STAT 206 Lab 10–Lihua Xu

Due Monday, December 11, 5:00 PM

General instructions for labs: You are encouraged to work in pairs to complete the lab. Labs must be completed as an R Markdown file. Be sure to include your lab partner (if you have one) and your own name in the file. Give the commands to answer each question in its own code block, which will also produce plots that will be automatically embedded in the output file. Each answer must be supported by written statements as well as any code used.

Agenda: Determine the full conditional distributions and simulate a Gibbs sampler

Gibbs Sampler

Suppose (Y_1, Y_2) are normally distributed with mean $\mu = (0, 0)$ and covariance matrix

$$\Sigma = \left(\begin{array}{cc} 1 & \rho \\ \rho & 1 \end{array} \right).$$

1. Find the full conditional distributions of $Y_1|Y_2$ and $Y_2|Y_1$.

   Solution :

   As in the problem, we have already get the covariance matrix: We know that $\sigma 1 = 1$ and $\sigma 2 = 1$. So the conditional distribution for $Y_1|Y_2$ is :

   $$f(Y_1|Y_2) = y2 \sim N(\mu 1 + \frac{\sigma 1}{\sigma 2}\rho(x_2 - \mu 2), (1 - \rho^2)\sigma 1^2)$$

   $$f(Y_2|Y_1) = y1 \sim N(\mu 2 + \frac{\sigma 2}{\sigma 1}\rho(x_1 - \mu 1), (1 - \rho^2)\sigma 2^2)$$

   ρ is the correlation coefficient as shown in the problem.

(Reference: wikipedia.Multivariate_normal_distribution.Bivariate case)

2. Write a Gibbs sampler using the full conditional distributions.

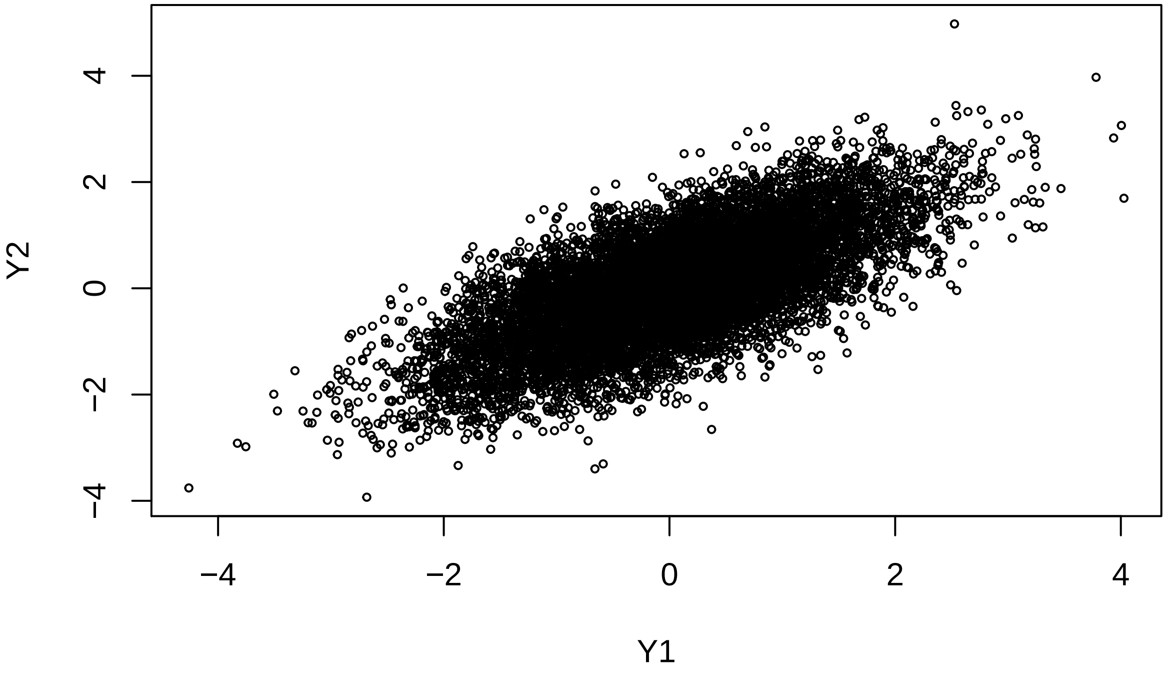
   Solution :

- 1)The bivariate distribution is (Y_1, Y_2). We use Y to represent.
- 2)Generating a starting point value $Y0 = (y1, y2)$
- 3)Generate in turn:
- (1)we create the $Y_1^*(1)$ from the conditional distribution $f(Y_1|y_2)$, then we assign the new value to the previous value as $y_1 = Y_1^*(1)$
- (2)we create the $Y_2^*(1)$ from the conditional distribution $f(Y_2|y_1)$, then we assign the new value to the previous value as $y_2 = Y_2^*(1)$
- (3)The new $Y(1) = (Y_1^*(1), Y_2^*(1))$ $(Y(1) = (y_1, y_2))$. This would be in the next cycle.
- 4)Repeat the 3), and at the same time incease the vale in the braket.

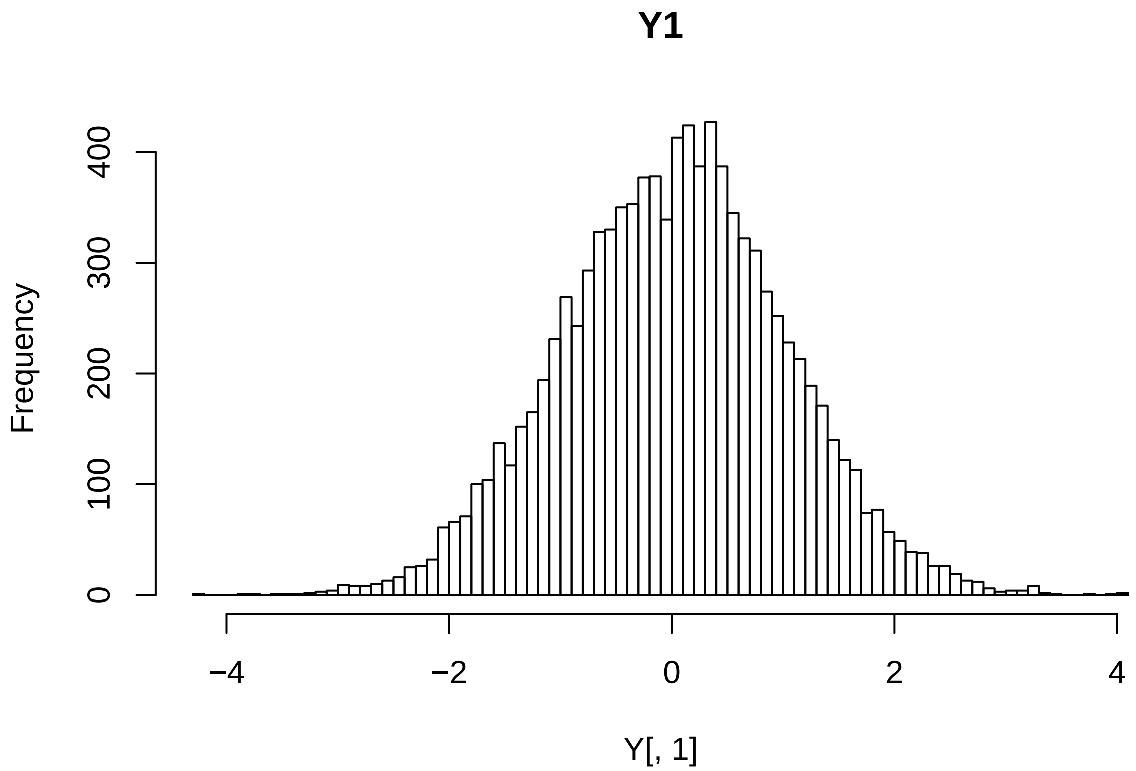
3. Generate 10000 draws from the bivariate normal distribution with $\rho = .7$.

```r
rho_value <- 0.7
N <- 10000
Y <- matrix(0,N,2)
mu_1 <- 0;mu_2 <- 0
sigma_1 <- 1; sigma_2 <-1
```

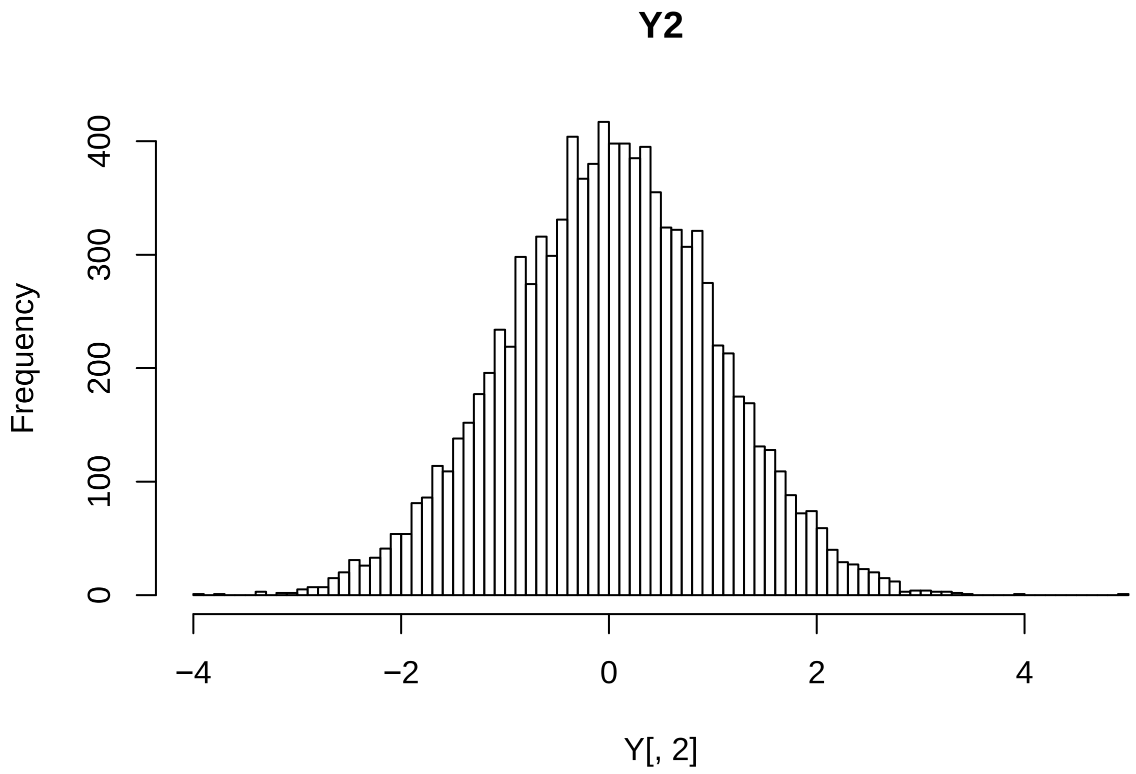
```
sigma_test_1 <- sqrt(1-rho_value^2)*sigma_1
sigma_test_2 <- sqrt(1-rho_value^2)*sigma_2
Y[1,] <- c(mu_1,mu_2)
for (i in 2:N) {
y2 <- Y[i-1, 2]
v1 <- mu_1 + rho_value * (y2 - mu_2) * sigma_1/sigma_2
Y[i, 1] <- rnorm(1, v1, sigma_test_1)
y1 <- Y[i, 1]
v2 <- mu_2 + rho_value * (y1 - mu_1) * sigma_2/sigma_1
Y[i, 2] <- rnorm(1, v2, sigma_test_2)
}
#10000 draws from the bivariate normal distribution is stored in thematrix Y, here I will not print it
plot(Y, cex=.5, xlab="Y1",ylab="Y2")
```



```
hist(Y[,1], breaks=100, main="Y1")
```

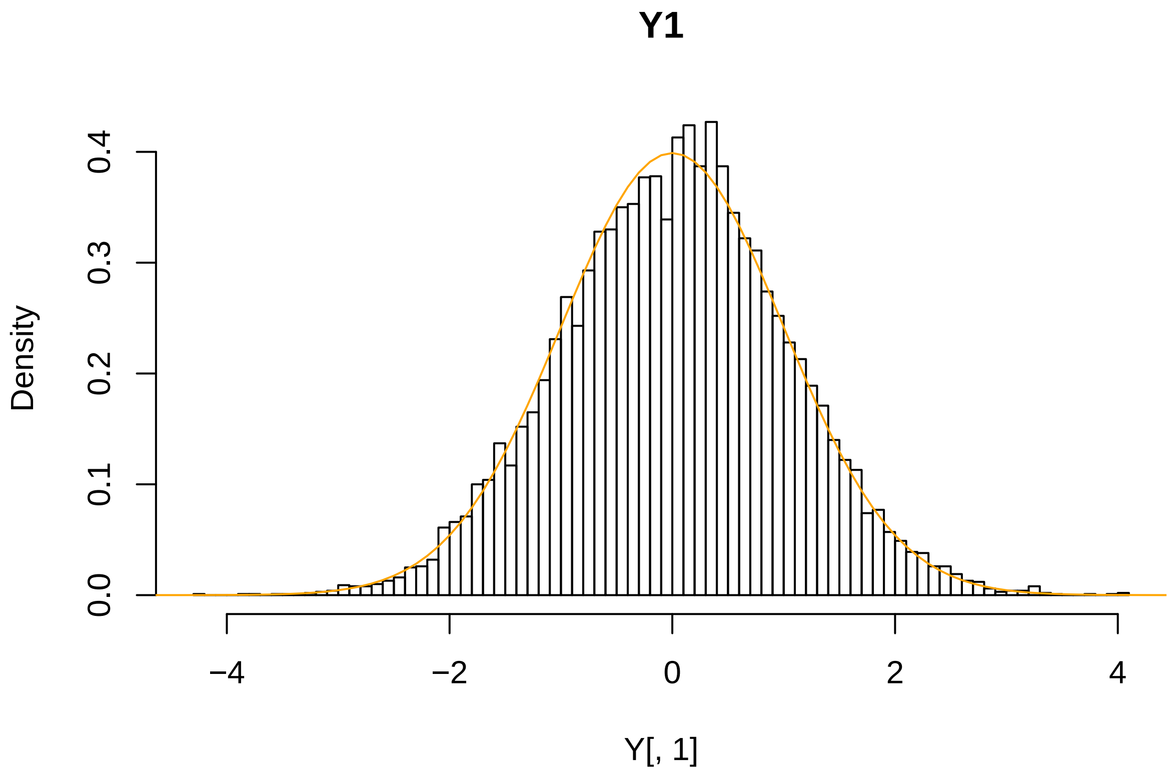


```
hist(Y[,2], breaks=100, main="Y2")
```

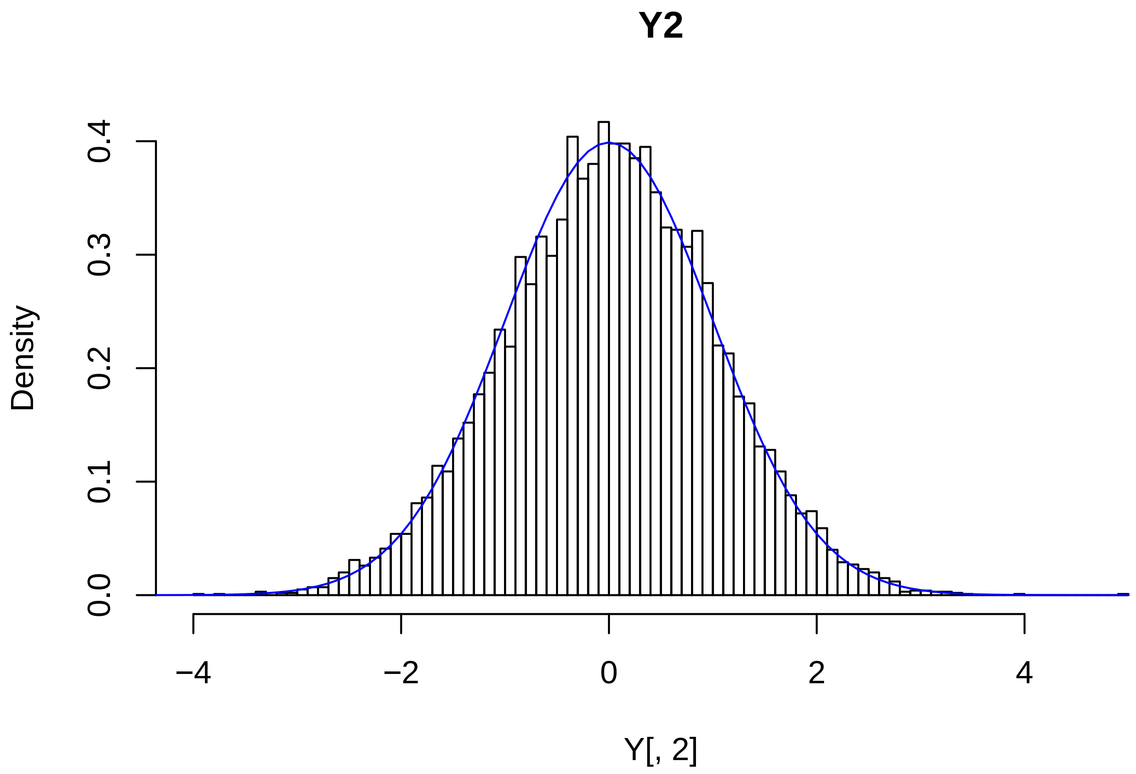


4. Plot estimates of the marginal distributions of Y_1 and Y_2 using the 10000 MCMC draws along with the true distribution. Comment on your findings.

```
hist(Y[,1], breaks=100, main="Y1",freq=FALSE)
curve(dnorm,-5,5, add=TRUE,col="orange")
```



```
hist(Y[,2], breaks=100, main="Y2",freq=FALSE)
curve(dnorm,-5,5, add=TRUE,col="blue")
```



5. Estimate the Effective Sample Size for estimating $E(Y_1)$ and $E(Y_2)$. Comment on your findings.

```
library("mcmcse")
```

```
## Warning: package 'mcmcse' was built under R version 3.4.3
```

```
## mcmcse: Monte Carlo Standard Errors for MCMC
## Version 1.3-2 created on 2017-07-03.
## copyright (c) 2012, James M. Flegal, University of California, Riverside
##                     John Hughes, University of Colorado, Denver
##                     Dootika Vats, University of Warwick
##                     Ning Dai, University of Minnesota
##  For citation information, type citation("mcmcse").
##  Type help("mcmcse-package") to get started.
```

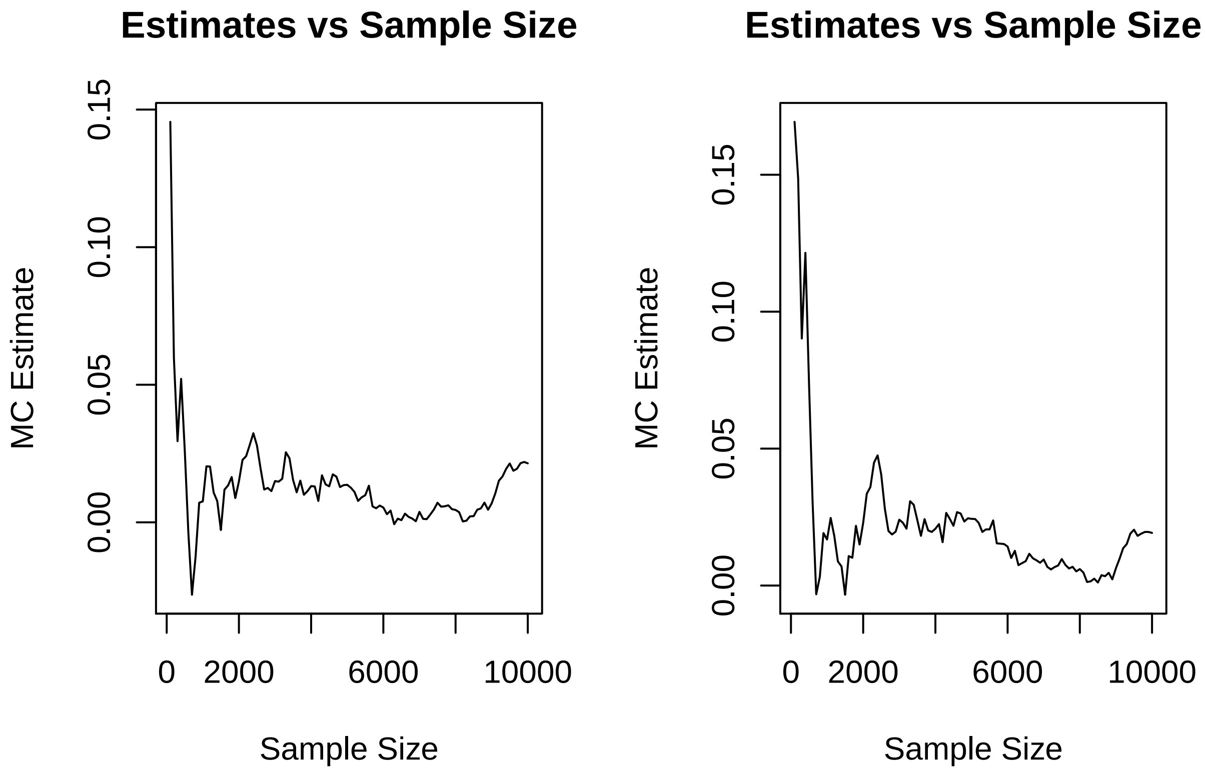
```
ess(Y[,1])
```

```
## [1] 4176.6
```

```
ess(Y[,2])
```

```
## [1] 3611.296
```

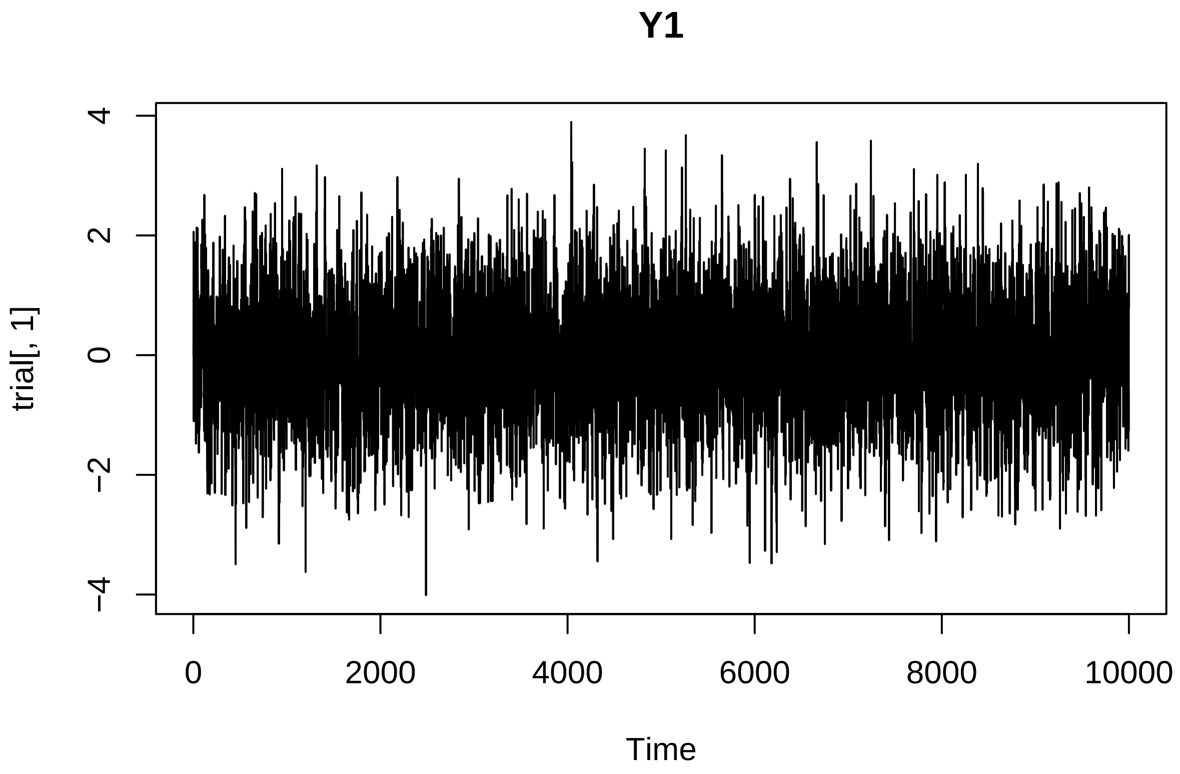
```
par(mfrow=c(1,2))
estvssamp(Y[,1])
estvssamp(Y[,2])
```



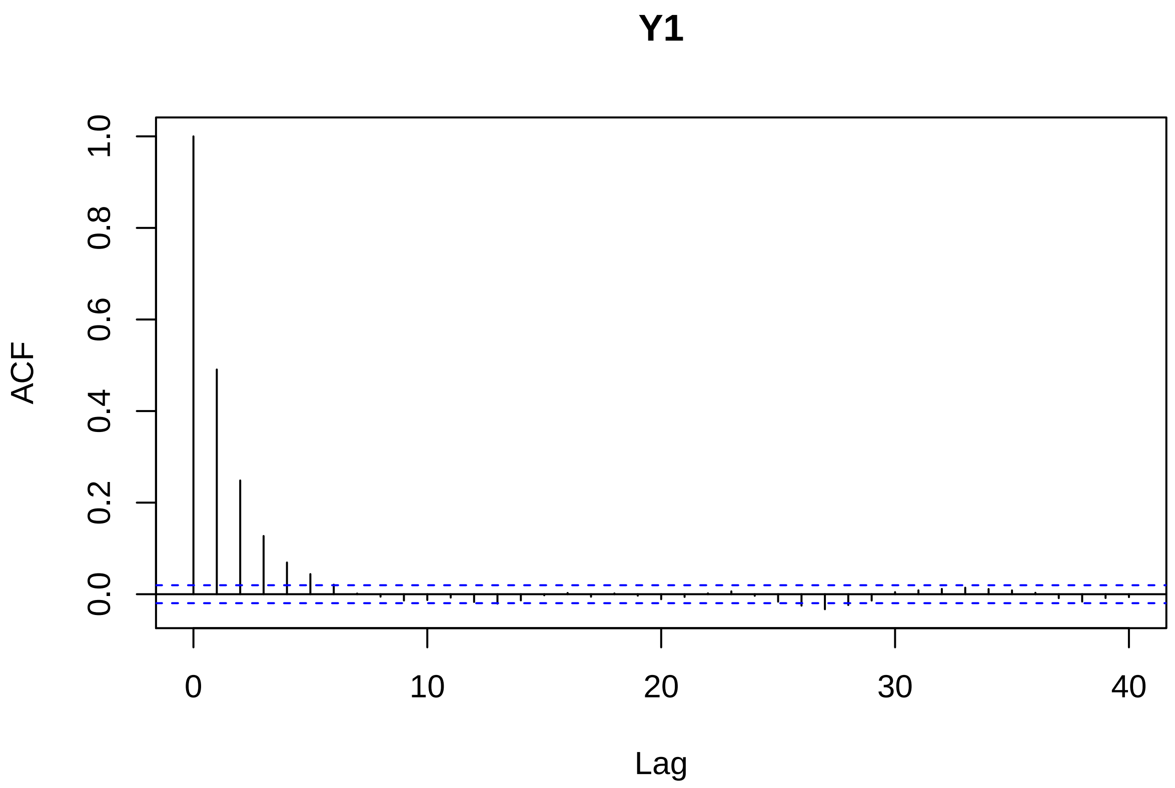
6. Comment on the mixing properties for your Gibbs sampler. Include at least one plot in support of your comments.

```
#Make the code in Problem 3 into a function
gibbs.chain <- function(rho_value=0.7,N=10000,mu_1=0,mu_2=0,sigma_1=1,sigma_2=1){
  Y <- matrix(0,N,2)
  sigma_test_1 <- sqrt(1-rho_value^2)*sigma_1
  sigma_test_2 <- sqrt(1-rho_value^2)*sigma_2
  for (i in 2:N) {
   y2 <- Y[i-1, 2]
   v1 <- mu_1 + rho_value * (y2 - mu_2) * sigma_1/sigma_2
   Y[i, 1] <- rnorm(1, v1, sigma_test_1)
   y1 <- Y[i, 1]
   v2 <- mu_2 + rho_value * (y1 - mu_1) * sigma_2/sigma_1
   Y[i, 2] <- rnorm(1, v2, sigma_test_2)}
  return(Y)
}
```

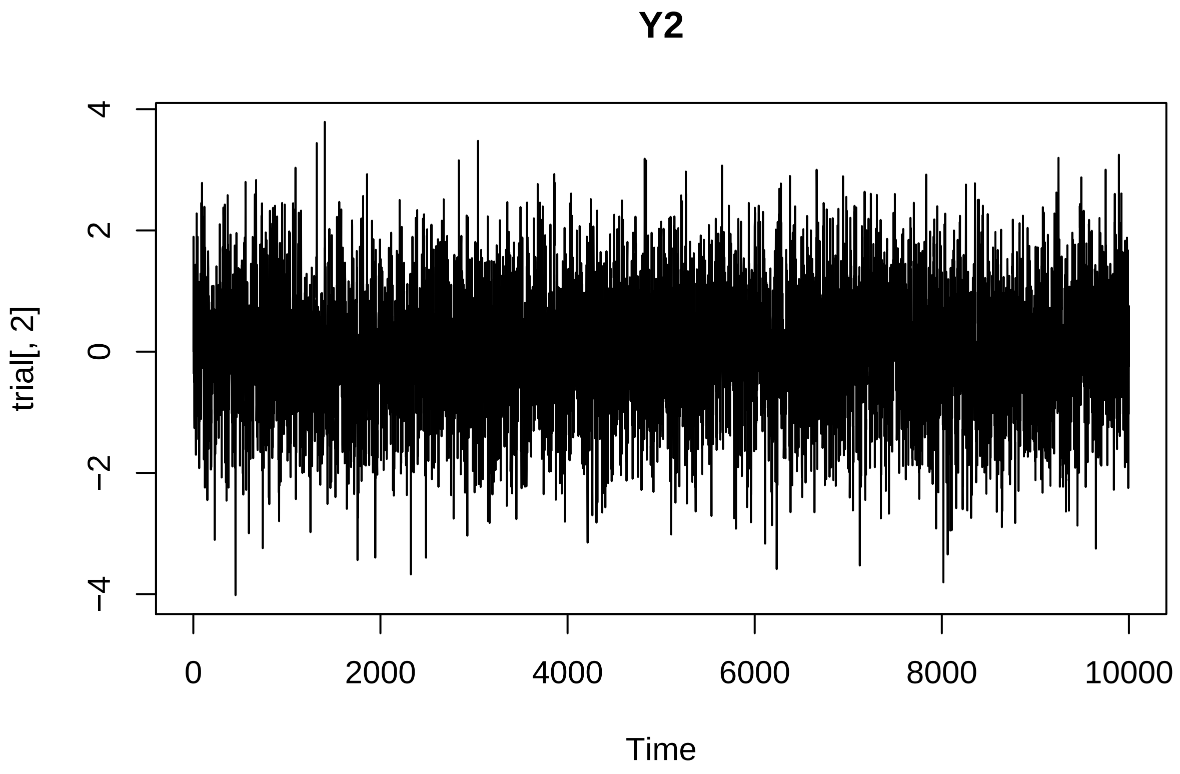
```
trial <- gibbs.chain()
plot.ts(trial[,1],main="Y1")
```



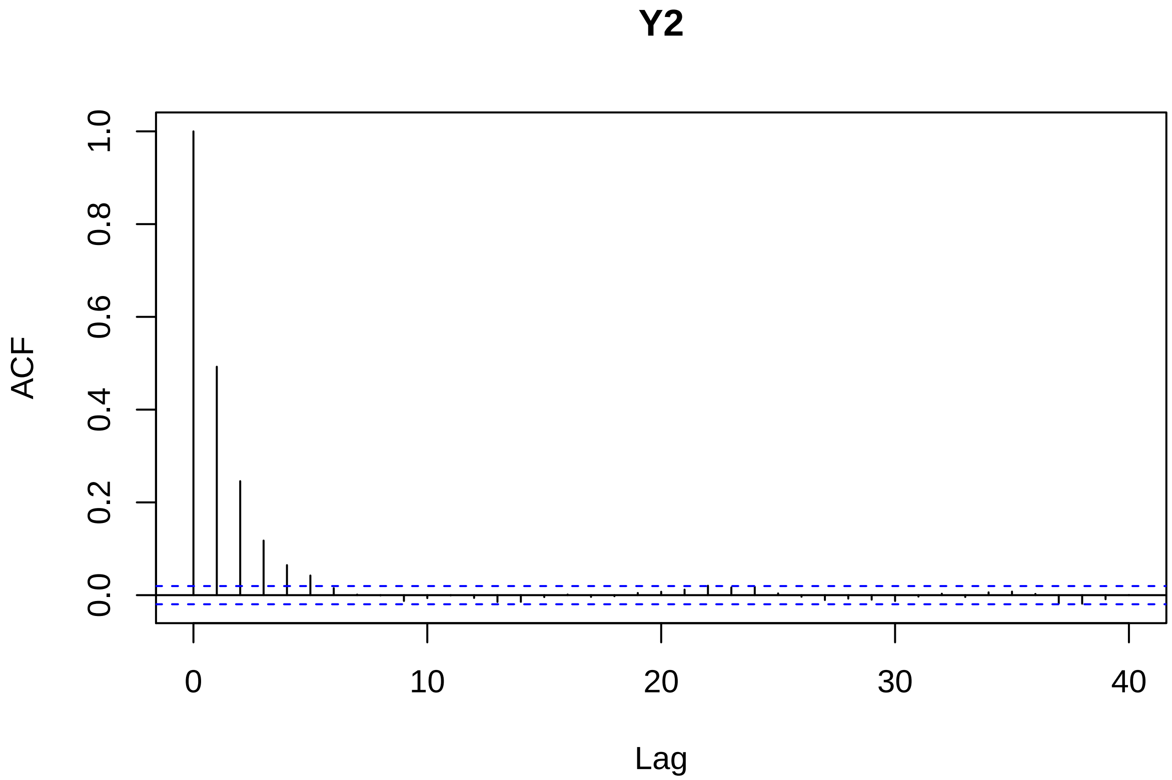
```r
acf(trial[,1],main="Y1")
```



```
plot.ts(trial[,2],main="Y2")
```



```r
acf(trial[,2],main="Y2")
```



```
#From thiese trace plots, we can judge the mixing properties for these data,
#If the distribution for the points is not changing along the chain,
#this chain might reach to stationarity situation.
#So from this plot, I think it is mixing well.
```