STA360 Homework 6 (Ken Ye)

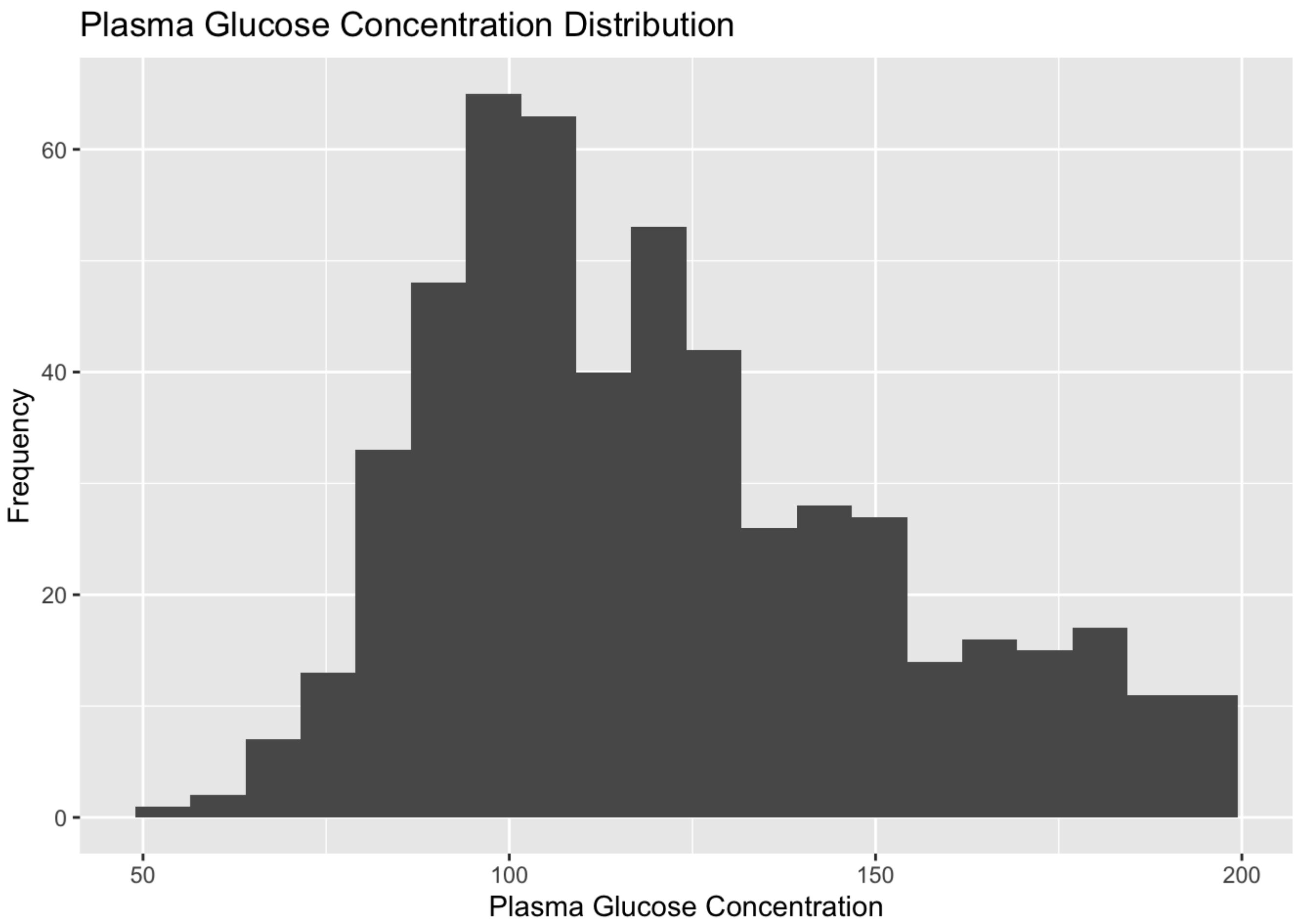
```
library(latex2exp)
library(coda)
library(ggplot2)
set.seed(0)
```

Exercise 6.2

```
# load data
glucose <- read.table("glucose.dat.txt")
```

Part a

```
# plot historgram
glucose |>
  ggplot(aes(x = V1)) +
  geom_histogram(bins = 20) +
  labs(title = "Plasma Glucose Concentration Distribution") +
  xlab("Plasma Glucose Concentration") +
  ylab("Frequency")
```



This empirical plasma glucose concentration distribution is not normal, as it”“s bimodal and significantly right-skewed.

Part b

(See hand-written page)

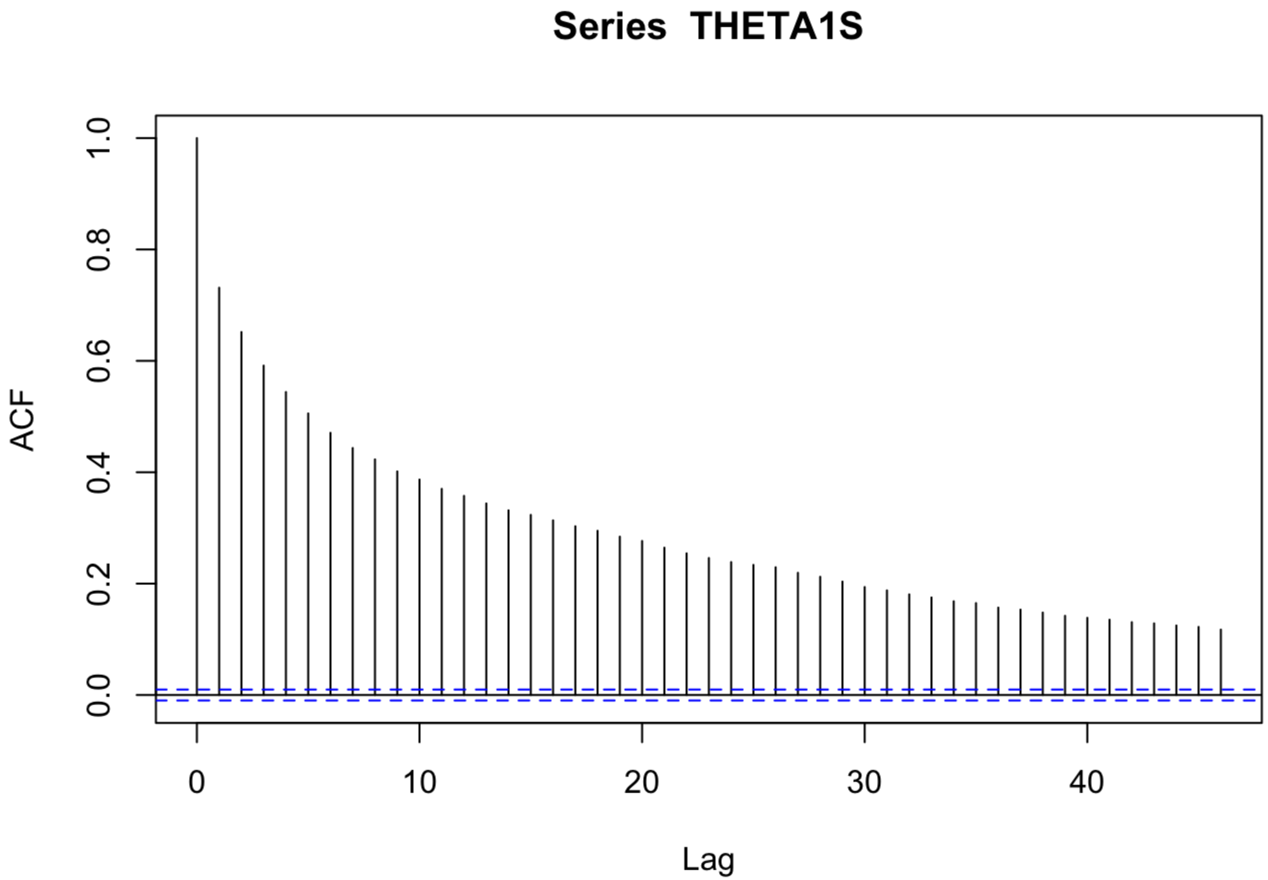
Part c

```r
Y <- glucose$V1
n <- length(Y)
nsim <- 5e4
burnin <- 1e4

# priors
a <- 1
b <- 1
mu0 <- 120
t20 <- 200
s20 <- 1000
nu0 <- 10

# storage vectors
THETA1 <- numeric(nsim - burnin)
THETA2 <- numeric(nsim - burnin)
Y.gibb <- numeric(nsim - burnin)

# starting values
p <- 0.5
theta1 <- mean(Y)
theta2 <- mean(Y)
s21 <- var(Y)
s22 <- var(Y)

# Gibbs sampling
for (s in 1:nsim) {
  # simulate X
  p.x1 <- p * dnorm(Y, theta1, sqrt(s21))
  p.x2 <- (1 - p) * dnorm(Y, theta2, sqrt(s22))
  p.xi <- p.x1 / (p.x1 + p.x2)
  X <- rbinom(n, 1, p.xi)

  # calcuate group-specific summary statistics
  n1 <- sum(X)
  n2 <- n - n1
```

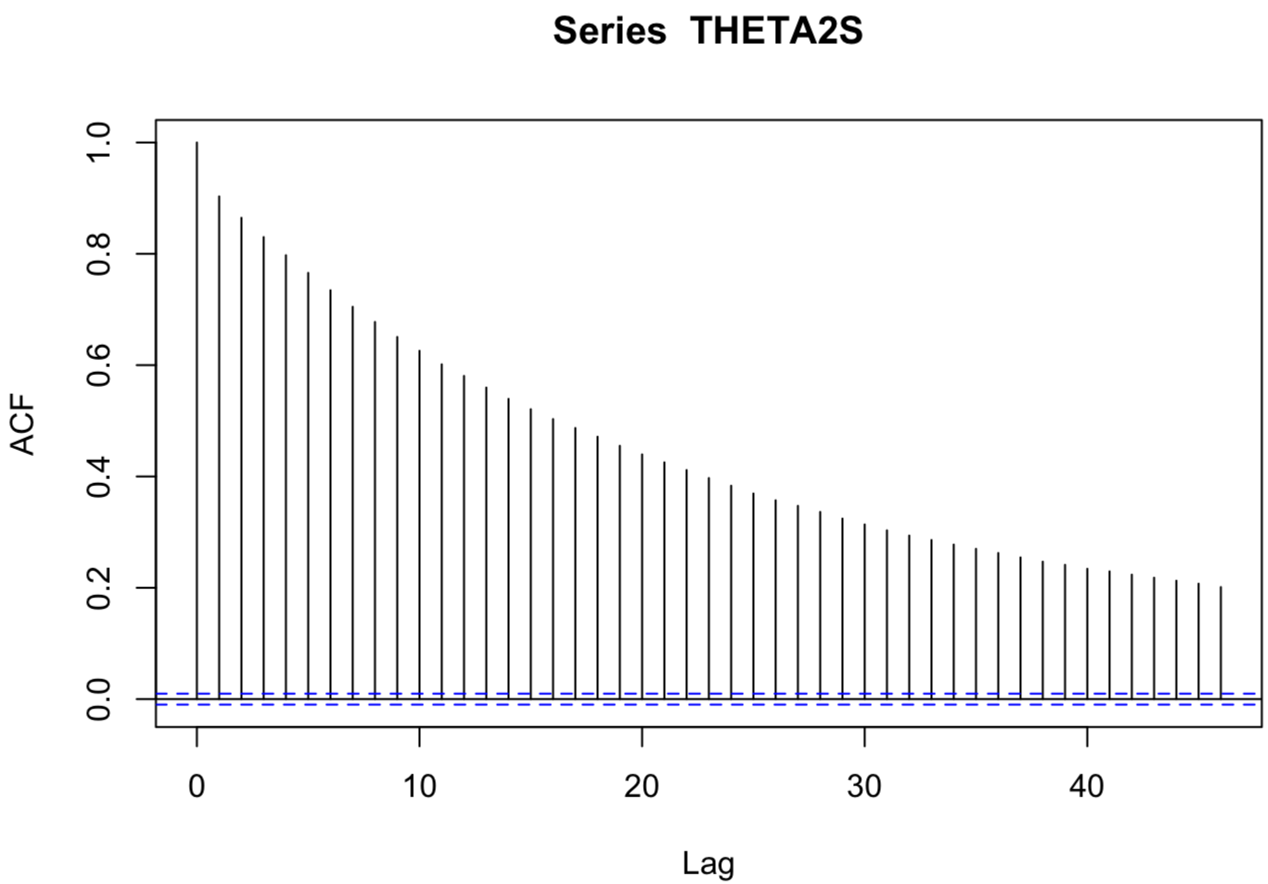
```r
y1 <- Y[X == 1]
y2 <- Y[X == 0]
ybar1 <- mean(y1)
ybar2 <- mean(y2)
yvar1 <- var(y1)
yvar2 <- var(y2)

# simulate p
p <- rbeta(1, a + n1, b + n2)

# simulate theta1
t2n1 <- 1 / (1 / t20 + n1 / s21)
mun1 <- (mu0 / t20 + n1 * ybar1 / s21) / (1 / t20 + n1 / s21)
theta1 <- rnorm(1, mun1, sqrt(t2n1))

# simulate theta2
t2n2 <- 1 / (1 / t20 + n2 / s22)
mun2 <- (mu0 / t20 + n2 * ybar2 / s22) / (1 / t20 + n2 / s22)
theta2 <- rnorm(1, mun2, sqrt(t2n2))

# simulate sigma^2 1
nun1 <- nu0 + n1
s2n1 <- (nu0 * s20 + (n1 - 1) * yvar1 + n1 * (ybar1 - theta1)^2) / nun1
s21 <- 1 / rgamma(1, nun1 / 2, s2n1 * nun1 / 2)

# simulate sigma^2 2
nun2 <- nu0 + n2
s2n2 <- (nu0 * s20 + (n2 - 1) * yvar2 + n2 * (ybar2 - theta2)^2) / nun2
s22 <- 1 / rgamma(1, nun2 / 2, s2n2 * nun2 / 2)

# simulate posterior
x.gibb <- runif(1) < p
y.gibb <- ifelse(x.gibb,
                rnorm(1, theta1, sqrt(s21)),
                rnorm(1, theta2, sqrt(s22)))

# store values
if (s > burnin){
```

```
    THETA1[s - burnin] <- theta1
    THETA2[s - burnin] <- theta2
    Y.gibb[s - burnin] <- y.gibb
  }
}
```

```
THETA1S <- pmin(THETA1, THETA2)
THETA2S <- pmax(THETA1, THETA2)

# plot autocorrelation
acf(THETA1S)
```




```
acf(THETA2S)
```




```
# print effective size
print("The effective size of theta (1) (s) is: ")
```

```
## [1] "The effective size of theta (1) (s) is: "
```

```
print(effectiveSize(THETA1S))
```

```
##     var1
## 1280.576
```

```
print("The effective size of theta (2) (s) is: ")
```

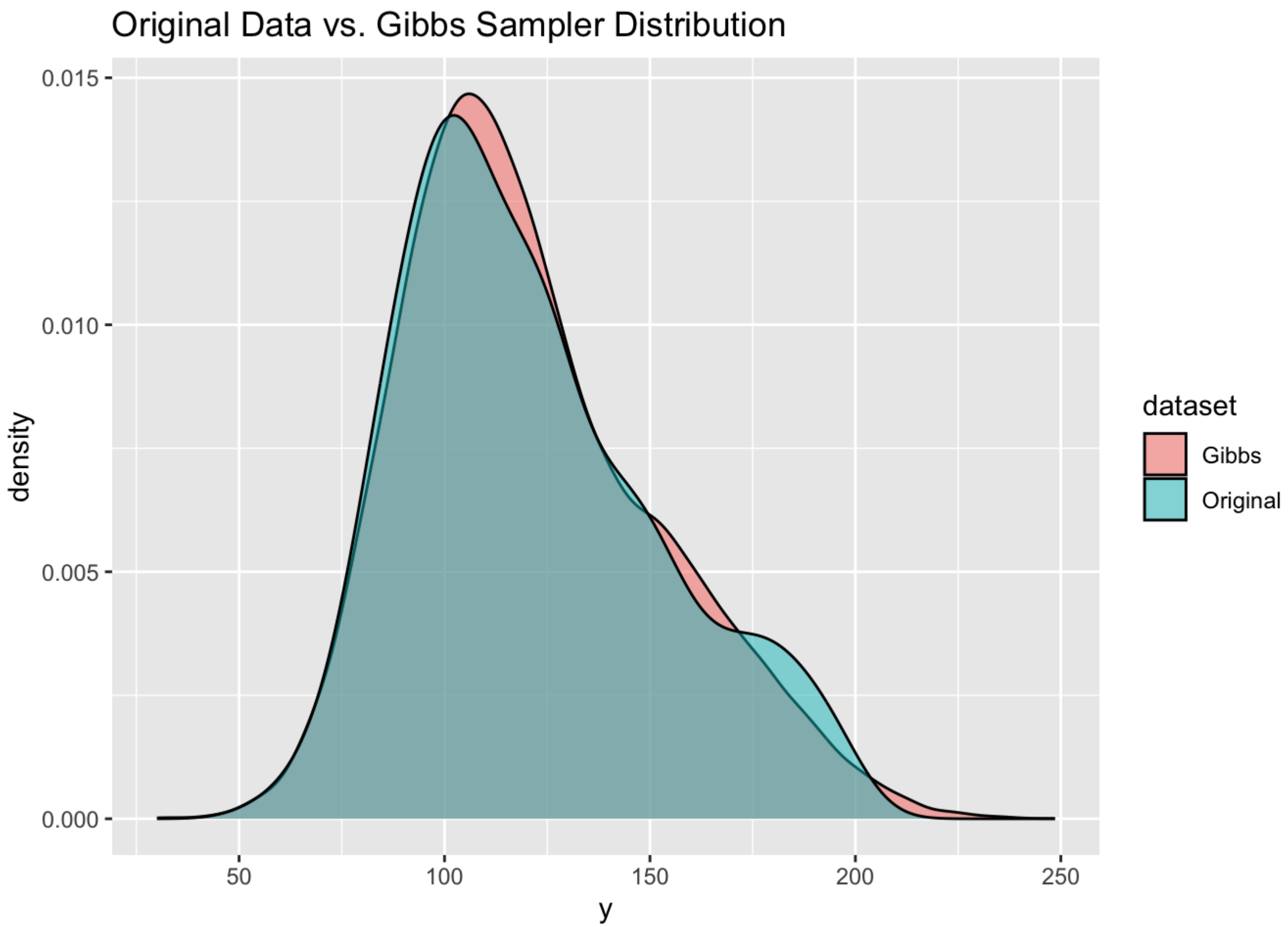
```
## [1] "The effective size of theta (2) (s) is: "
```

```
print(effectiveSize(THETA2S))
```

```
##     var1
## 875.6929
```

Part d

```
#
Y.compare = rbind(data.frame(y = Y.gibb, dataset = "Gibbs"),
              data.frame(y = Y, dataset = "Original"))
# plot
ggplot(Y.compare, aes(x = y, fill = dataset)) +
  geom_density(alpha = 0.5) +
  labs(title = "Original Data vs. Gibbs Sampler Distribution")
```



Based on the graph, the densities of the original and the Gibbs sampler are very close, indicating that this two-component mixture model is a good fit for the glucose data.

Exercise 6.3

```
# load data
divorce <- read.table("divorce.dat.txt")
```

```
# function for simulating from a constrained normal distribution with mean mean and standard deviation sd, constr
ained to lie in the interval (a,b)
rcnorm <- function(n, mean = 0, sd = 1, a = -Inf, b = Inf){
  u <- runif(n, pnorm((a - mean) / sd), pnorm((b - mean) / sd) )
  mean + sd * qnorm(u)
}
```

Part a

(See hand-written page)

Part b

(See hand-written page)

Part c

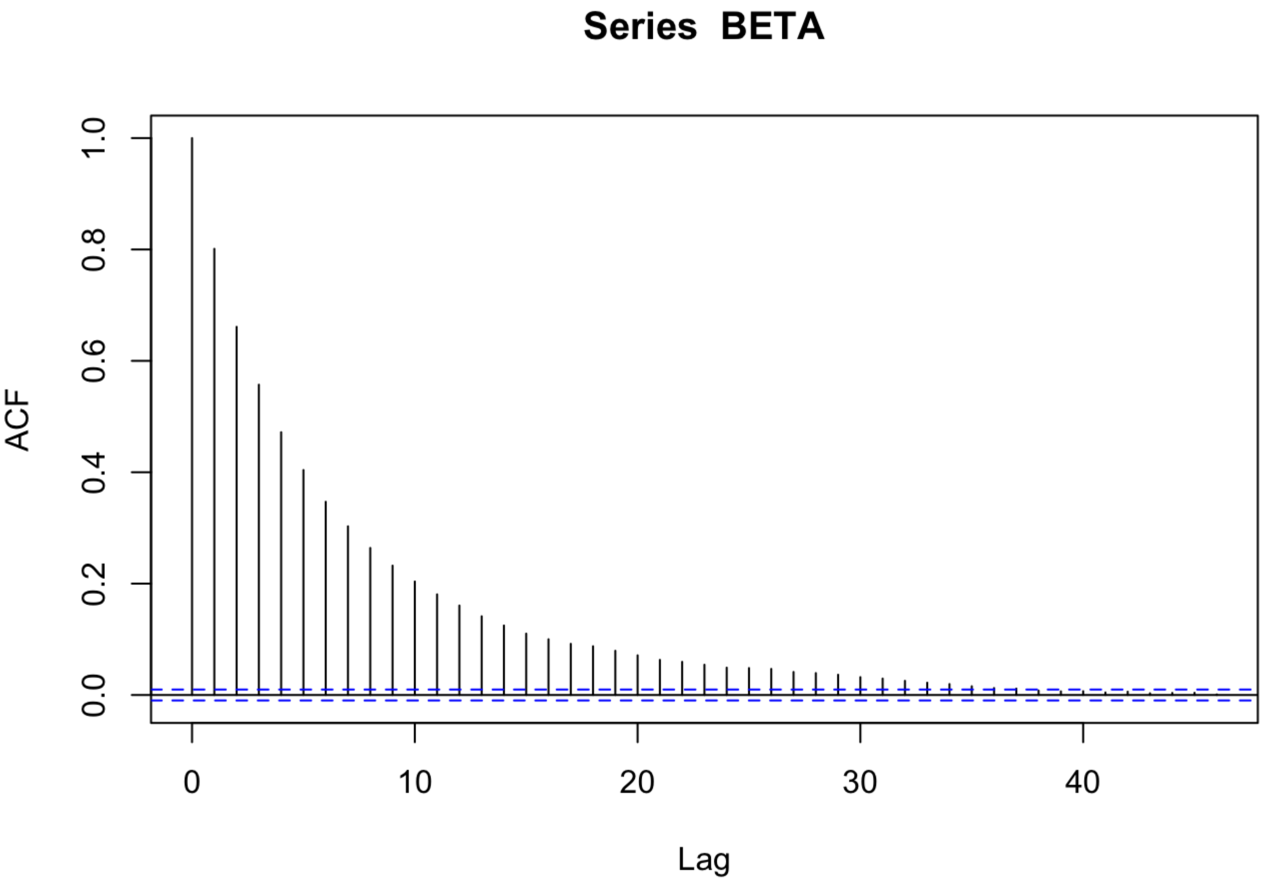
```
Y <- divorce$V1
X <- divorce$V2
n <- length(Y)
nsim <- 5e4
burnin <- 1e4

# priors
t2b <- 16
t2c <- 16

# storage vectors
Z <- rep(list(0 * length(X)), times = nsim - burnin)
BETA <- numeric(nsim - burnin)
C <- numeric(nsim - burnin)

# starting values
z <- rep(0, n)
beta <- 0
c <- 0

# Gibbs sampling
for (s in 1:nsim) {
  # simulate beta
  mubn <- sum(z*X) / (sum(X^2) + 1/ (t2b))
  t2bn <- 1 / (sum(X^2) - 1 / t2b)
  beta <- rnorm(1, mubn, sqrt(t2bn))

  # simulate c
  z0 <- subset(z, Y == 0)
  z1 <- subset(z, Y == 1)
  a <- max(z0)
  b <- min(z1)
  c <- rcnorm(1, 0, sqrt(t2c), a, b)

  # simulate z
  for (i in 1:n){
```

```
    if(Y[i] == 1){
      z[i] <- rcnorm(1, beta * X[i], 1, c, Inf)
    }
    else{
      z[i] <- rcnorm(1, beta * X[i], 1, -Inf, c)
    }
  }

  # store values
  if (s > burnin){
    Z[[s - burnin]] <- z
    BETA[s - burnin] <- beta
    C[s - burnin] <- c
  }
}
```

```
# beta effective size and autocorrelation
effectiveSize(BETA)
```

```
##     var1
## 3338.234
```

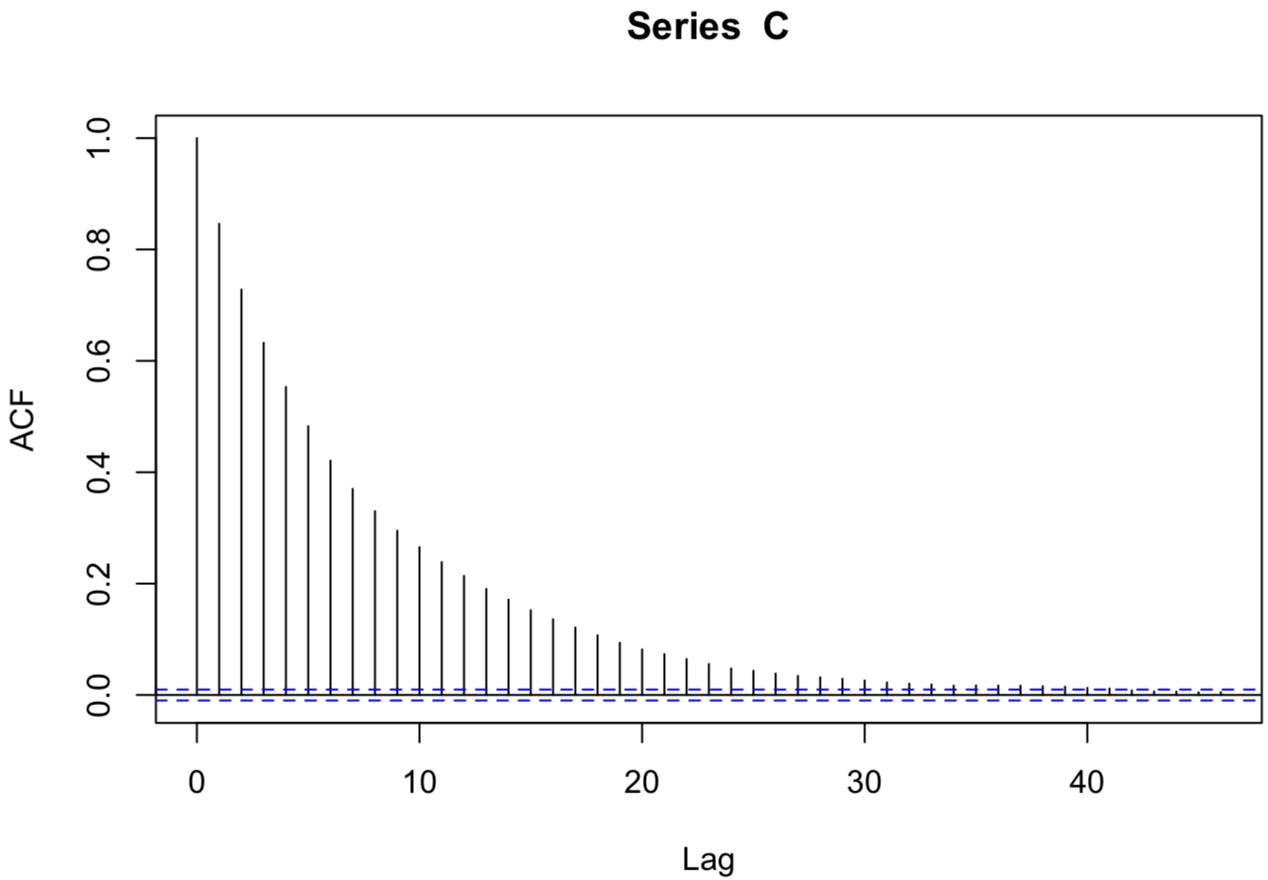
```
acf(BETA)
```




```
# c effective size and autocorrelation
effectiveSize(C)
```

```
##     var1
## 2611.212
```

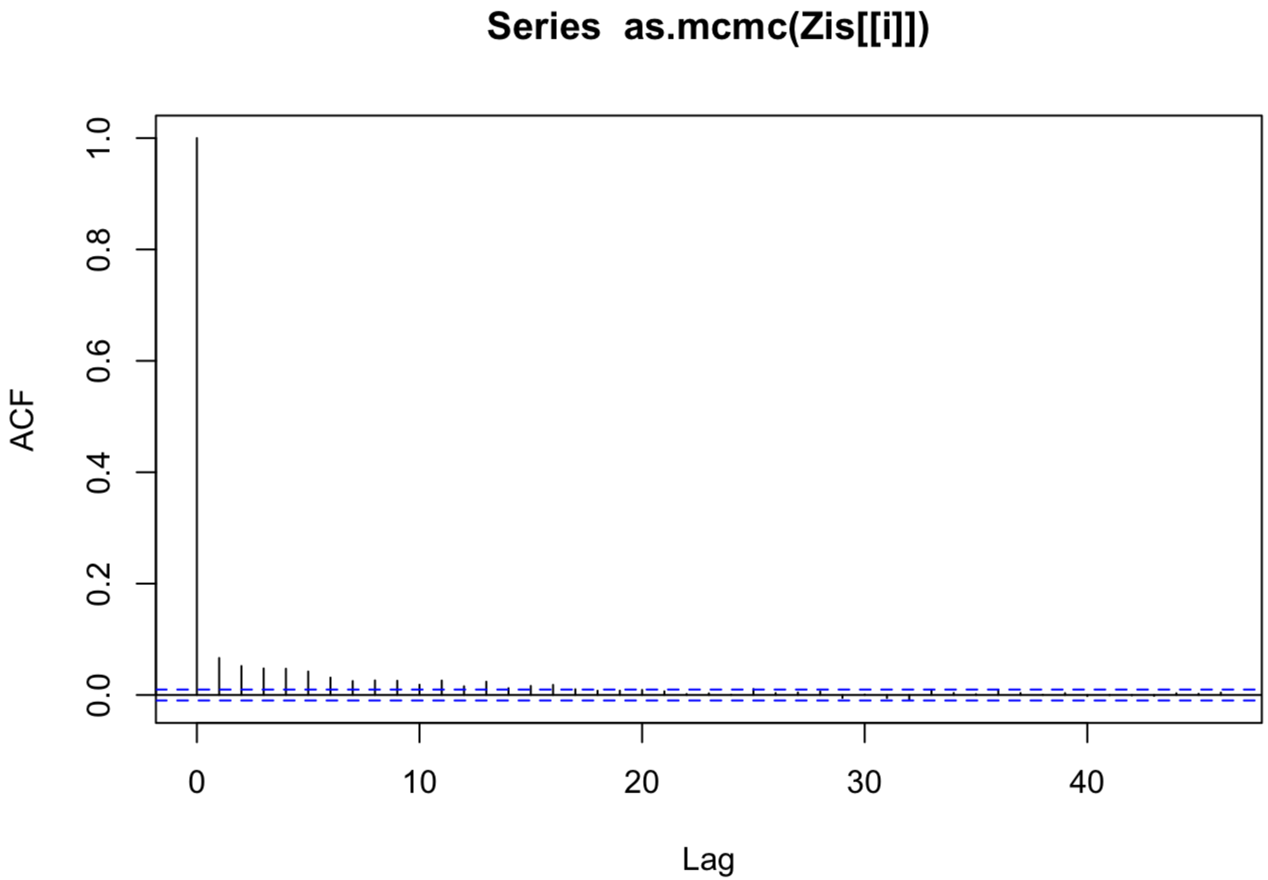
```
acf(C)
```

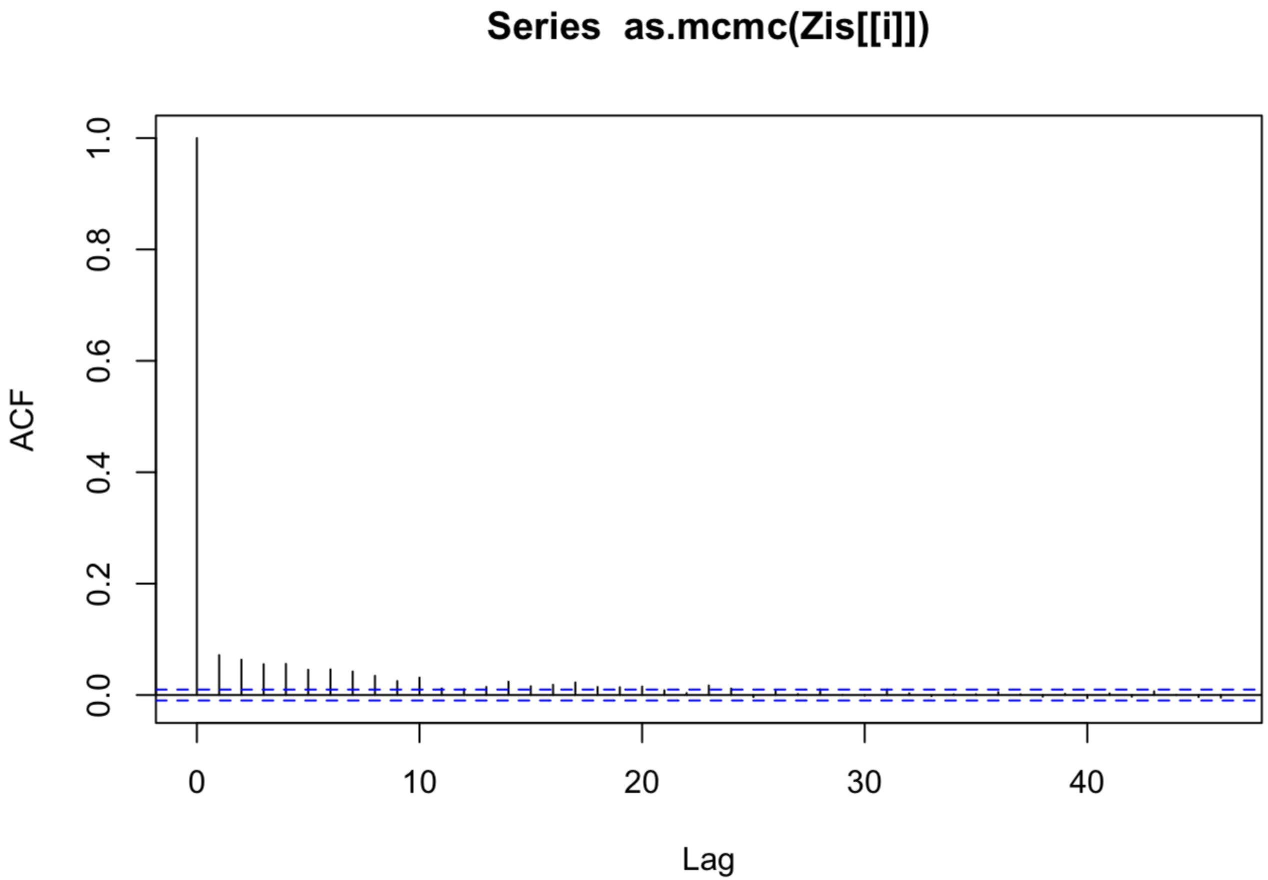

Series C
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
10
20
30
40
Lag

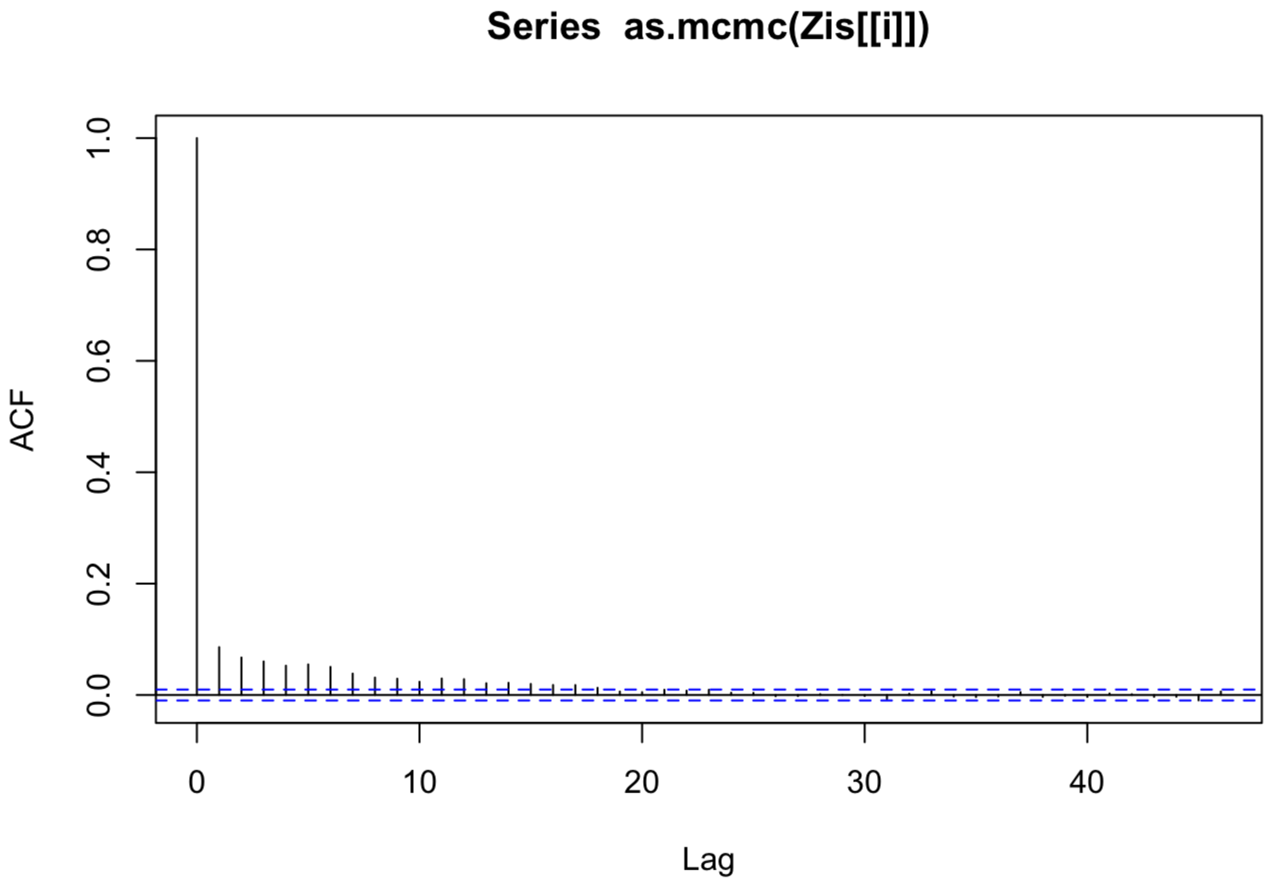
```r
# zi's effective size
Zis <- rep(list(0 * length(Z)), times = n)
for (i in 1:length(Z)){
  for (j in 1:n){
     Zis[[j]][i] <- Z[[i]][j]
  }
}
Zis.eff <- rep(0, times = length(Zis))
for (i in 1:length(Zis)){
  Zis.eff[i] <- effectiveSize(unlist(Zis[[i]]))
}
Zis.eff
```

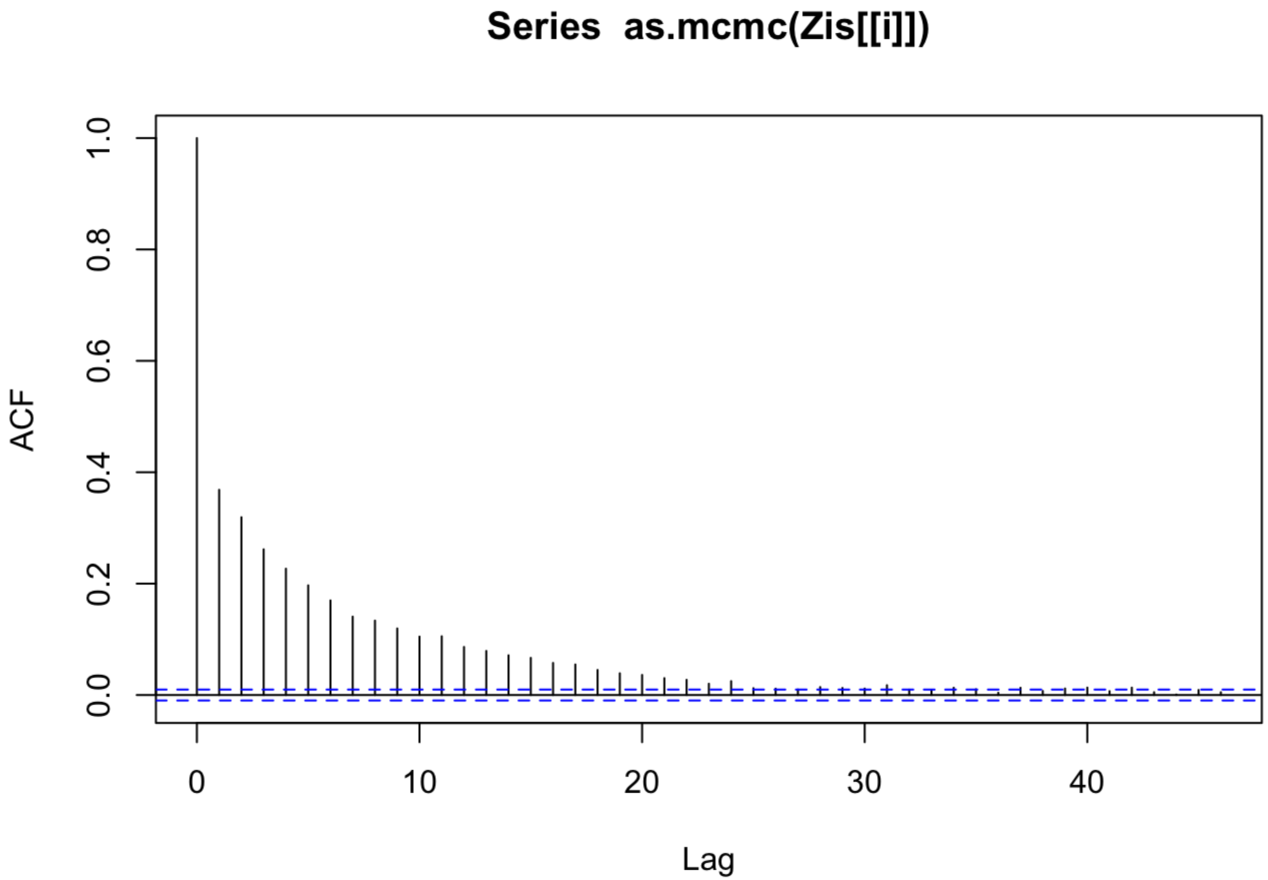
```
##  [1] 19027.034 17233.683 16214.720  5989.295 17960.810 19064.730 18205.878
##  [8]  6995.131 18500.697  6292.201 19794.650  3498.864  6996.895  7192.662
## [15]  6925.164 19898.214  6836.595  6516.958 17812.840 18809.866  6719.969
## [22] 19622.109 19797.833  6620.454 19957.361
```

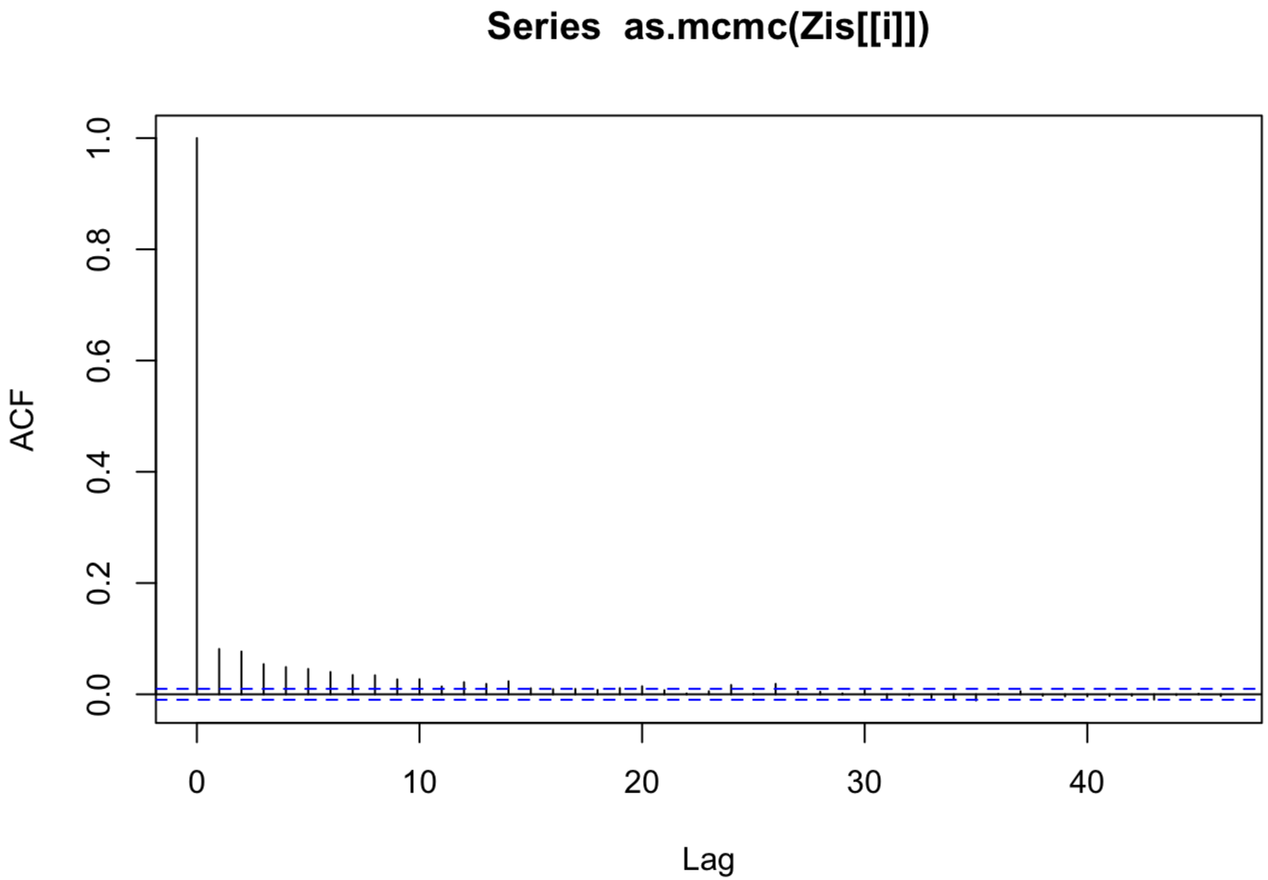
```r
# zi's autocorrelation
for (i in 1:length(Zis)){
  acf(as.mcmc(Zis[[i]]))
}
```

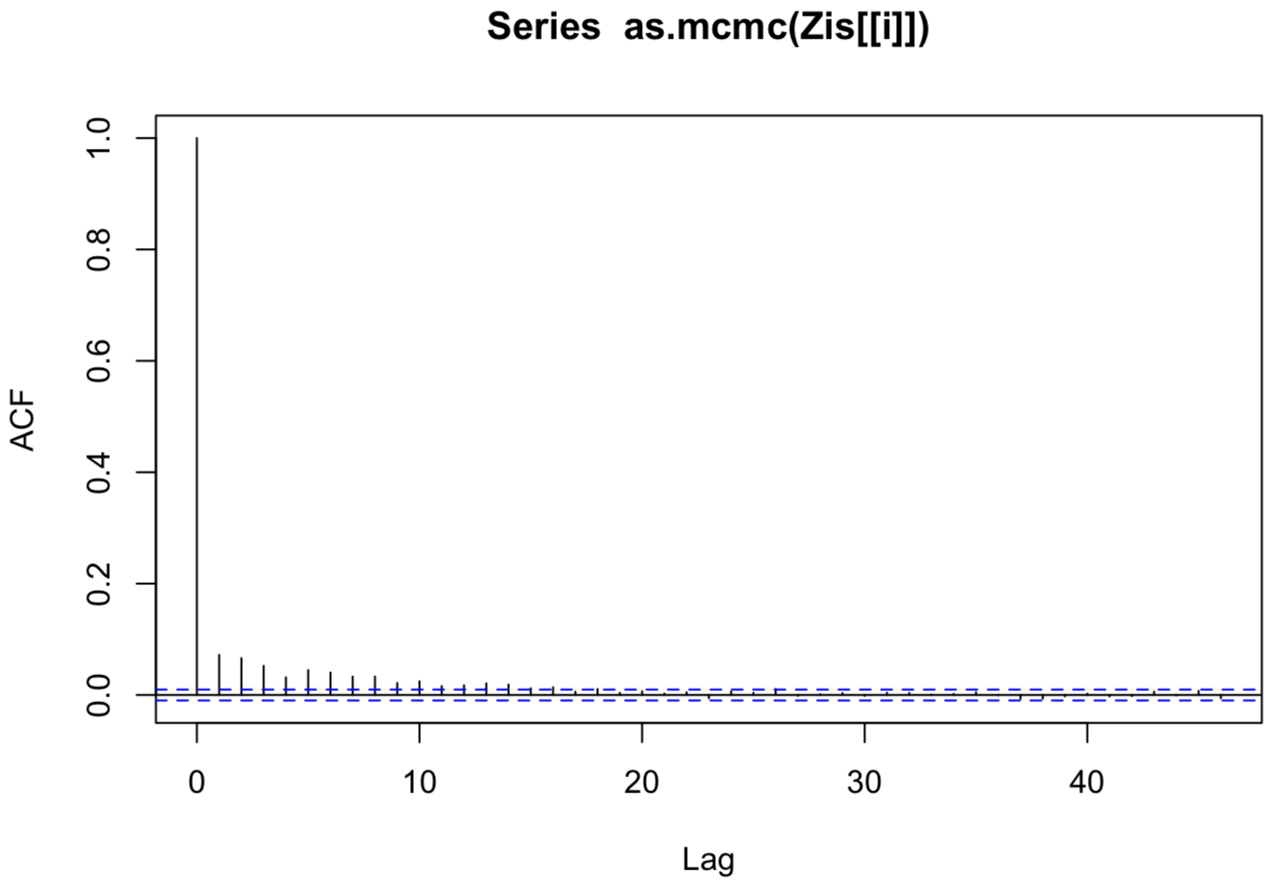

Series as.mcmc(Zis[[i]])
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
10
20
30
40
Lag

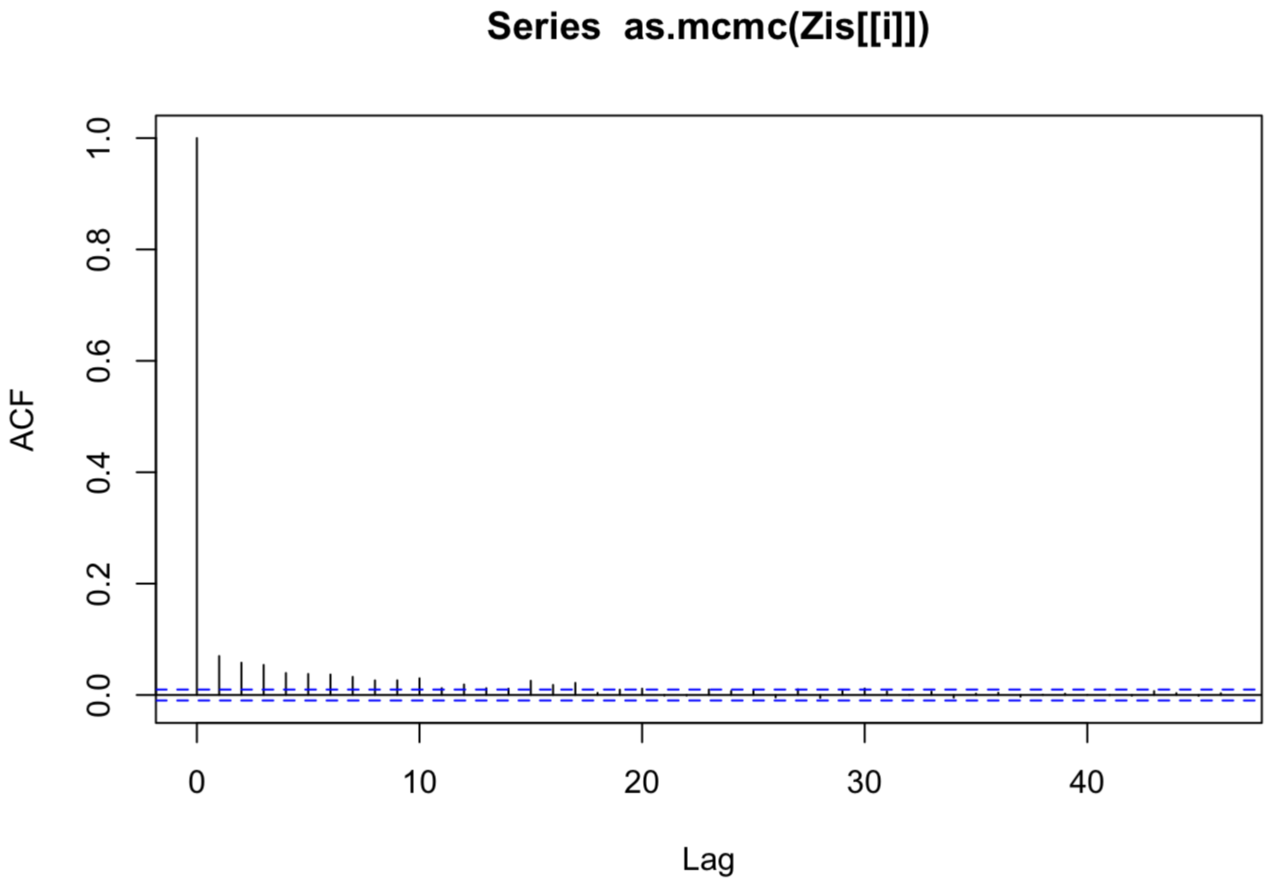

Series as.mcmc(Zis[[i]])
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
10
20
30
40
Lag

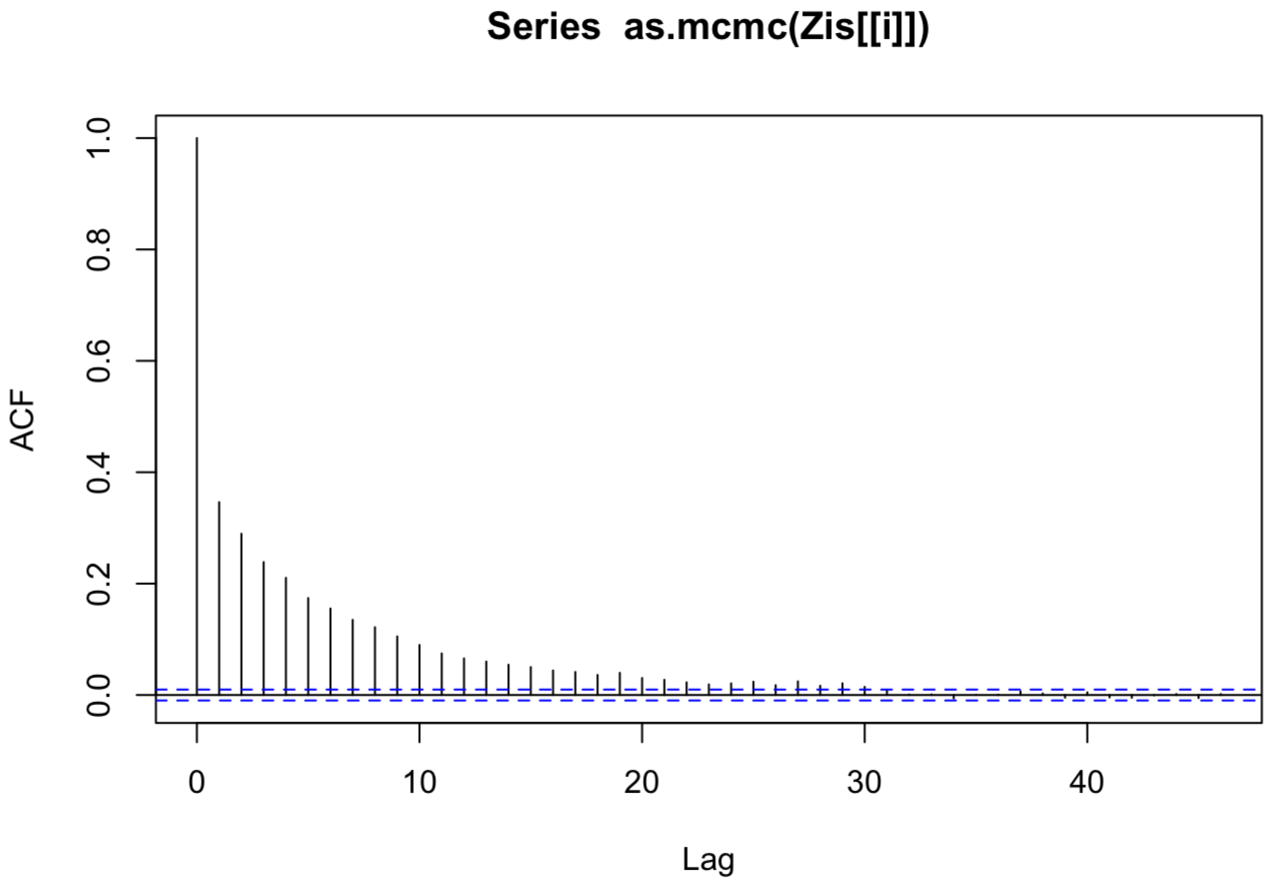

Series as.mcmc(Zis[[i]])
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
10
20
30
40
Lag

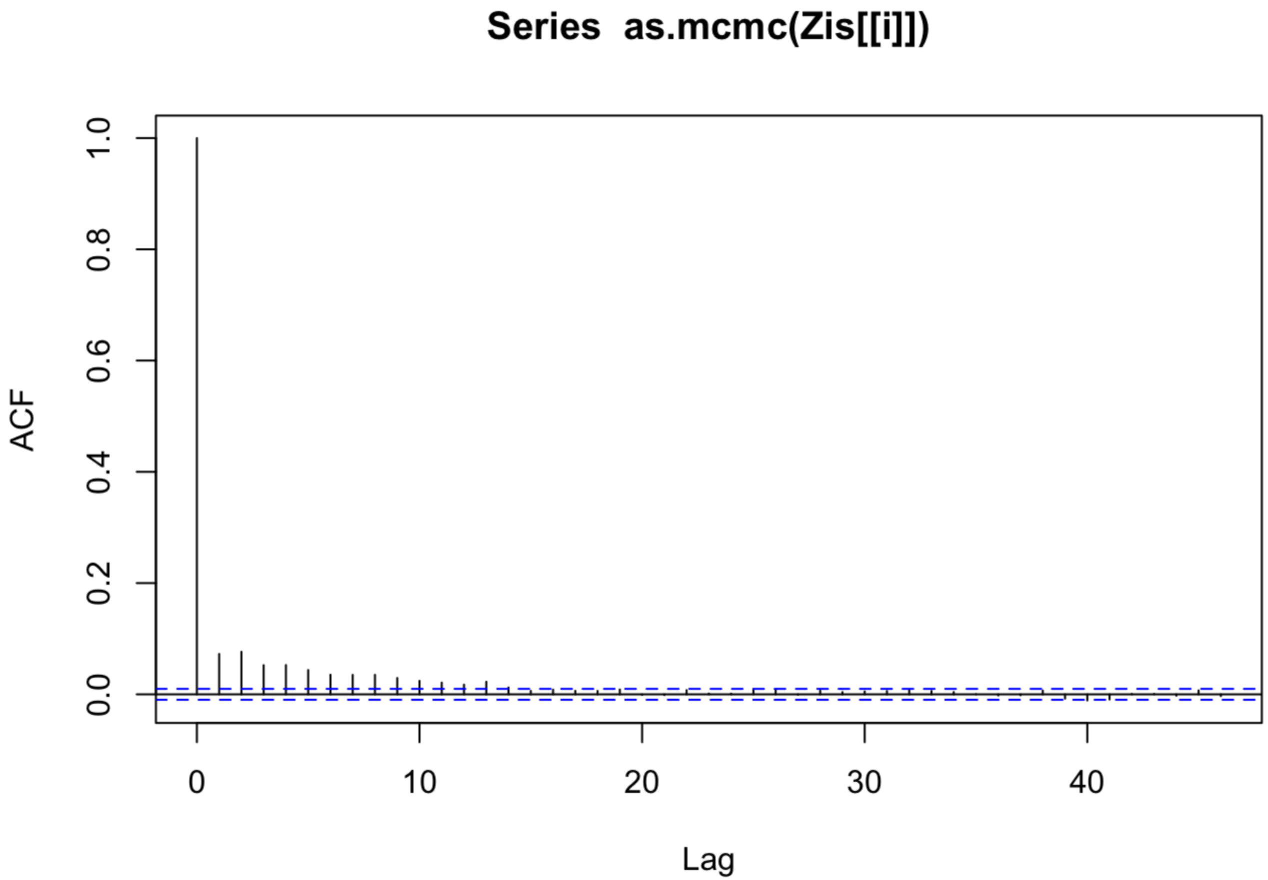

Series as.mcmc(Zis[[i]])
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
10
20
30
40
Lag

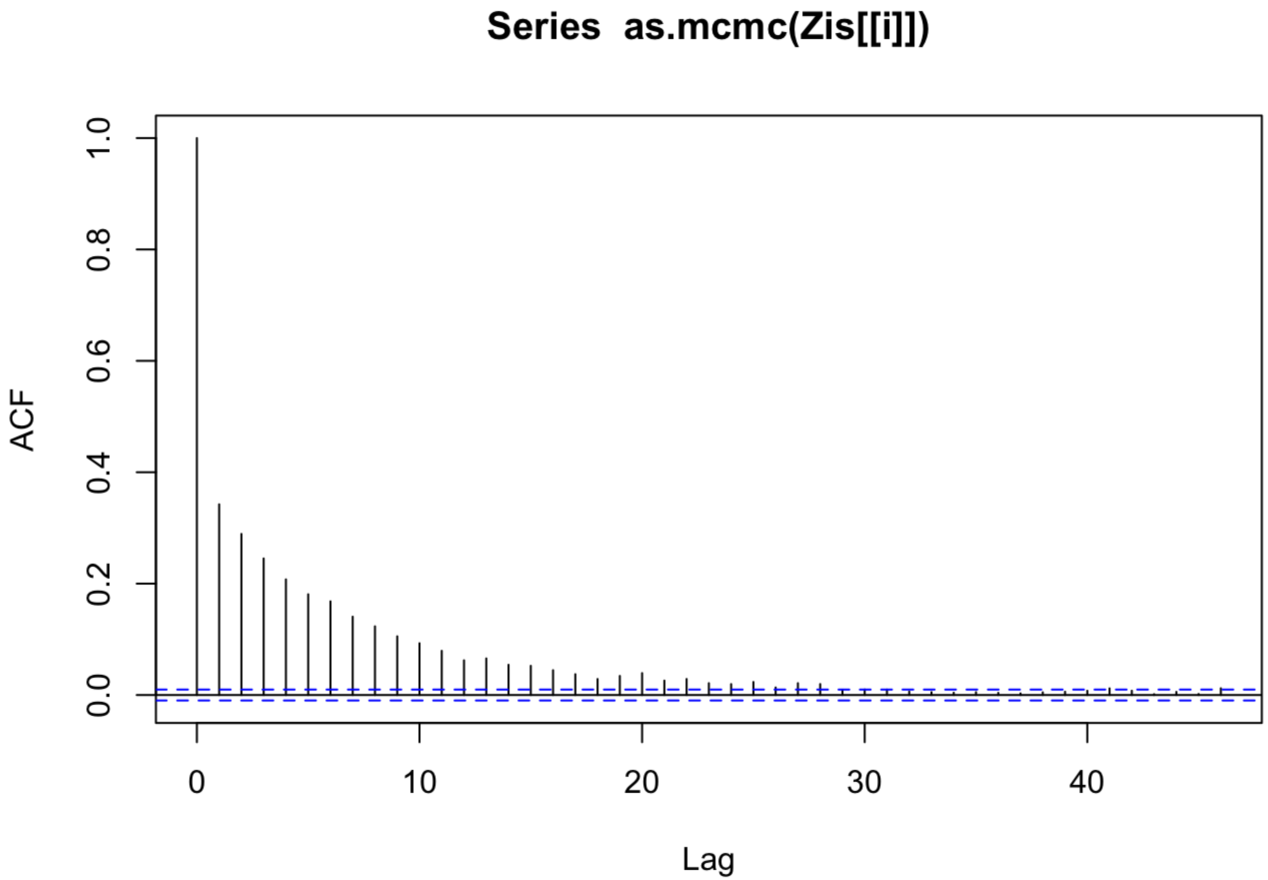

Series as.mcmc(Zis[[i]])
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
10
20
30
40
Lag

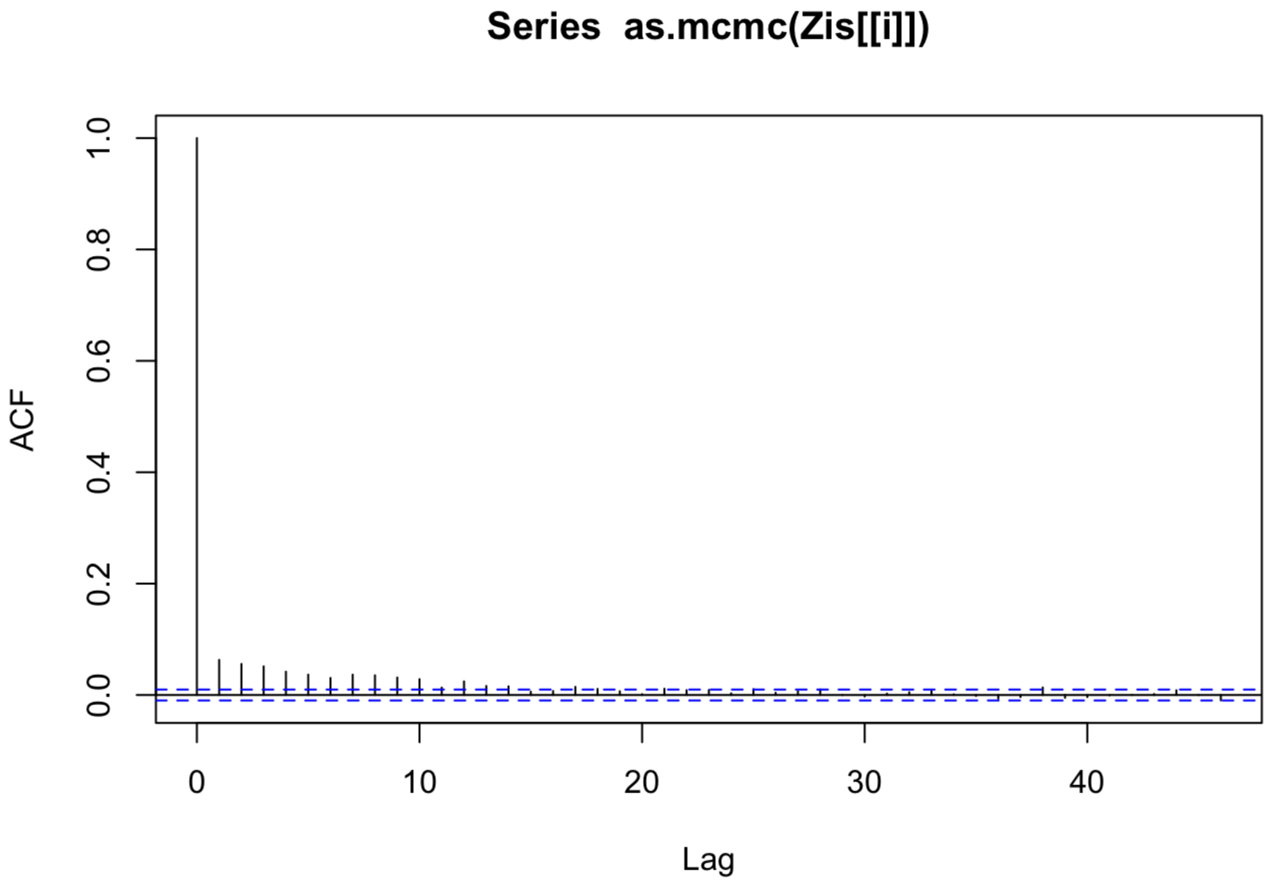

Series as.mcmc(Zis[[i]])
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
10
20
30
40
Lag

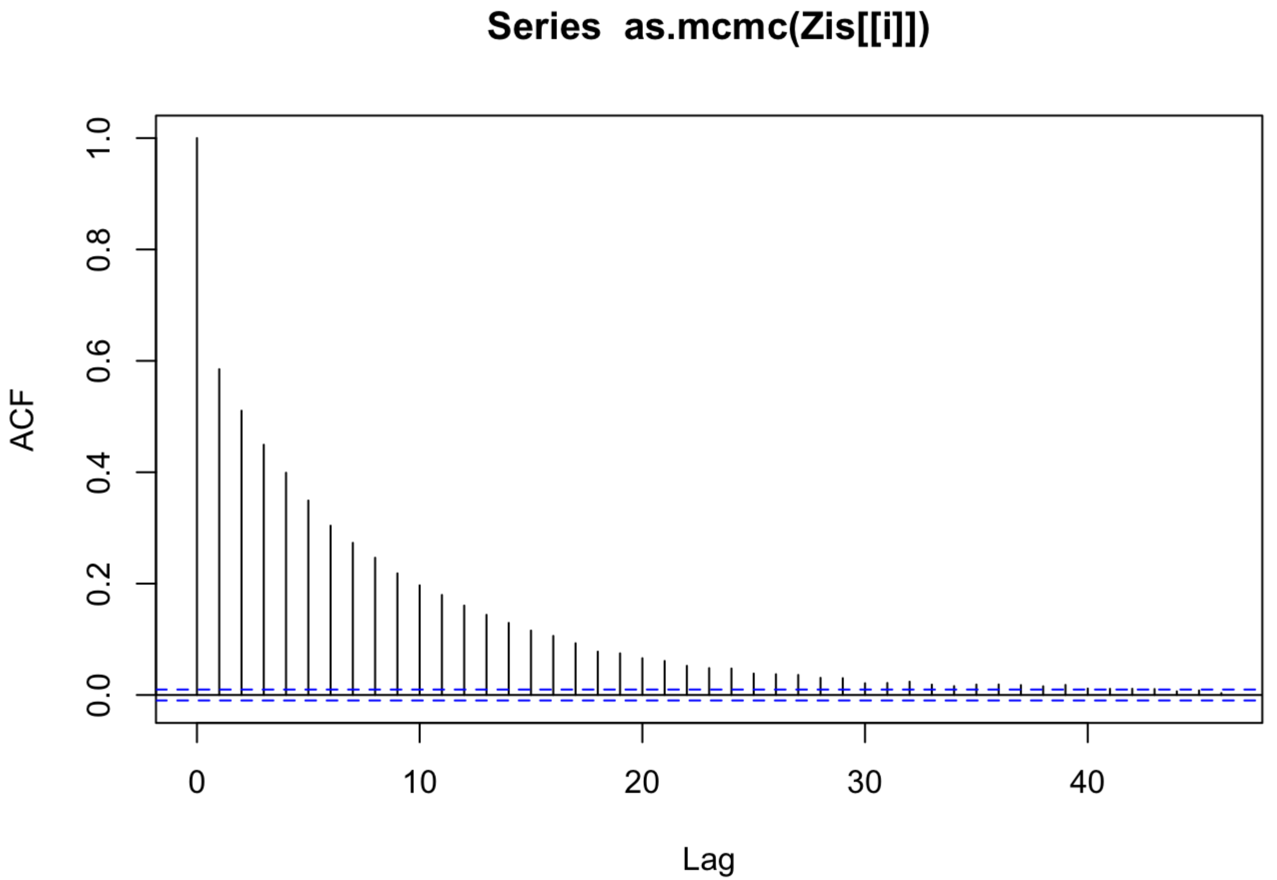

Series as.mcmc(Zis[[i]])
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
10
20
30
40
Lag

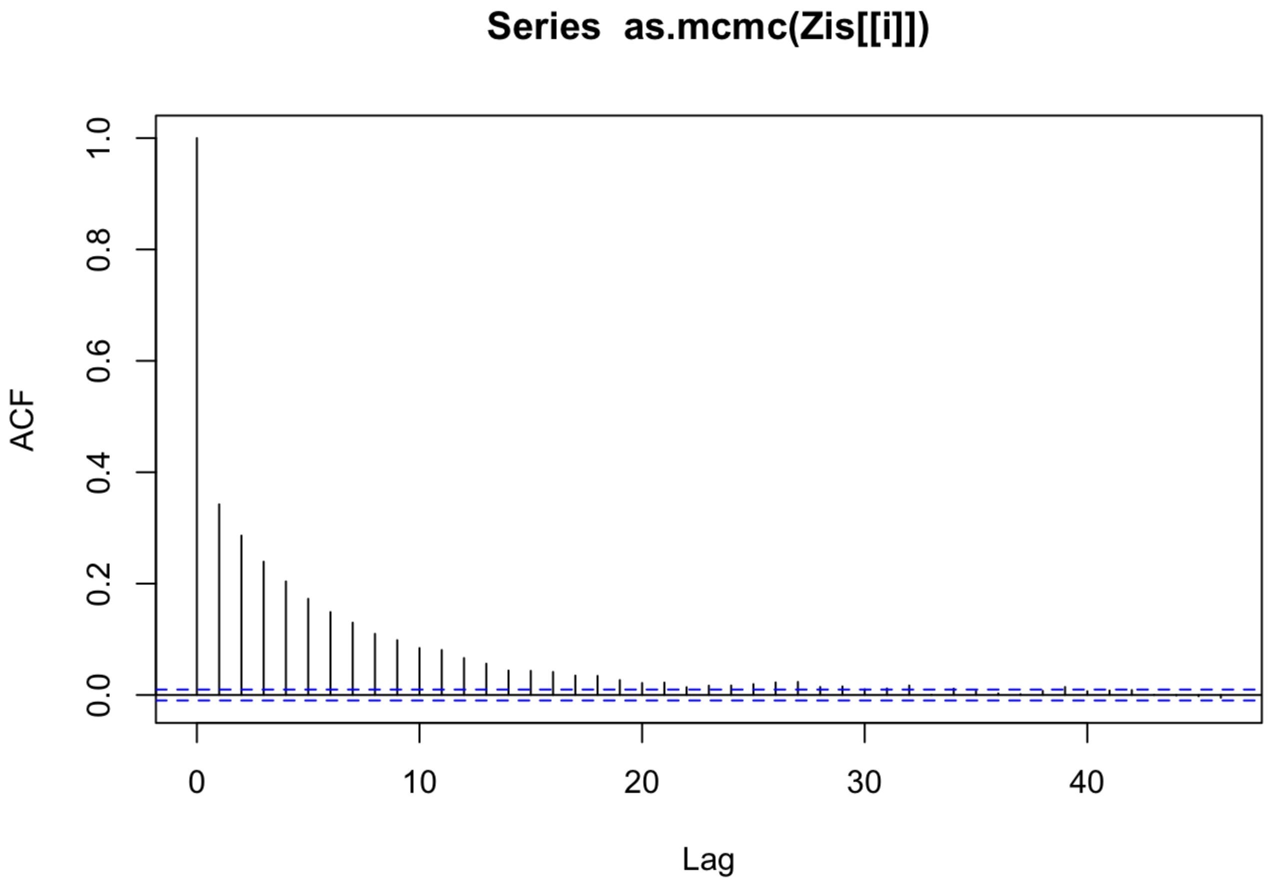

Series as.mcmc(Zis[[i]])
ACF
Lag
0.0
0.2
0.4
0.6
0.8
1.0
0
10
20
30
40

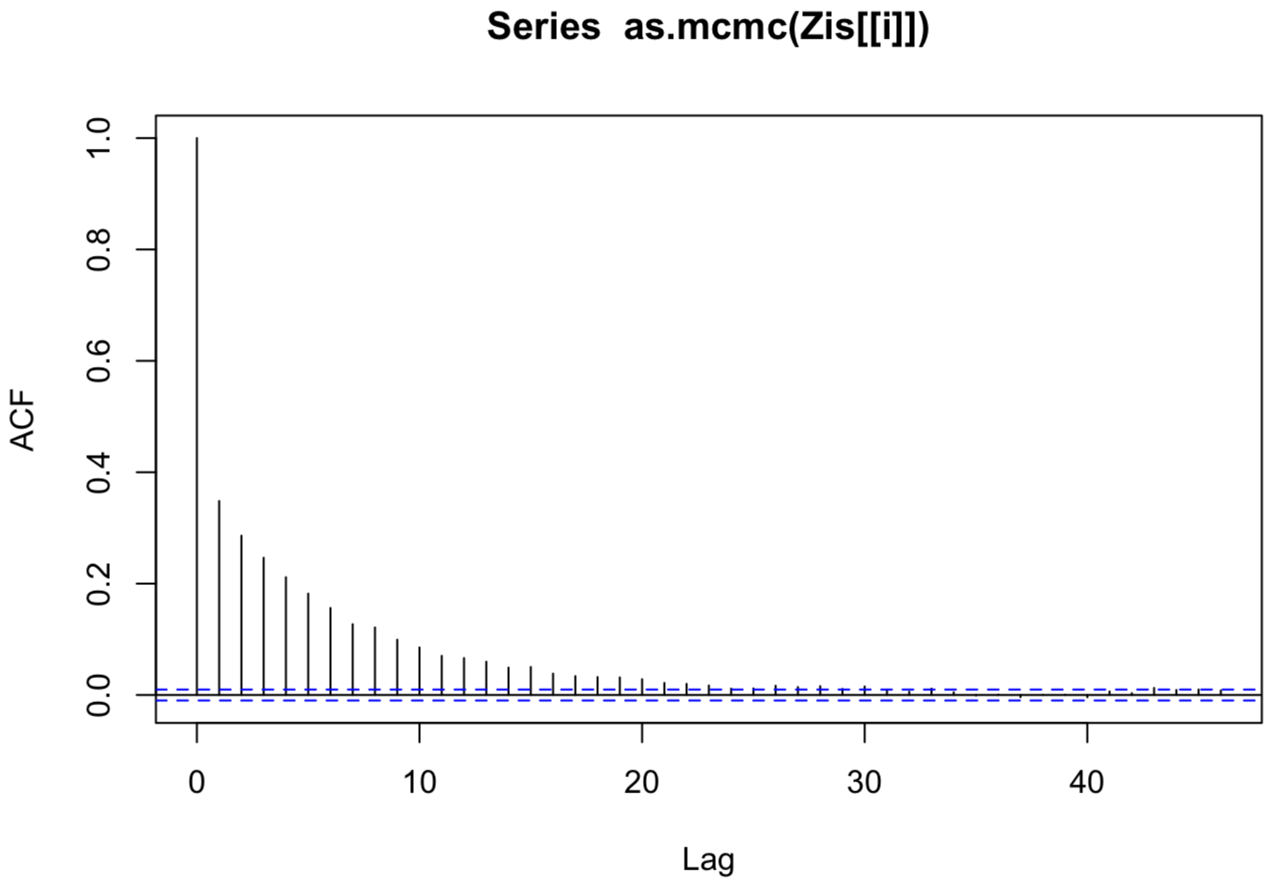

Series as.mcmc(Zis[[i]])
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
10
20
30
40
Lag

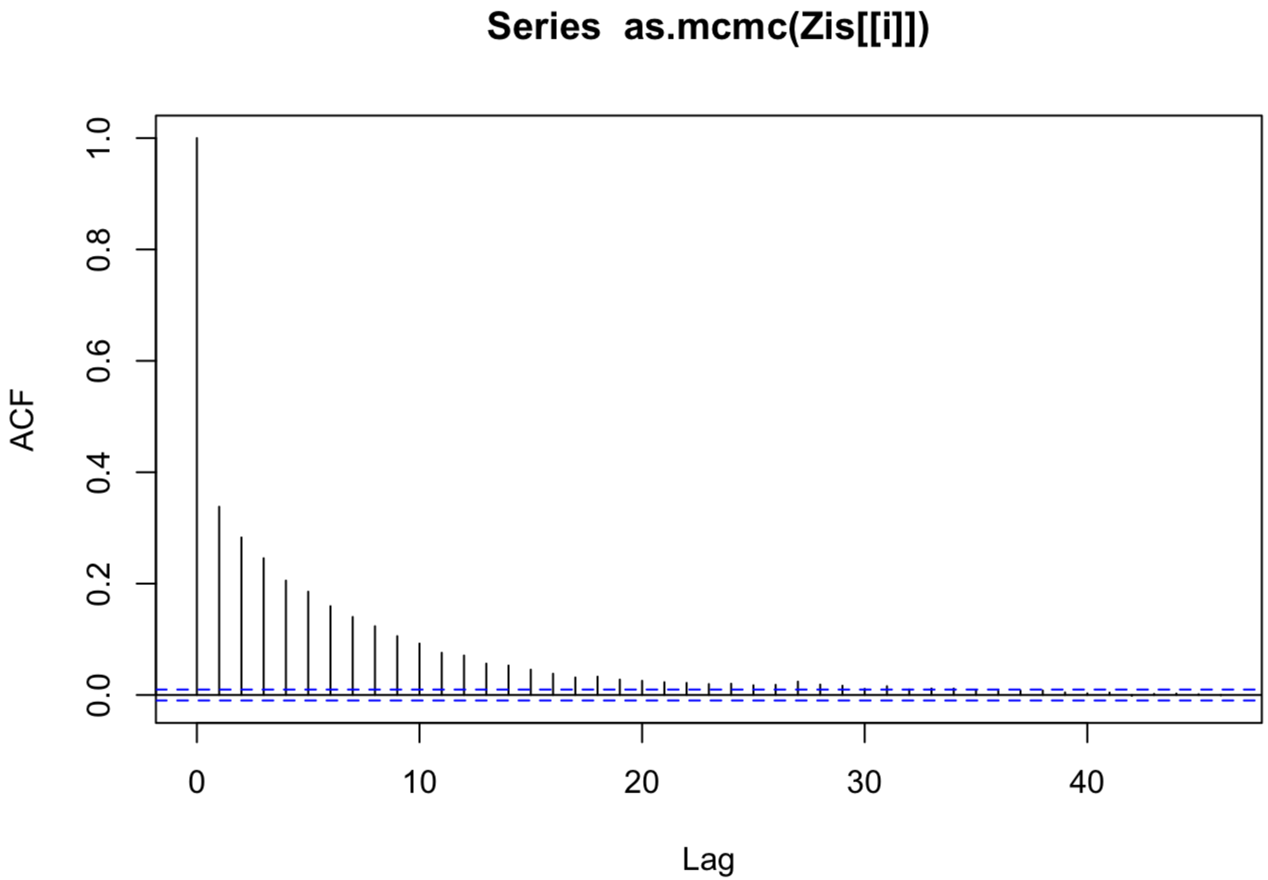

Series as.mcmc(Zis[[i]])
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
10
20
30
40
Lag

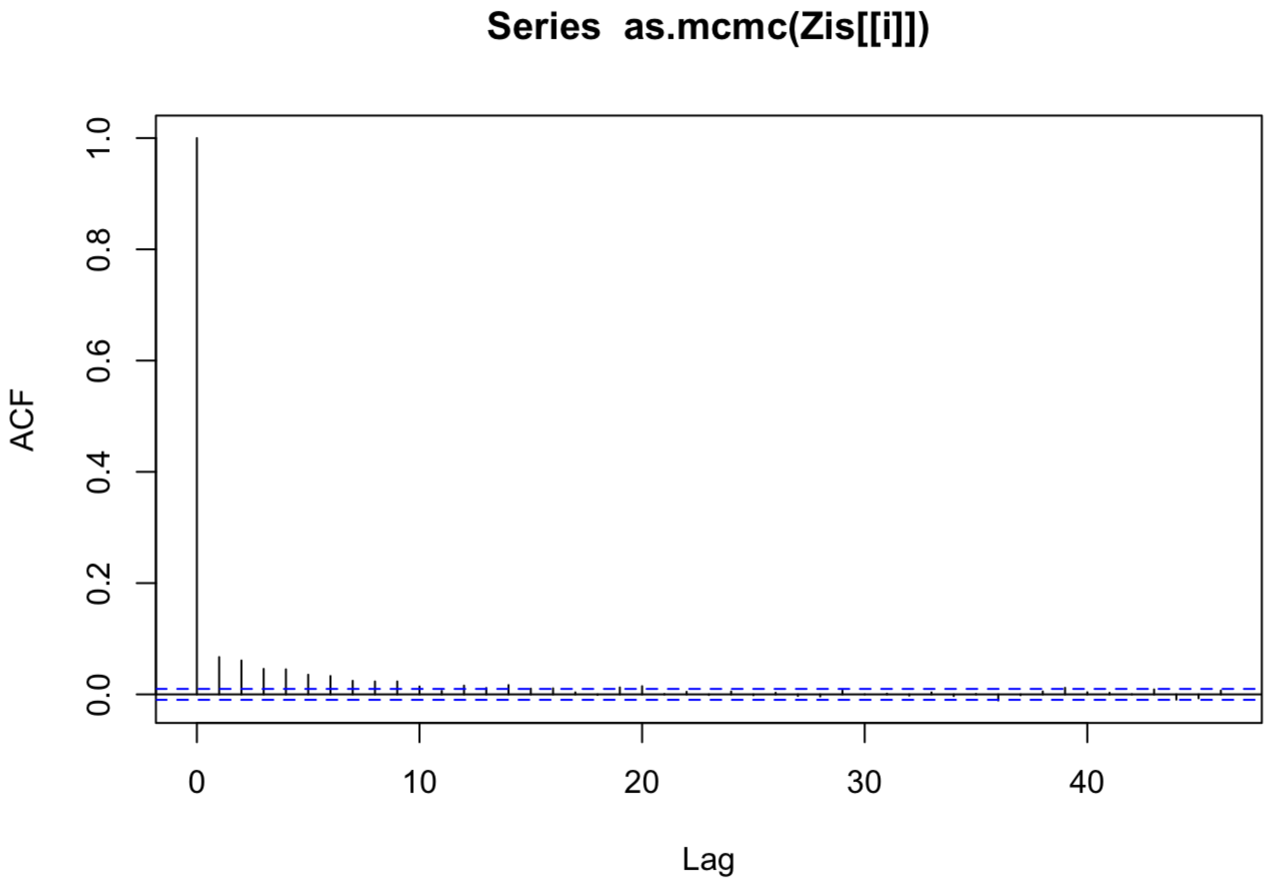

Series as.mcmc(Zis[[i]])
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
10
20
30
40
Lag

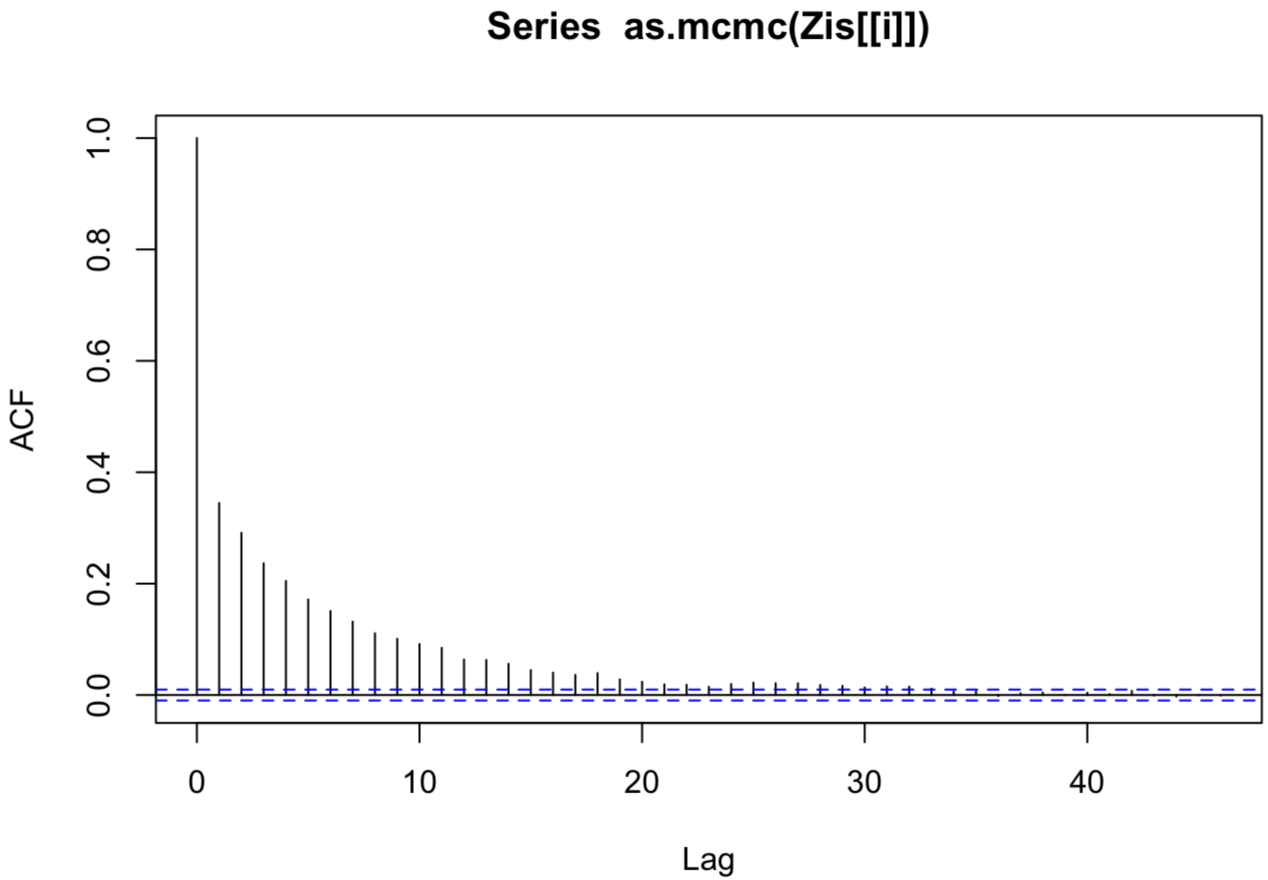

Series as.mcmc(Zis[[i]])
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
10
20
30
40
Lag

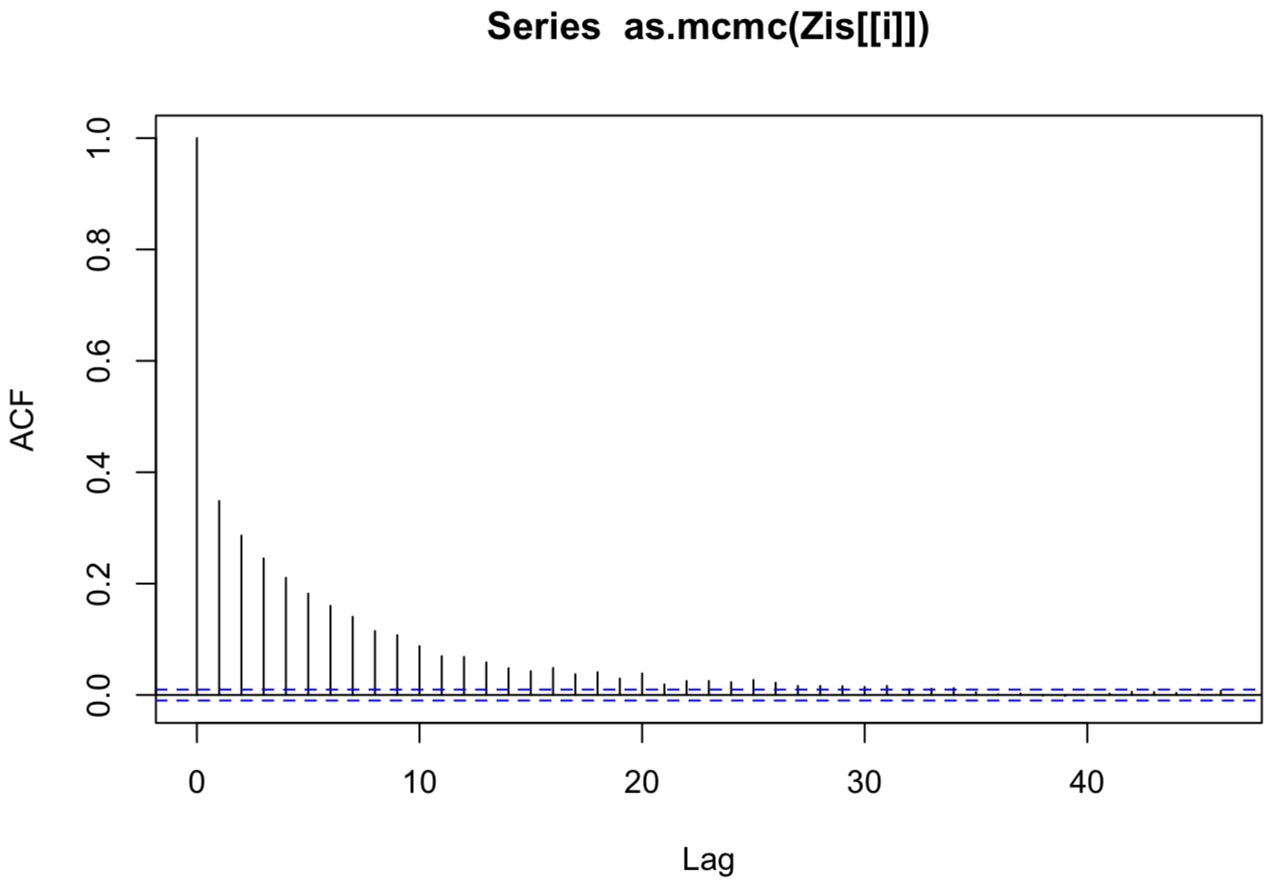

Series as.mcmc(Zis[[i]])
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
10
20
30
40
Lag

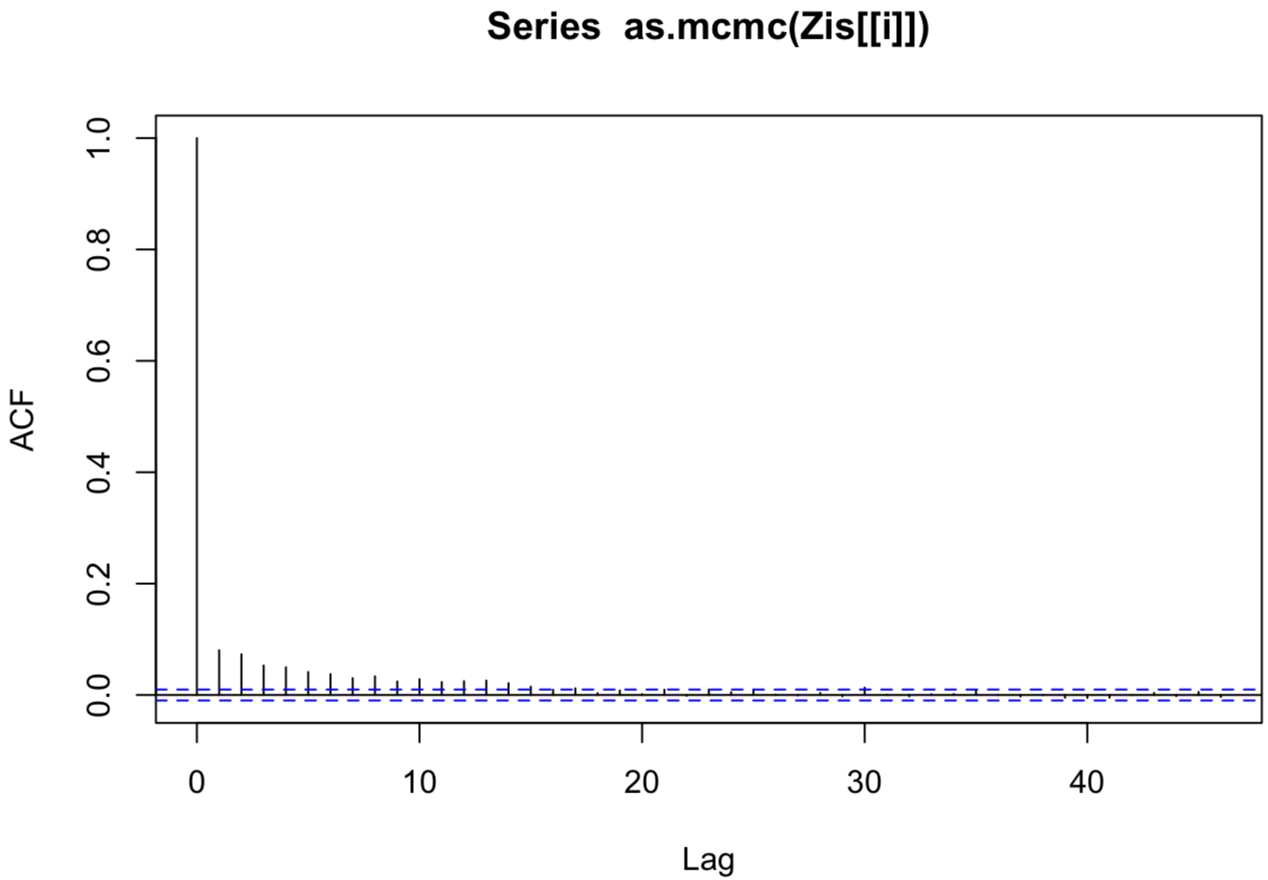

Series as.mcmc(Zis[[i]])
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
10
20
30
40
Lag

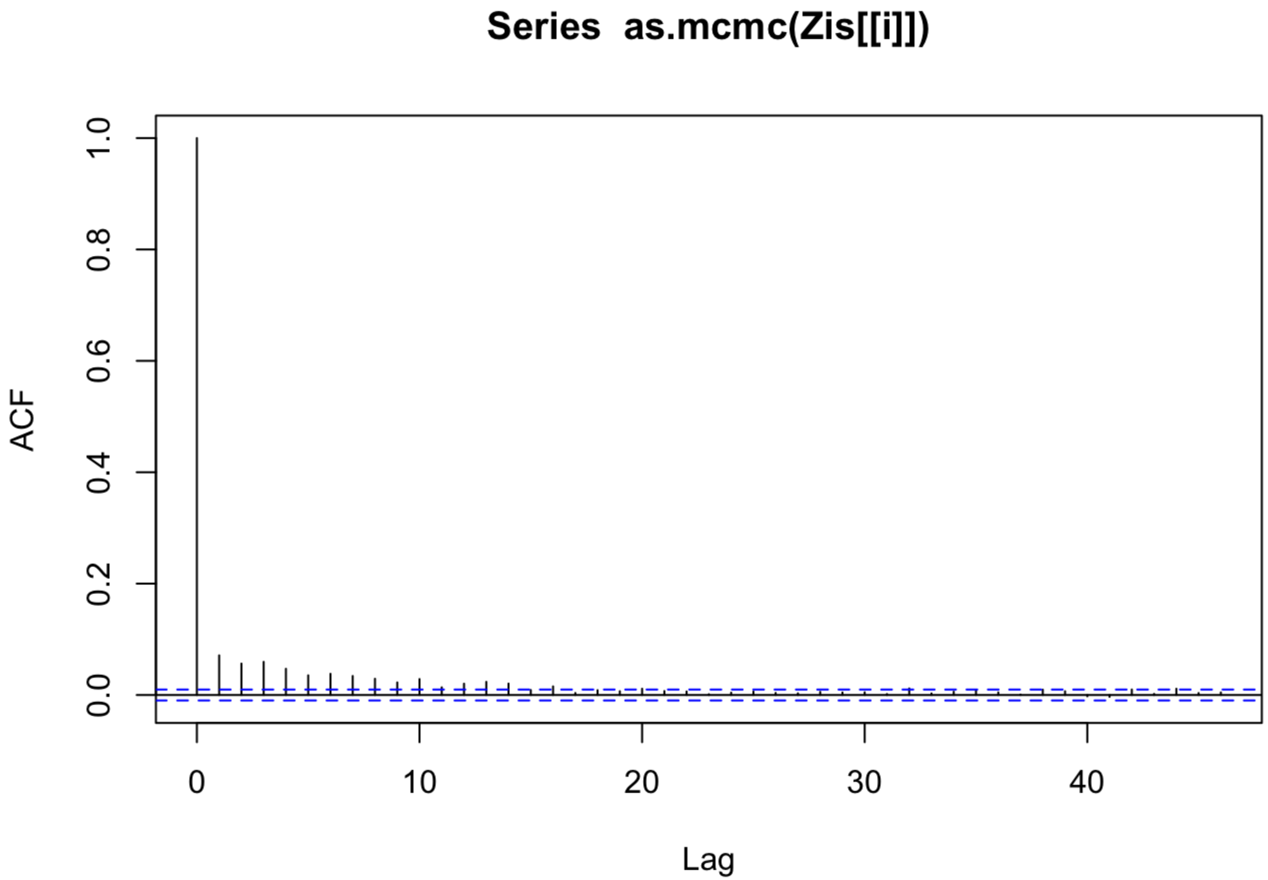

Series as.mcmc(Zis[[i]])
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
10
20
30
40
Lag

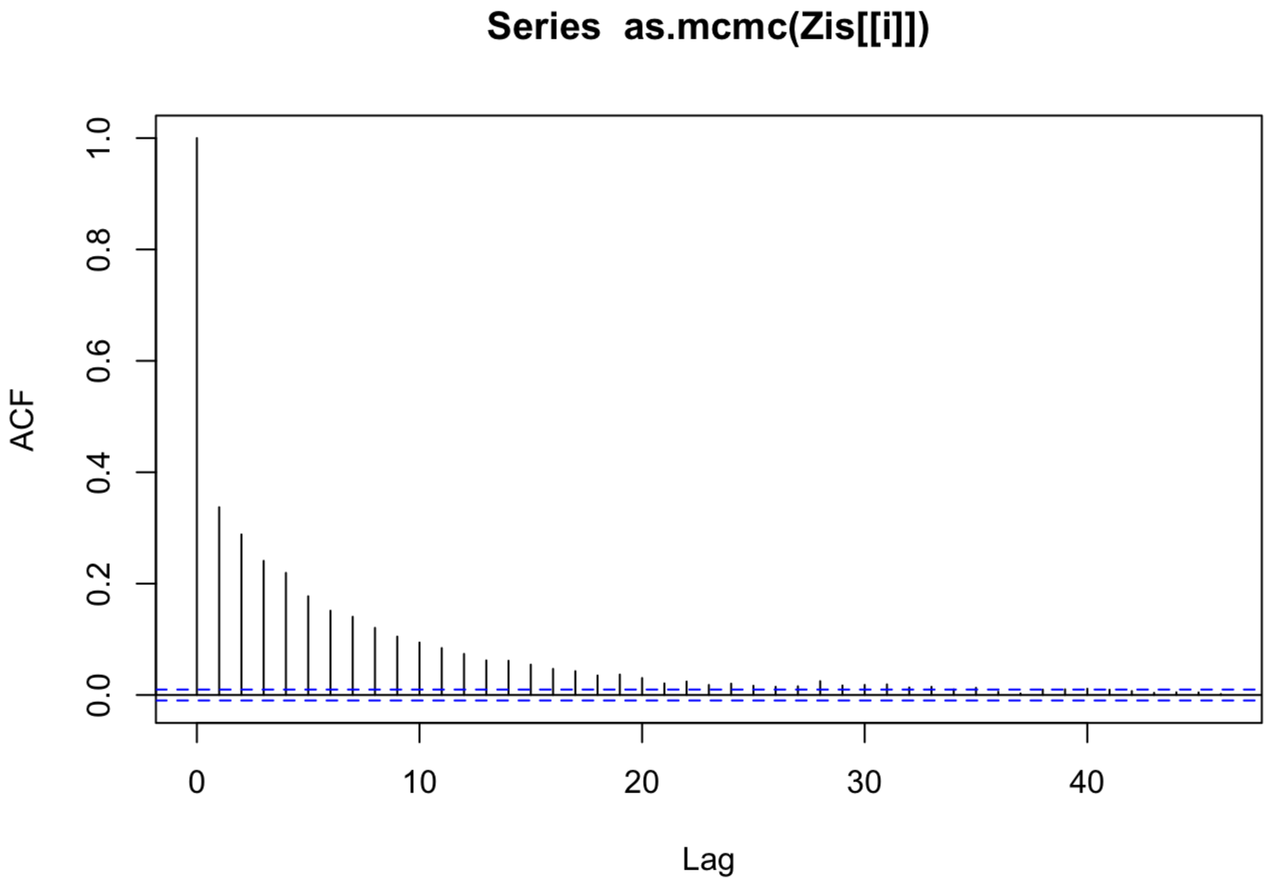

Series as.mcmc(Zis[[i]])
ACF
Lag
0.0
0.2
0.4
0.6
0.8
1.0
0
10
20
30
40

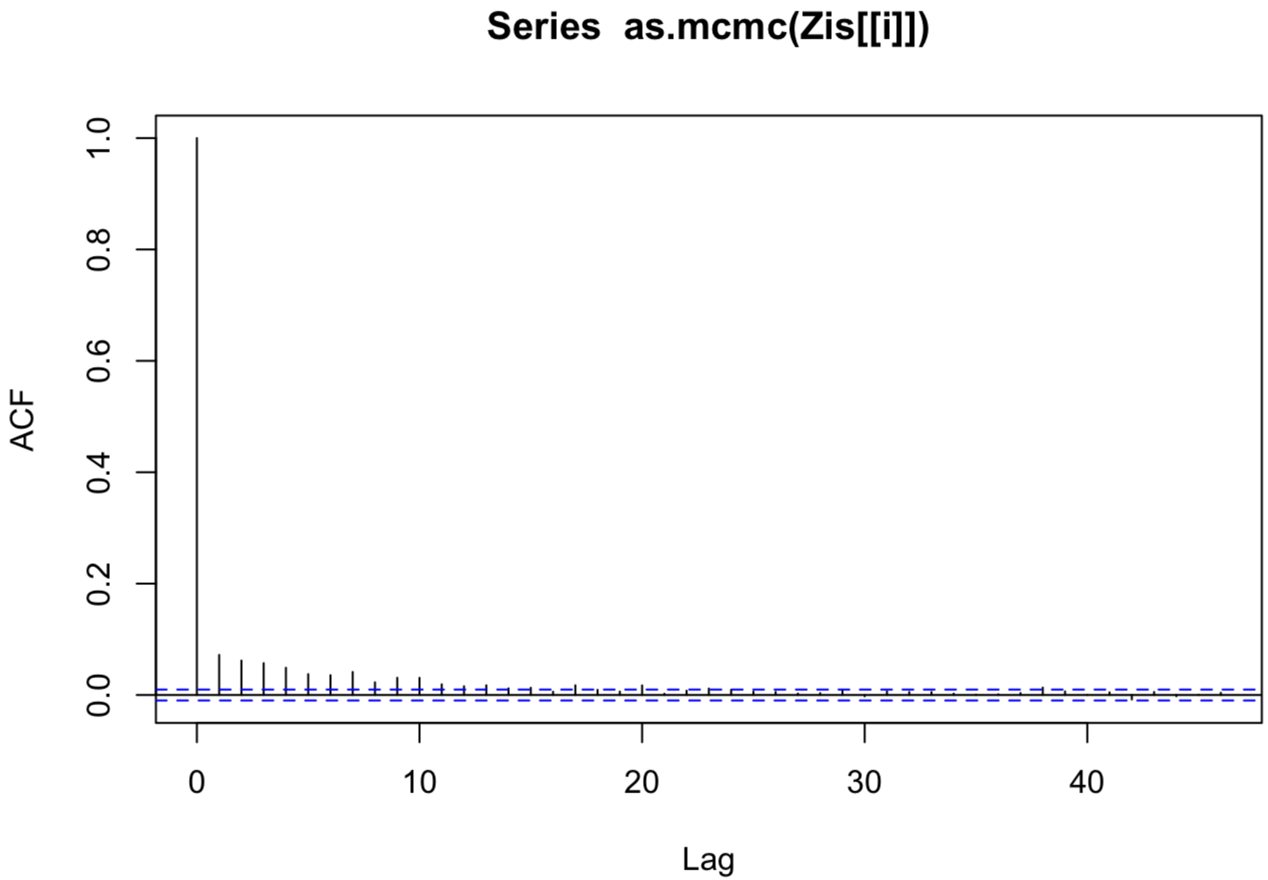

Series as.mcmc(Zis[[i]])
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
10
20
30
40
Lag

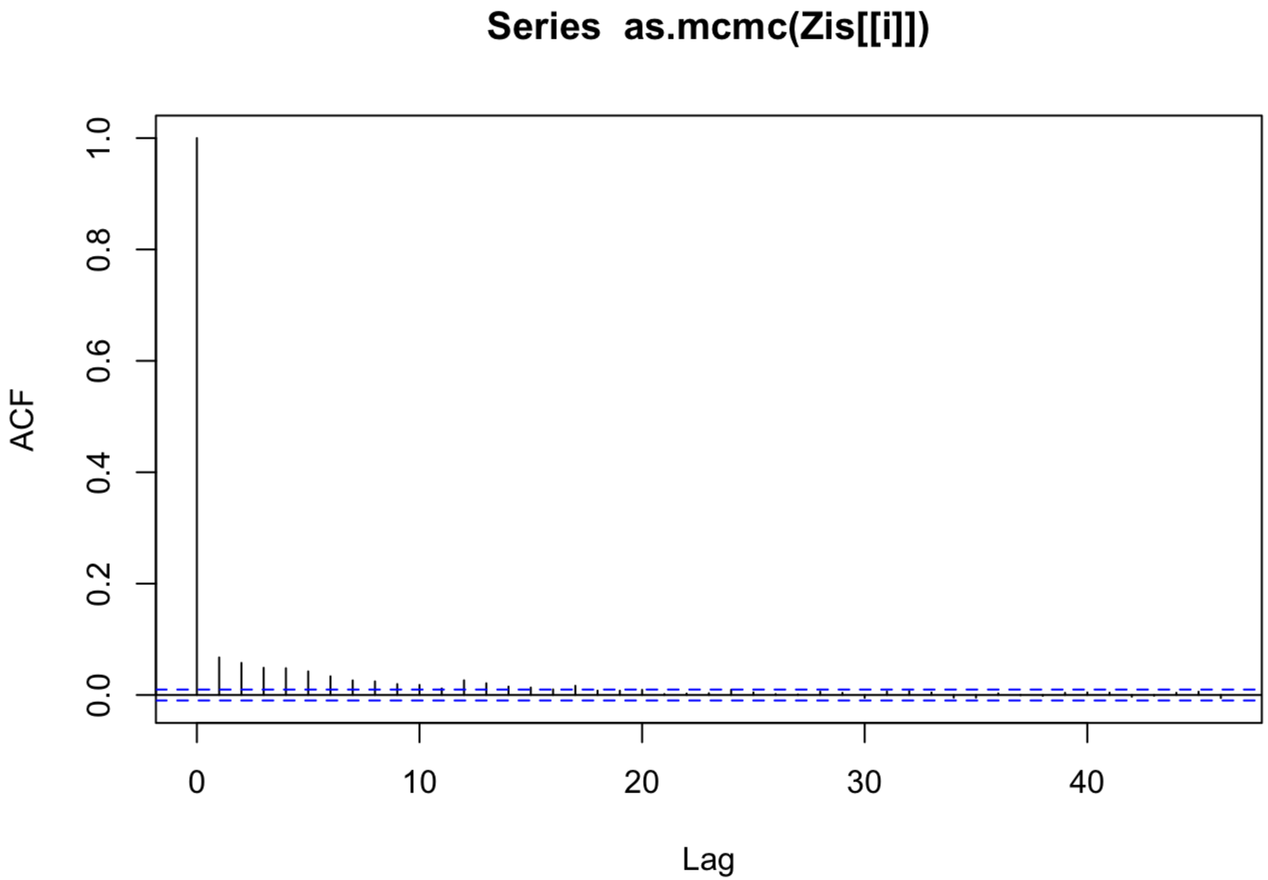

Series as.mcmc(Zis[[i]])
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
10
20
30
40
Lag

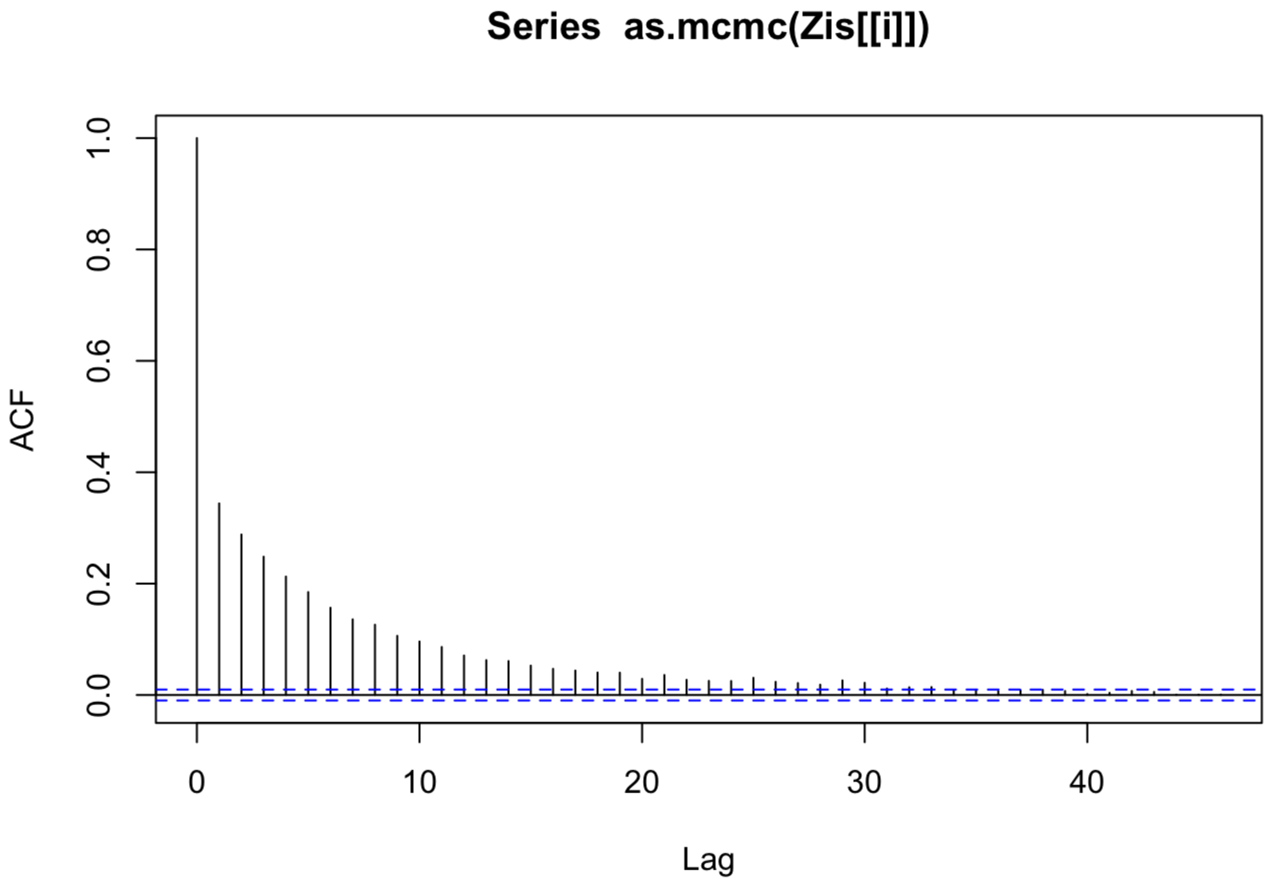

Series as.mcmc(Zis[[i]])
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
10
20
30
40
Lag

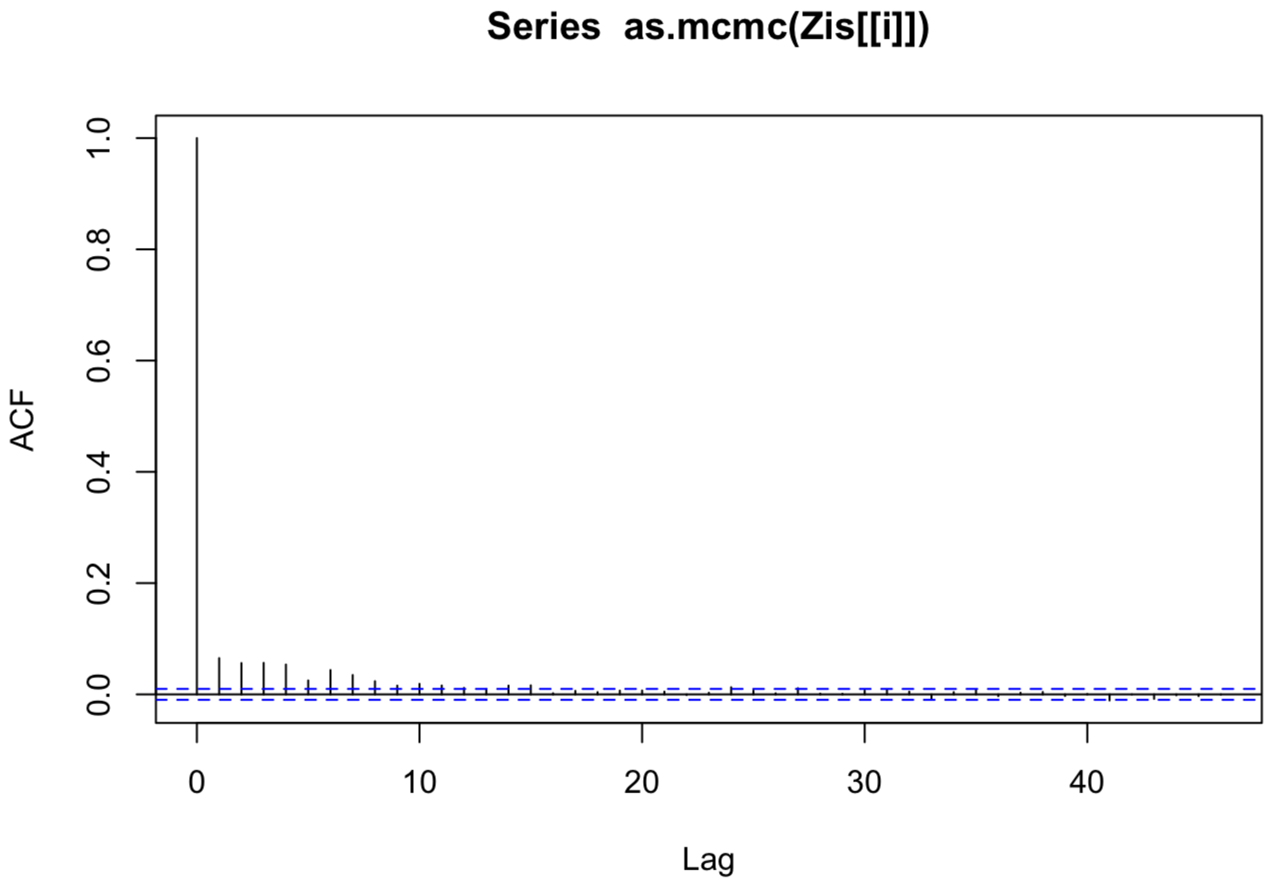

Series as.mcmc(Zis[[i]])
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
10
20
30
40
Lag


Series as.mcmc(Zis[[i]])
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
10
20
30
40
Lag


Series as.mcmc(Zis[[i]])
ACF
Lag
0.0
0.2
0.4
0.6
0.8
1.0
0
10
20
30
40


Series as.mcmc(Zis[[i]])
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
10
20
30
40
Lag


Series as.mcmc(Zis[[i]])
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
10
20
30
40
Lag



According to the autocorrelation plots, all values decreases and converges to 0 as we perform sufficient number of simulations , indicating that the mixing of the Markov Chain has converged to a steady distribution.

Part d

```
# 95% CI for beta
CI <- quantile(BETA, c(0.025, 0.975))
print(CI)
```

```
##       2.5%      97.5%
## -2.1479751  0.7194671
```

```
# Pr(beta > 0 | y, x)
prob <- mean(BETA > 0)
print(prob)
```

```
## [1] 0.1723
```

#6.2

$\theta_1 = \theta_A,\ \theta_2 = \theta_B,\ \sigma_1 = \sigma_A,\ \sigma_2 = \sigma_B$

b) $p(\underset{\sim}{y}, \theta_A, \sigma_B^2, \theta_A, \sigma_B^2, \underset{\sim}{x}, p)$

$= p(\underset{\sim}{y} \mid \theta_A, \sigma_A^2, \theta_B, \sigma_B^2, \underset{\sim}{x}, p)\, p(\underset{\sim}{x} \mid p, \theta_A, \theta_B, \sigma_A^2, \sigma_B^2)\, p(\theta_A) p(\theta_B) p(\sigma_A^2) p(\sigma_B^2) p(p)$

$= \left\{\prod_{i:x_i=1} p(y_i \mid \theta_A, \sigma_A^2)\right\} \left\{\prod_{i:x_i=2} p(y_i \mid \theta_B, \sigma_B^2)\right\} \left\{p(\underset{\sim}{x} \mid p)\right\} p(\theta_A) p(\theta_B) p(\sigma_A^2) p(\sigma_B^2) p(p)$

$= \left\{\prod_{i:x_i=1} p(y_i \mid \theta_A, \sigma_A^2)\right\} p(\theta_A) p(\sigma_A^2) \cdot$ ①

$\left\{\prod_{i:x_i=2} p(y_i \mid \theta_B, \sigma_B^2)\right\} p(\theta_B) p(\sigma_B^2) \cdot$ ②

$\left\{p(\underset{\sim}{x} \mid p)\, p(p)\right\}$ ③

$x_i = \begin{cases} 1 \\ 2 \end{cases} \Rightarrow x_i = \begin{cases} 1 \\ 0 \end{cases}$ binary

$\boxed{\text{FC of } (x_1, \dots, x_n)}$:

$p(\underset{\sim}{x} \mid \underset{\sim}{y}, \theta_A, \sigma_A^2, \theta_B, \sigma_B^2, p) \propto p(\underset{\sim}{y}, \theta_A, \sigma_A^2, \theta_B, \sigma_B^2, \underset{\sim}{x}, p)$

$\propto p(\underset{\sim}{y} \mid \theta_A, \sigma_A^2, \theta_B, \sigma_B^2, \underset{\sim}{x}, p)\, p(\underset{\sim}{x} \mid p, \theta_A, \theta_B, \sigma_A^2, \sigma_B^2)$

$\propto p(\underset{\sim}{y} \mid \theta_A, \sigma_A^2, \theta_B, \sigma_B^2, \underset{\sim}{x})\, p(\underset{\sim}{x} \mid p)$

$= \prod_{i=1}^{n} p(y_i \mid \theta_A, \sigma_A^2, \theta_B, \sigma_B^2, x_i)\, p(x_i \mid p)$

$= \prod_{i=1}^{n} [p \cdot \text{dnorm}(y_i, \theta_A, \sigma_A)]^{x_i} \cdot [(1-p)\, \text{dnorm}(y_i, \theta_B, \sigma_B)]^{1-x_i}$

$Pr(x_i = 0 \mid p) = 1-p$

$Pr(x_i = 1 \mid p) = p$

$Pr(x_i = x_i \mid p) = p^{x_i} (1-p)^{1-x_i}$

so marginally, $x_i \sim \text{Bernoulli}\left(\dfrac{p \cdot \text{dnorm}(y_i, \theta_A, \sigma_A)}{p \cdot \text{dnorm}(y_i, \theta_A, \sigma_A) + (1-p)\, \text{dnorm}(y_i, \theta_B, \sigma_B)}\right)$

Let $n_1 = \Sigma x_i$ (# of 1, in X), $n_2 = n - n_1$ (# of 0, in X)

$\boxed{\text{FC of } p}$:

$$p(p \mid \underset{\sim}{X}, \underset{\sim}{Y}, \theta_A, \theta_B, \sigma_A^2, \sigma_B^2) \propto p(\underset{\sim}{X}, \underset{\sim}{Y}, \theta_A, \theta_B, \sigma_A^2, \sigma_B^2, p)$$

$$\propto p(p)\, p(\underset{\sim}{X} \mid p) \quad \text{by } ③$$

$$\propto \text{dbeta}(p, a, b)\ \text{dbinom}(n_1, n, p)$$

$$\propto p^{a-1}(1-p)^{b-1} p^{n_1}(1-p)^{n_2}$$

$$= p^{a+n_1-1}(1-p)^{b+n_2-1}$$

$$= \text{dbeta}(p, a+n_1, b+n_2)$$

$\boxed{\text{FC of } \theta_A}$:

$$p(\theta_A \mid \underset{\sim}{X}, \underset{\sim}{Y}, p, \theta_B, \sigma_A^2, \sigma_B^2) \propto p(\underset{\sim}{X}, \underset{\sim}{Y}, \theta_A, \theta_B, \sigma_A^2, \sigma_B^2, p)$$

$$\propto \left\{\prod_{i: x_i = 1}^{n} p(y_i \mid \theta_A, \sigma_A^2)\right\} p(\theta_A)\, p(\sigma_A^2) \quad \text{by } ①$$

$$\propto \left\{\prod_{i: x_i=1}^{n} \text{dnorm}(y_i, \theta_A, \sigma_A)\right\} \cdot \text{dnorm}(\theta_A, \mu_0, \tau_0)$$

$$\propto \text{dnorm}(\theta_A, \mu_{n_1}, \tau_{n_1}) \quad \text{by textbook p89}$$

where $\mu_{n_1} = \dfrac{\frac{1}{\tau_0^2}\mu_0 + \frac{n_1}{\sigma_A^2}\bar{y}}{\frac{1}{\tau_0^2} + \frac{n_1}{\sigma_A^2}}$ and $\tau_{n_1} = \left(\frac{1}{\tau_0^2} + \frac{n_1}{\sigma_A^2}\right)^{-1}$

$\boxed{\text{FC of } \theta_B}$

similar to θ_A, $\theta_B \sim \text{dnorm}(\theta_B, \mu_{n_2}, \tau_{n_2})$,

where $\mu_{n_2} = \dfrac{\frac{1}{\tau_0^2}\mu_0 + \frac{n_2}{\sigma_B^2}\bar{y}}{\frac{1}{\tau_0^2} + \frac{n_2}{\sigma_B^2}}$ and $\tau_{n_2} = \left(\frac{1}{\tau_0^2} + \frac{n_2}{\sigma_B^2}\right)^{-1}$

page 3

$\boxed{\text{Fc of } \sigma_A^2}$

$$p(\sigma_A^2 \mid \underset{\sim}{x}, \underset{\sim}{y}, \theta_A, \theta_B, \sigma_B^2, p) \propto p(\underset{\sim}{x}, \underset{\sim}{y}, \theta_A, \theta_B, \sigma_A^2, \sigma_B^2, p)$$

$$\propto \left\{ \prod_{i: x_i = 1}^{n} p(y_i \mid \theta_A, \sigma_A^2) \right\} p(\theta_A) \, p(\sigma_A^2) \text{ by } ①$$

$$\propto \left\{ \prod_{i: x_i = 1}^{n} \text{dnorm}(y_i, \theta_A, \sigma_A) \right\} \cdot \text{inverse-gam}(\sigma_A^2, \nu_0, \sigma_0^2/2)$$

$$\propto \text{inverse-gam}\left(\nu_{n_1}/2, \ \sigma_{n_1}^2(\theta_A)\nu_{n_1}/2\right) \text{ by textbook p93}$$

where $\nu_{n_1} = \nu_0 + n_1$,

and $\sigma_{n_1}^2(\theta_A) = \frac{1}{\nu_{n_1}}\left[\nu_0 \sigma_0^2 + n_1 s_{n_1}^2(\theta_A)\right]$

$\boxed{\text{Fc of } \sigma_B^2}$

similar to σ_A^2, $\sigma_B^2 \sim \text{inverse-gam}(\nu_{n_2}/2, \sigma_{n_2}^2(\theta_B)\nu_{n_2}/2)$,

where $\nu_{n_2} = \nu_0 + n_2$,

and $\sigma_{n_2}^2(\theta_B) = \frac{1}{\nu_{n_2}}\left[\nu_0 \sigma_0^2 + n_2 s_{n_2}^2(\theta_B)\right]$

6.3

a) $z_i|\beta \sim N(\beta x_i, 1)$, $\epsilon_1, \ldots, \epsilon_n \sim iid\ N(0,1)$, $\beta \sim N(0, \tau_\beta^2)$

$$P(\beta|y, x, z, c) \propto p(\beta, y, x, z, c)$$

$$\propto P(z|\beta, y, x, c)\, P(\beta)$$

$$\propto \left\{\prod_{i=1}^{n} dnorm(z_i, \beta x_i, 1)\right\} \cdot dnorm(\beta, 0, \tau_\beta)$$

$$\propto \left\{\prod_{i=1}^{n} e^{-\frac{1}{2}(z_i - \beta x_i)^2}\right\} e^{-\frac{1}{2}\left(\frac{\beta}{\tau_\beta}\right)^2}$$

$$\propto \exp\left\{-\frac{1}{2}\left[\cancel{\Sigma z_i^2} + \beta^2 \Sigma x_i^2 - 2\beta \Sigma(z_i x_i) + \frac{\beta^2}{\tau_\beta^2}\right]\right\}$$

$$\propto \exp\left\{-\frac{1}{2}\left[\beta^2\left(\Sigma x_i^2 - \frac{1}{\tau_\beta^2}\right) - 2\beta\, \Sigma(z_i x_i)\right]\right\}$$

$$\propto \exp\left\{-\frac{1}{2}\left(\Sigma x_i^2 - \frac{1}{\tau_\beta^2}\right)\left(\beta - \frac{\Sigma(z_i x_i)}{\Sigma x_i^2 + \frac{1}{\tau_\beta^2}}\right)^2\right\}$$

$$\sim N(\mu_{\beta_n}, \tau_{\beta_n}^2)$$

where $\mu_{\beta_n} = \dfrac{\Sigma(z_i x_i)}{\Sigma x_i^2 + \frac{1}{\tau_\beta^2}}$,

and $\tau_{\beta_n}^2 = \dfrac{1}{\Sigma x_i^2 - \frac{1}{\tau_\beta^2}}$

#6.3

b) $c \sim N(0, \tau_c^2)$

$\boxed{c}$ Given everything else, the distribution of c will only depend on y and z.

Given $Y=y$ and $Z=z$, c must $>$ all z_i's for which $y_i=0$ and $\leq$ all z_i's for which $y_i=1$.

Let $a = \max\{z_i : y_i = 0\}$ and $b = \min\{z_i : y_i = 1\}$.

The FC of c is the $\propto p(c)$ but constrained to (a,b):

$$p(c \mid y, x, z, \beta) = p(c \mid y, z)$$
$$\propto p(c)\, p(y \mid z, c)$$
$$= \text{dnorm}(c, 0, \tau_c)\, \delta_{(a,b)}(c)$$

Therefore, $p(c \mid y, x, z, \beta)$ is a constrained normal density with a support on (a,b).

$\boxed{z_i}$ $z_i \mid \beta \sim N(\beta x_i, 1)$

Given c, observing $Y_i = y_i$ will tell us that z_i must lie in the interval y_i belongs to $\Rightarrow (-\infty, c)$ if $y_i = 0$ and (c, ∞) if $y_i=1$

Thus, the FC of z_i is:

$$p(z_i \mid y, x, z_{-i}, \beta, c) \propto p(z_i \mid \beta, x_i)\, p(y_i \mid z_i, c)$$
$$= \begin{cases} \text{dnorm}(z_i, \beta x_i, 1)\, \delta_{(c,\infty)}(z_i) & \text{if } y_i = 1 \\ \text{dnorm}(z_i, \beta x_i, 1)\, \delta_{(-\infty, c)}(z_i) & \text{if } y_i = 0 \end{cases}$$

Therefore, $p(z_i \mid y, x, z_{-i}, \beta, c)$ is proportional to a normal density but constrained to either above c or below c, depending on y_i.