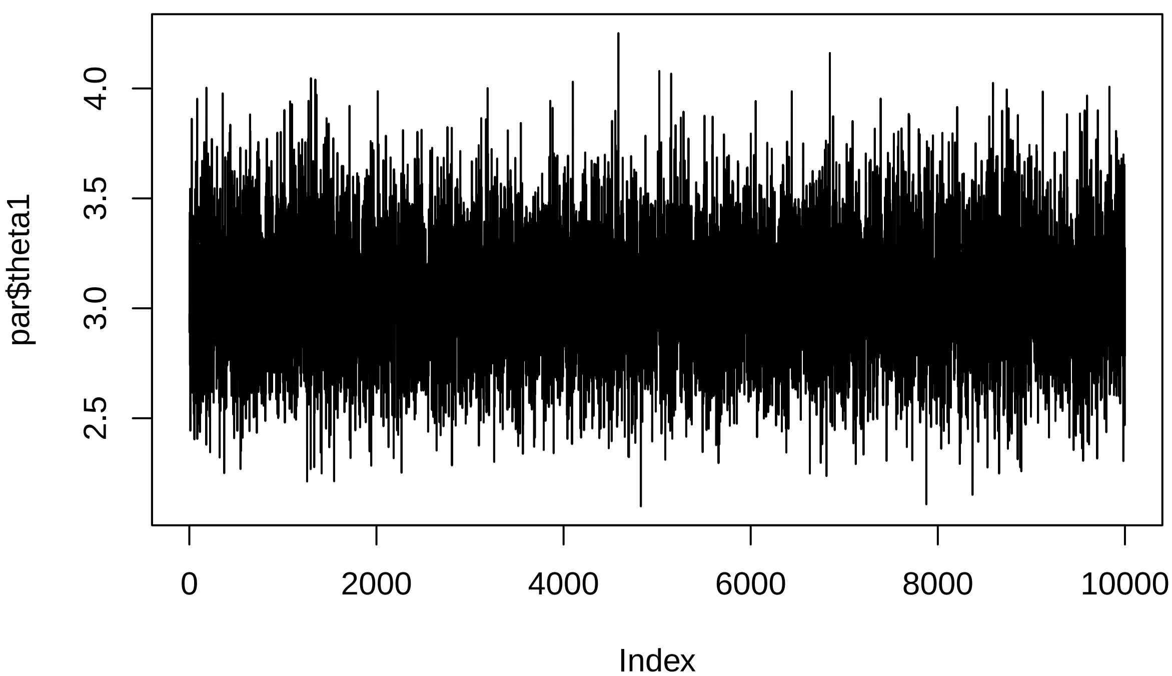
We can run this algorithm on the data:

```r
N<-10000
alpha10<-0.5
alpha20<-0.5
beta10<-1
beta20<-1
par<-gibbs(x=coal$disasters,N,alpha10,beta10,alpha20,beta20)
```

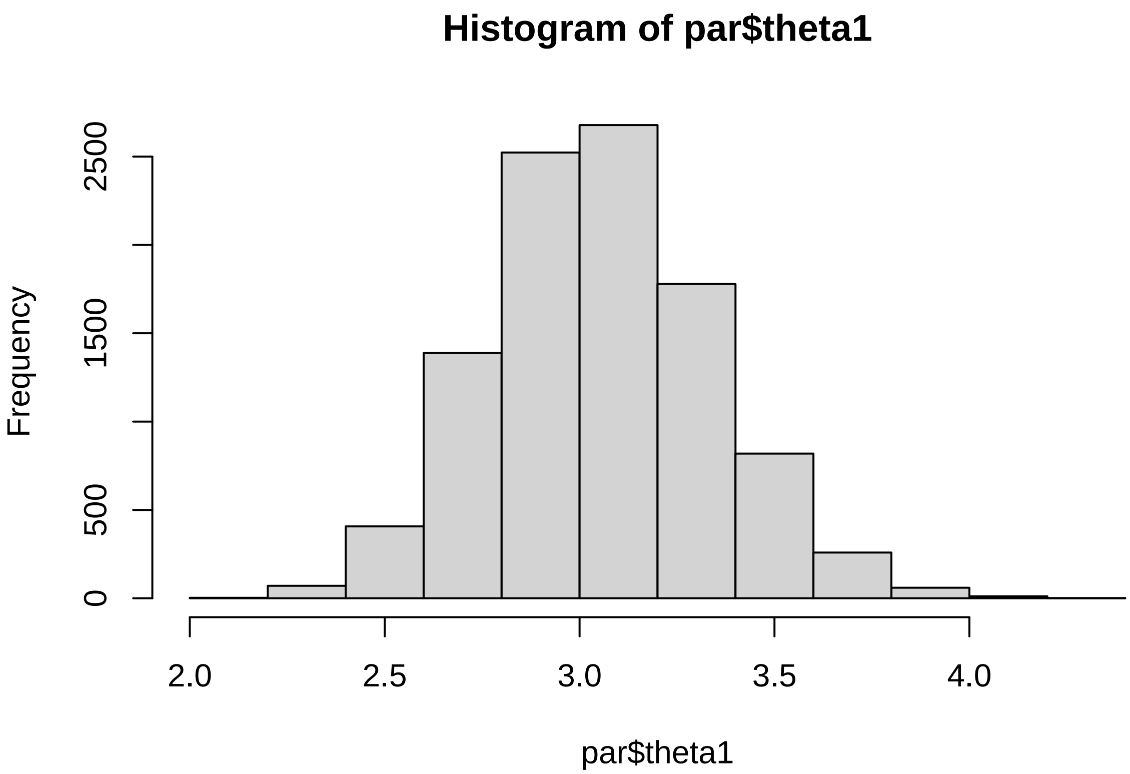
Question d

Plots for θ_1:

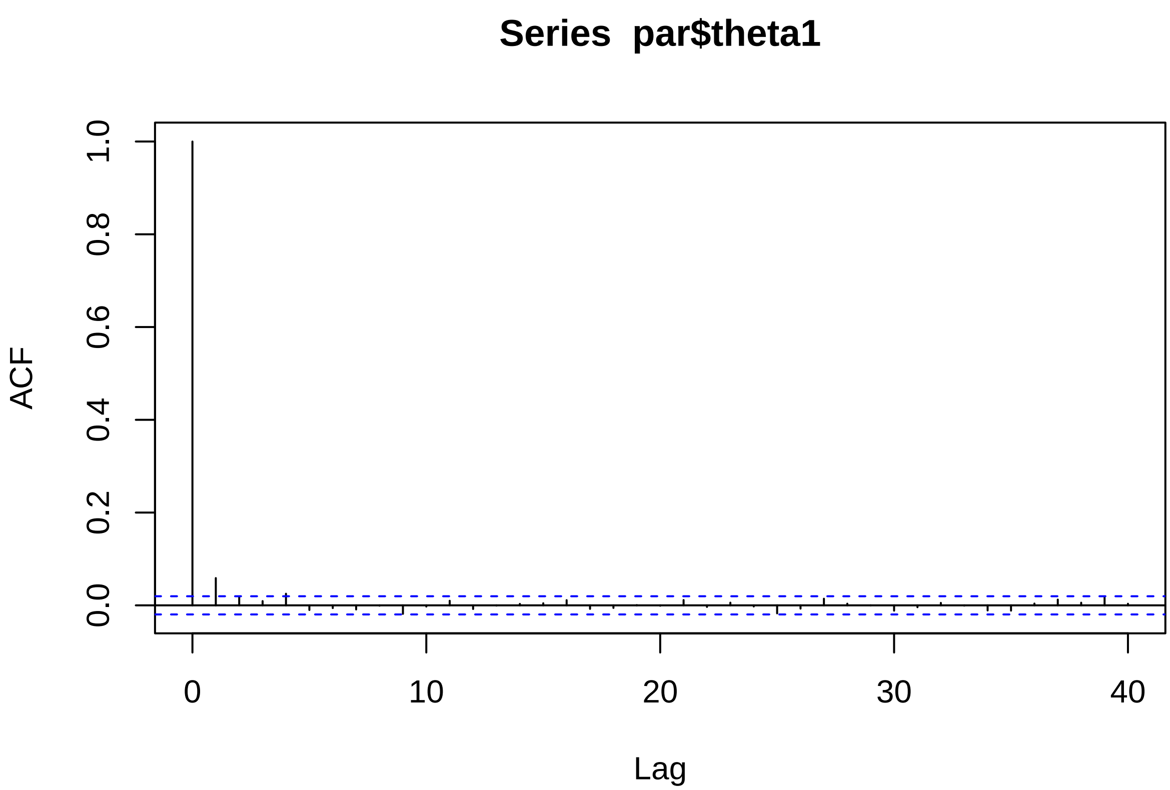
```r
plot(par$theta1,type="l")
```




```r
hist(par$theta1)
```

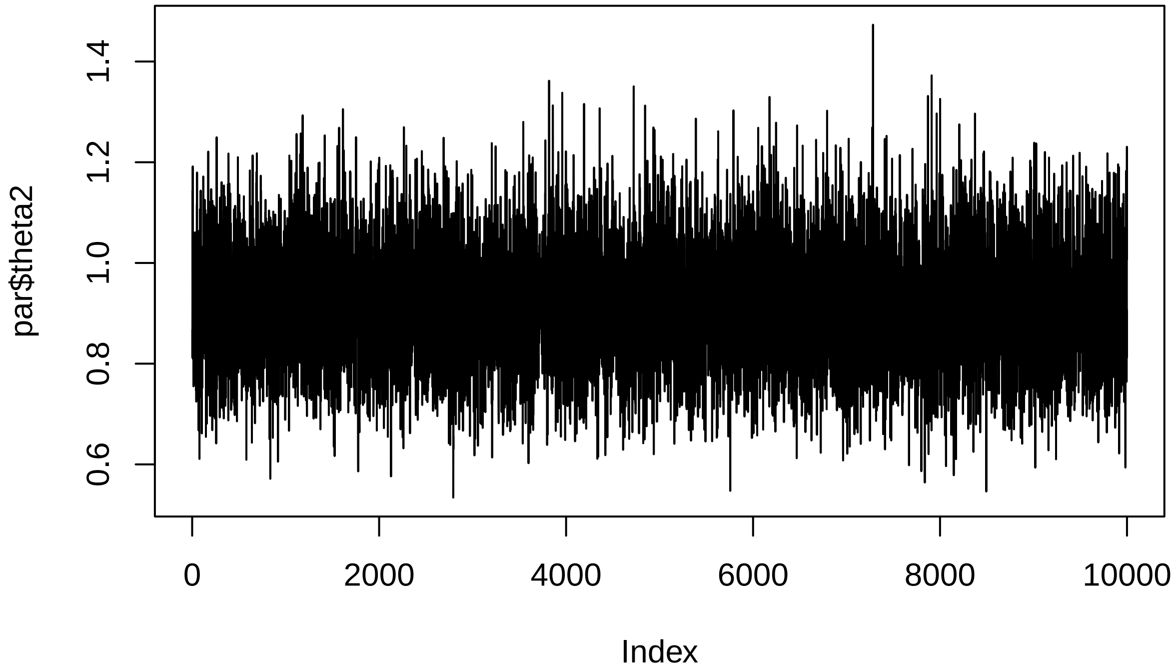



```r
acf(par$theta1)
```

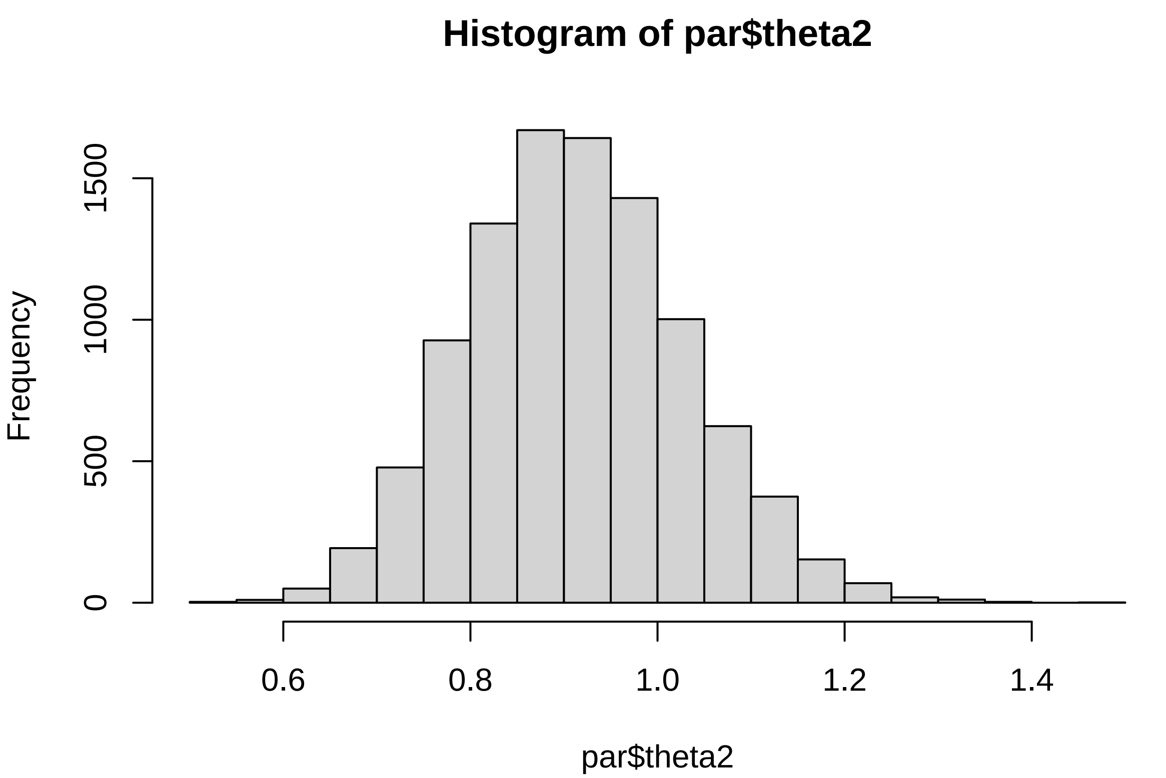


Plots for θ_2:

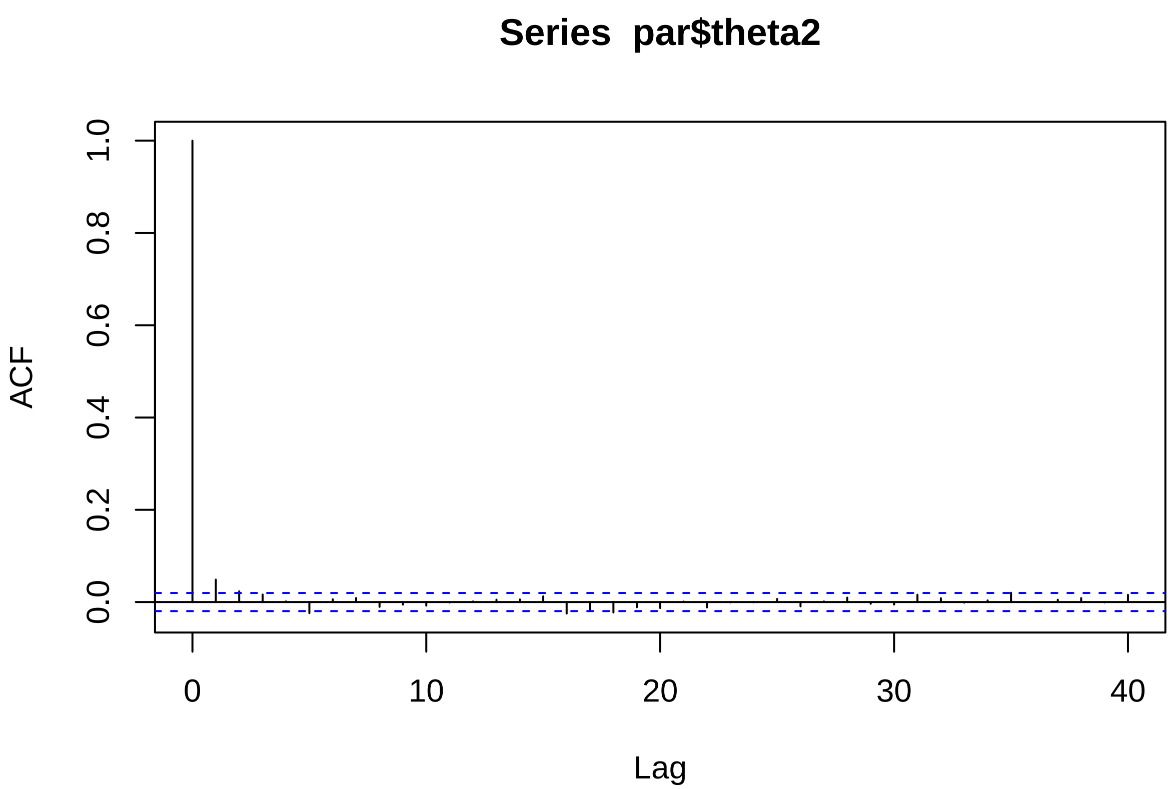
```r
plot(par$theta2,type="l")
```




```r
hist(par$theta2)
```

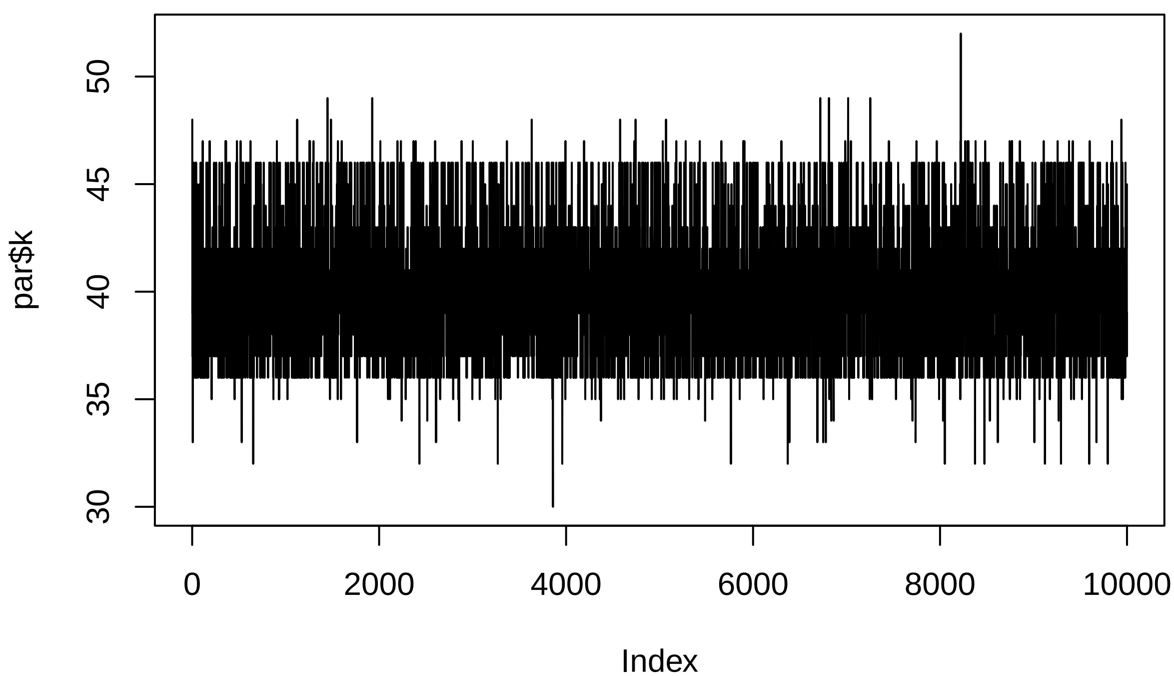



```r
acf(par$theta2)
```

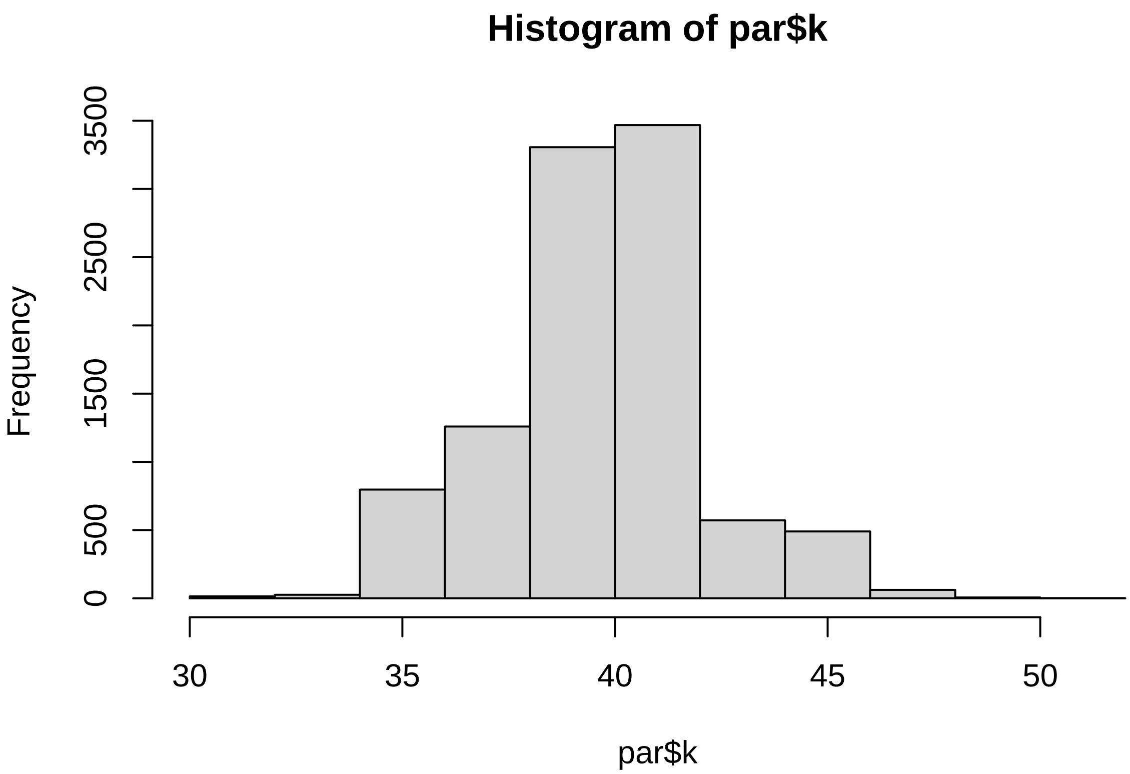


Plots for k:

```
plot(par$k,type="l")
```

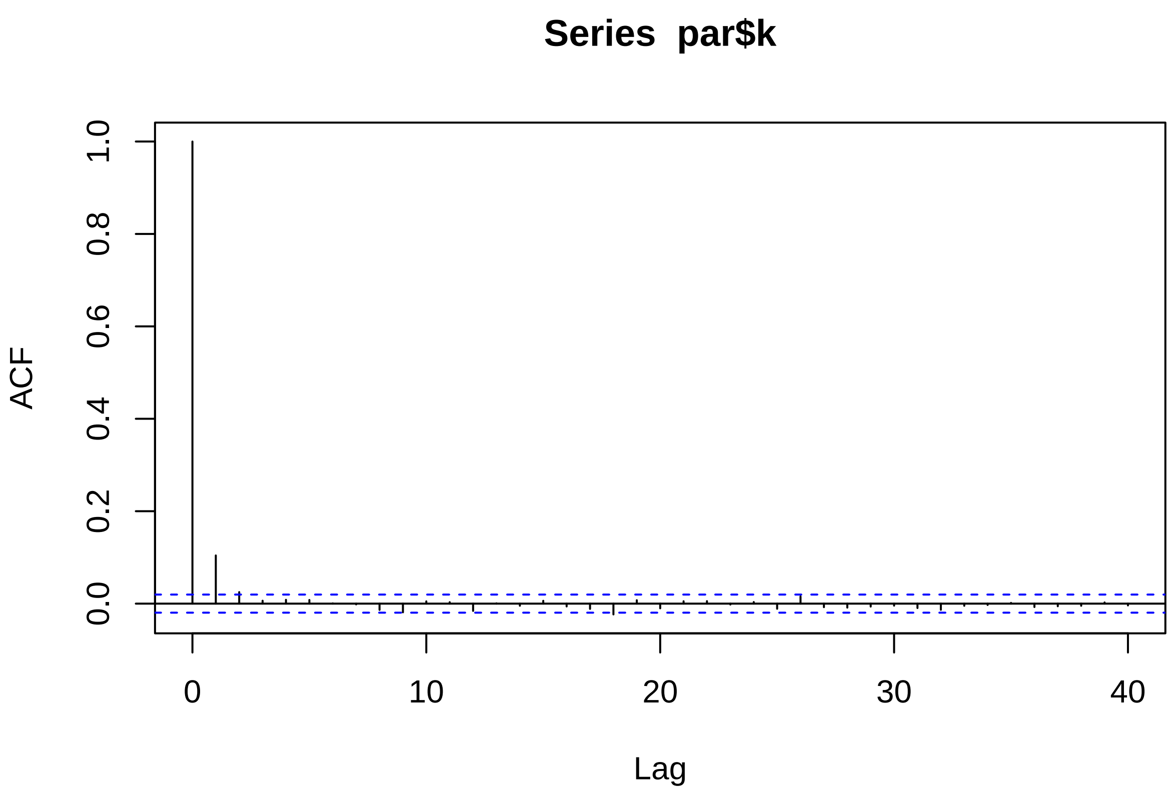



```
hist(par$k)
```



Histogram of par$k
Frequency
0
500
1500
2500
3500
30
35
40
45
50
par$k


```
acf(par$k)
```



Series par$k
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
10
20
30
40
Lag

Question e

We set the lag to $k_0 = 10$ and the burn-in period to $D = 1000$, and we compute the number B of batches:

```
k0 <- 10 # lag
D <- 1000 # burn in
B<-floor((N-D)/k0)  # number of batches
```

Estimated conditional expectation and simulated standard error for θ_1:

```
Z<-vector(B,mode="numeric")
for(b in (1:B)) Z[b]<-mean(par$theta1[(D+(b-1)*k0+1):(D+b*k0)])
se <- sd(Z)/sqrt(B)
cat("mean=", mean(par$theta1[(D+1):N]),"se =",se)
```

```
## mean= 3.051553 se = 0.003175602
```

Estimated conditional expectation and simulated standard error for θ_2:

```
for(b in (1:B)) Z[b]<-mean(par$theta2[(D+(b-1)*k0+1):(D+b*k0)])
se <- sd(Z)/sqrt(B)
cat("mean=", mean(par$theta2[(D+1):N]),"se =",se)
```

```
## mean= 0.9144713 se = 0.001345767
```

Estimated conditional expectation and simulated standard error for k:

```
for(b in (1:B)) Z[b]<-mean(par$k[(D+(b-1)*k0+1):(D+b*k0)])
se <- sd(Z)/sqrt(B)
cat("mean=", mean(par$k[(D+1):N]),"se =",se)
```

```
## mean= 40.15444 se = 0.02908424
```

The change is estimated to have taken place in the year

```
cat(coal$year[round(mean(par$k[(D+1):N]))])
```

```
## 1890
```

Exercise 4

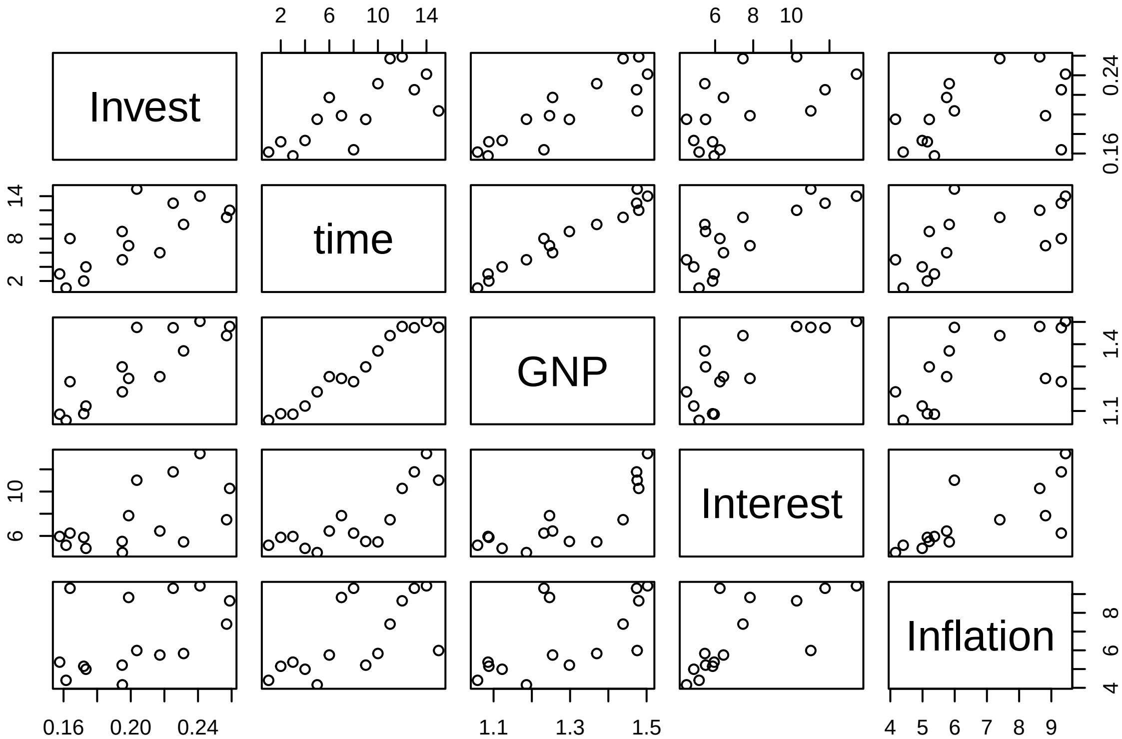
Question a

Let us first read and transform the data:

```
data <- read.table("/Users/Thierry/Documents/R/Data/Compstat/investment.txt",header=TRUE)
attach(data)
y <- Invest/(CPI*10)
time <- 1:15
GNP1<-data$GNP/(CPI*10)
X <- cbind(rep(1,15),time,GNP1,Interest,Inflation)
data1 <- data.frame(Invest=y,time, GNP=GNP1,Interest,Inflation)
```

Let us plot the data:

```
plot(data1)
```



Finally, we perform classical linear regression on these data:

```r
reg <- lm(Invest~.,data=data1)
summary(reg)
```

```
##
## Call:
## lm(formula = Invest ~ ., data = data1)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0100886 -0.0024959  0.0004332  0.0028830  0.0079354
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -5.091e-01  5.393e-02  -9.439 2.69e-06 ***
## time        -1.659e-02  1.929e-03  -8.598 6.23e-06 ***
## GNP          6.703e-01  5.380e-02  12.459 2.05e-07 ***
## Interest    -2.428e-03  1.193e-03  -2.035   0.0692 .
## Inflation    6.402e-05  1.319e-03   0.049   0.9622
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.006571 on 10 degrees of freedom
## Multiple R-squared:  0.9735, Adjusted R-squared:  0.9629
## F-statistic: 91.83 on 4 and 10 DF,  p-value: 7.672e-08
```

Question b

The joint posterior pdf of (β, σ^2) is

$$f(\beta, \sigma^2|\boldsymbol{y}) \propto f(\boldsymbol{y}|\beta, \sigma^2)f(\beta)f(\sigma^2),$$

that is,

$$f(\beta, \sigma^2|\boldsymbol{y}) \propto \left(\frac{1}{\sigma^2}\right)^{n/2} \exp\left[-\frac{1}{2\sigma^2}(\boldsymbol{y} - \boldsymbol{X}\beta)^T(\boldsymbol{y} - \boldsymbol{X}\beta)\right] \times \\ \exp\left[-\frac{1}{2}(\beta - \beta_0)^T \boldsymbol{B}_0^{-1}(\beta - \beta_0)\right] \times \frac{1}{(\sigma^2)^{\alpha_0/2+1}} \exp\left[-\delta_0/(2\sigma^2)\right].$$

To compute $f(\beta|\sigma^2, \boldsymbol{y})$, we consider only the terms in the posterior that contain β. We get

$$f(\beta|\sigma^2, \boldsymbol{y}) \propto \exp\left[-\frac{1}{2\sigma^2}(\boldsymbol{y} - \boldsymbol{X}\beta)^T(\boldsymbol{y} - \boldsymbol{X}\beta) - \frac{1}{2}(\beta - \beta_0)^T \boldsymbol{B}_0^{-1}(\beta - \beta_0)\right].$$

Developing and rearranging the terms in the exponential, we get

$$\begin{aligned} f(\beta|\sigma^2, \boldsymbol{y}) &\propto \exp\left\{-\frac{1}{2}\left[\beta^T(\sigma^{-2}\boldsymbol{X}^T\boldsymbol{X} + \boldsymbol{B}_0^{-1})\beta - 2\beta^T(\sigma^{-2}\boldsymbol{X}\boldsymbol{y} + \boldsymbol{B}_0^{-1}\beta_0)) + \sigma^{-2}\boldsymbol{y}^T\boldsymbol{y} + \beta_0^T\boldsymbol{B}_0^{-1}\beta_0\right]\right\} \\ &\propto \exp\left\{-\frac{1}{2}(\beta - \overline{\beta})^T\boldsymbol{B}_1^{-1}(\beta - \overline{\beta})\right\}, \end{aligned}$$

with

$$\boldsymbol{B}_1 = (\sigma^{-2}\boldsymbol{X}^T\boldsymbol{X} + \boldsymbol{B}_0^{-1})^{-1}$$

and

$$\overline{\beta} = \boldsymbol{B}_1(\sigma^{-2}\boldsymbol{X}^T\boldsymbol{y} + \boldsymbol{B}_0^{-1}\beta_0).$$

Similarly,

$$\begin{aligned} \sigma^2|\beta, \boldsymbol{y}) &\propto \left(\frac{1}{\sigma^2}\right)^{n/2} \exp\left[-\frac{1}{2\sigma^2}(\boldsymbol{y} - \boldsymbol{X}\beta)^T(\boldsymbol{y} - \boldsymbol{X}\beta)\right] \times \frac{1}{(\sigma^2)^{\alpha_0/2+1}} \exp\left[-\delta_0/(2\sigma^2)\right] \\ &\propto \left(\frac{1}{\sigma^2}\right)^{\alpha_1/2+1} \exp\left[-\delta_1/(2\sigma^2)\right], \end{aligned}$$

with $\alpha_1 = \alpha_0 + n$ and

$$\delta_1 = \delta_0 + (\boldsymbol{y} - \boldsymbol{X}\beta)^T(\boldsymbol{y} - \boldsymbol{X}\beta).$$

Question c

As the conditional posterior distributions of β given σ^2 and of σ^2 given β have standard form, they can be easily sampled and a Gibbs sampler can be easily implemented. To sample from a multivariate normal distribution, we use function `mvtnorm` from package `MASS`:

```
library(MASS)
```

To sample from the inverse gamma distribution, we notice that $\sigma^2 \sim IG(\alpha, \beta)$ iff $1/\sigma^2 \sim G(\alpha, \beta)$. We can thus sample $1/\sigma^2$ from the Gamma distribution. Here is an implementation of the Gibbs sampler for this problem:

```r
gibbs_reg<- function(y,X,beta0,B0,alpha0,delta0,D=1000,L=10000){
  n<-nrow(X)
  p<-ncol(X)
  B0inv<-solve(B0)
  alpha1=alpha0+n
  N <- D+L
  beta<-matrix(0,nrow=N,ncol=p)
  sigma2<-vector(length=N,mode="numeric")
  # Initialization
  sigma20<-1/rgamma(1, shape=alpha0/2, rate = delta0/2)
  for(t in 1:N){
    # Generation of beta given sigma^2 and y
    B1<-solve(sigma20^{-1}* t(X)%*%X + B0inv)
    beta1<- B1 %*% (sigma20^{-1}*t(X)%*%y + B0inv%*%beta0)
    beta[t,]<- mvrnorm(n=1,mu=beta1,Sigma=B1)
    # Generation of sigma^2 given beta and y
    delta1<- delta0+t(y-X%*%beta[t,])%*%(y-X%*%beta[t,])
    sigma2[t]<- 1/rgamma(1, shape=alpha1/2, rate = delta1/2)
    sigma20<-sigma2[t]
  }
  return(list(beta=beta[(D+1):N,],sigma2=sigma2[(D+1):N]))
}
```

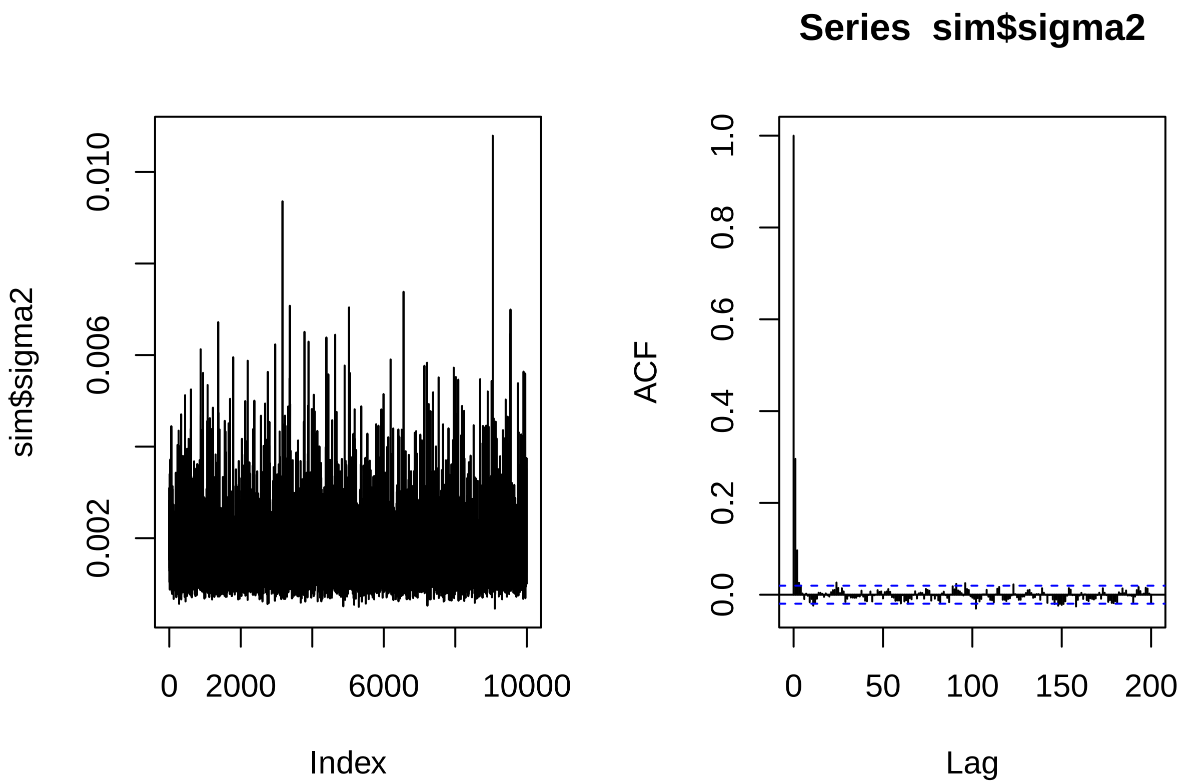
Question d

Let us assume $\beta_0 = (0, 0, 0, 0, 0)$, $\boldsymbol{B}_0 = 10\boldsymbol{I}_5$. For the prior on σ^2, we assume $E = 0.01$ and $V = 1000$. We will generate a chain of size $L = 10000$ with a burn-in period of length $D = 1000$:

```r
p <- ncol(X)
V <- 1000
E <- 0.01
alpha0 <- 2*(E^2/V+2)
delta0 <- 2*E*(E^2/V+1)
B0 <- 10*diag(p)
beta0 <- rep(0,p)
D <- 1000
L <- 10000
sim <- gibbs_reg(y,X,beta0,B0,alpha0,delta0,D,L)
```

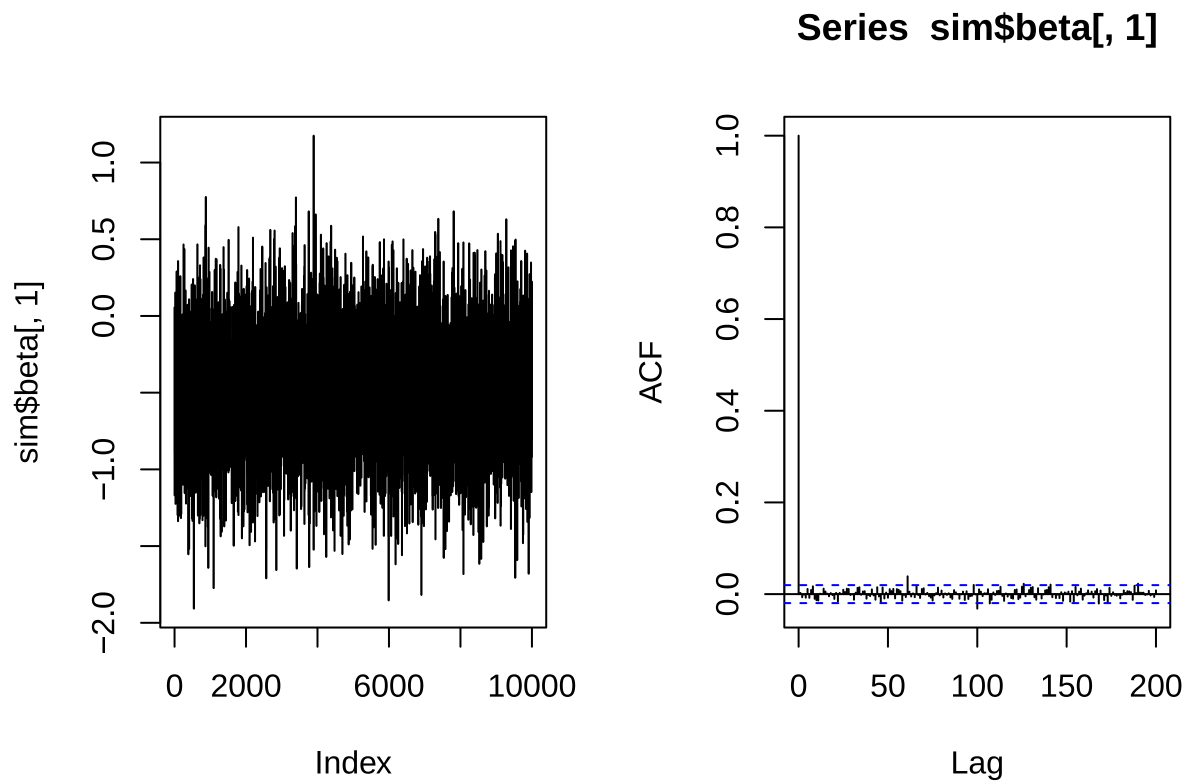
The sample paths and autocorrelation plots show a good mixing of the chain:

```r
par(mfrow=c(1,2))
plot(sim$sigma2,type="l")
acf(sim$sigma2,lag.max=200)
```




```
par(mfrow=c(1,1))
```

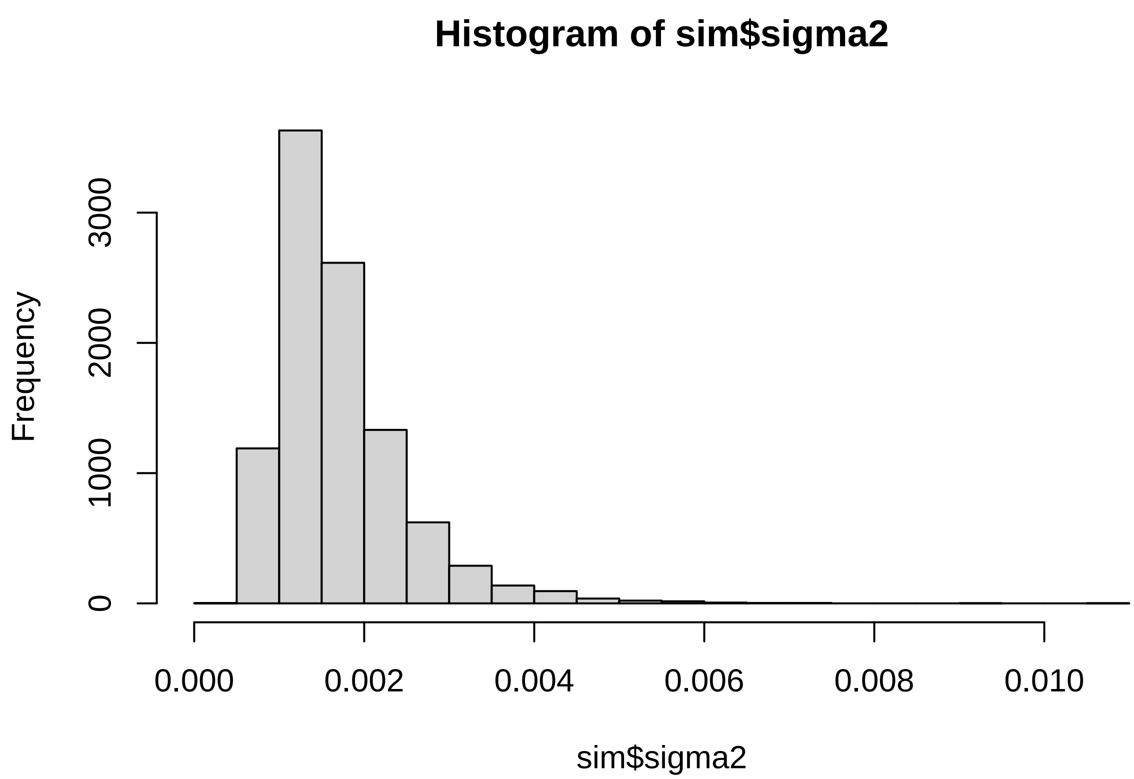
```
par(mfrow=c(1,2))
plot(sim$beta[,1],type="l")
acf(sim$beta[,1],lag.max=200)
```




```r
par(mfrow=c(1,1))
```

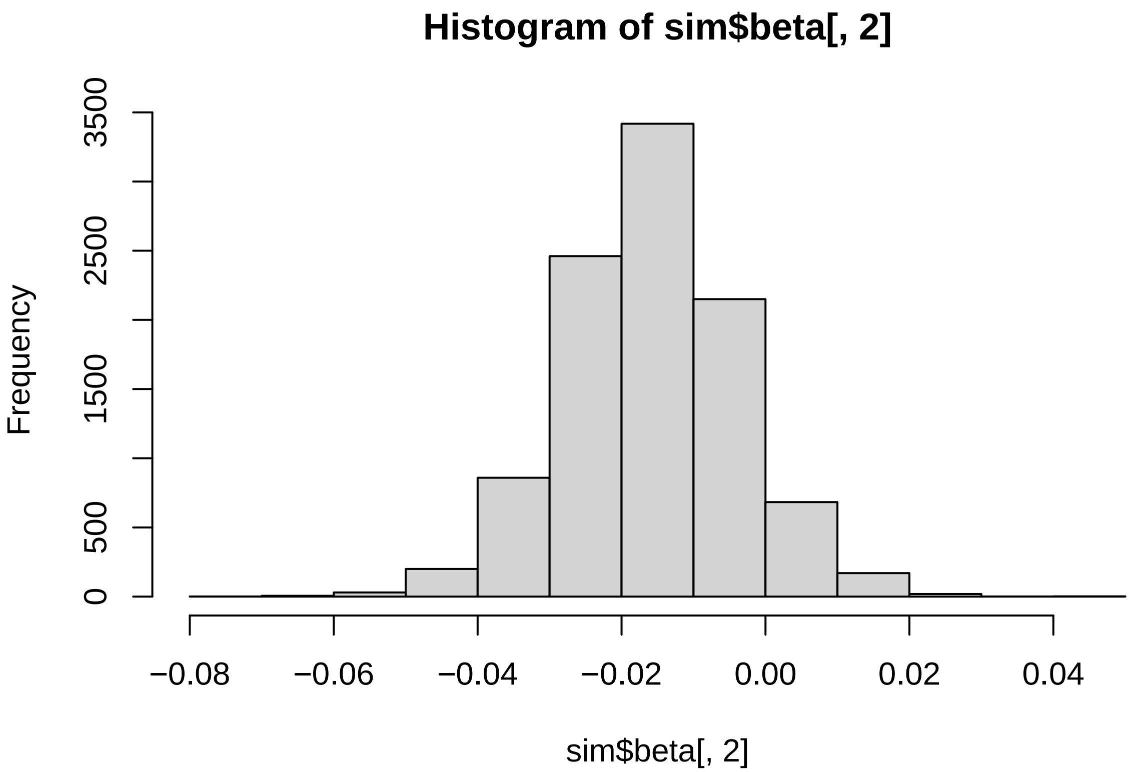
Here are, for instance, the histograms of the posterior distribution of σ^2

```r
hist(sim$sigma2)
```



and β_1:

```
hist(sim$beta[,2])
```



We can compute 95% credible intervals for the six parameters:

```
for(i in 1:5) print(quantile(sim$beta[,i],probs = c(0.025,0.975)),2)
```

```
##  2.5% 97.5%
## -1.16  0.17
##    2.5%   97.5%
## -0.0398  0.0082
##   2.5%  97.5%
## -0.014  1.315
##   2.5%  97.5%
## -0.017  0.012
##   2.5%  97.5%
## -0.016  0.017
```

```
print(quantile(sim$sigma2,probs = c(0.025,0.975)),2)
```

```
##    2.5%   97.5%
## 0.00079 0.00371
```

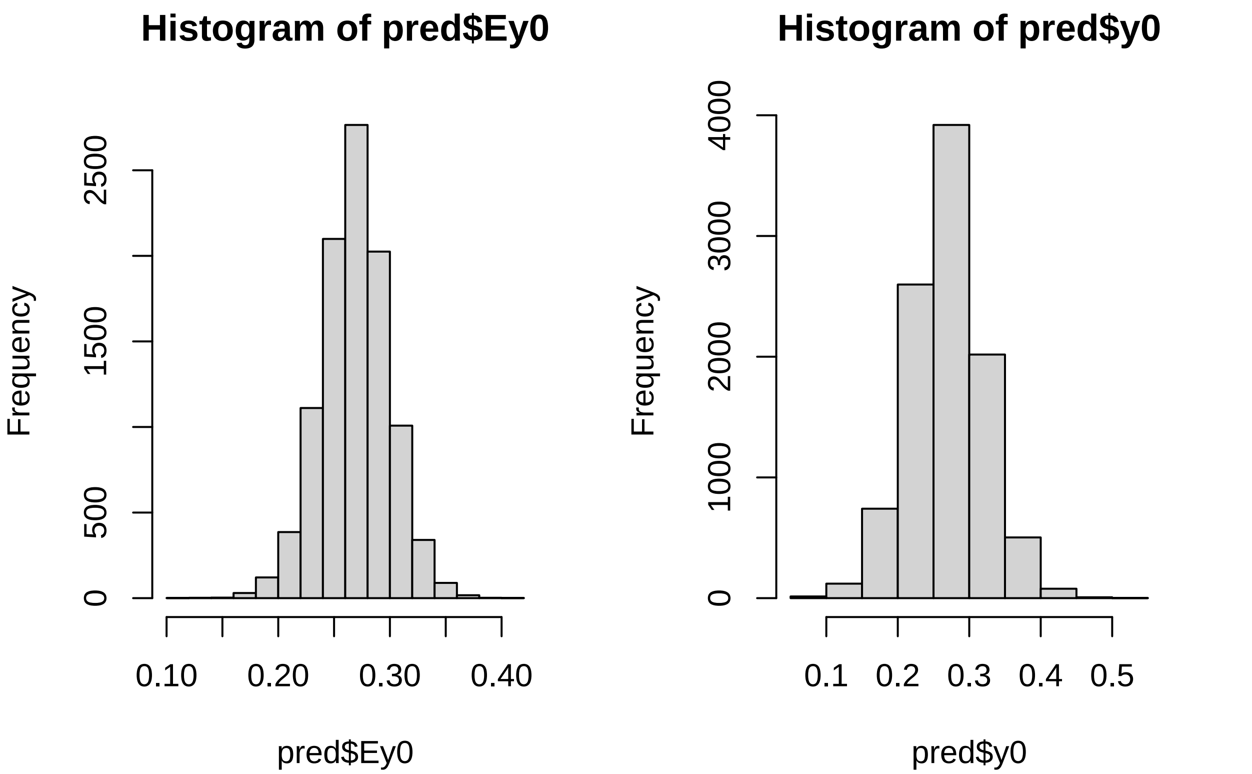
Question e

To sample from the posterior distribution of $E(y_0) = x_0^T\beta$, we simply sample from the posterior distribution of β and we multiply each vector by x_0. To sample from the posterior distribution of $y_0 = x_0^T\beta + \sigma u$, we sample independently β from its posterior distribution and u from the standard normal distribution:

```r
predict_reg <- function(x0,sim){
  N <- length(sim$sigma2)
  Ey0 <- vector(length=N,mode="numeric")
  y0 <- Ey0
  for(t in 1:N){
    Ey0[t]<- t(x0)%*%sim$beta[t,]
    y0[t] <- Ey0[t] + sqrt(sim$sigma2[t])*rnorm(1)
  }
  return(list(Ey0=Ey0,y0=y0))
}
```

Let us plot side by side the two histograms for $x_0 = (1, 16, 1.6, 12, 7)$:

```r
x0<-c(1,16,1.6,12,7)
pred<-predict_reg(x0,sim)
par(mfrow=c(1,2))
hist(pred$Ey0)
hist(pred$y0)
```




```r
par(mfrow=c(1,1))
```

Let us compute 95% credible intervals on $E(y_0)$ and y_0:

```r
print(quantile(pred$Ey0,probs = c(0.025,0.975)),2)
```

```
##  2.5% 97.5%
##  0.21  0.33
```

```r
print(quantile(pred$y0,probs = c(0.025,0.975)),2)
```

```
##  2.5% 97.5%
##  0.17  0.37
```