

Homework 7 - STAT 5361 Statistical Computing

*Sen Yang**

27 October 2018

Abstract

This is homework 7 for STAT 5361 - Statistical Computing.

Contents

1	Markov Chain Monte Carlo sampling	2
1.1	Posterior inference of Normal mixture	2
1.2	MCMC process in R	2

*sen.2.yang@uconn.edu; M.S. student at Department of Statistics, University of Connecticut.

1 Markov Chain Monte Carlo sampling

1.1 Posterior inference of Normal mixture

For a Normal mixture of $N(\mu_1, \sigma_1^2)$ and $N(\mu_2, \sigma_2^2)$ with propotion of first Normal distribution δ , we have

$$X_i \sim \delta N(\mu_1, \sigma_1^2) + (1 - \delta)N(\mu_2, \sigma_2^2)$$

Then the likelihood function of $X_i|\mu_1, \mu_2, \sigma_1^2, \sigma_2^2, \delta$ is given as

$$L(X_i|\mu_1, \mu_2, \sigma_1^2, \sigma_2^2, \delta) = \prod_{i=1}^n [\delta \cdot \phi(x_i|\mu_1, \sigma_1^2) + (1 - \delta) \cdot \phi(x_i|\mu_2, \sigma_2^2)]$$

which can be rewritten as

$$f(x|\theta) \propto \prod_{i=1}^n \left[\frac{\delta}{\sigma_1} \exp\left(-\frac{(x_i - \mu_1)^2}{\sigma_1^2}\right) + \frac{1 - \delta}{\sigma_2} \exp\left(-\frac{(x_i - \mu_2)^2}{\sigma_2^2}\right) \right]$$

Since the prior for $1/\sigma_1^2$ and $1/\sigma_2^2$ are $\Gamma(a, b)$ with shape $a = 0.5$ and scale $b = 10$, then $\sigma_1^2 \sim Inv\text{-Gamma}(a, 1/b)$ and $\sigma_2^2 \sim Inv\text{-Gamma}(a, 1/b)$. Also, the prior for μ_1 and μ_2 are $N(0, 10^2)$ and all the priors are independent. The posterior density of $\theta = (\mu_1, \mu_2, \sigma_1^2, \sigma_2^2, \delta)$ is

$$q(\mu_1, \mu_2, \sigma_1^2, \sigma_2^2, \delta|x) \propto \prod_{i=1}^n \left[\frac{\delta}{\sigma_1} \exp\left(-\frac{(x_i - \mu_1)^2}{\sigma_1^2}\right) + \frac{1 - \delta}{\sigma_2} \exp\left(-\frac{(x_i - \mu_2)^2}{\sigma_2^2}\right) \right] \cdot \exp(\mu_1^2) \cdot \exp(\mu_2^2) \cdot (\sigma_1^2)^{-1.5} \exp\left(-\frac{1}{\sigma_1^2}\right) \cdot (\sigma_2^2)^{-1.5} \exp\left(-\frac{1}{\sigma_2^2}\right)$$

The full conditional distributions of $\theta = (\mu_1, \mu_2, \sigma_1^2, \sigma_2^2, \delta)$ can be shown to be log-concave, which allows adaptive rejection alogorithm.

1.2 MCMC process in R

- Firstly, generate some data from a mixtrure Normal of of $N(-1, 0.1)$ and $N(5, 0.5)$ with propotion of first Normal distribution $\delta = 0.7$.

```
## generate a random sample from a mixture normal with true parameter values
delta <- 0.7 # true value to be estimated based on the data
n <- 100
set.seed(123)
u <- rbinom(n, prob = delta, size = 1)
x <- rnorm(n, ifelse(u == 1, 7, 10), 0.5)
mydata <- data.frame(x)
```

- Secondly, calculate the posterior density.

```

## posterior density
## posterior density
logpost <- function(theta,data) {
  mu_1 <- theta[1]; mu_2 <- theta[2];
  sigma2_1 <- theta[3]; sigma2_2 <- theta[4]; delta <- theta[5]
  x <- data$x
  library("invgamma")
  return(sum(log(delta*dnorm(x,mu_1,sigma2_1^0.5)+(1-delta)*dnorm(x,mu_2,sigma2_2^0.5))) +
           dnorm(mu_1,0,10,log = T) + dnorm(mu_2,0,10,log = T) +
           dinvgamma(sigma2_1,shape=0.5,scale=10,log = T) +
           dinvgamma(sigma2_2,shape=0.5,scale=10,log = T))
}

```

- Thirdly, An MCMC based the Gibbs sampler uses the ARMS algorithm from R package **HI**.

```

## MCMC
mymcmc <- function(niter, thetaInit, data, nburn) {
  p <- length(thetaInit)
  thetaCurrent <- thetaInit
  ## define a function for full conditional sampling
  logFC <- function(th, idx) {
    theta <- thetaCurrent
    theta[idx] <- th
    logpost(theta, data)
  }
  out <- matrix(thetaInit, niter, p, byrow = TRUE)
  ## Gibbs sampling
  for (i in 2:niter) {
    for (j in 1:p) {
      ## general-purpose arms algorithm
      # Indicator function
      indF <- function(x, idx) {
        if (idx==1 | idx==2) {(x<30)*(x>-30)}
        else if (idx==3 | idx==4) {(x<10)*(x>0)}
        else {(x<1)*(x>=0)}
      }
      out[i, j] <- thetaCurrent[j] <-
        HI::arms(thetaCurrent[j], logFC, indF, 1, idx = j)
    }
  }
  out[:(1:nburn), ]
}

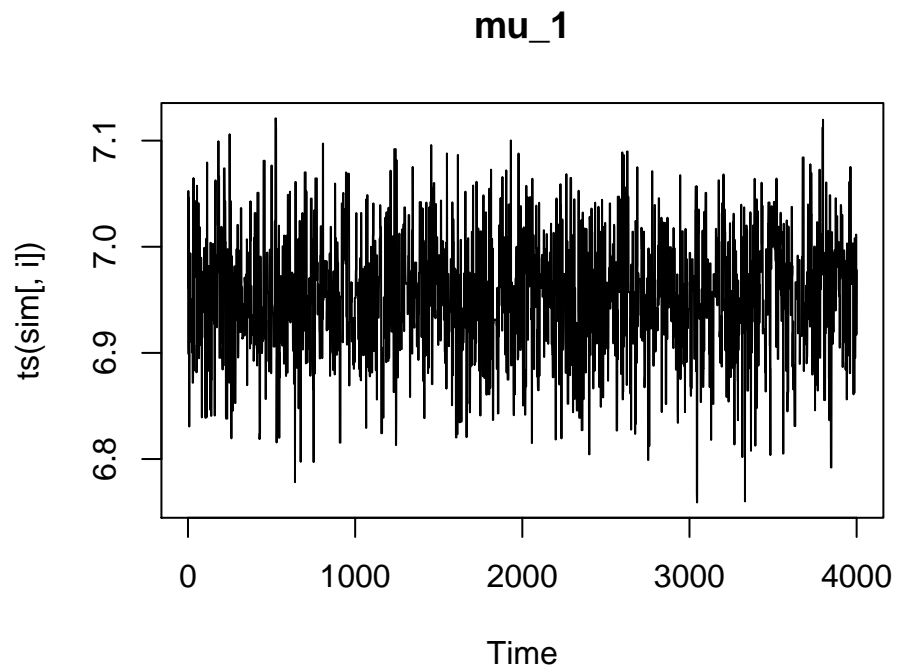
```

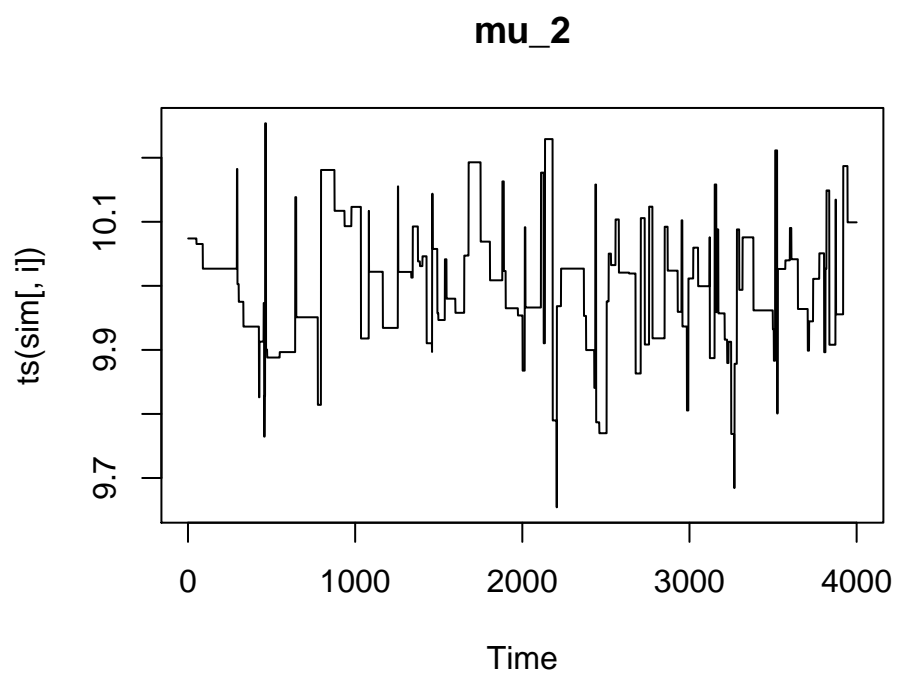
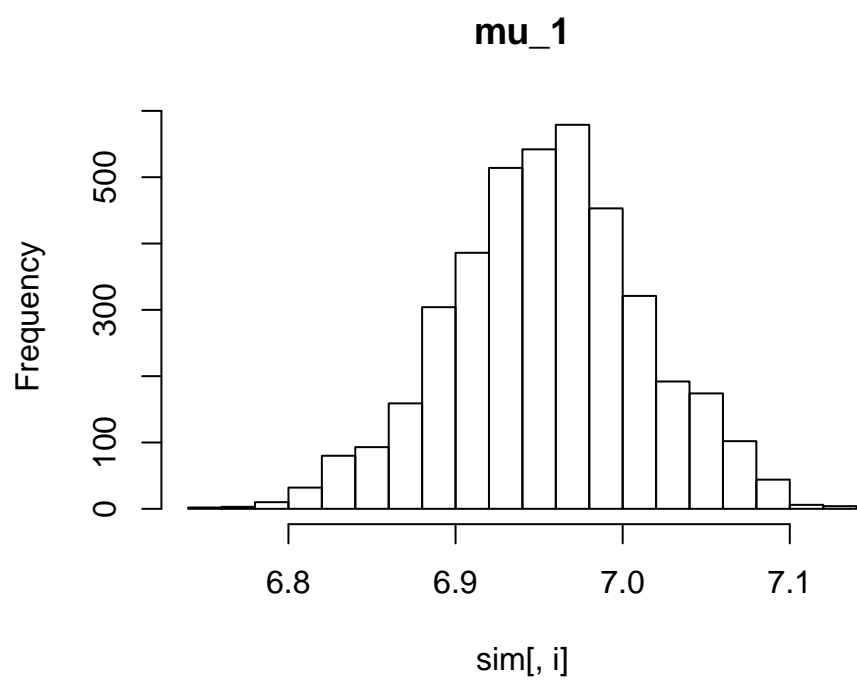
- Finally, run a simulation, drop the first 1000 observations and draw corresponding histogram for all parameters.

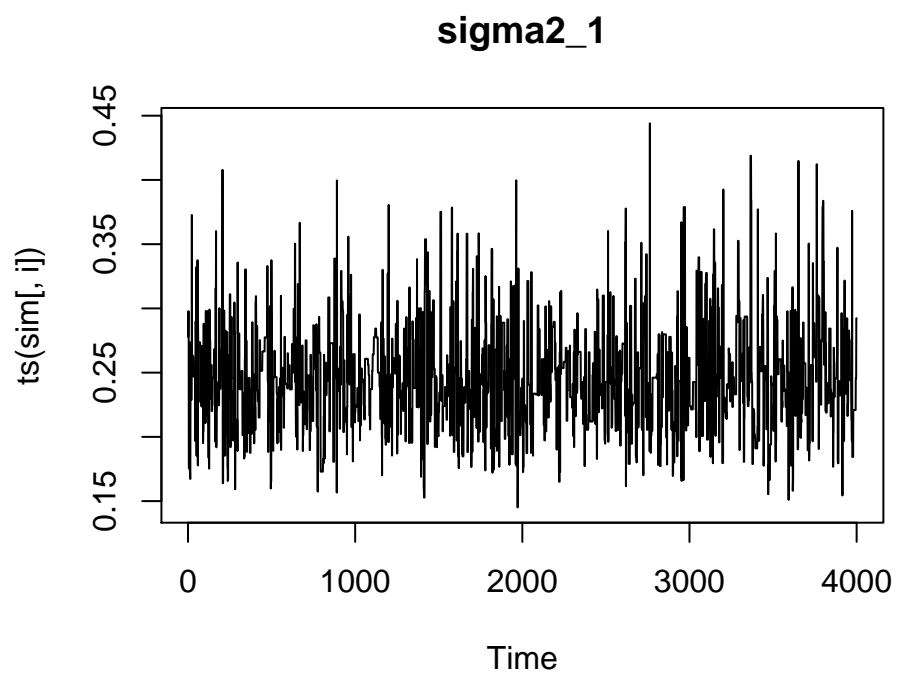
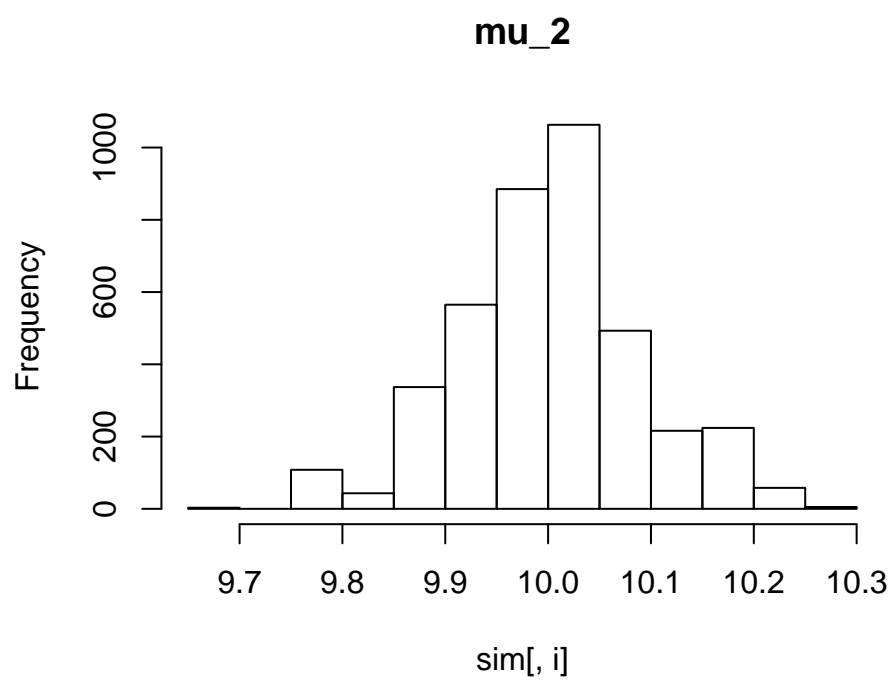
```

## simulation
niter <- 5000; nburn <- 1000
thetaInit <- c(0,0,0.2,0.2,0.5)
sim <- mymcmc(niter, thetaInit, mydata, nburn)
name_para <- c("mu_1","mu_2","sigma2_1","sigma2_2","delta")
for (i in 1:5) {
  plot(ts(sim[,i]),main=(paste(name_para[i])))
  hist(sim[,i],main=(paste(name_para[i])))
}

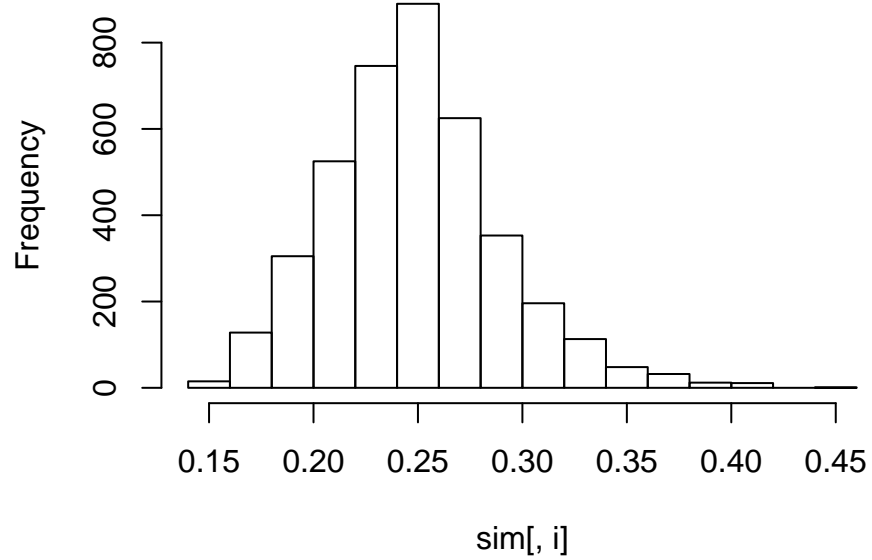
```



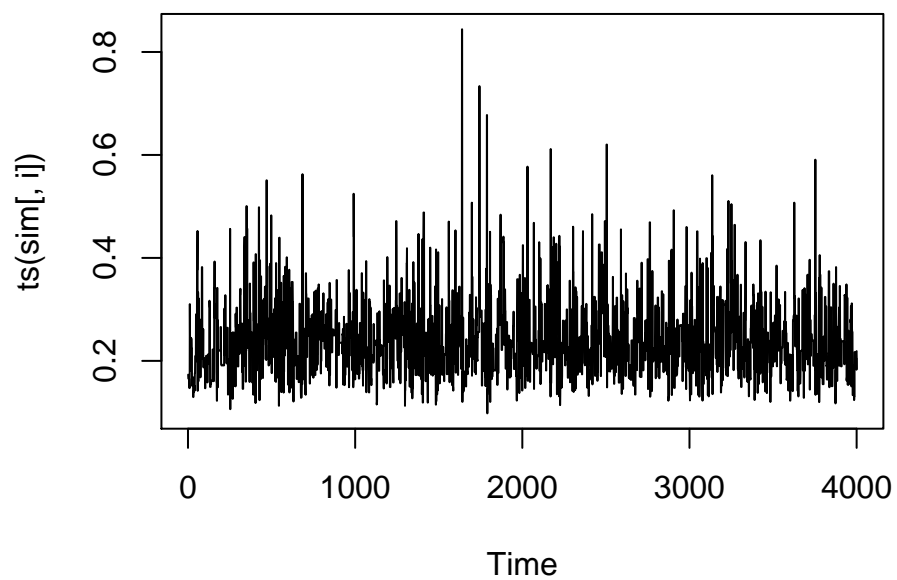




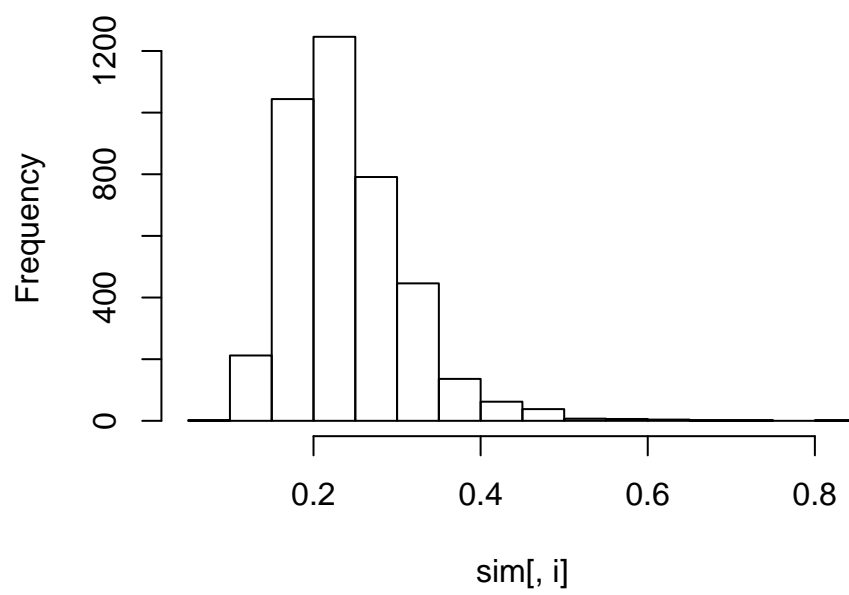
sigma2_1



sigma2_2



sigma2_2



delta

