

# Stat 651 Project

Talmage Hilton

2025-04-08

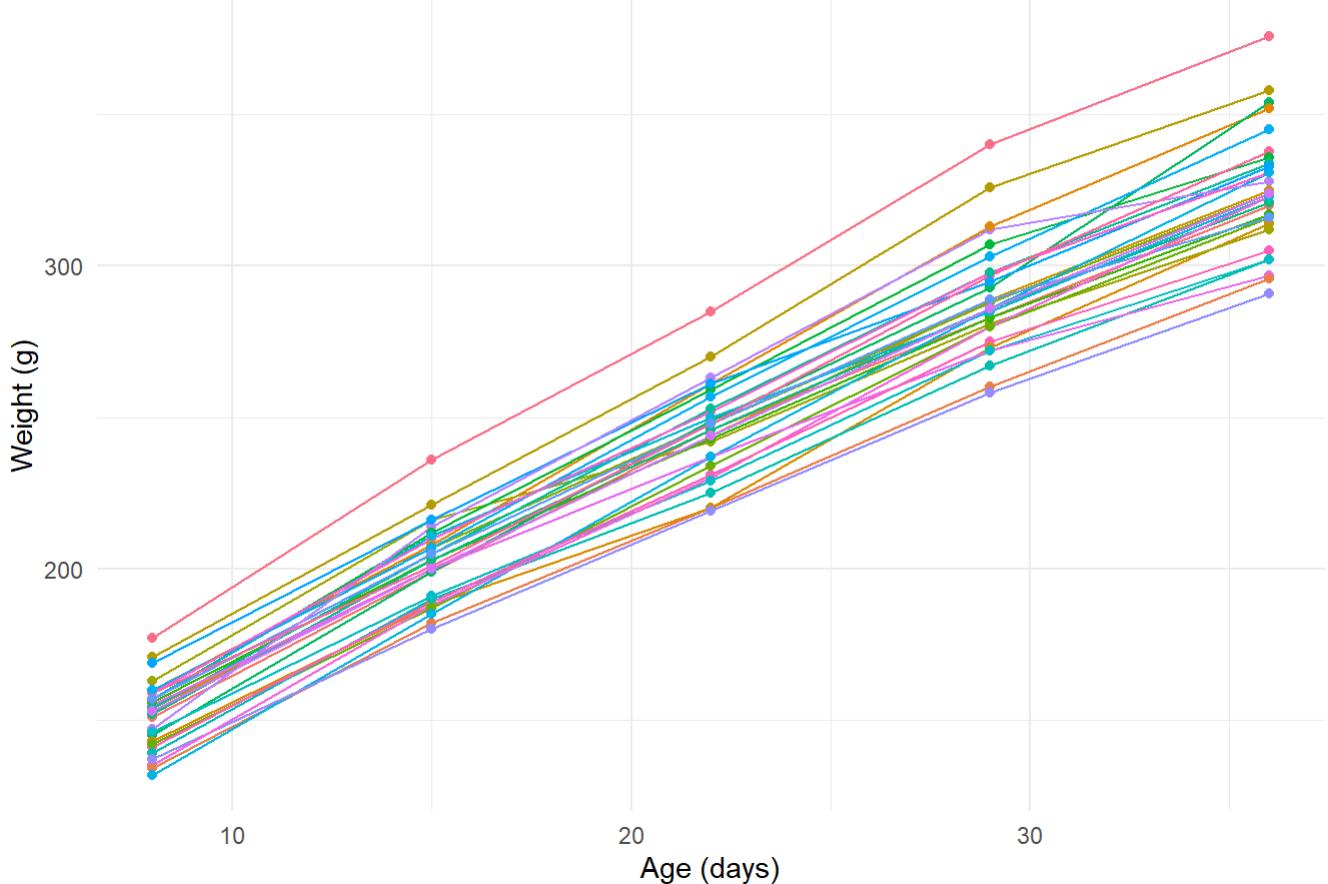
```
# Read in data
ratdata <- read_excel("ratdata.xlsx")

# Pivot Longer to get tidy data
rat_long <- ratdata %>%
  pivot_longer(cols = starts_with("rat"), names_to = "rat", values_to = "weight")
```

## EDA

```
# Weights of each rat
ggplot(rat_long, aes(x = age, y = weight, group = rat, color = rat)) +
  geom_line() +
  geom_point() +
  labs(title = "Growth Curves for Each Rat", x = "Age (days)", y = "Weight (g)") +
  theme_minimal() +
  theme(legend.position = "none")
```

Growth Curves for Each Rat

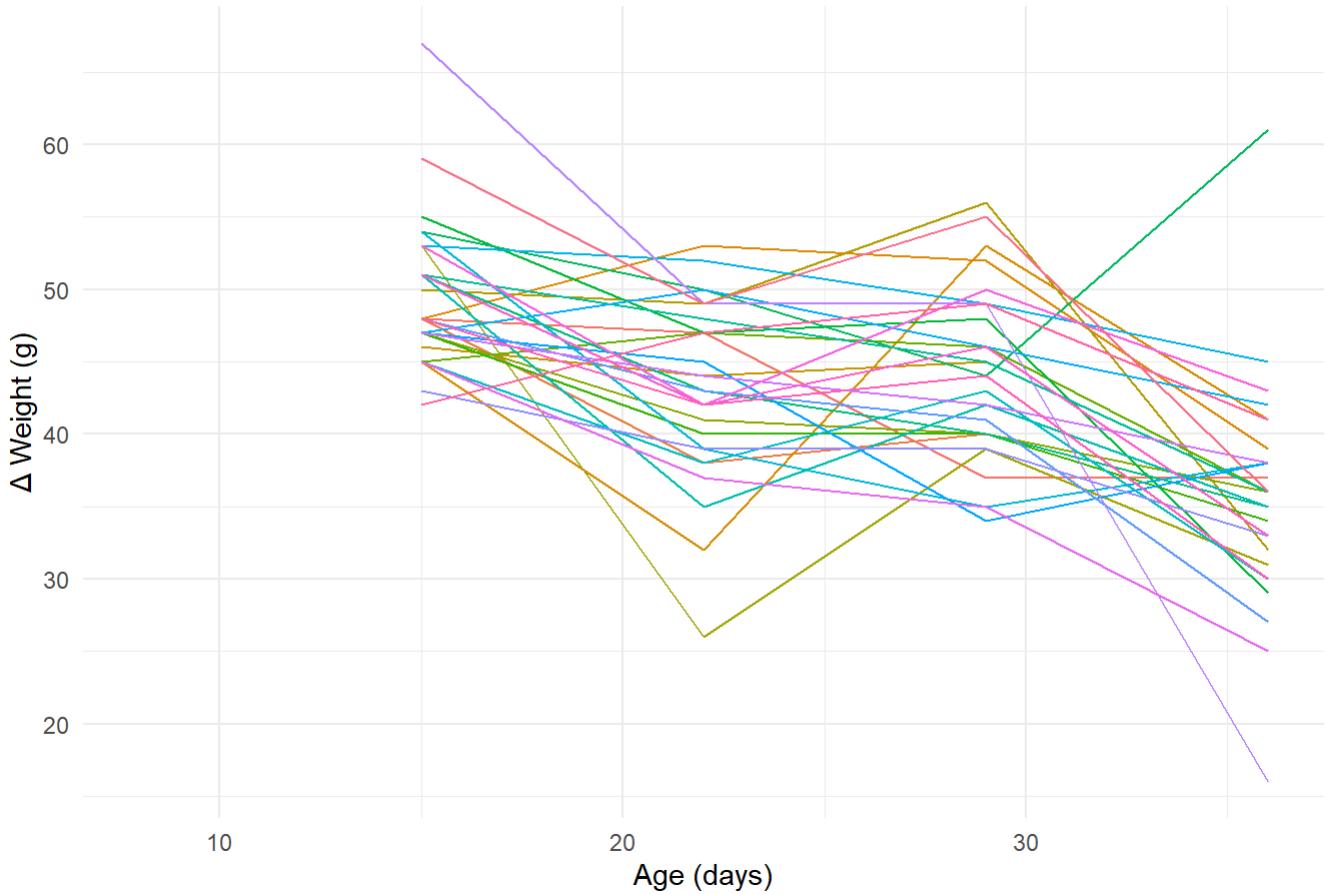


```
# Growth rate for each rat
rat_diff <- ratdata %>%
  dplyr::select(-age) %>%
  mutate(across(everything(), ~ c(NA, diff(.)))) %>%
  mutate(age = ratdata$age)

rat_diff_long <- rat_diff %>%
  pivot_longer(cols = -age, names_to = "rat", values_to = "growth_rate")

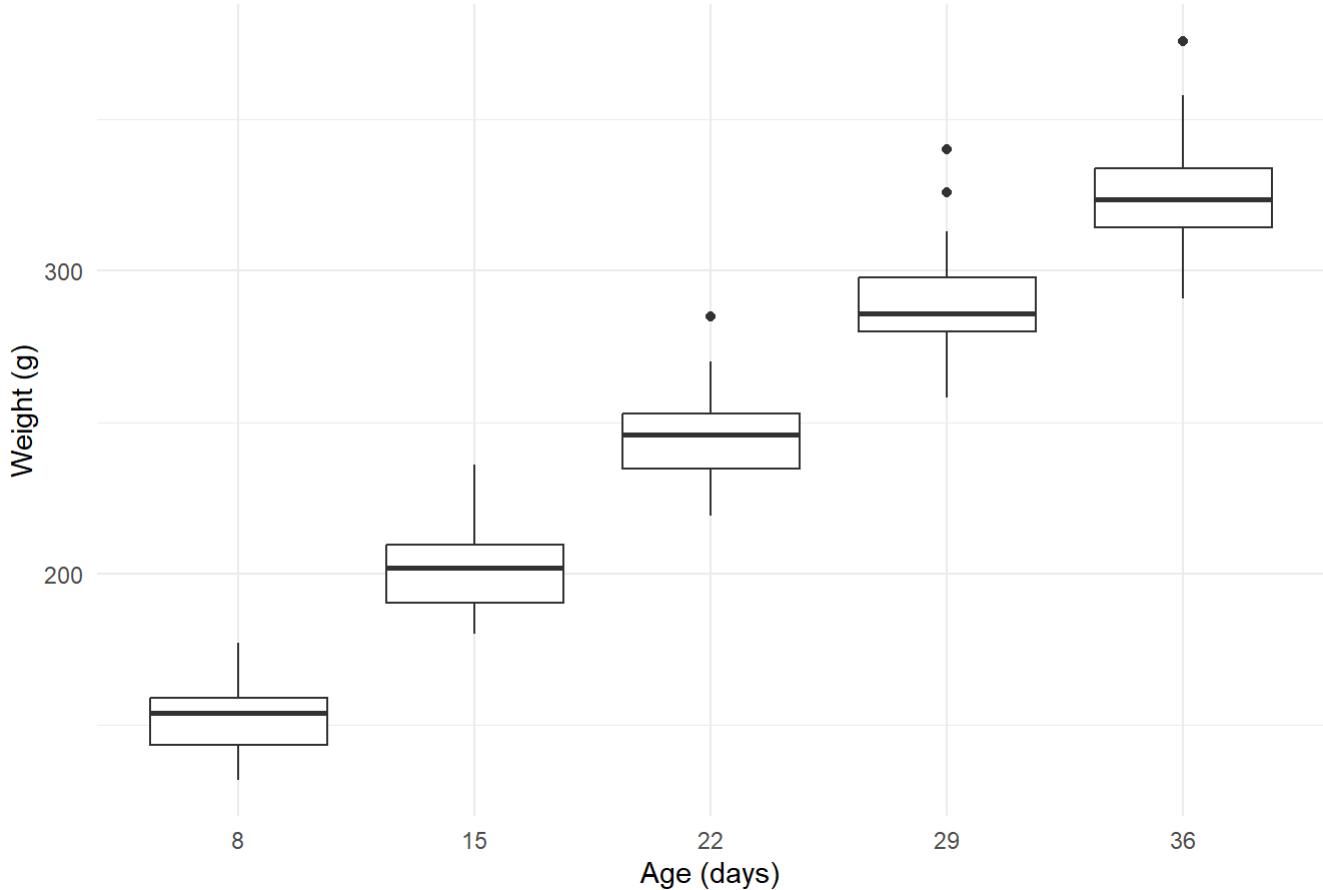
ggplot(rat_diff_long, aes(x = age, y = growth_rate, group = rat, color = rat)) +
  geom_line() +
  labs(title = "Growth Rate by Age", x = "Age (days)", y = " $\Delta$  Weight (g)") +
  theme_minimal() +
  theme(legend.position = "none")
```

Growth Rate by Age



```
# Boxplots at each age
ggplot(rat_long, aes(x = factor(age), y = weight)) +
  geom_boxplot() +
  labs(title = "Distribution of Rat Weights at Each Age", x = "Age (days)", y = "Weight (g)") +
  theme_minimal()
```

### Distribution of Rat Weights at Each Age



The growth curves show that the rats grow in a mostly linear fashion. Obviously there is some variability, but overall I'd say that it's pretty linear. The growth rates are not very consistent (lots of ups and downs), but overall they're all pretty similar. For the most part the growth rate slows down between days 15-22, then speed up between days 22-29, and then decrease again from days 29-36. However, there are a few rate that increase in growth rate during the last time period. There are some exceptions to the rule here, but overall I'd say that the Normal model for  $Y_{ij}$  is fairly reasonable.

## Question 2

```

# Nest the data by rat
rat_nested <- rat_long %>%
  group_by(rat) %>%
  nest()

# Fit regression to each nested dataset
rat_models <- rat_nested %>%
  mutate(model = map(data, ~lm(weight ~ age, data = .)))

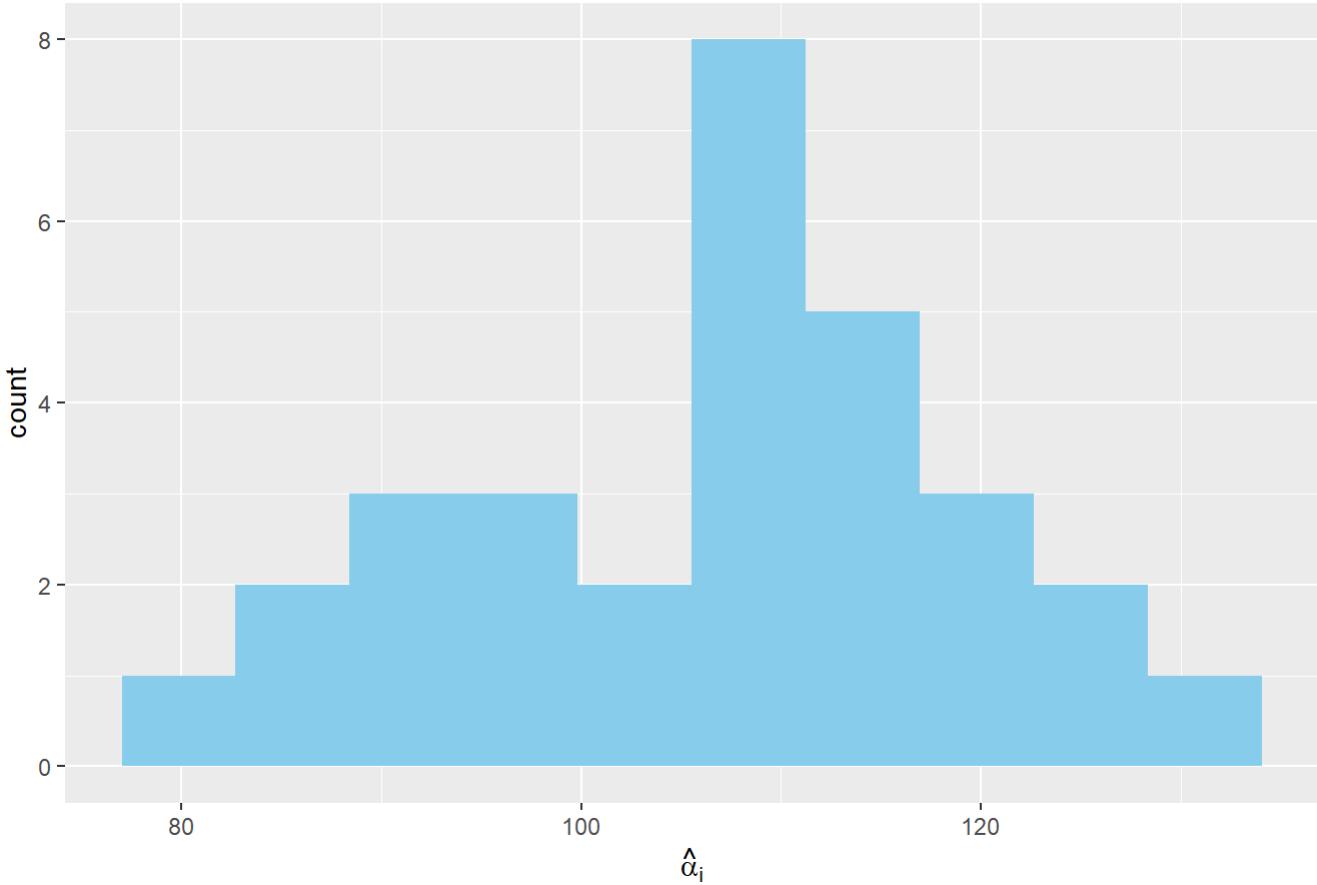
# Extract coefficients from each model
rat_coefs <- rat_models %>%
  mutate(coefs = map(model, broom::tidy)) %>%
  unnest(coefs) %>%
  dplyr::select(rat, term, estimate) %>%
  pivot_wider(names_from = term, values_from = estimate) %>%
  rename(alpha_hat = `(Intercept)`, beta_hat = age)

# Check Normality of thetas

# Histogram of intercepts
ggplot(rat_coefs, aes(x = alpha_hat)) +
  geom_histogram(bins = 10, fill = "skyblue") +
  labs(title = "Histogram of Intercepts ( $\hat{\alpha}_i$ )", x = expression(hat(alpha)[i]))

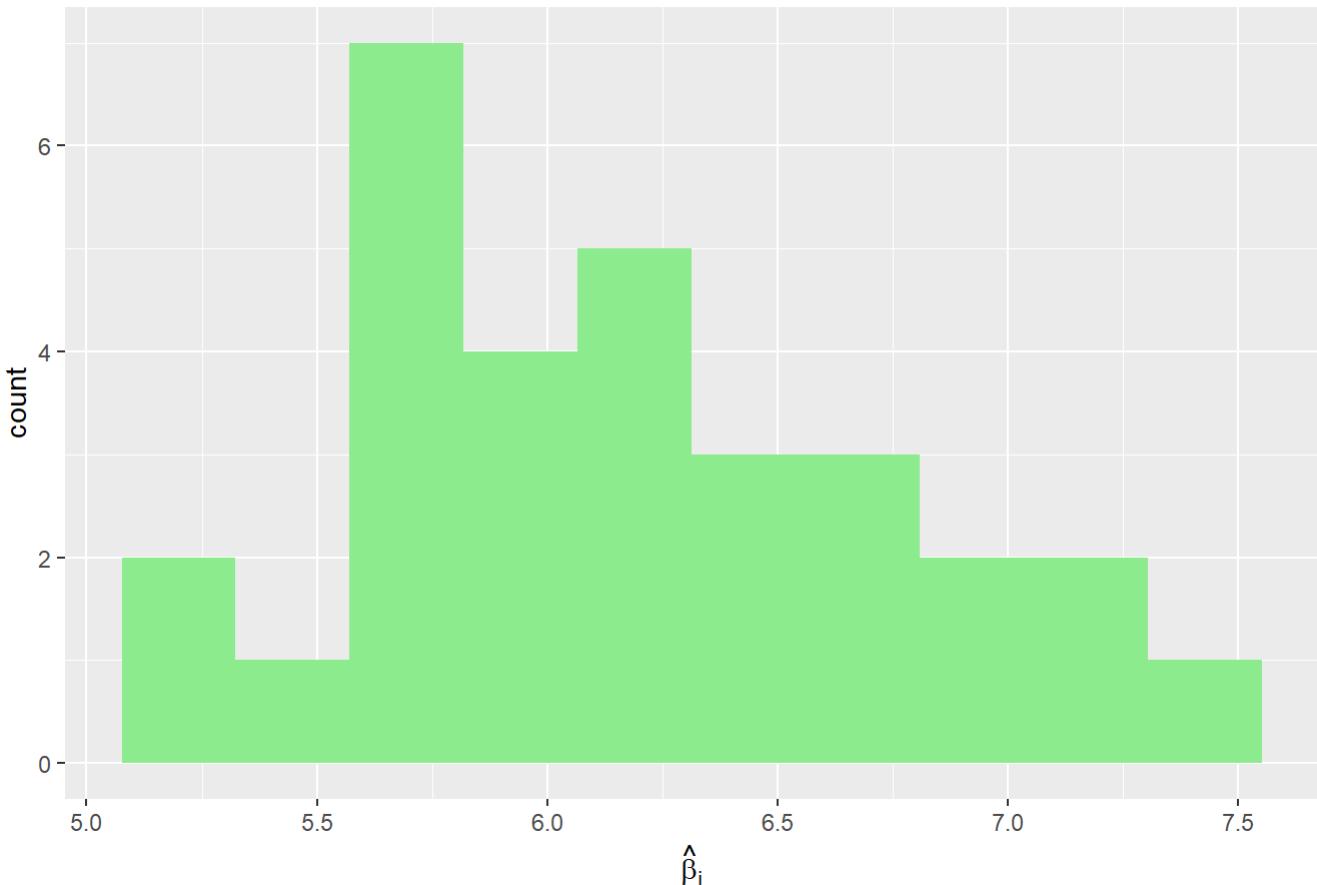
```

Histogram of Intercepts ( $\hat{\alpha}_i$ )

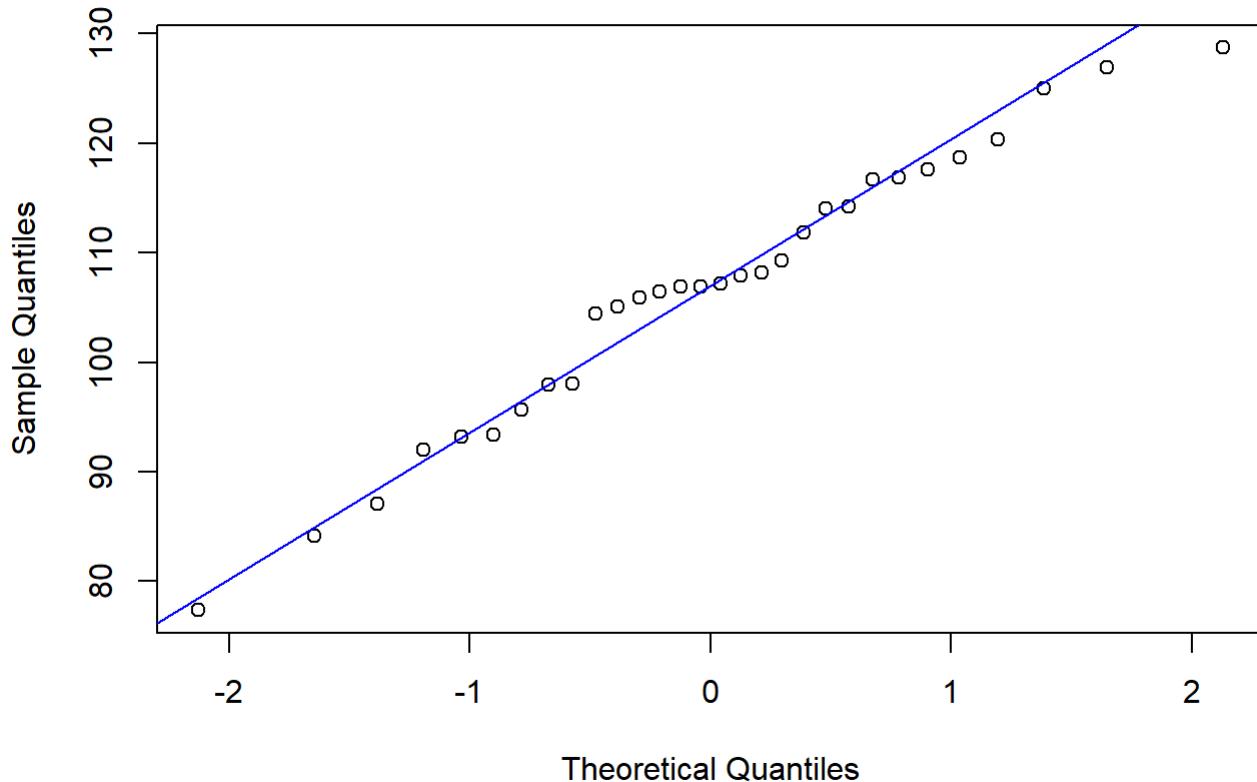


```
# Histogram of slopes
ggplot(rat_coefs, aes(x = beta_hat)) +
  geom_histogram(bins = 10, fill = "lightgreen") +
  labs(title = "Histogram of Slopes (beta_i)", x = expression(hat(beta)[i]))
```

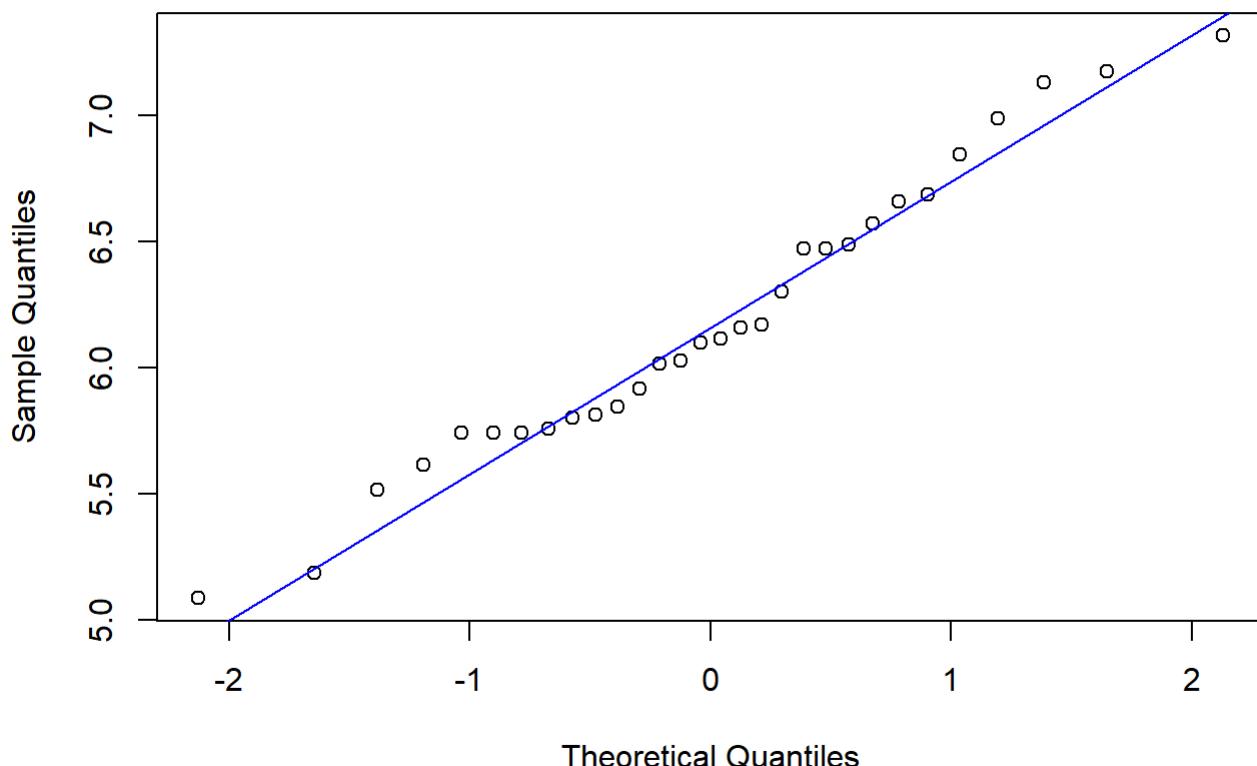
Histogram of Slopes (beta\_i)



```
# QQ plots
qqnorm(rat_coefs$alpha_hat); qqline(rat_coefs$alpha_hat, col = "blue")
```

**Normal Q-Q Plot**

```
qqnorm(rat_coefs$beta_hat); qqline(rat_coefs$beta_hat, col = "blue")
```

**Normal Q-Q Plot**

I would say that the normality assumption is reasonable here. The distribution of intercepts is approximately normal. The distribution of slopes is a little right skewed, but I wouldn't say it's too egregious. The Q-Q plots also aren't too terrible.

# Gibbs Sampler

```

# Priors
lambda0 <- 0.1
nu0 <- 0.1
eta <- c(0, 0)
Sigma <- matrix(c(10, 0, 0, 10), nrow=2, ncol=2)
C <- matrix(c(5, 0, 0, 5), nrow=2, ncol=2)

set.seed(123)

# Prepare data
Y <- as.matrix(ratdata[, 1:30])
age <- ratdata$age
n <- ncol(Y) # number of rats = 30
t <- length(age) # number of time points = 5

# Design matrix for each rat (same across rats)
X_list <- lapply(1:n, function(i) cbind(1, age))
Y_list <- lapply(1:n, function(i) Y[, i])

# Hyperparameters
n_iter <- 10000
burn_in <- 1000

# Storage
theta <- array(NA, dim = c(n, 2, n_iter)) # alpha, beta for each rat
mu <- matrix(NA, nrow = n_iter, ncol = 2)
tau <- numeric(n_iter)

# Initialize
theta_mean <- matrix(0, nrow = n, ncol = 2)
mu[1, ] <- c(0, 0)
tau[1] <- 1

# Gibbs sampler
for (iter in 2:n_iter) {
  # --- 1. Sample theta_i / Y_i, mu, tau ---
  for (i in 1:n) {
    Xi <- X_list[[i]]
    Yi <- Y_list[[i]]
    D_inv <- (1/tau[iter - 1]) * t(Xi) %*% Xi + solve(Sigma)
    D <- solve(D_inv)
    m <- D %*% (tau[iter - 1] * t(Xi) %*% Yi + solve(Sigma) %*% mu[iter - 1, ])
    theta[i, , iter] <- mvtnorm(1, m, D)
  }
  # --- 2. Sample mu / theta, tau ---
  theta_bar <- colMeans(theta[, , iter])
  V_mu_inv <- 30 * solve(Sigma) + solve(C)
  V_mu <- solve(V_mu_inv)
  m_mu <- V_mu %*% (30 * solve(Sigma) %*% theta_bar + solve(C) %*% eta)
  mu[iter, ] <- mvtnorm(1, m_mu, V_mu)
  # --- 3. Sample tau / theta ---
  rss <- 0
}

```

```

for (i in 1:n) {
  Xi <- X_list[[i]]
  Yi <- Y_list[[i]]
  res <- Yi - Xi %*% theta[i, , iter]
  rss <- rss + sum(res^2)
}
shape <- (t * n + nu0) / 2
rate <- (rss + nu0 * lambda0) / 2
tau[iter] <- rgamma(1, shape, rate)
}

# POSTERIOR INFERENCE

# Remove burn-in
theta_post <- theta[, , (burn_in+1):n_iter]
mu_post <- mu[(burn_in+1):n_iter, ]
tau_post <- tau[(burn_in+1):n_iter]

# Means and credible intervals
mu_est <- colMeans(mu_post)
mu_ci <- apply(mu_post, 2, quantile, probs = c(0.025, 0.975))

tau_est <- mean(tau_post)
tau_ci <- quantile(tau_post, probs = c(0.025, 0.975))

cat("Posterior Mean of mu_1:\n", mu_est[1], "\n")

```

```

## Posterior Mean of mu_1:
## -0.01029279

```

```

cat("Posterior Mean of mu_2:\n", mu_est[2], "\n")

```

```

## Posterior Mean of mu_2:
## 0.004852225

```

```

cat("95% CI for mu_1:\n", mu_ci[c(1,2)], "\n")

```

```

## 95% CI for mu_1:
## -1.122027 1.078195

```

```

cat("95% CI for mu_2:\n", mu_ci[c(3,4)], "\n\n")

```

```

## 95% CI for mu_2:
## -1.107626 1.112315

```

```

cat("Posterior Mean of tau:\n", tau_est, "\n")

```

```
## Posterior Mean of tau:  
## 1.596345e-05
```

```
cat("95% CI for tau:\n", tau_ci, "\n")
```

```
## 95% CI for tau:  
## 1.25635e-05 1.976318e-05
```

```
# Number of rats and iterations  
n_rats <- dim(theta_post)[1]  
n_samples <- dim(theta_post)[3]  
  
# Create summary table  
theta_summary <- data.frame(  
  Rat = 1:n_rats,  
  Alpha_Mean = numeric(n_rats),  
  Alpha_Lower = numeric(n_rats),  
  Alpha_Upper = numeric(n_rats),  
  Beta_Mean = numeric(n_rats),  
  Beta_Lower = numeric(n_rats),  
  Beta_Upper = numeric(n_rats)  
)  
  
for (i in 1:n_rats) {  
  alpha_samples <- theta_post[i, 1, ]  
  beta_samples <- theta_post[i, 2, ]  
  
  theta_summary$Alpha_Mean[i] <- mean(alpha_samples)  
  theta_summary$Alpha_Lower[i] <- quantile(alpha_samples, 0.025)  
  theta_summary$Alpha_Upper[i] <- quantile(alpha_samples, 0.975)  
  
  theta_summary$Beta_Mean[i] <- mean(beta_samples)  
  theta_summary$Beta_Lower[i] <- quantile(beta_samples, 0.025)  
  theta_summary$Beta_Upper[i] <- quantile(beta_samples, 0.975)  
}  
  
# View summary  
print(theta_summary)
```

```

##      Rat   Alpha_Mean  Alpha_Lower Alpha_Upper      Beta_Mean    Beta_Lower
## 1     1  4.619471e-05 -0.008491087 0.008621121 -1.777549e-06 -0.0003566357
## 2     2  3.606484e-05 -0.008351269 0.008326511 -9.964905e-07 -0.0003471703
## 3     3 -3.902461e-05 -0.008558640 0.008467757  2.893401e-06 -0.0003512254
## 4     4 -2.979620e-05 -0.008599818 0.008405099  1.443878e-06 -0.0003557182
## 5     5 -3.603031e-05 -0.008575212 0.008536674  1.880454e-06 -0.0003563655
## 6     6 -1.450320e-05 -0.008461858 0.008462936  8.715088e-07 -0.0003531072
## 7     7 -1.382361e-05 -0.008568449 0.008503774 -2.383730e-07 -0.0003563909
## 8     8  3.375434e-07 -0.008720677 0.008585512  1.081288e-06 -0.0003555654
## 9     9 -1.538607e-05 -0.008608573 0.008585751  1.996573e-08 -0.0003538189
## 10   10  1.028568e-04 -0.008504174 0.008790182 -3.808050e-06 -0.0003618211
## 11   11 -5.401498e-05 -0.008615182 0.008475227  2.230997e-06 -0.0003485266
## 12   12  4.404498e-05 -0.008488300 0.008678446  1.571809e-07 -0.0003519078
## 13   13  2.492878e-05 -0.008504112 0.008531132 -1.353204e-06 -0.0003550300
## 14   14 -2.039371e-06 -0.008457965 0.008387726  4.461496e-07 -0.0003447940
## 15   15 -1.361326e-05 -0.008491993 0.008541378  8.081598e-07 -0.0003604161
## 16   16  1.040321e-04 -0.008446620 0.008781731 -3.321102e-06 -0.0003633760
## 17   17 -1.200747e-04 -0.008720090 0.008351969  4.692874e-06 -0.0003516303
## 18   18  2.150627e-07 -0.008481905 0.008287032  1.341648e-06 -0.0003420218
## 19   19 -2.978918e-05 -0.008526028 0.008291418  1.731428e-06 -0.0003486097
## 20   20  3.419670e-05 -0.008520439 0.008596648 -1.380269e-06 -0.0003515108
## 21   21  5.090546e-05 -0.008458966 0.008486989 -3.424439e-06 -0.0003573162
## 22   22  7.081519e-05 -0.008516264 0.008612679 -3.661250e-06 -0.0003652566
## 23   23  1.005823e-04 -0.008461843 0.008652124 -4.167574e-06 -0.0003569698
## 24   24 -1.386620e-05 -0.008574114 0.008750792  1.173981e-06 -0.0003596361
## 25   25 -3.481321e-05 -0.008345844 0.008489572  1.929241e-06 -0.0003490963
## 26   26 -8.883413e-06 -0.008770076 0.008663992  4.542192e-07 -0.0003609524
## 27   27 -1.250280e-05 -0.008648351 0.008645494 -4.199386e-07 -0.0003531757
## 28   28 -2.911668e-05 -0.008570166 0.008557858  1.139721e-06 -0.0003503626
## 29   29  4.352090e-05 -0.008367961 0.008408712 -1.477505e-06 -0.0003468140
## 30   30  2.205926e-05 -0.008342812 0.008392142 -9.249064e-07 -0.0003557689
##          Beta_Upper
## 1  0.0003524221
## 2  0.0003538972
## 3  0.0003485344
## 4  0.0003559096
## 5  0.0003570466
## 6  0.0003550267
## 7  0.0003661701
## 8  0.0003650451
## 9  0.0003513963
## 10 0.0003508431
## 11 0.0003565551
## 12 0.0003494206
## 13 0.0003565821
## 14 0.0003535280
## 15 0.0003545424
## 16 0.0003458320
## 17 0.0003535480
## 18 0.0003513804
## 19 0.0003533791
## 20 0.0003503306
## 21 0.0003523615
## 22 0.0003468725
## 23 0.0003517315

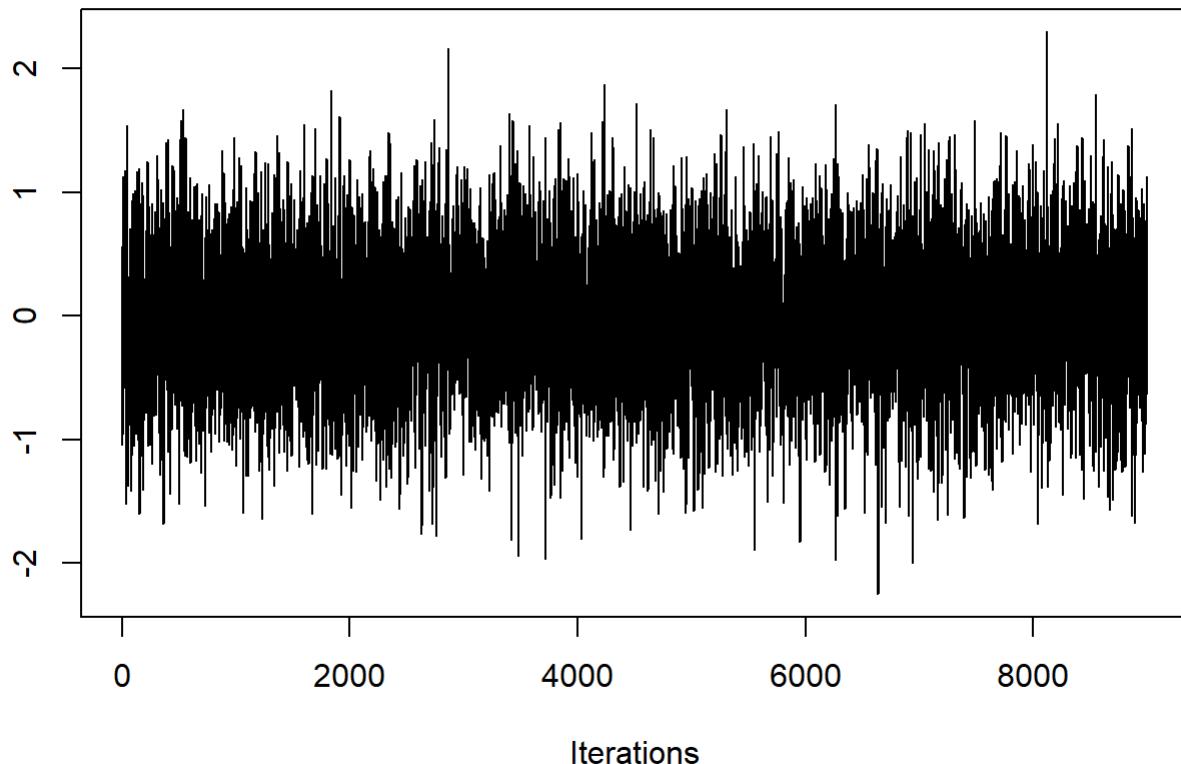
```

```
## 24 0.0003563169  
## 25 0.0003475793  
## 26 0.0003581488  
## 27 0.0003580739  
## 28 0.0003610941  
## 29 0.0003457764  
## 30 0.0003550135
```

```
# CONVERGENCE DIAGNOSTICS
```

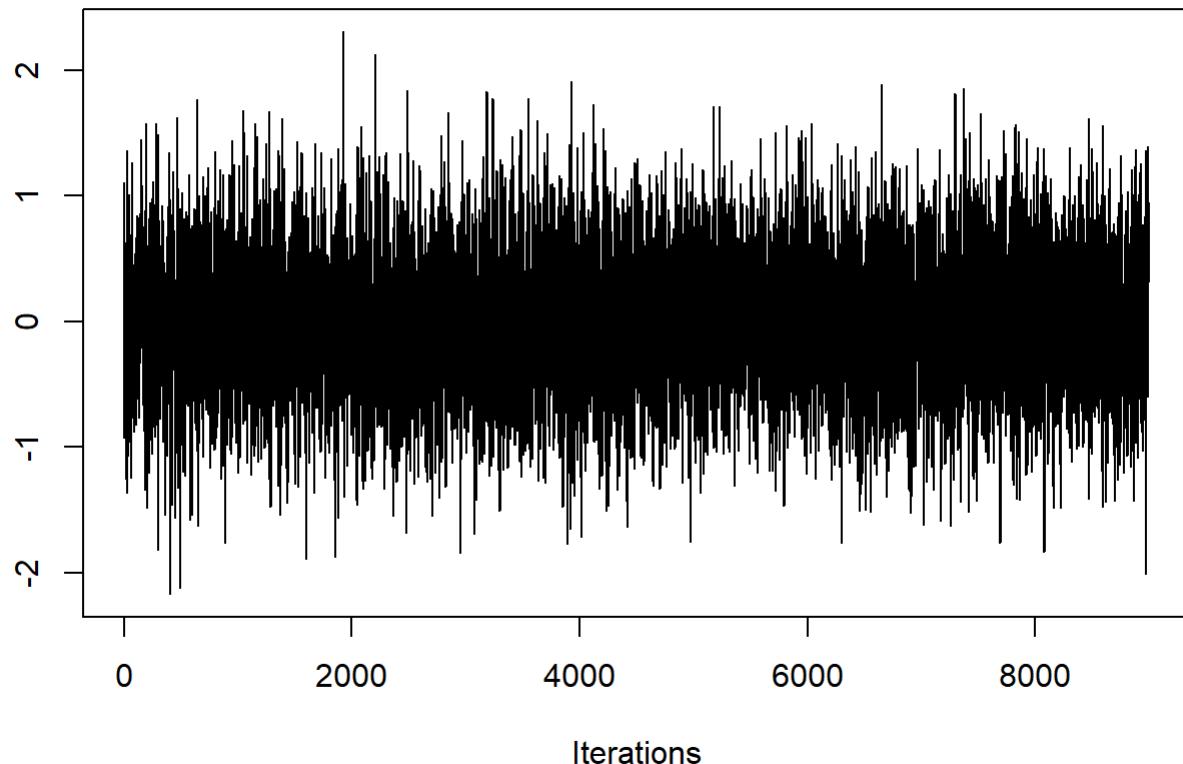
```
mcmc_mu <- mcmc(mu_post)  
mcmc_tau <- mcmc(tau_post)  
  
# Trace plots  
traceplot(mcmc_mu[,1], main = "Traceplot of mu_1")
```

**Traceplot of mu\_1**



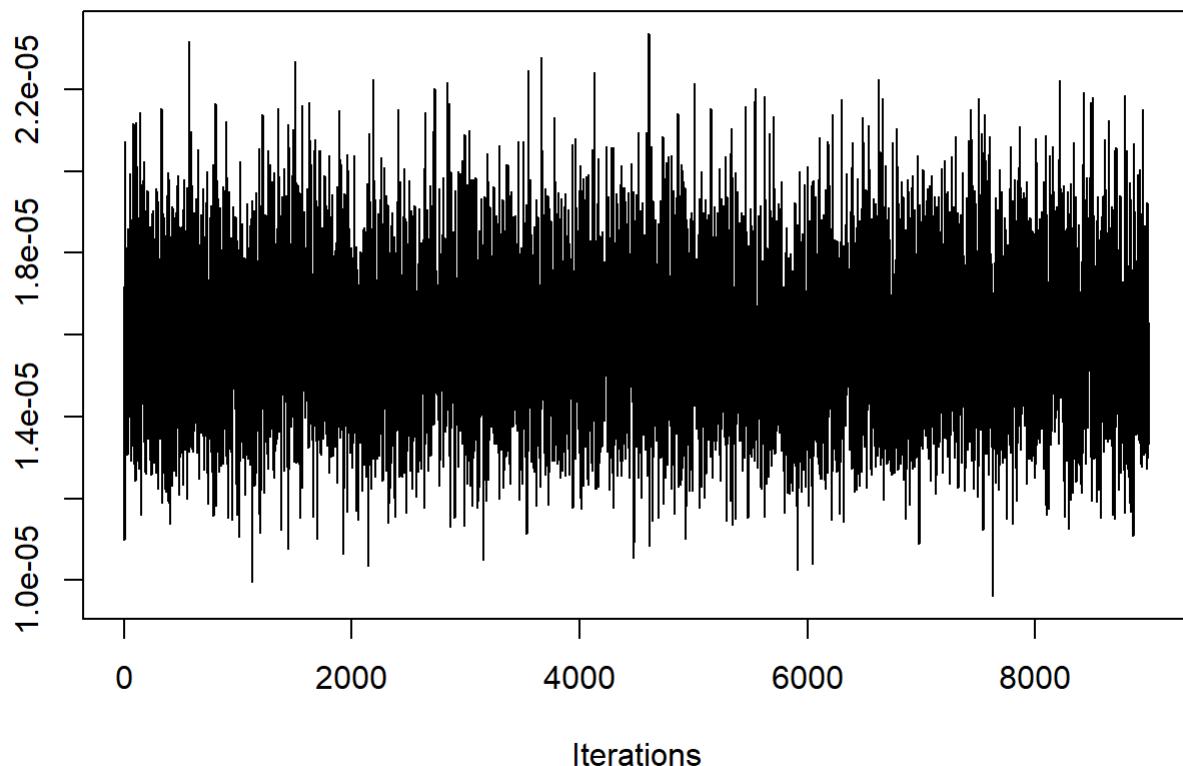
```
traceplot(mcmc_mu[,2], main = "Traceplot of mu_2")
```

### Traceplot of mu\_2

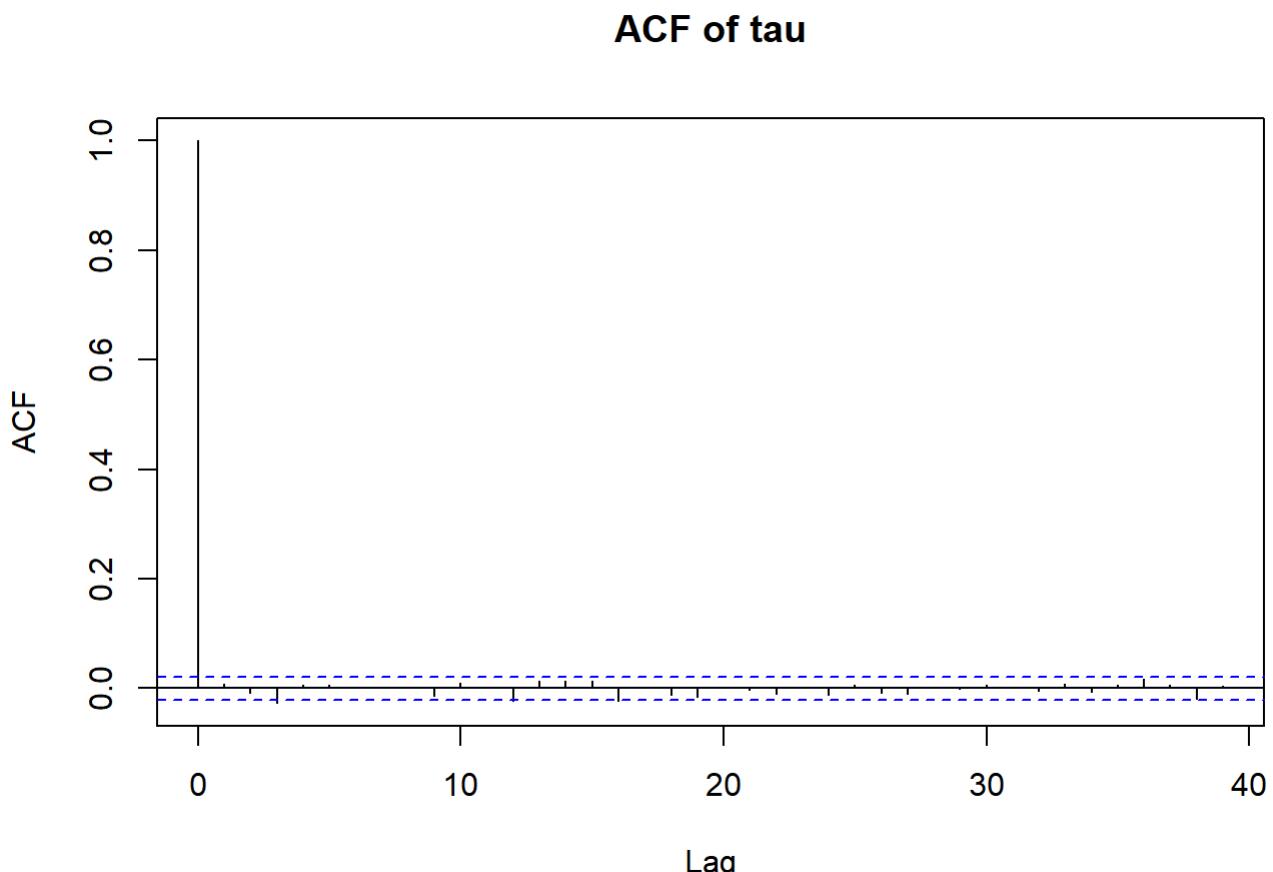


```
traceplot(mcmc_tau, main = "Traceplot of tau")
```

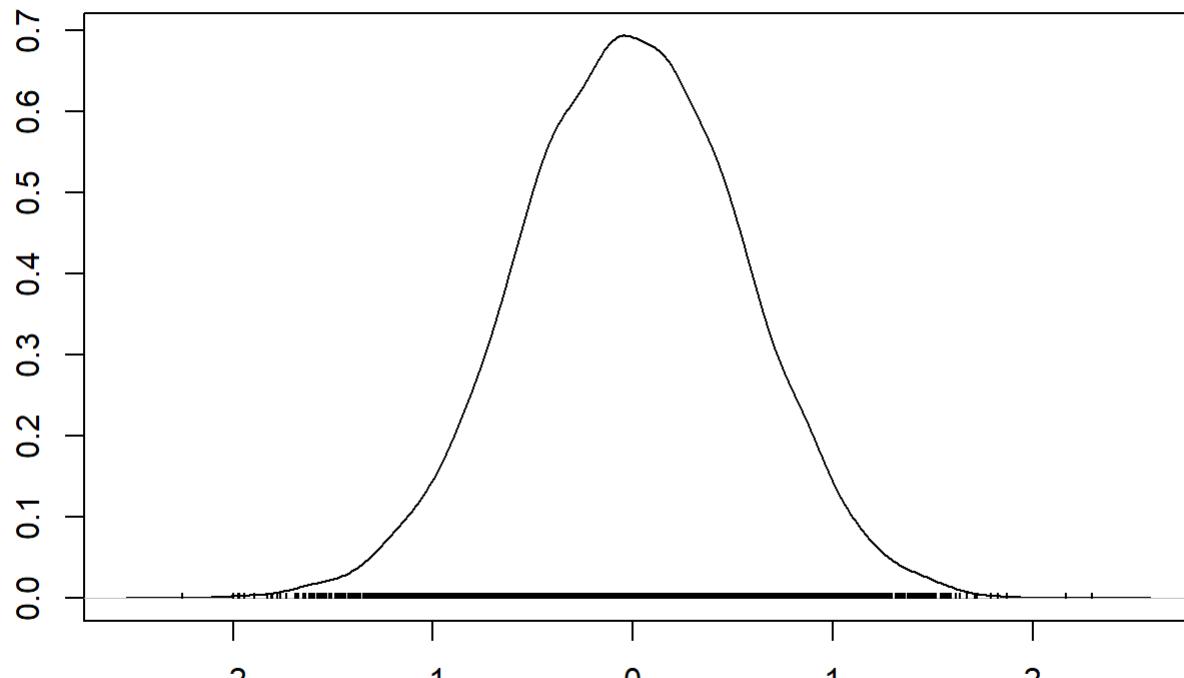
### Traceplot of tau



```
# Autocorrelation  
acf(mcmc_tau, main = "ACF of tau")
```

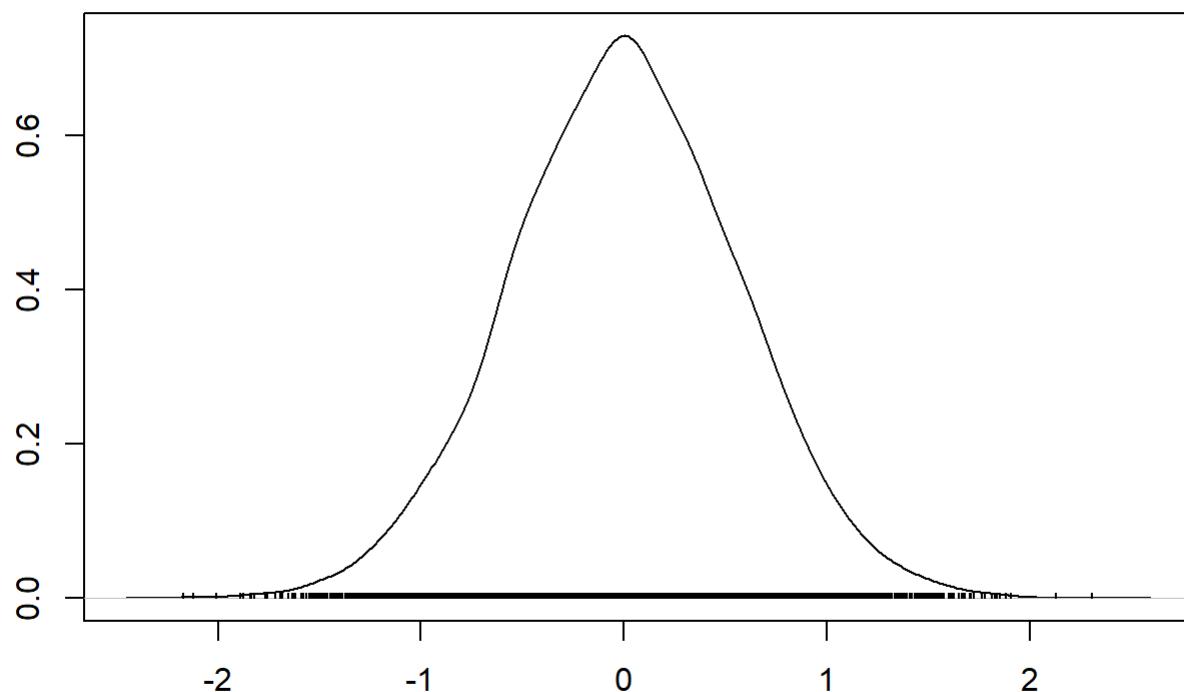


```
# Gelman diagnostics (if using multiple chains)  
# gelman.diag(as.mcmc.list(list(chain1, chain2)))  
  
# Posterior density  
densplot(mcmc_mu[,1], main = "Posterior density of mu[1]")
```

**Posterior density of mu[1]**

N = 9000 Bandwidth = 0.0961

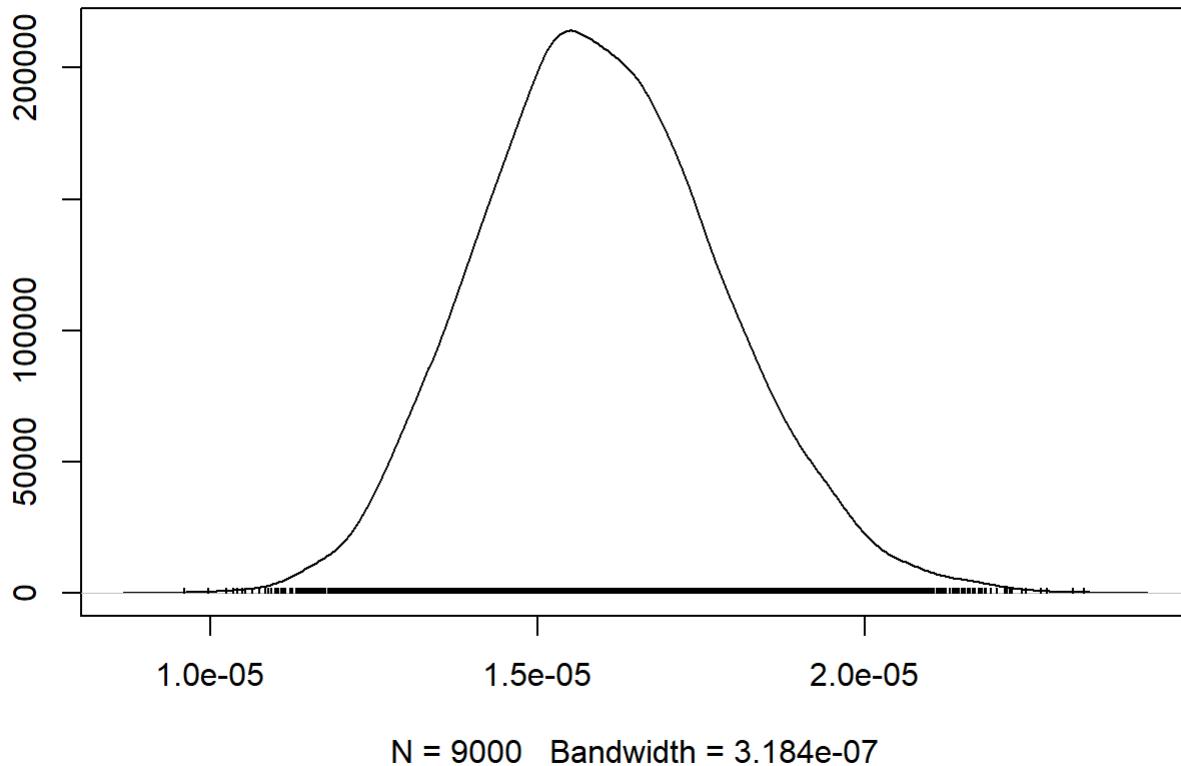
```
densplot(mcmc_mu[,2], main = "Posterior density of mu[2]")
```

**Posterior density of mu[2]**

N = 9000 Bandwidth = 0.09579

```
densplot(mcmc_tau, main = "Posterior density of tau")
```

### Posterior density of tau



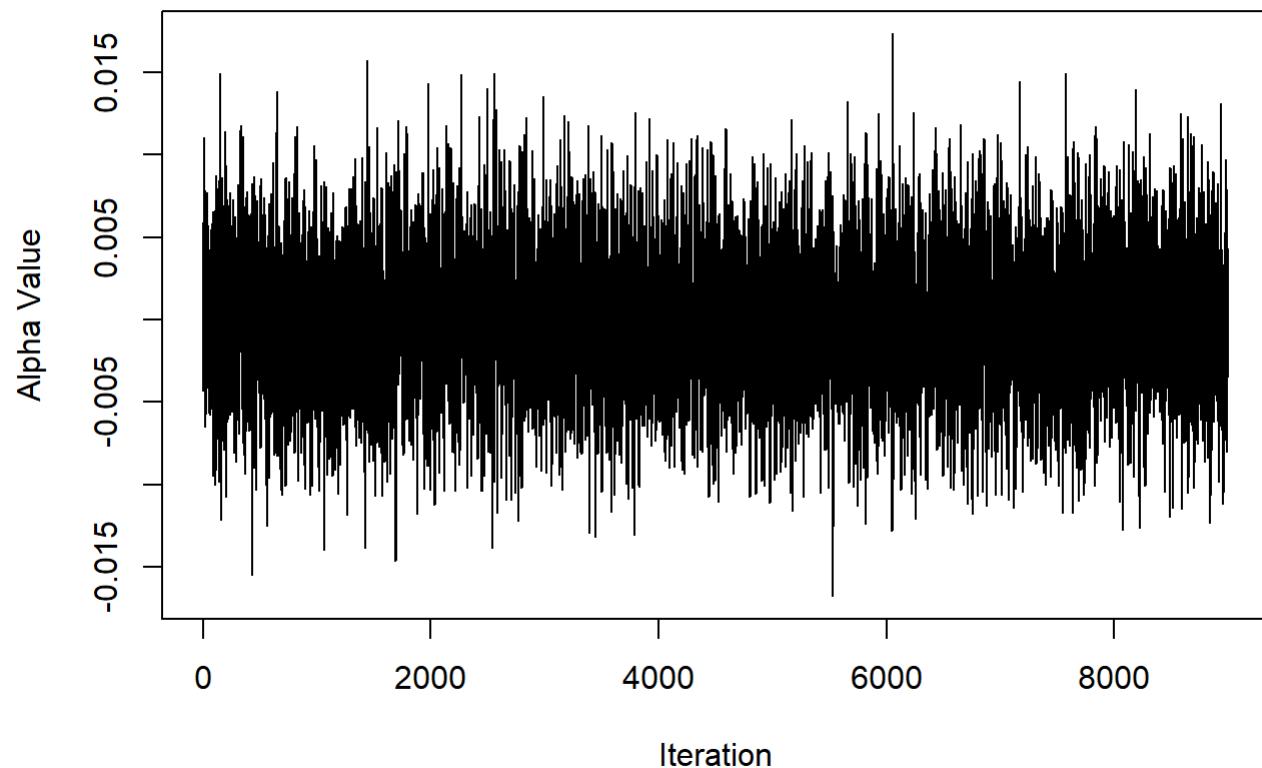
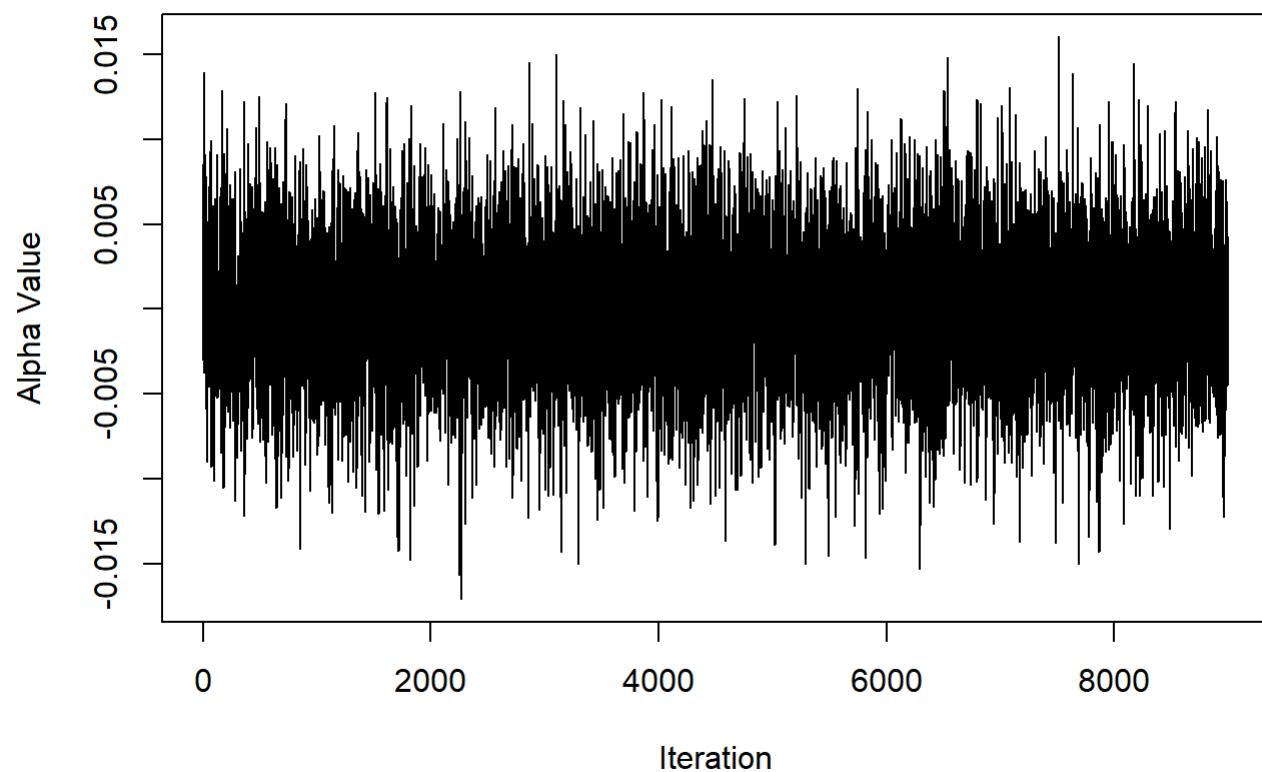
```
# Get alpha and beta
alpha_post <- theta_post[, 1, ]
beta_post <- theta_post[, 2, ]

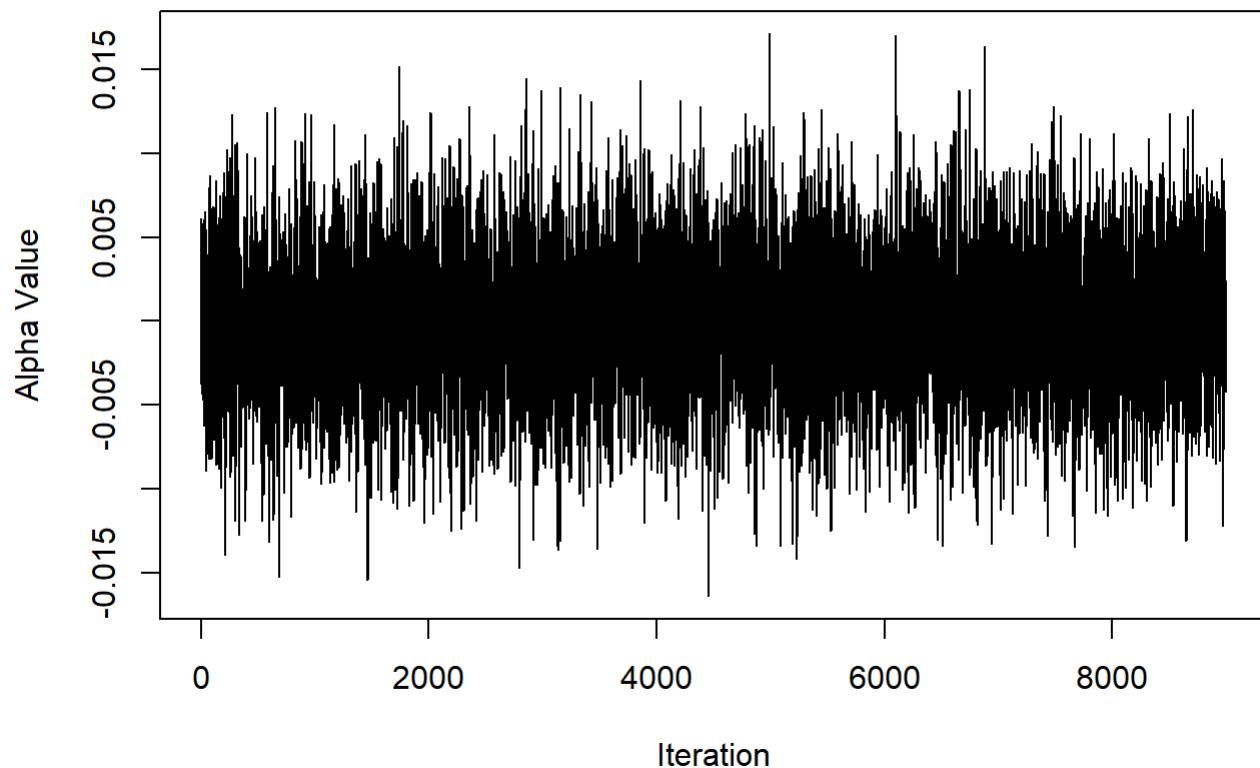
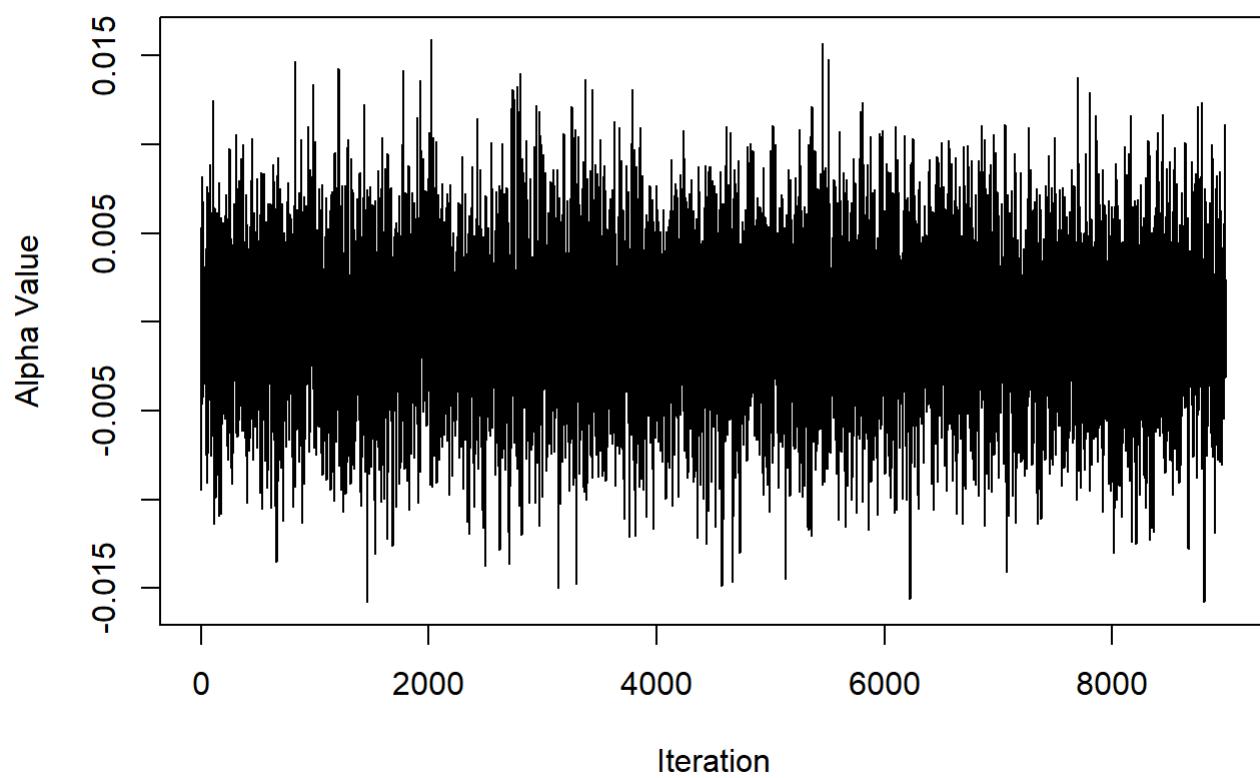
# Reshape the alpha and beta samples into a long format for plotting
alpha_df <- data.frame(iteration = rep(1:9000, each = 30),
                       value = as.vector(t(alpha_post)), # Transpose to stack the columns (observations)
                       observation = rep(1:30, times = 9000),
                       param = "alpha")

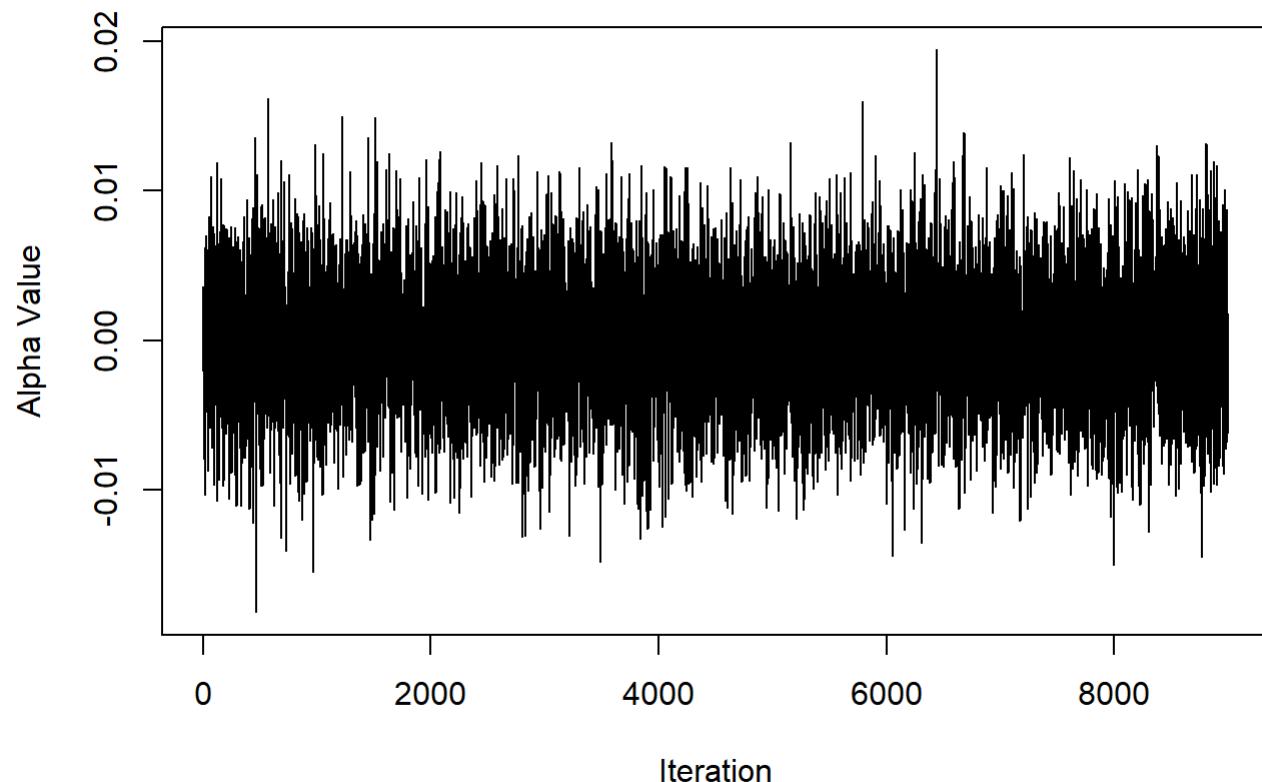
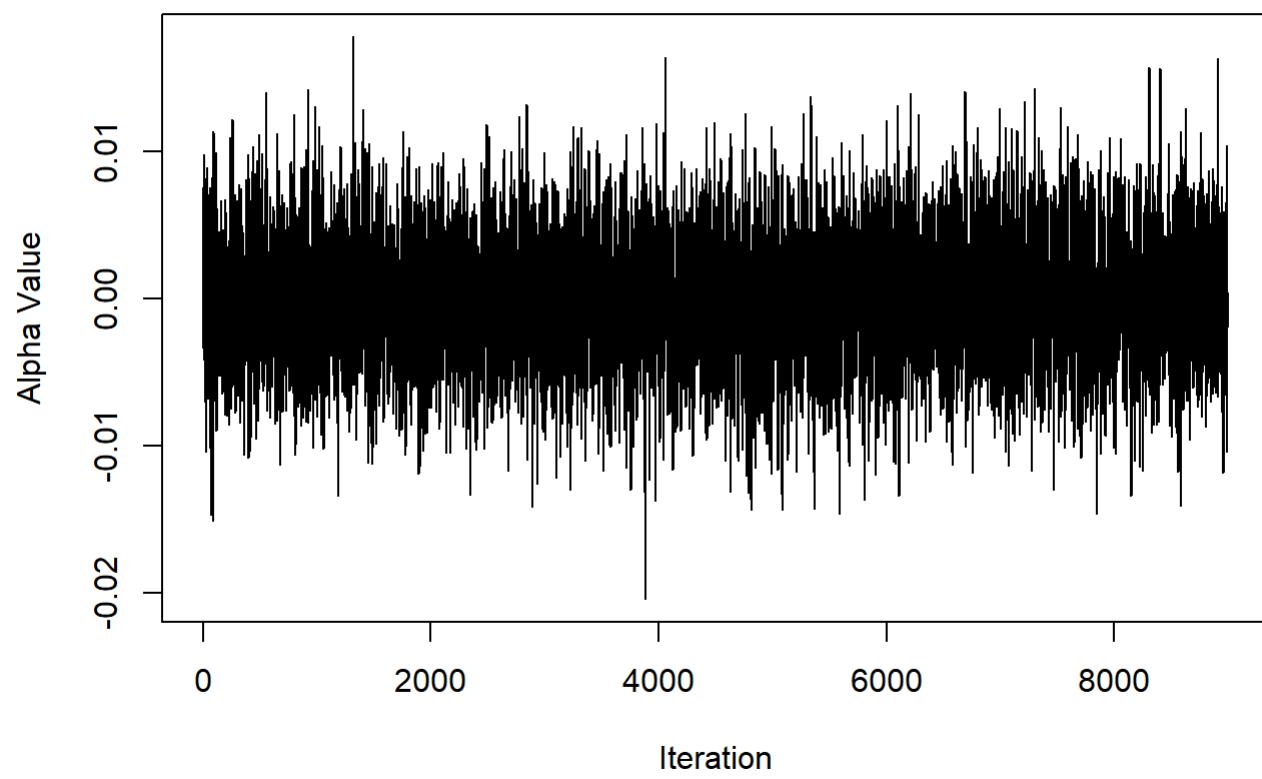
beta_df <- data.frame(iteration = rep(1:9000, each = 30),
                       value = as.vector(t(beta_post)),
                       observation = rep(1:30, times = 9000),
                       param = "beta")

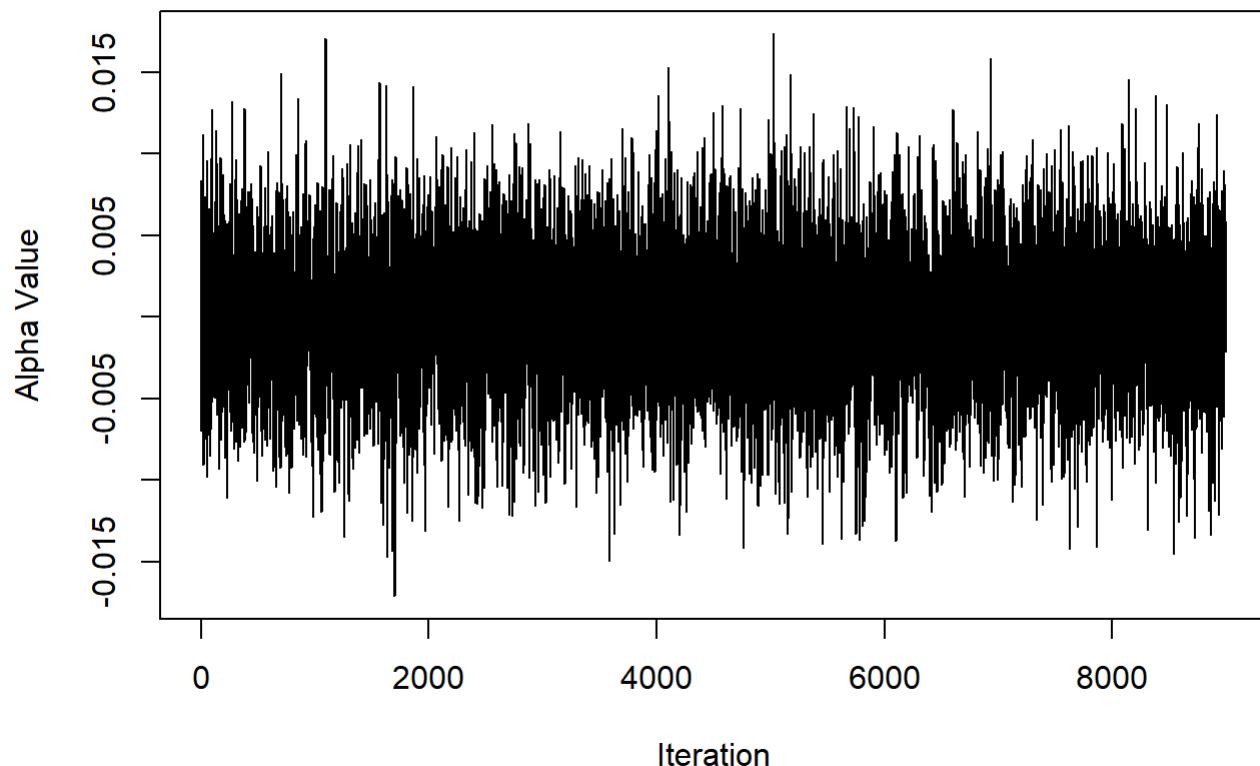
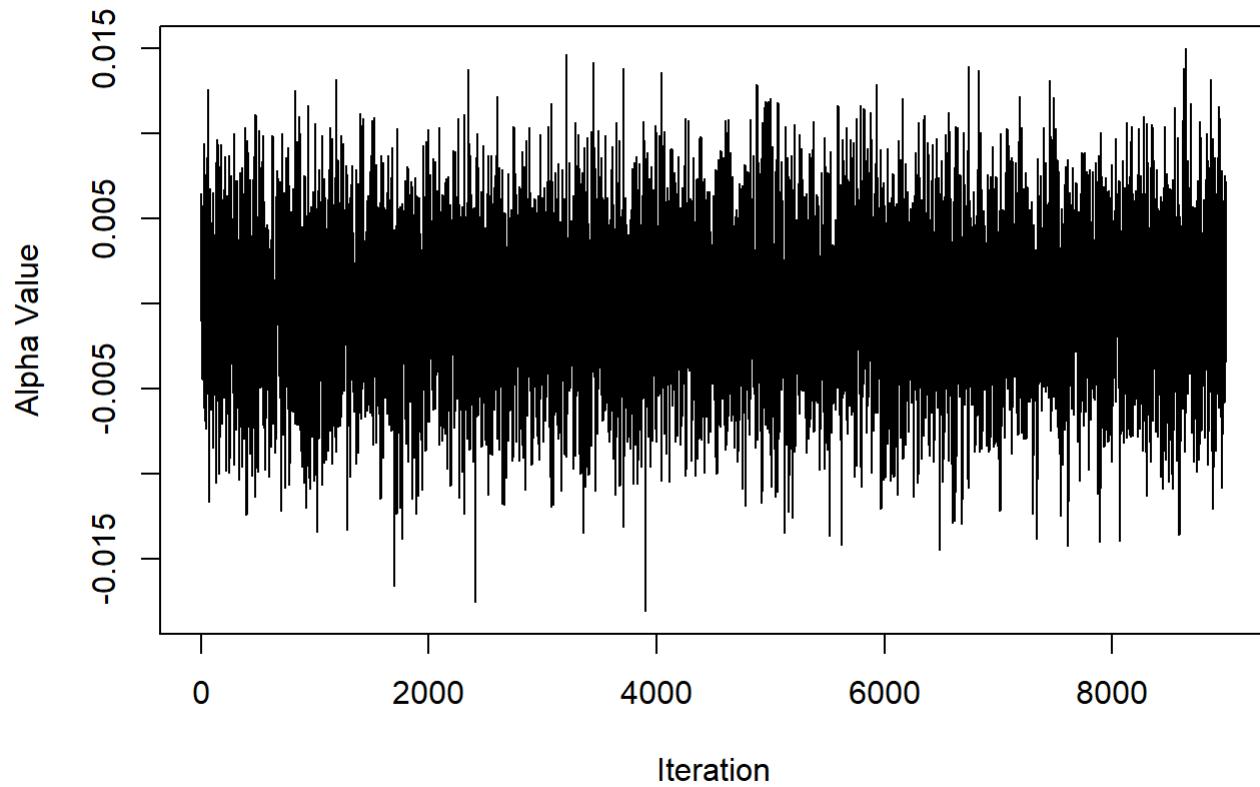
# Combine the data frames for plotting
theta_df <- rbind(alpha_df, beta_df)

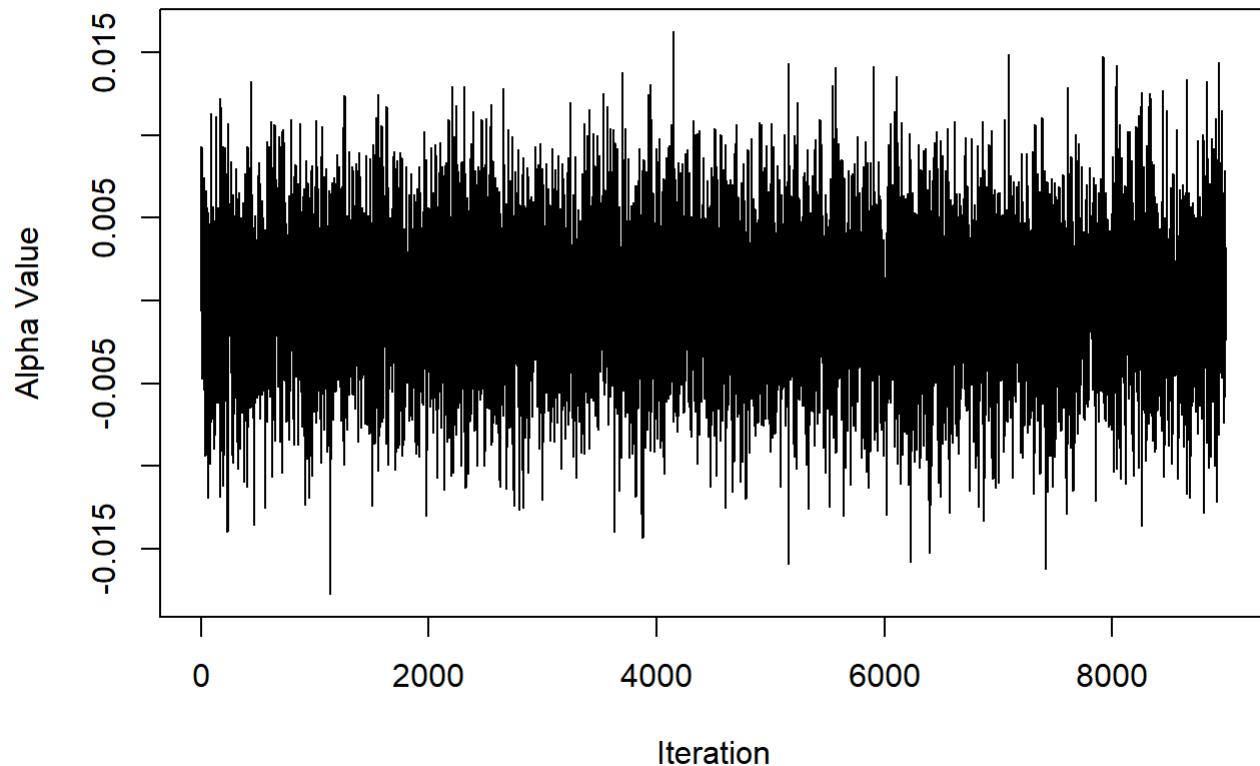
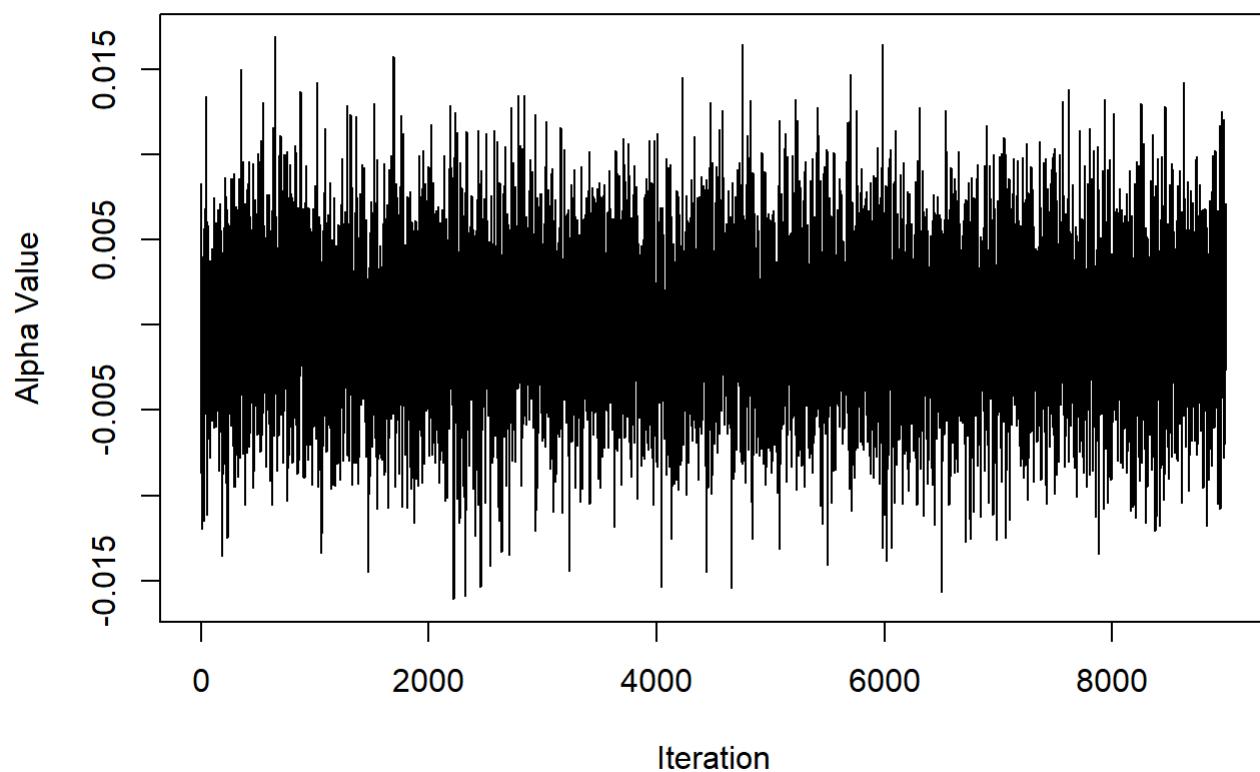
# Trace plots for alpha
for (i in 1:30) {
  # Plot trace for alpha (i-th observation)
  plot(1:9000, alpha_post[i, ], type = "l",
        main = paste("Trace plot for Alpha (Observation", i, ")"),
        xlab = "Iteration", ylab = "Alpha Value", col = "black")
}
```

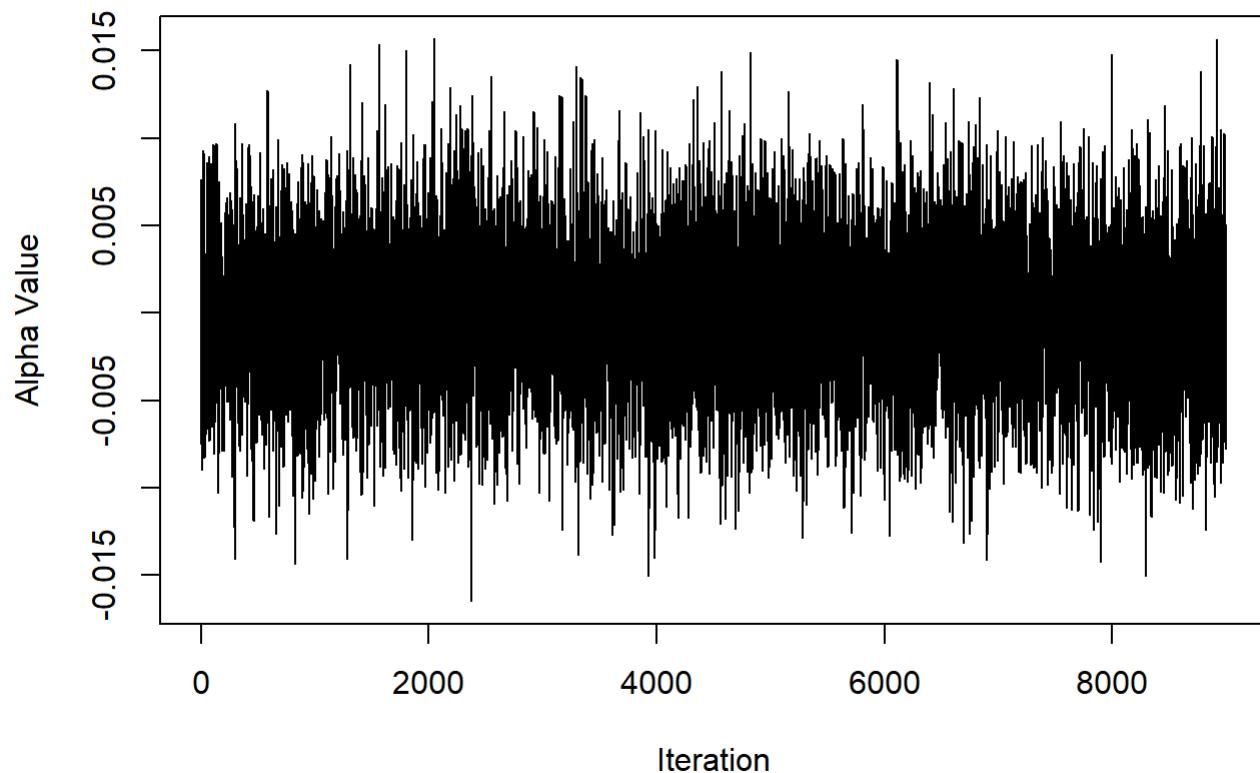
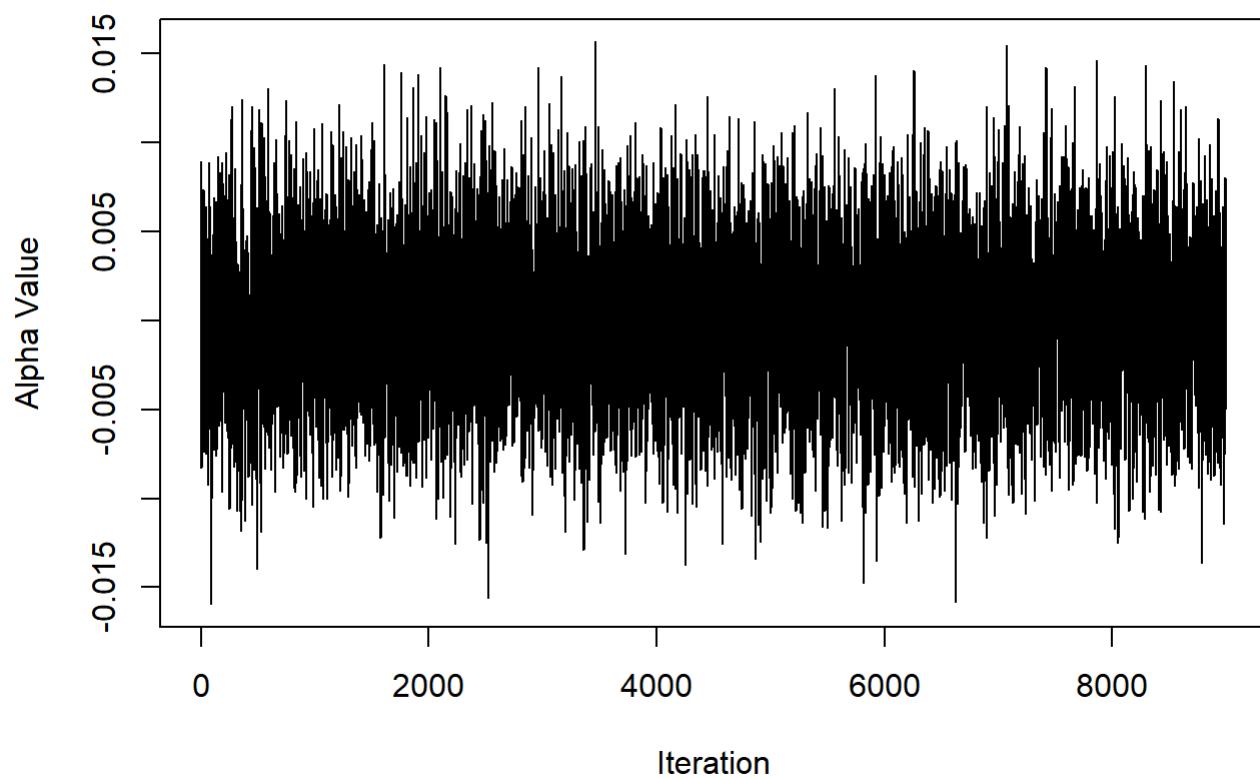
**Trace plot for Alpha (Observation 1 )****Trace plot for Alpha (Observation 2 )**

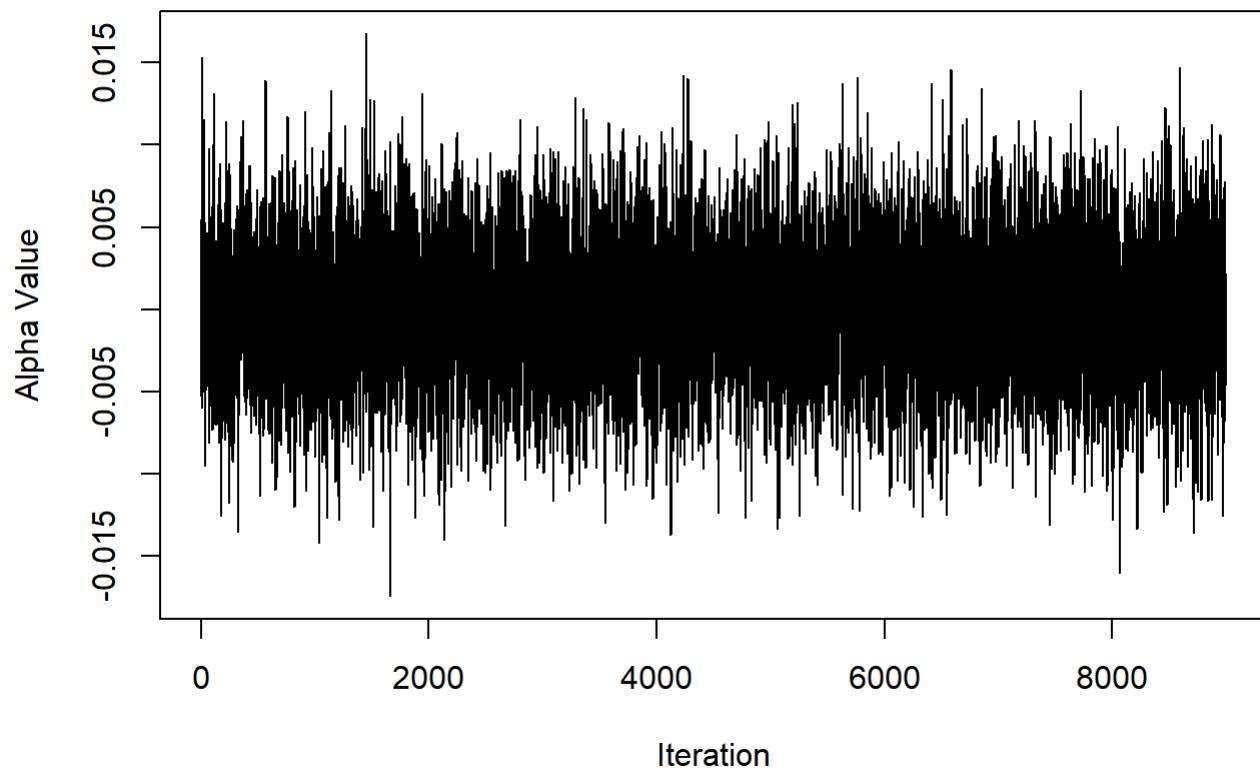
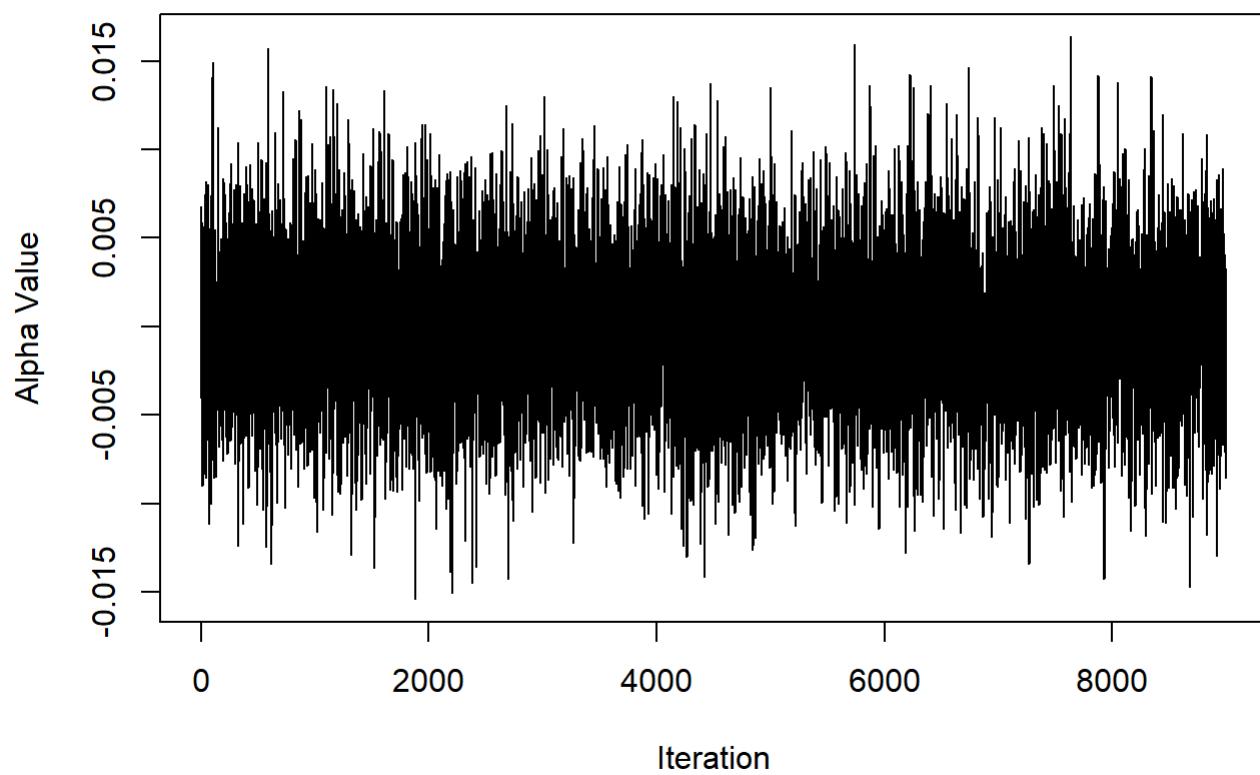
**Trace plot for Alpha (Observation 3 )****Trace plot for Alpha (Observation 4 )**

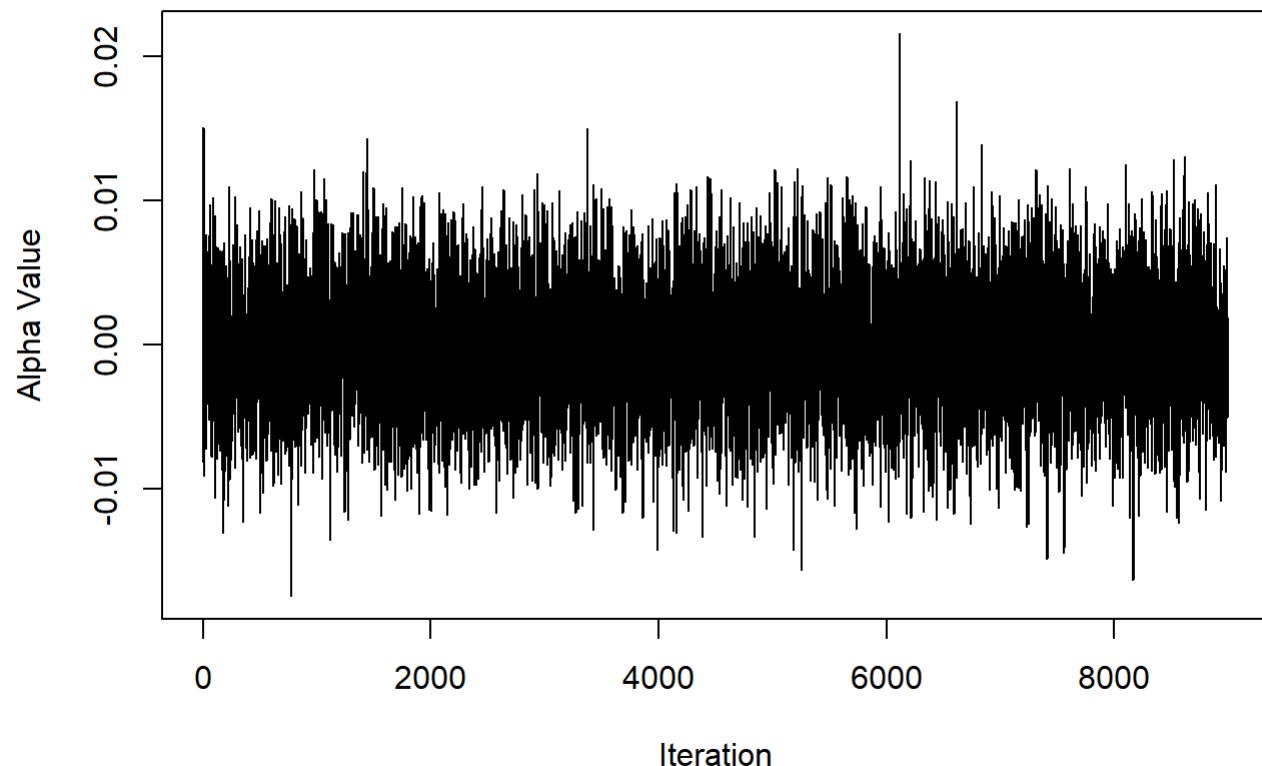
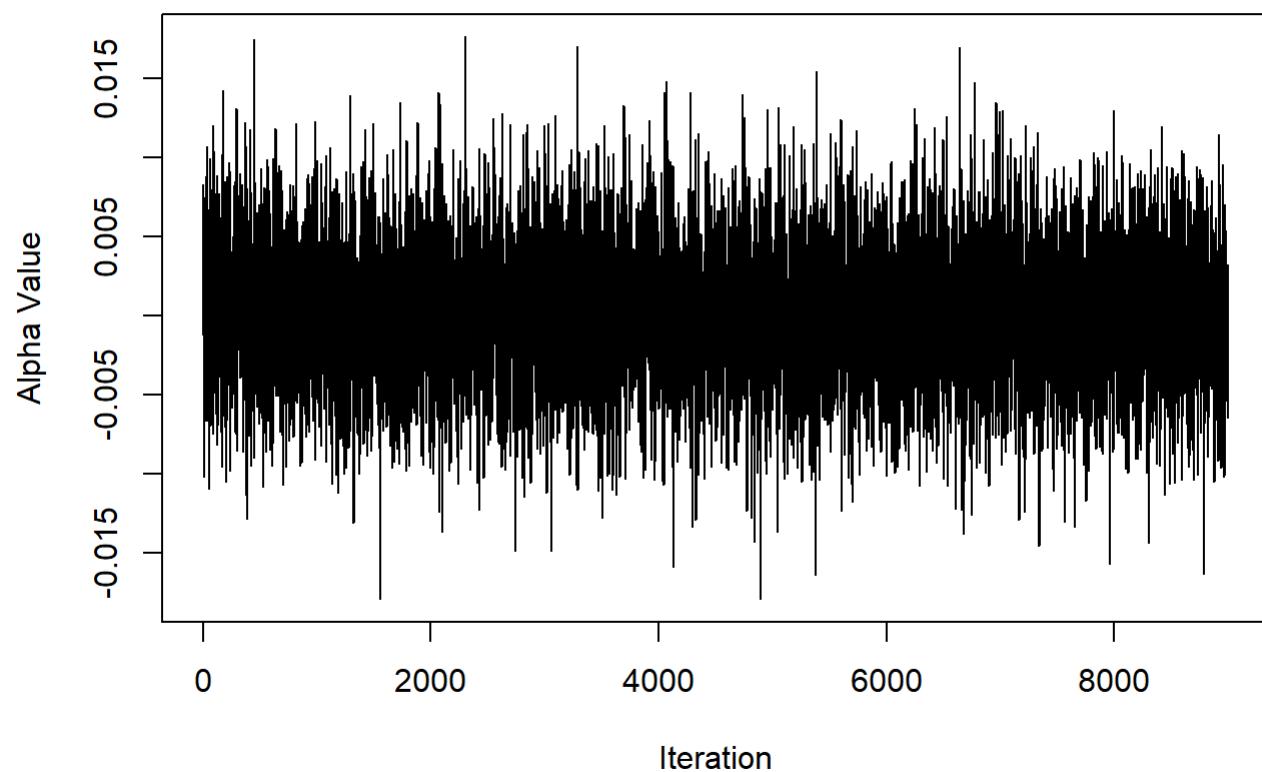
**Trace plot for Alpha (Observation 5 )****Trace plot for Alpha (Observation 6 )**

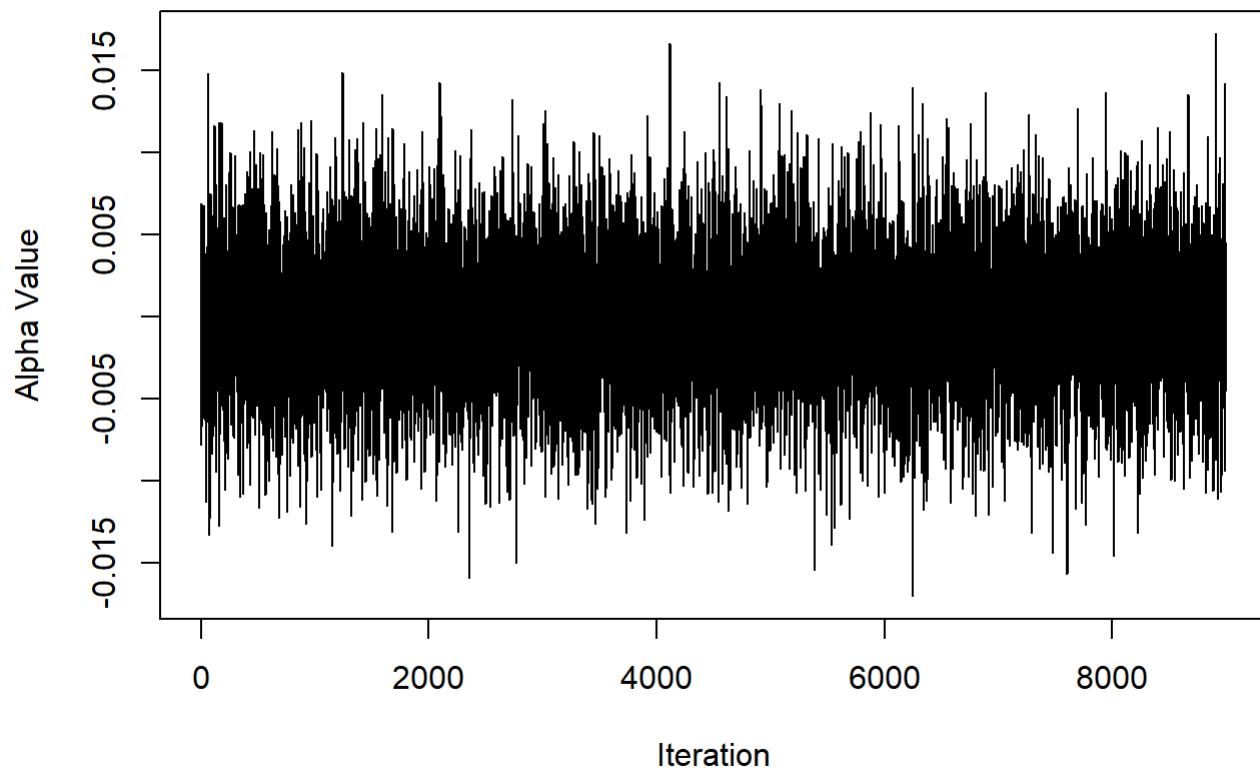
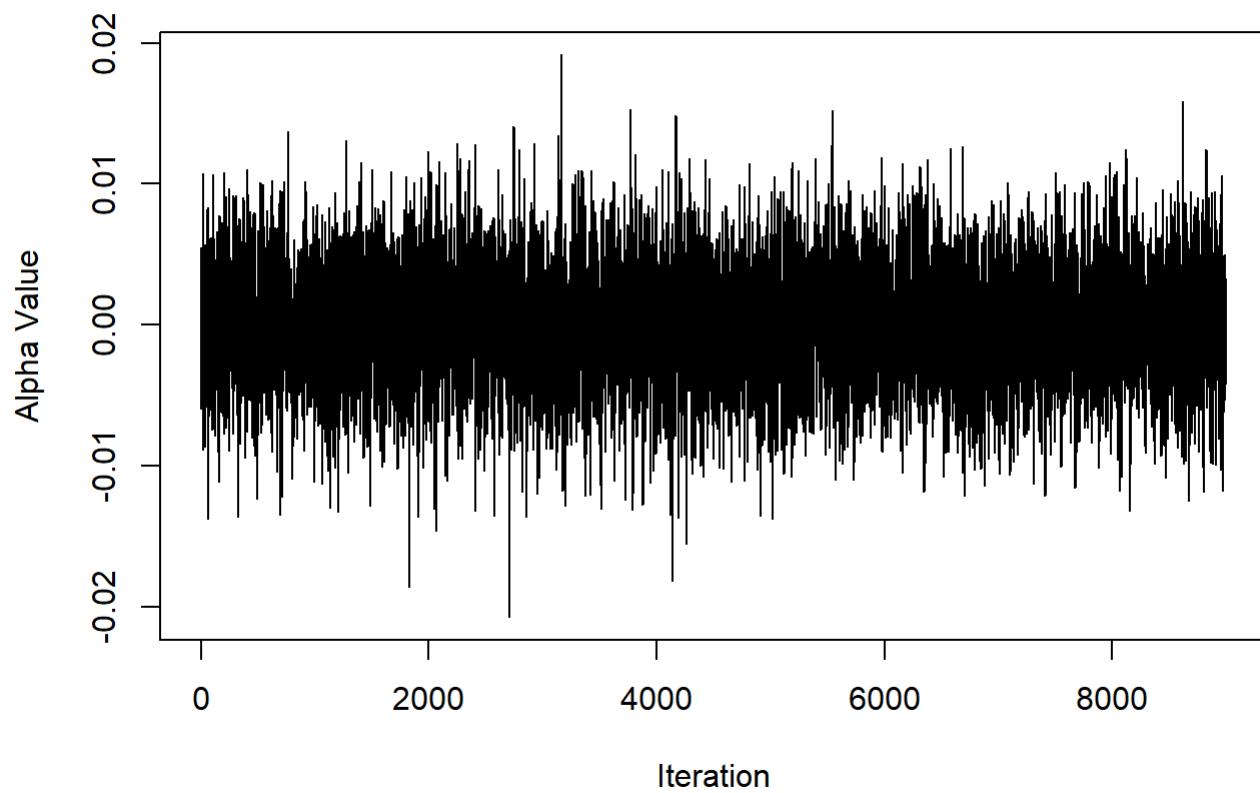
**Trace plot for Alpha (Observation 7 )****Trace plot for Alpha (Observation 8 )**

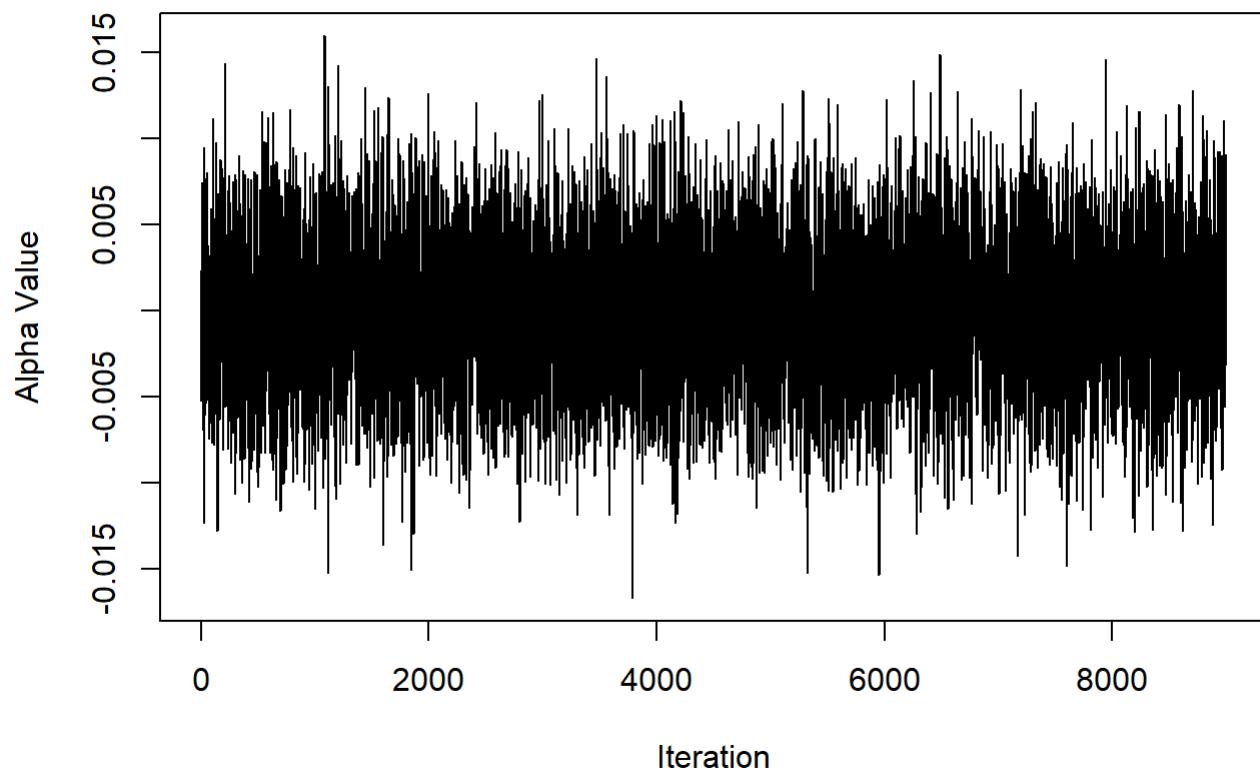
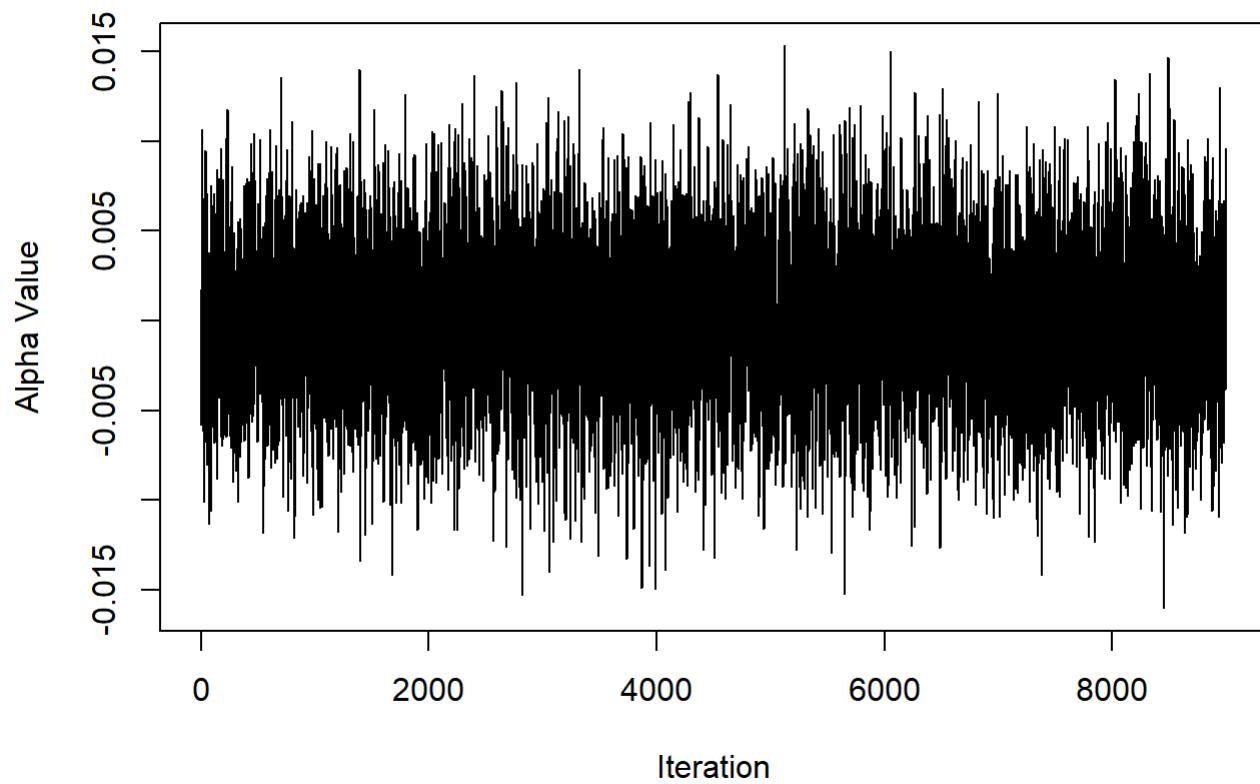
**Trace plot for Alpha (Observation 9 )****Trace plot for Alpha (Observation 10 )**

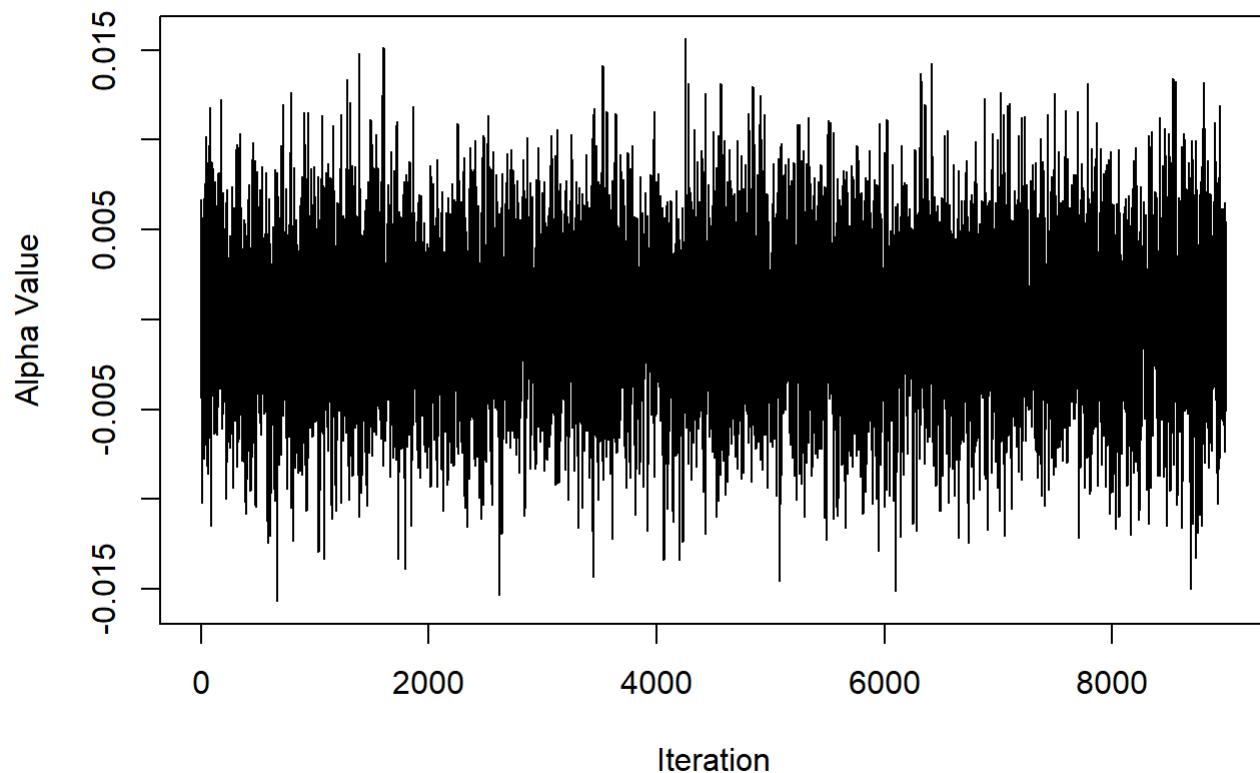
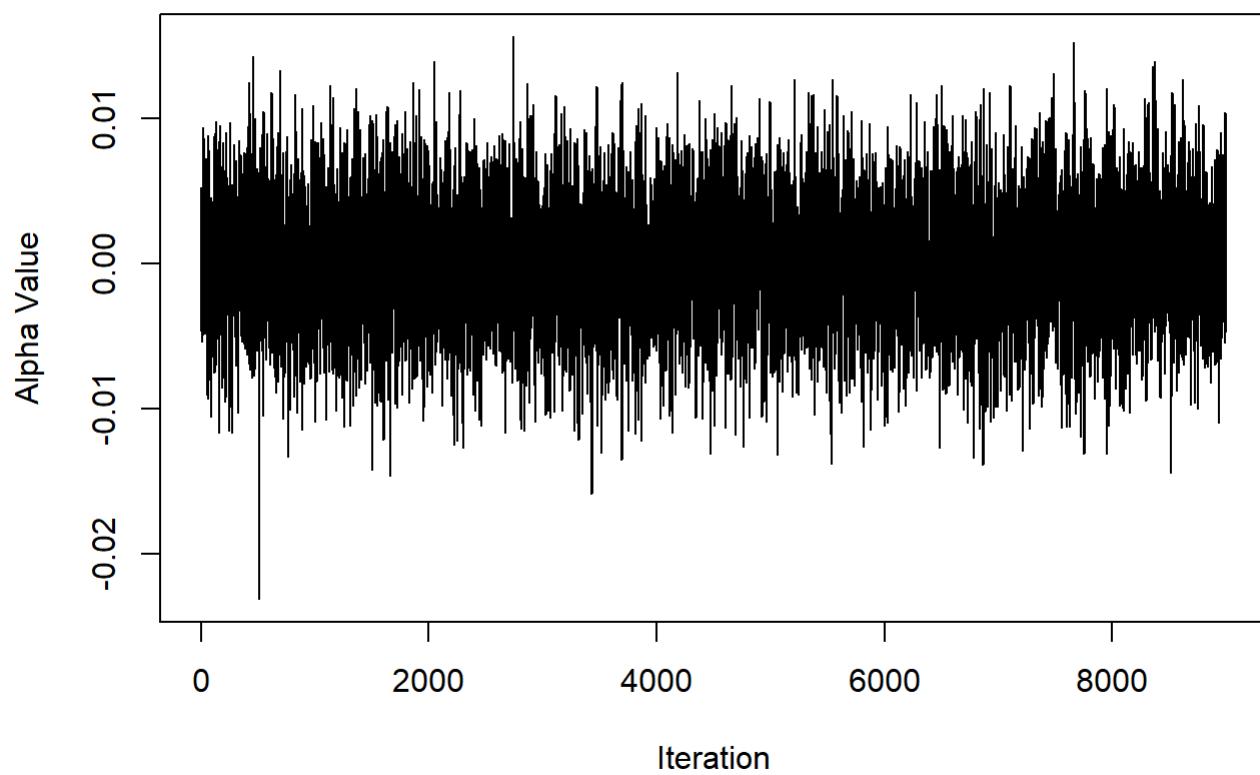
**Trace plot for Alpha (Observation 11 )****Trace plot for Alpha (Observation 12 )**

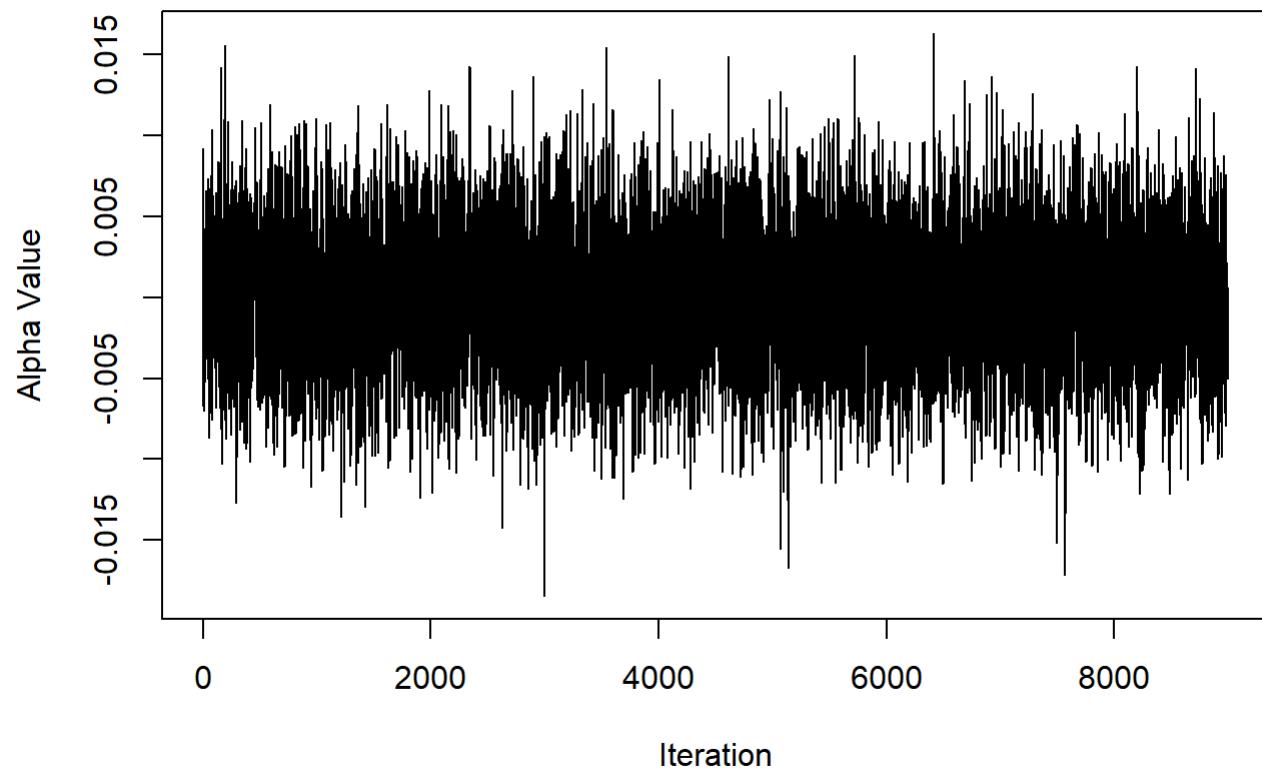
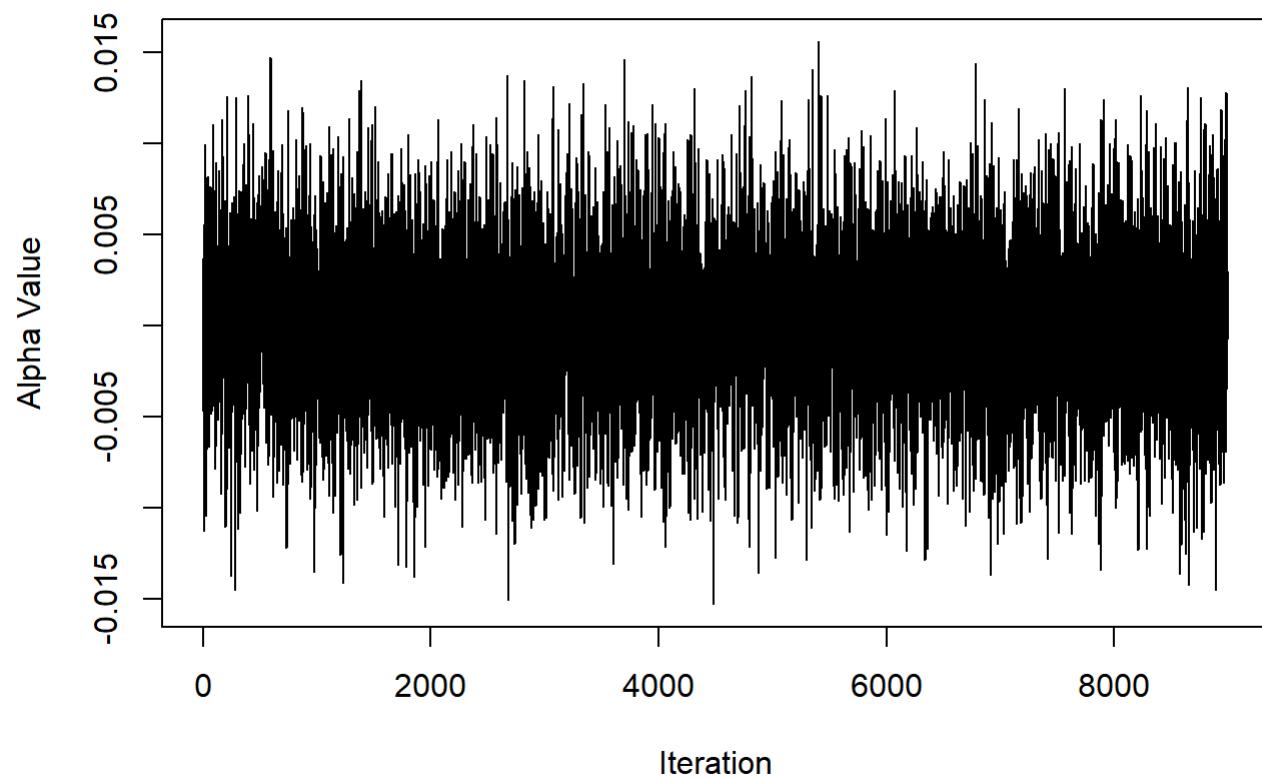
**Trace plot for Alpha (Observation 13 )****Trace plot for Alpha (Observation 14 )**

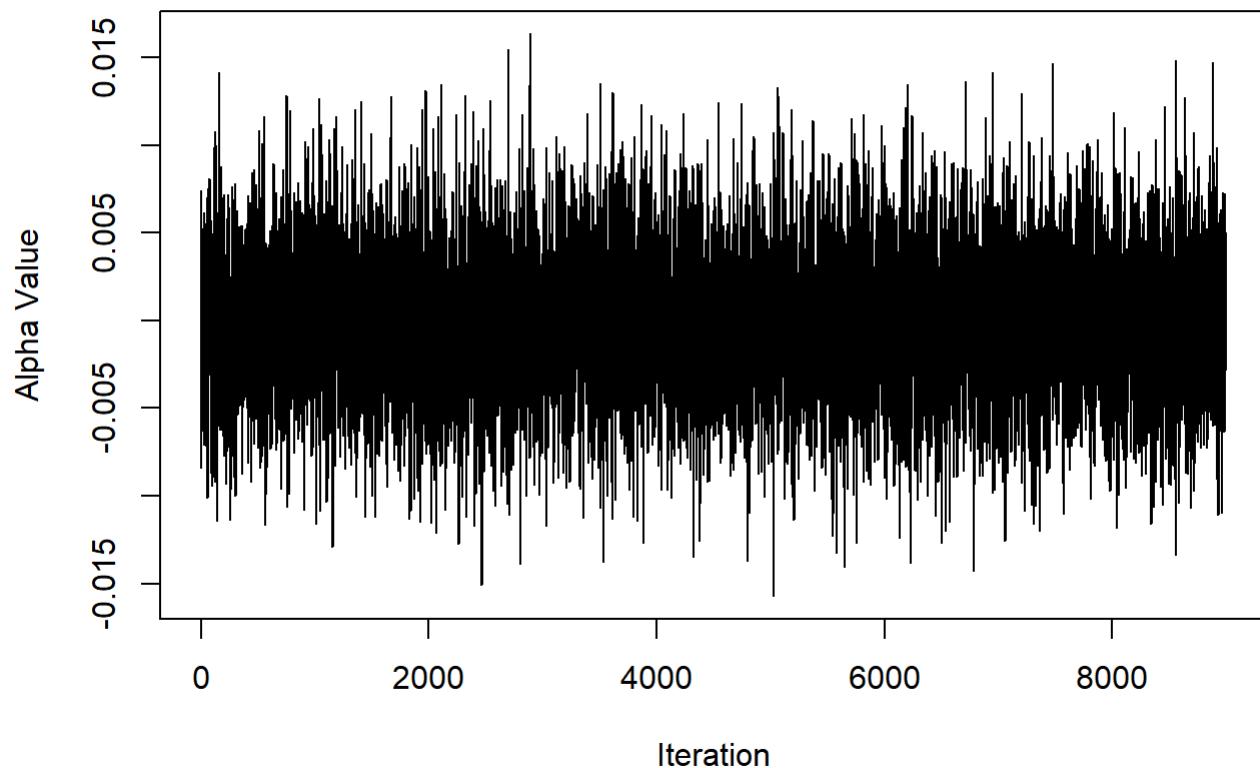
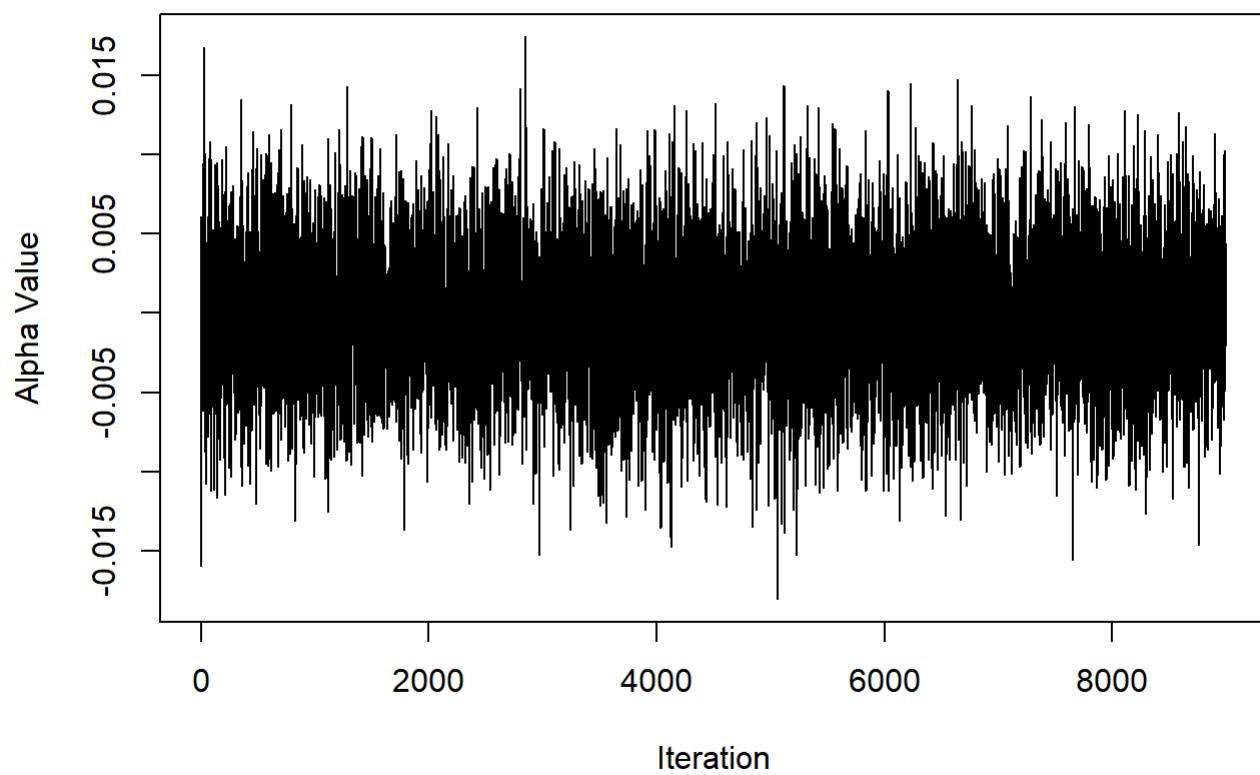
**Trace plot for Alpha (Observation 15 )****Trace plot for Alpha (Observation 16 )**

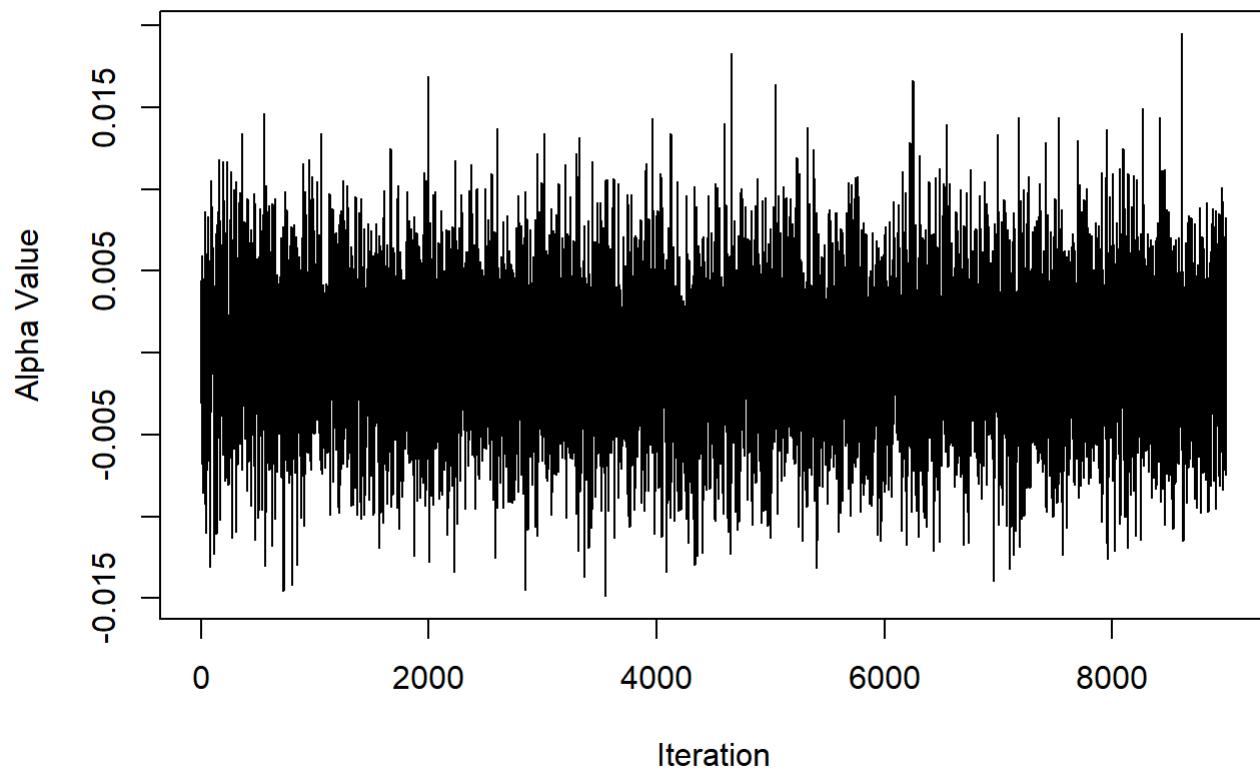
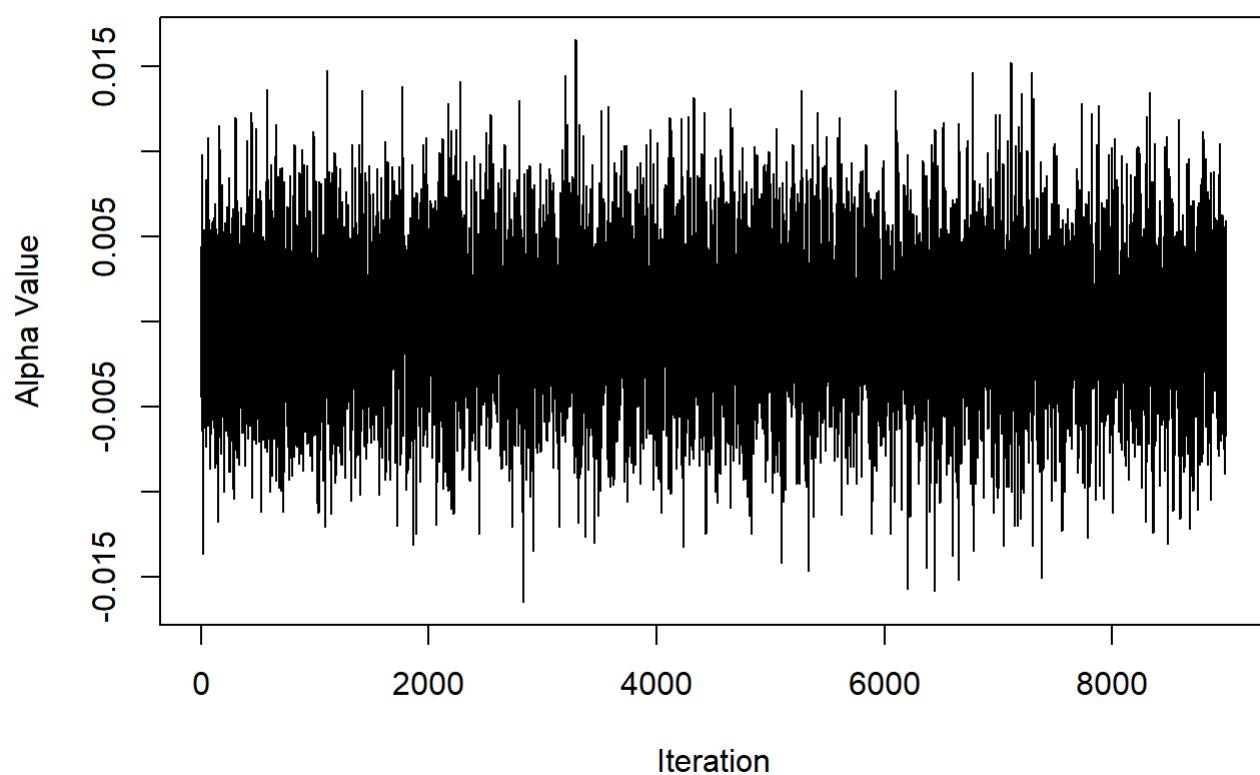
**Trace plot for Alpha (Observation 17 )****Trace plot for Alpha (Observation 18 )**

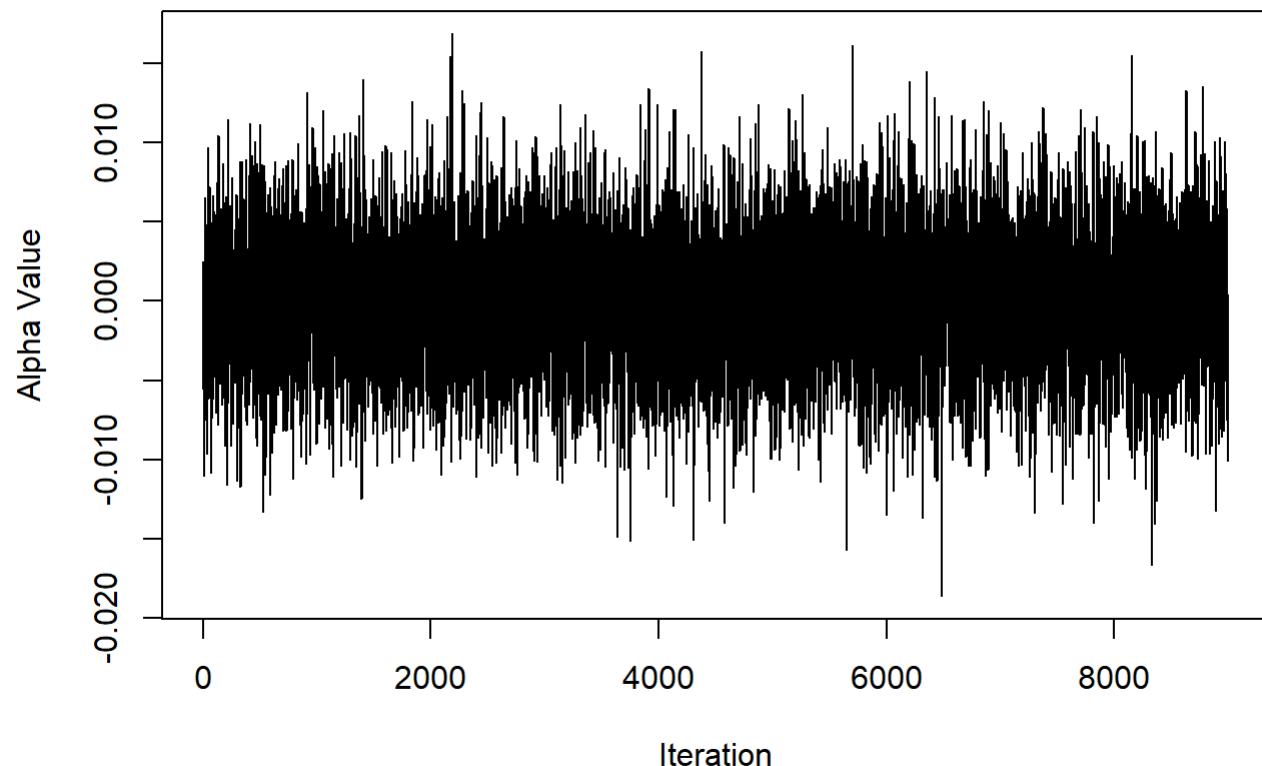
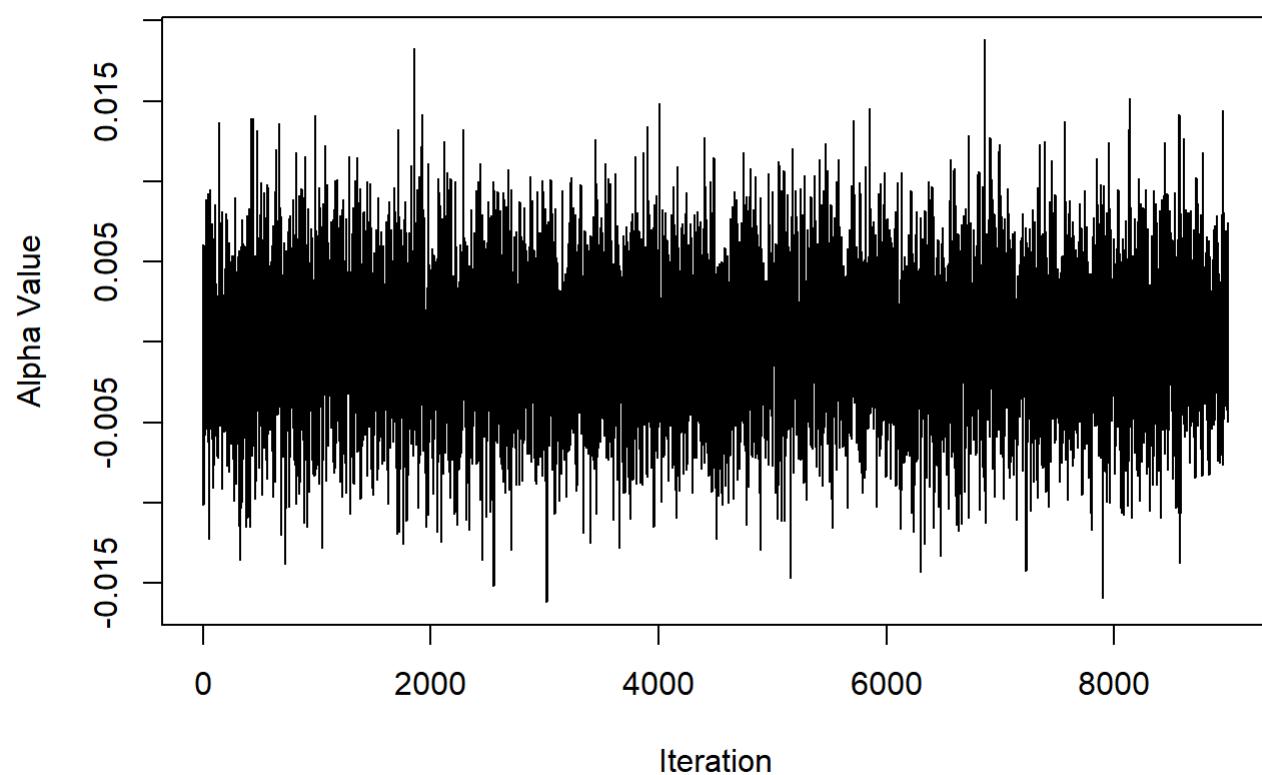
**Trace plot for Alpha (Observation 19 )****Trace plot for Alpha (Observation 20 )**

**Trace plot for Alpha (Observation 21 )****Trace plot for Alpha (Observation 22 )**

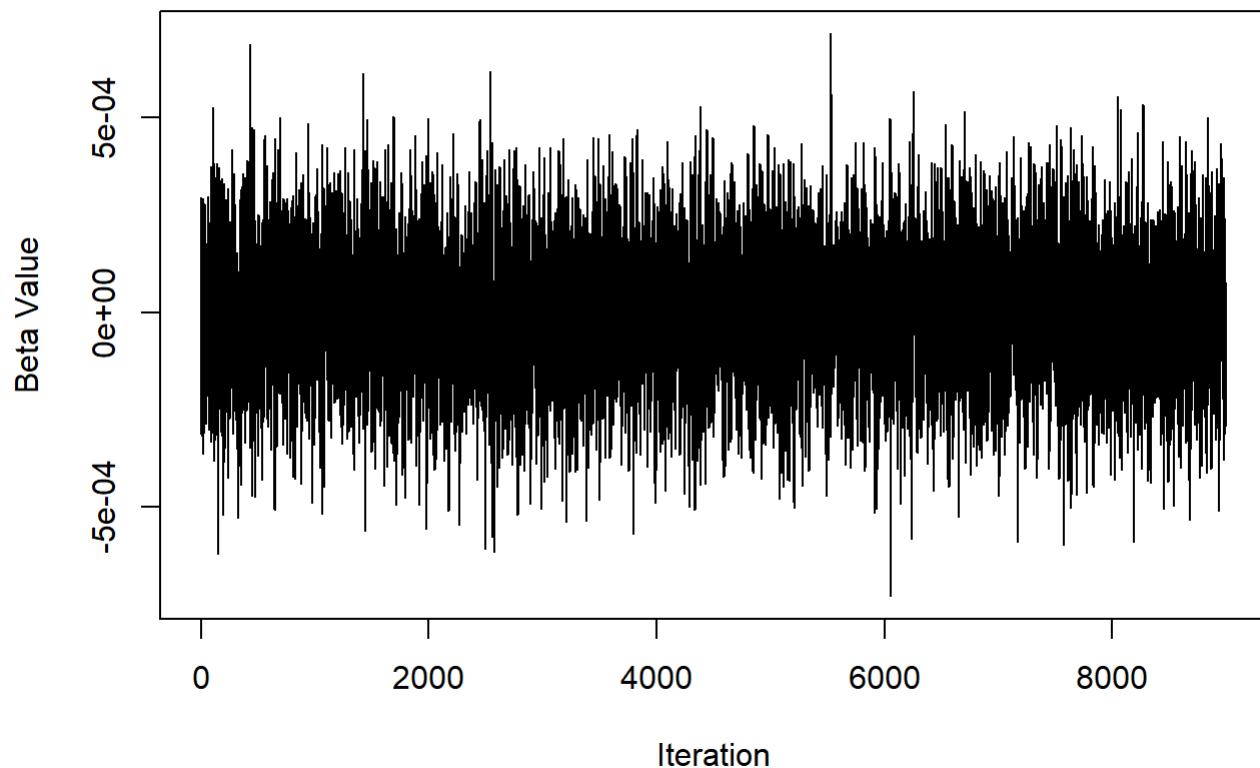
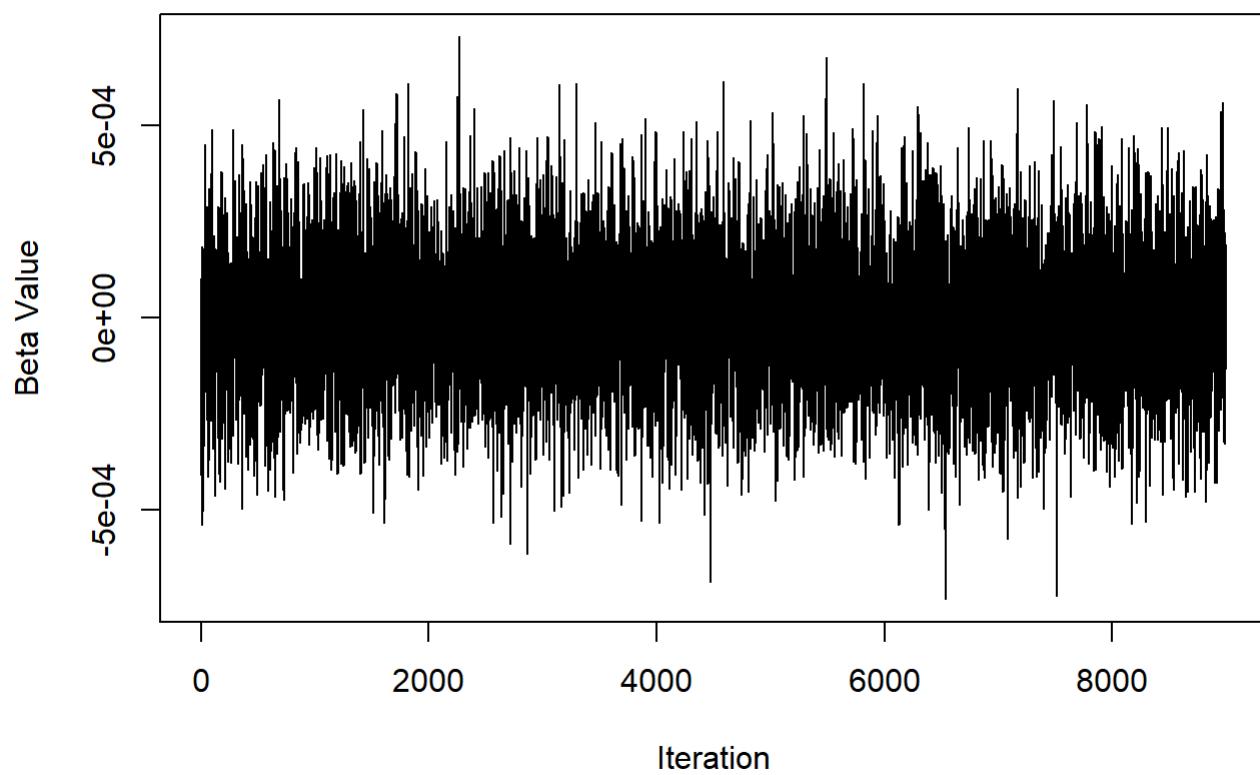
**Trace plot for Alpha (Observation 23 )****Trace plot for Alpha (Observation 24 )**

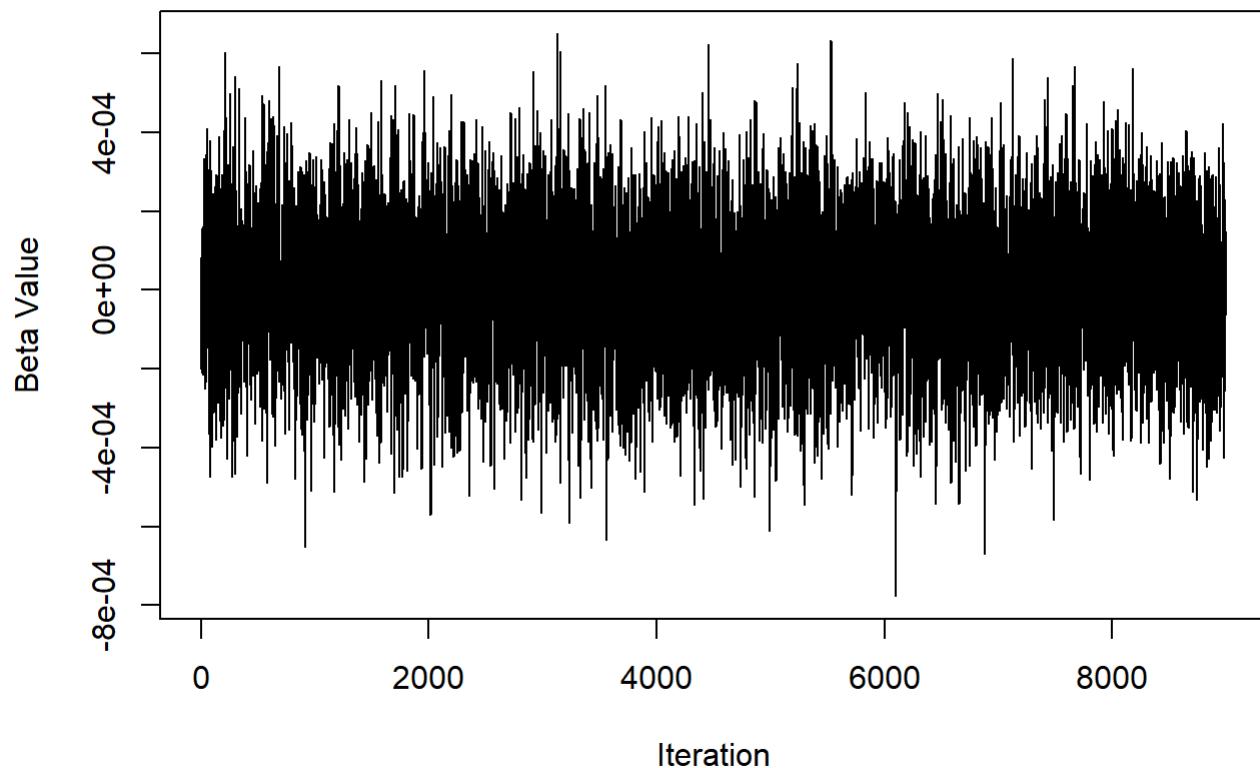
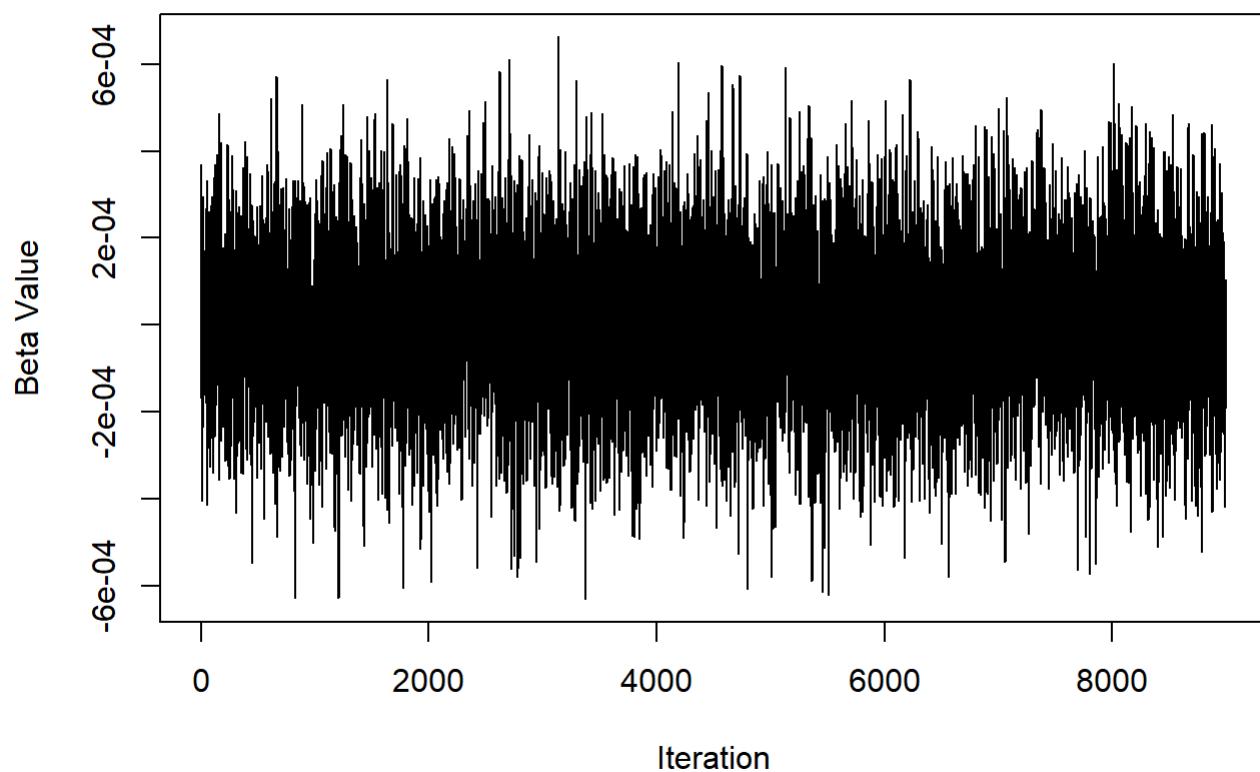
**Trace plot for Alpha (Observation 25 )****Trace plot for Alpha (Observation 26 )**

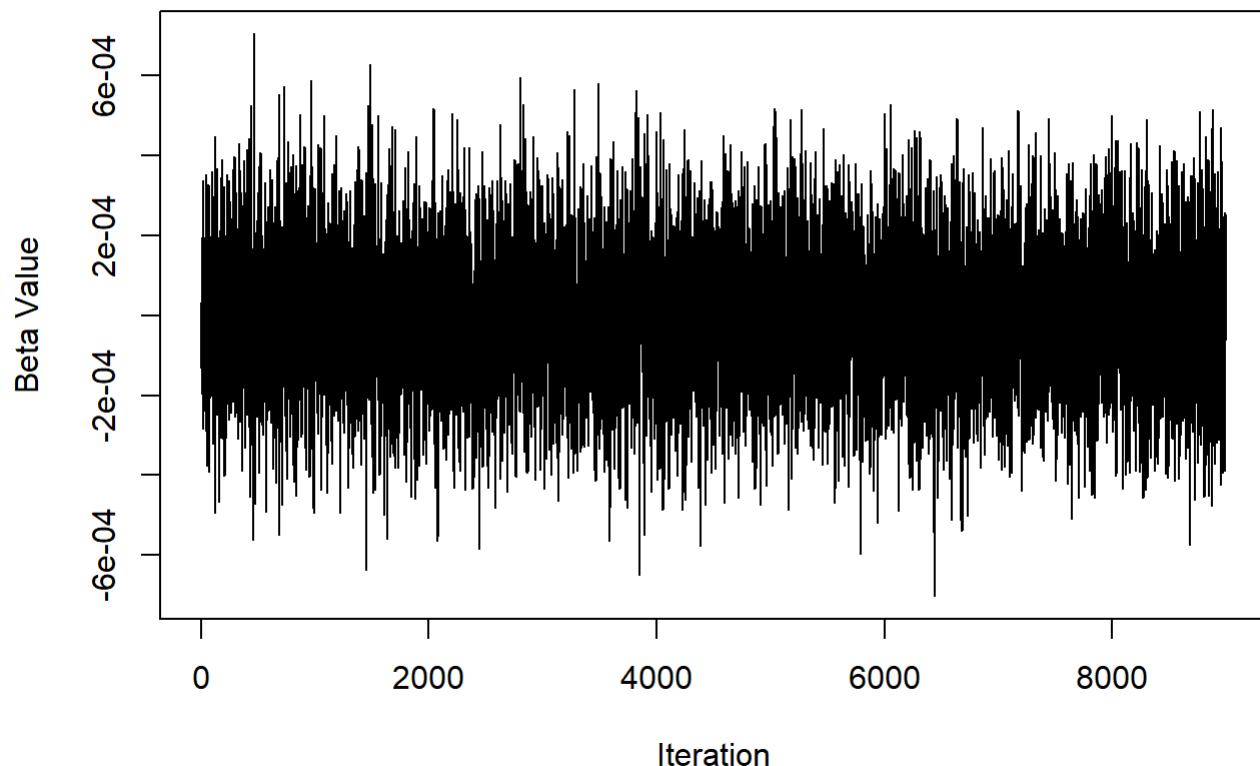
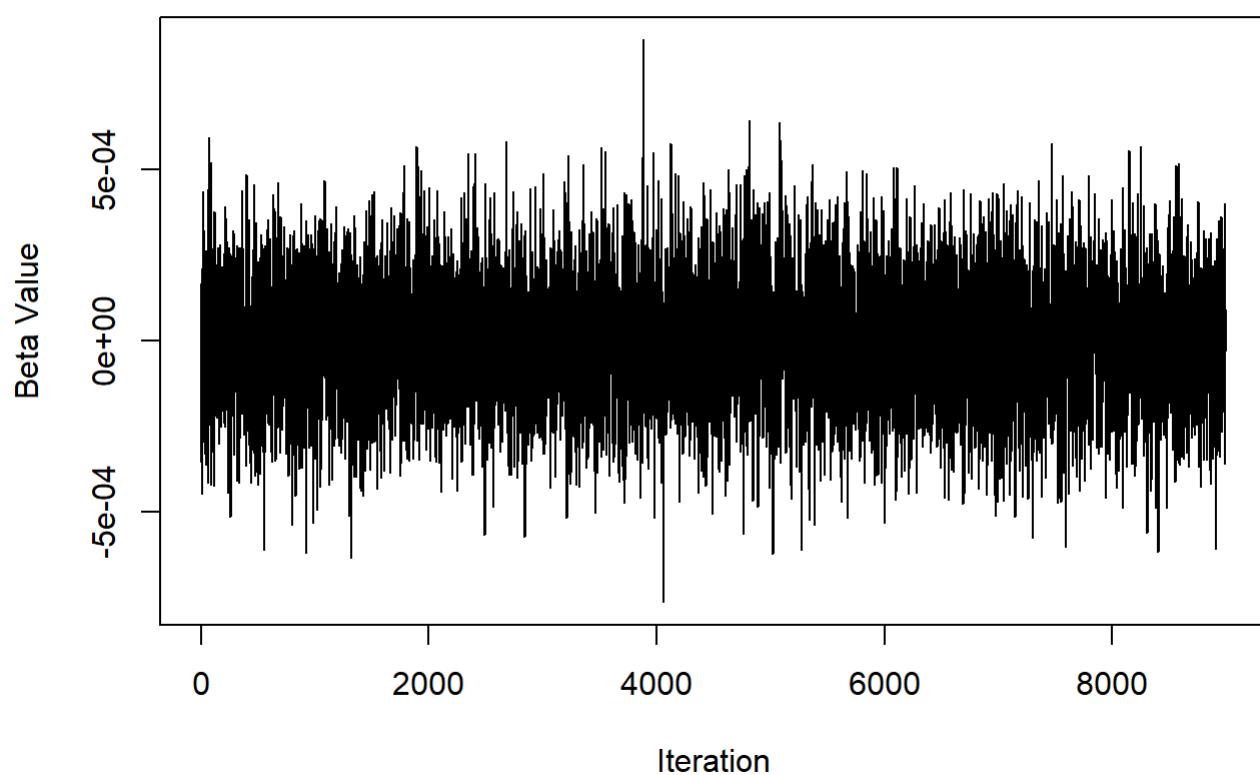
**Trace plot for Alpha (Observation 27 )****Trace plot for Alpha (Observation 28 )**

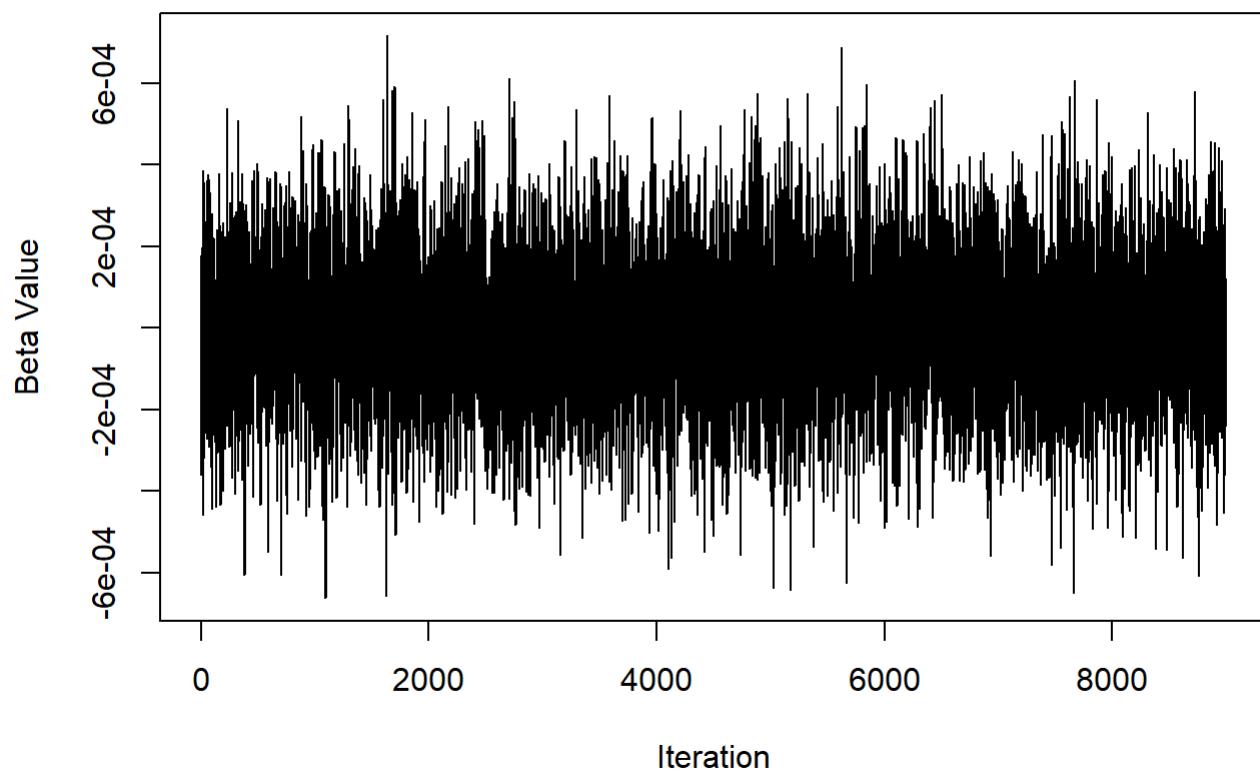
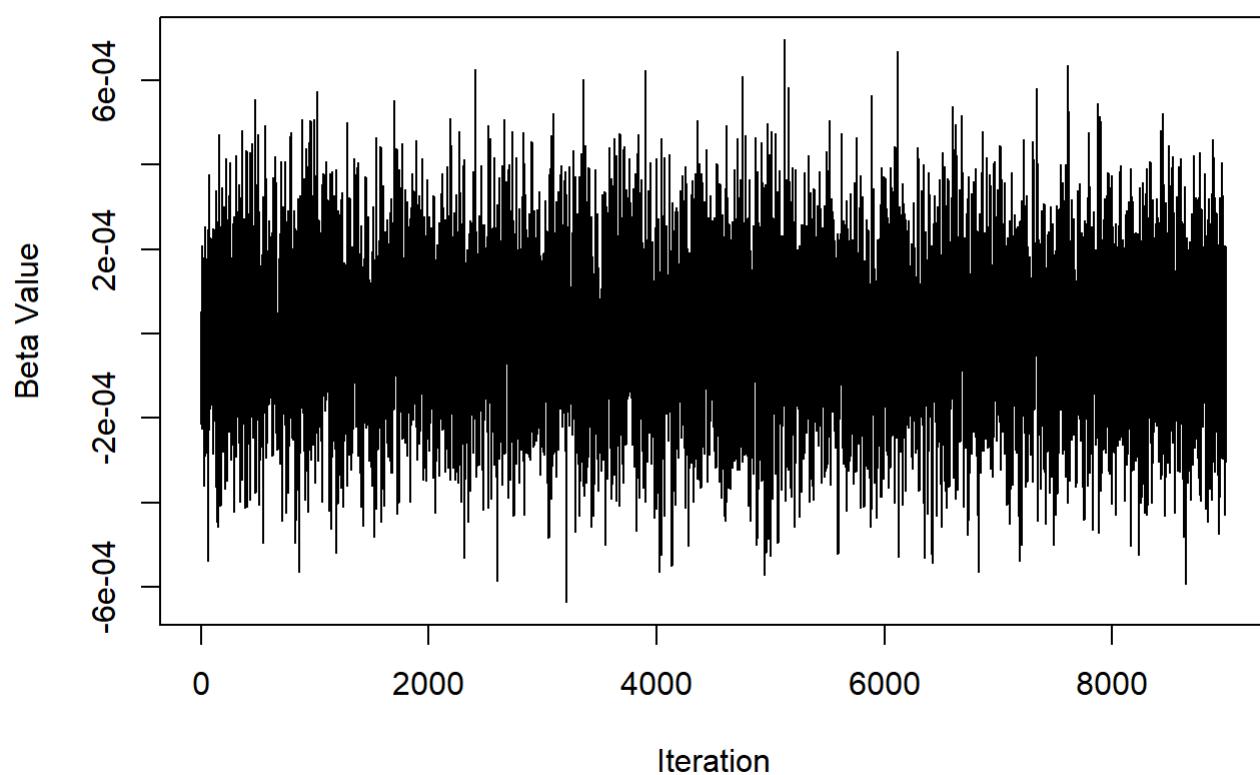
**Trace plot for Alpha (Observation 29 )****Trace plot for Alpha (Observation 30 )**

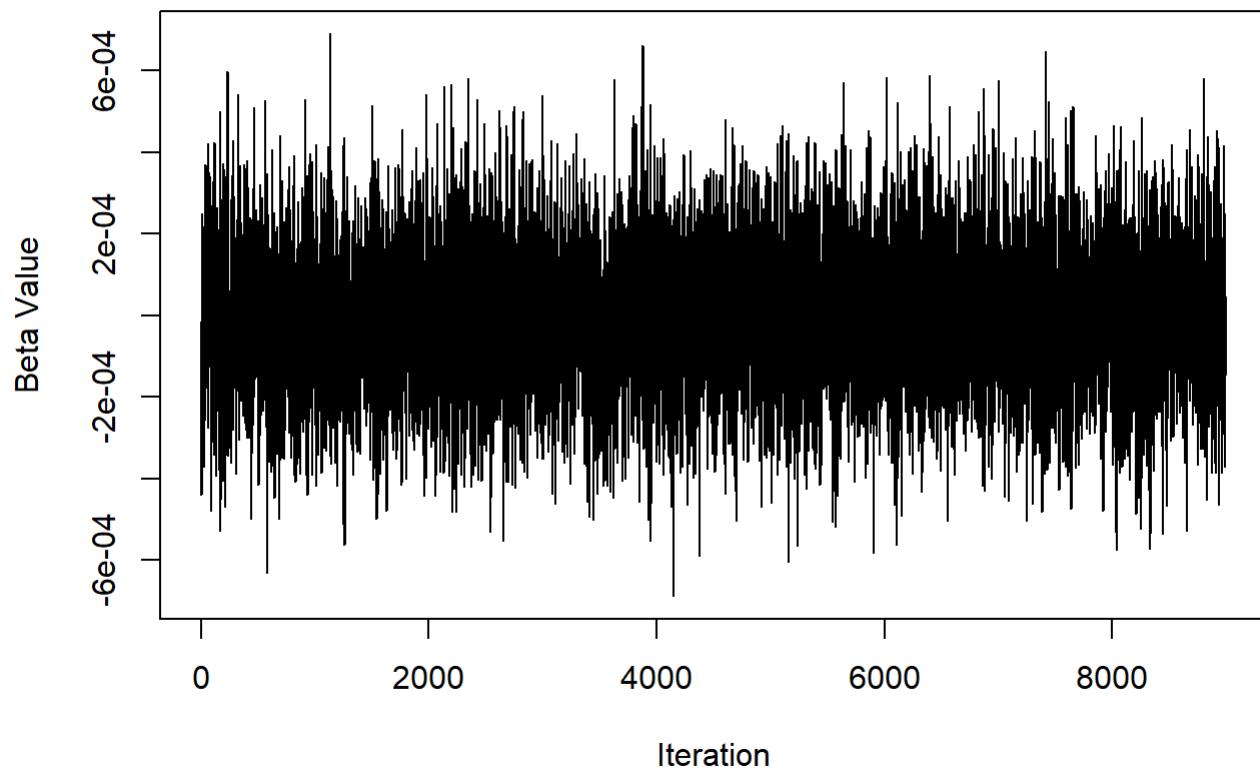
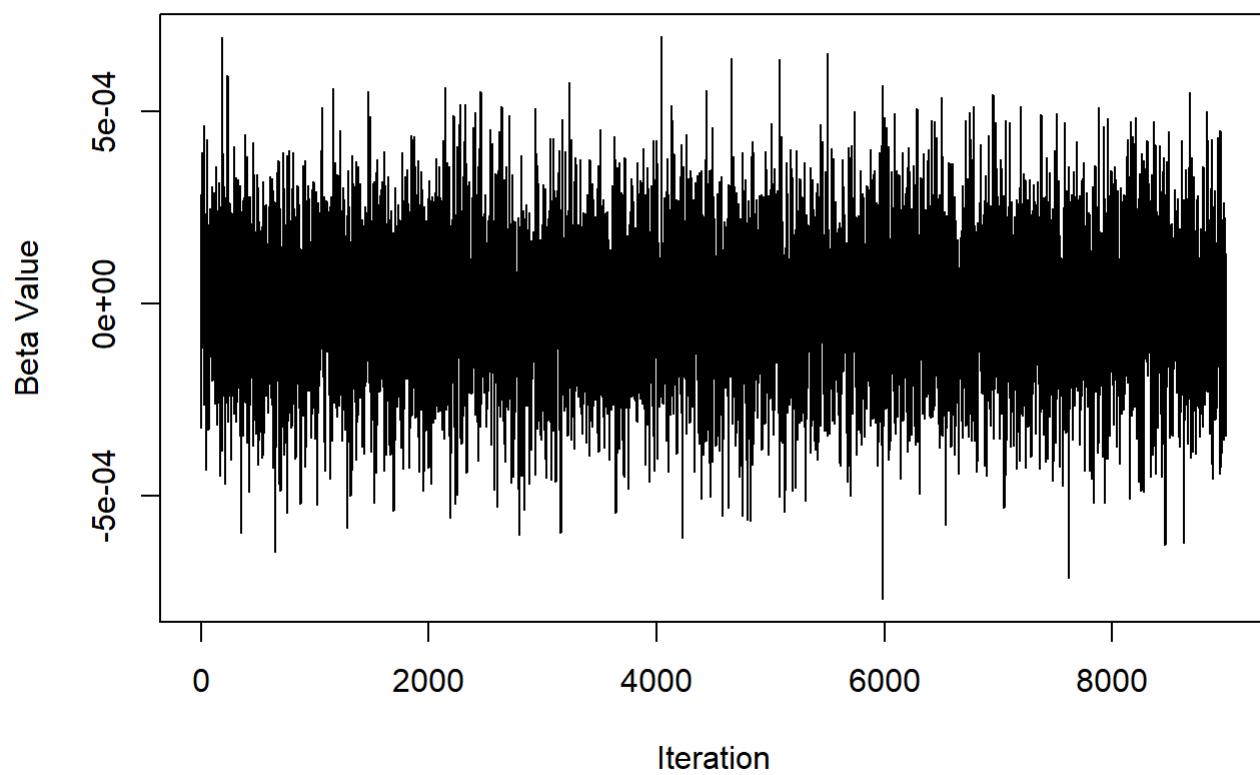
```
# Trace plots for beta
for (i in 1:30) {
  # Plot trace for beta (i-th observation)
  plot(1:9000, beta_post[i, ], type = "l",
    main = paste("Trace plot for Beta (Observation", i, ")"),
    xlab = "Iteration", ylab = "Beta Value", col = "black")
}
```

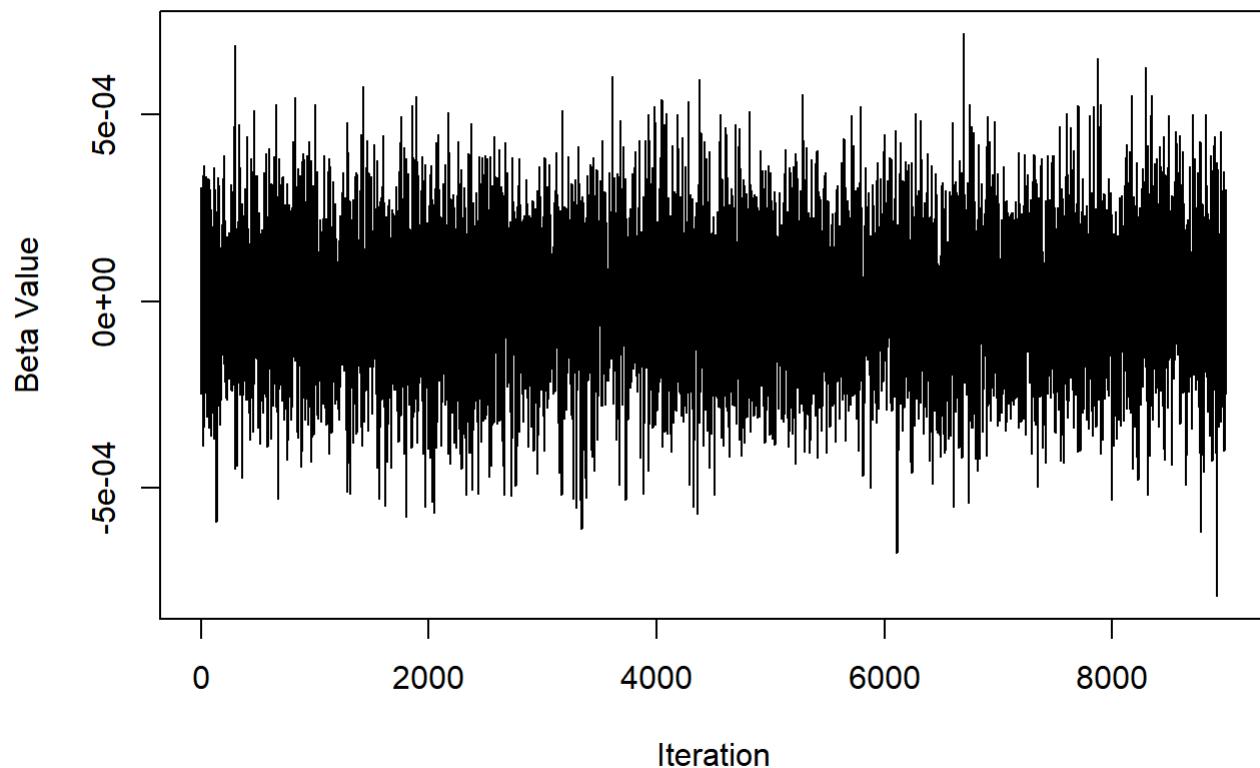
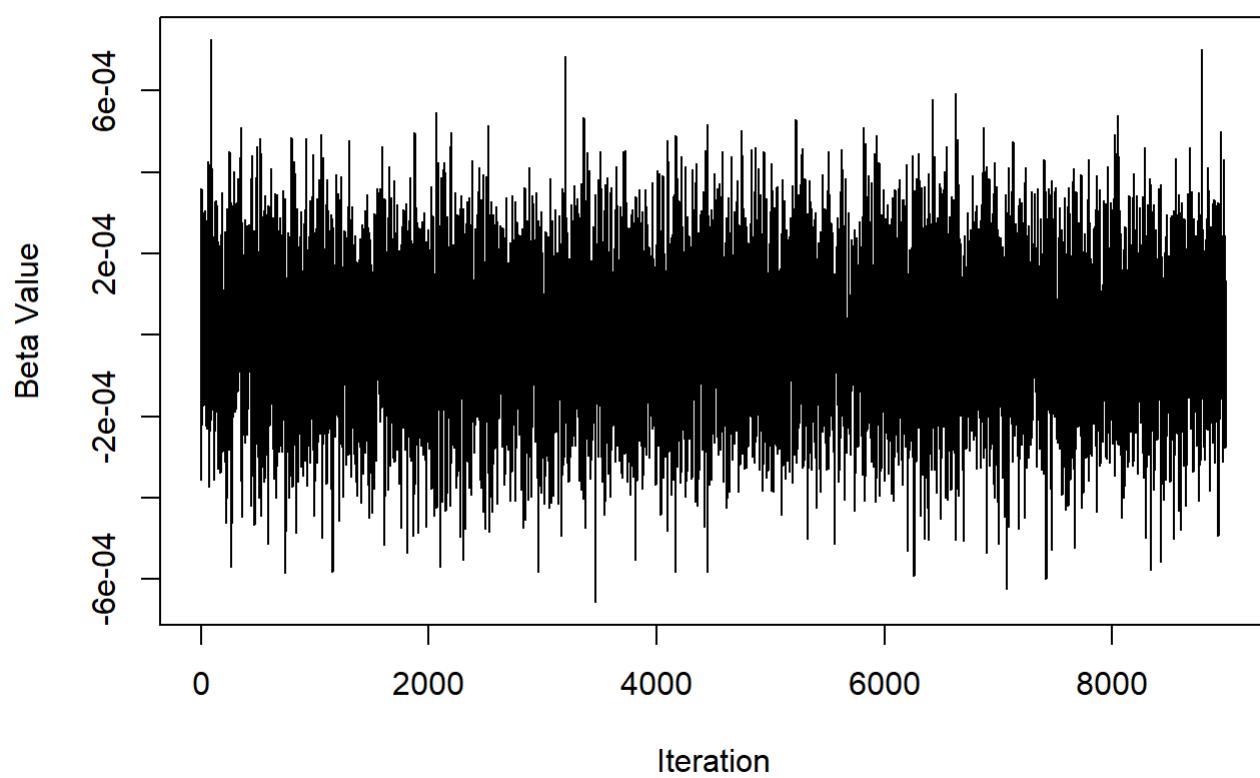
**Trace plot for Beta (Observation 1 )****Trace plot for Beta (Observation 2 )**

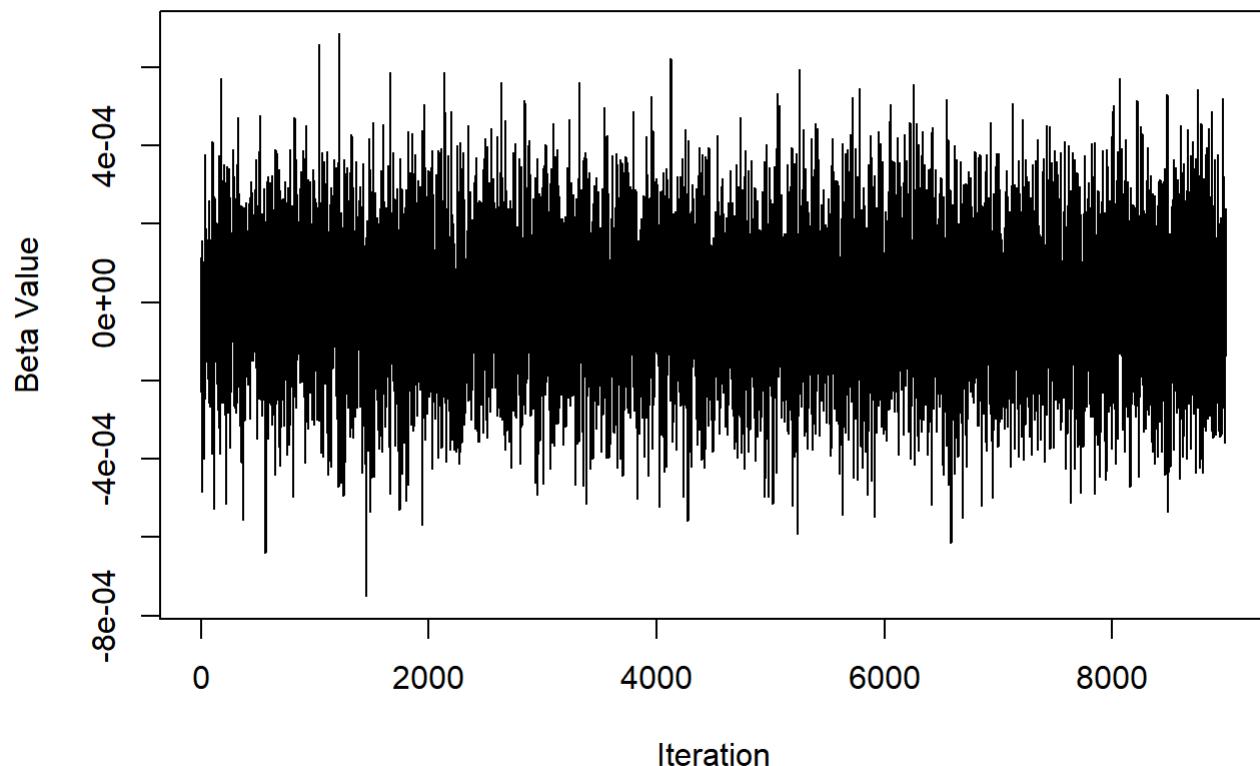
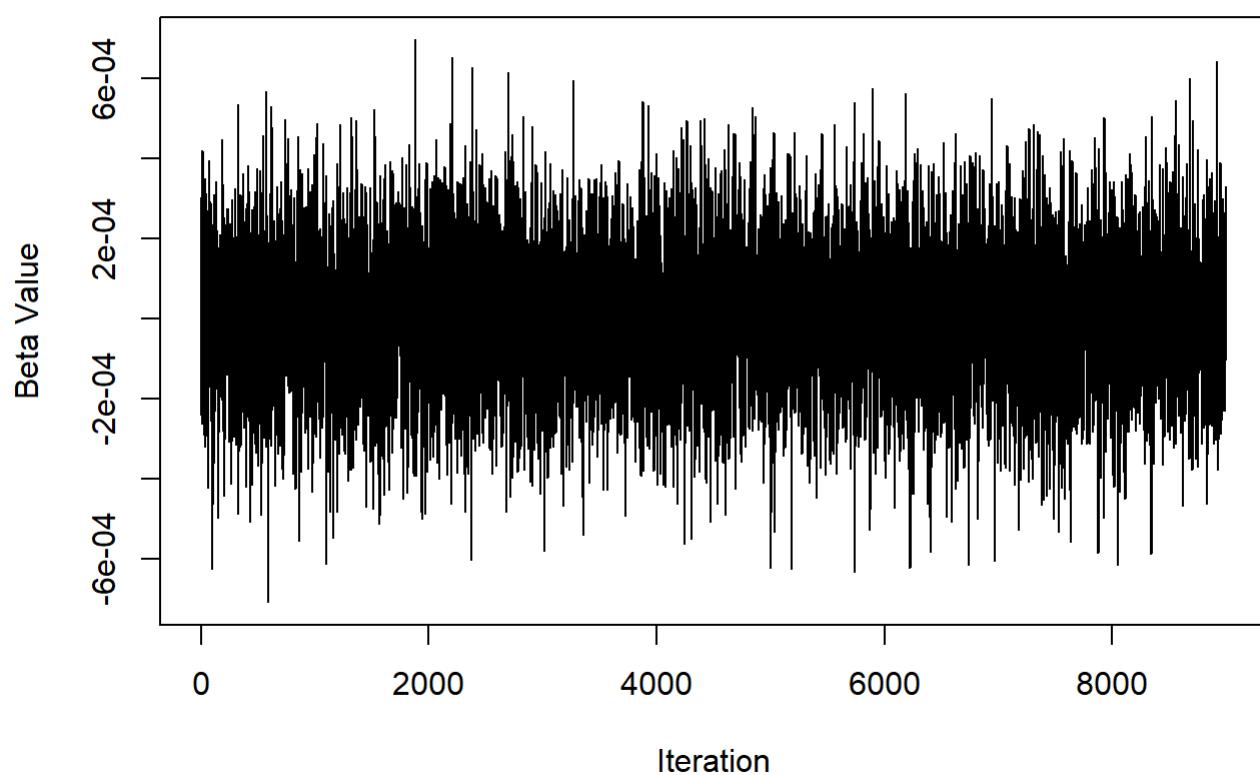
**Trace plot for Beta (Observation 3 )****Trace plot for Beta (Observation 4 )**

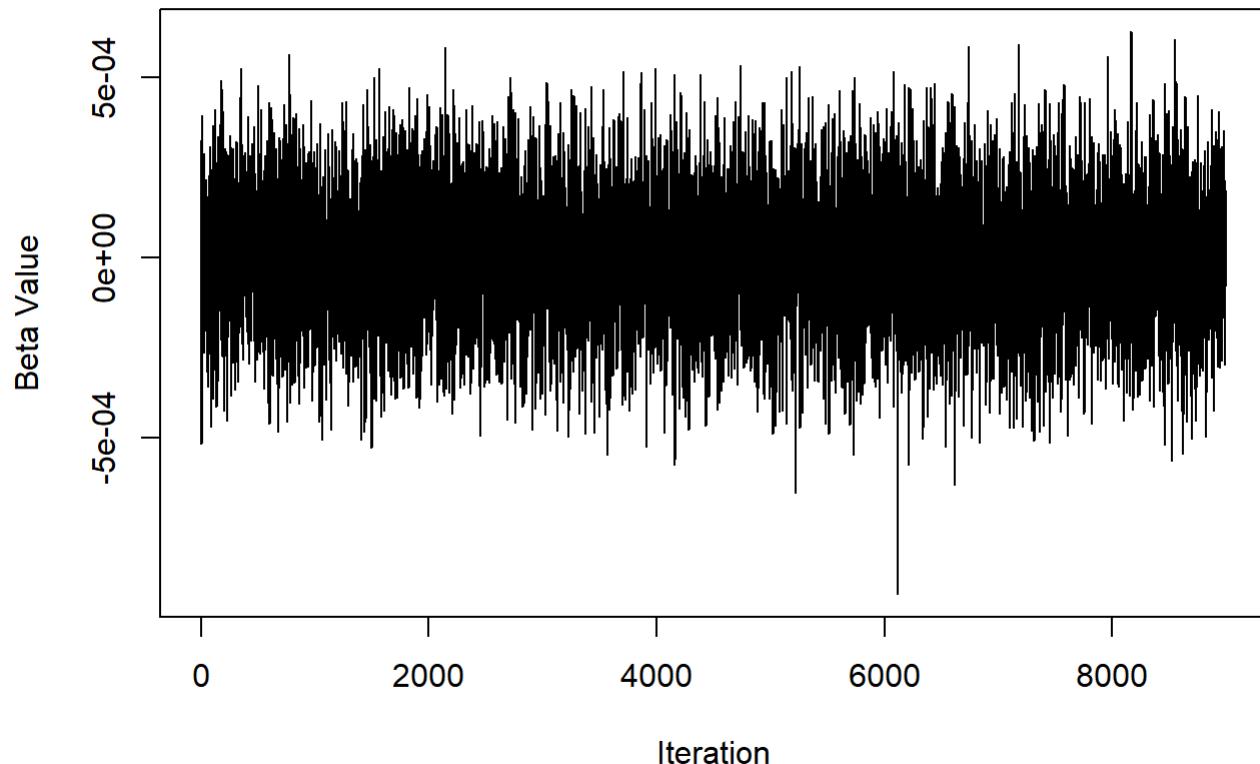
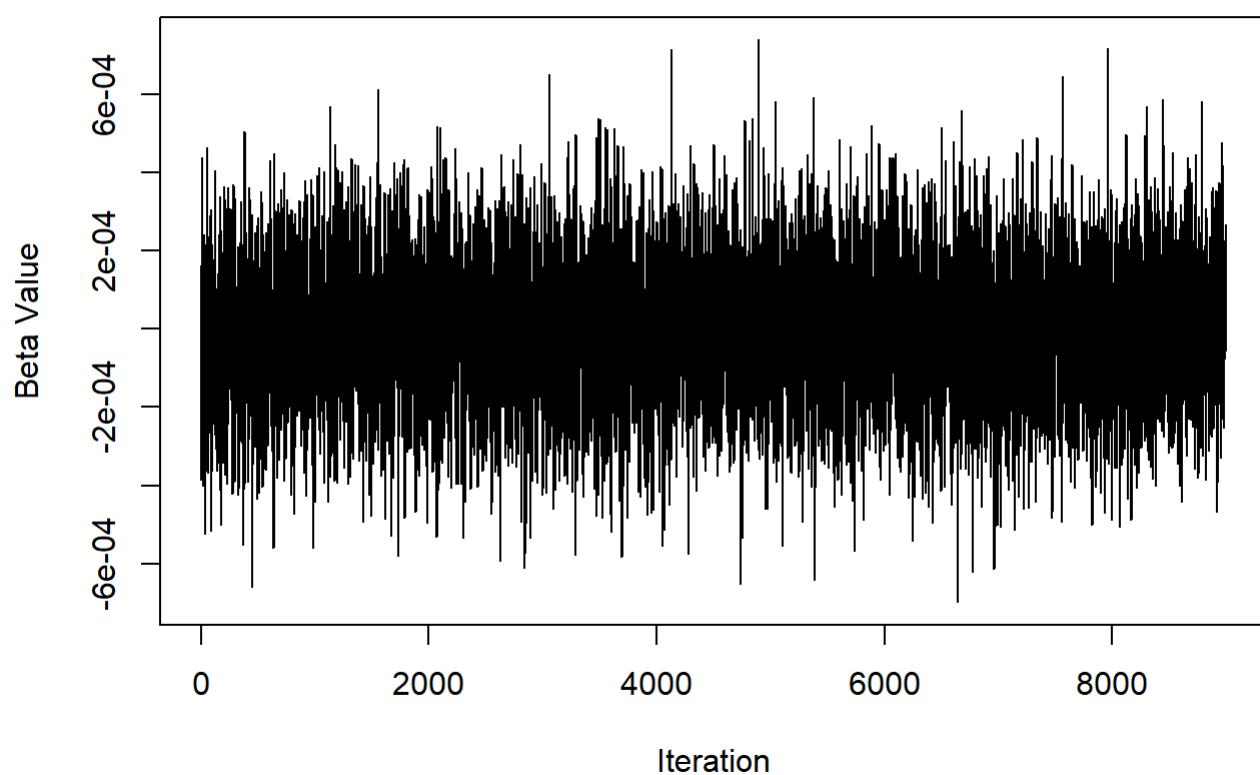
**Trace plot for Beta (Observation 5 )****Trace plot for Beta (Observation 6 )**

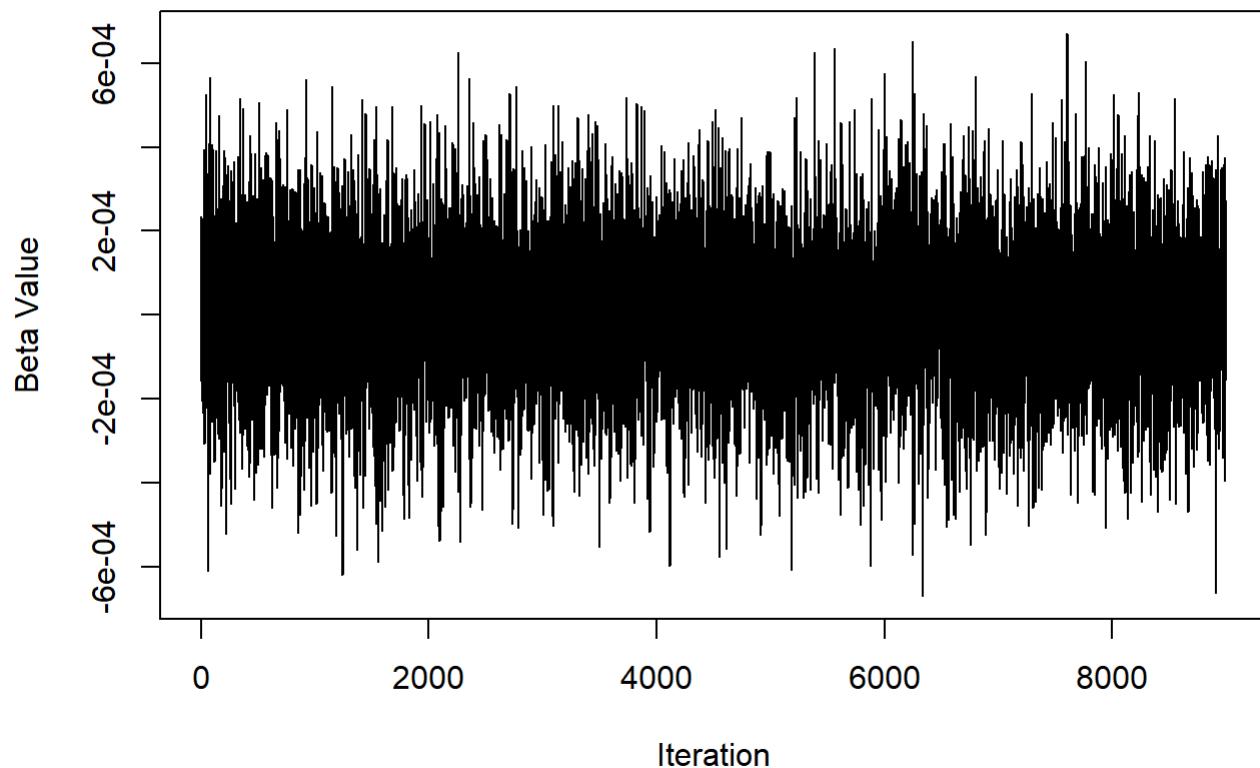
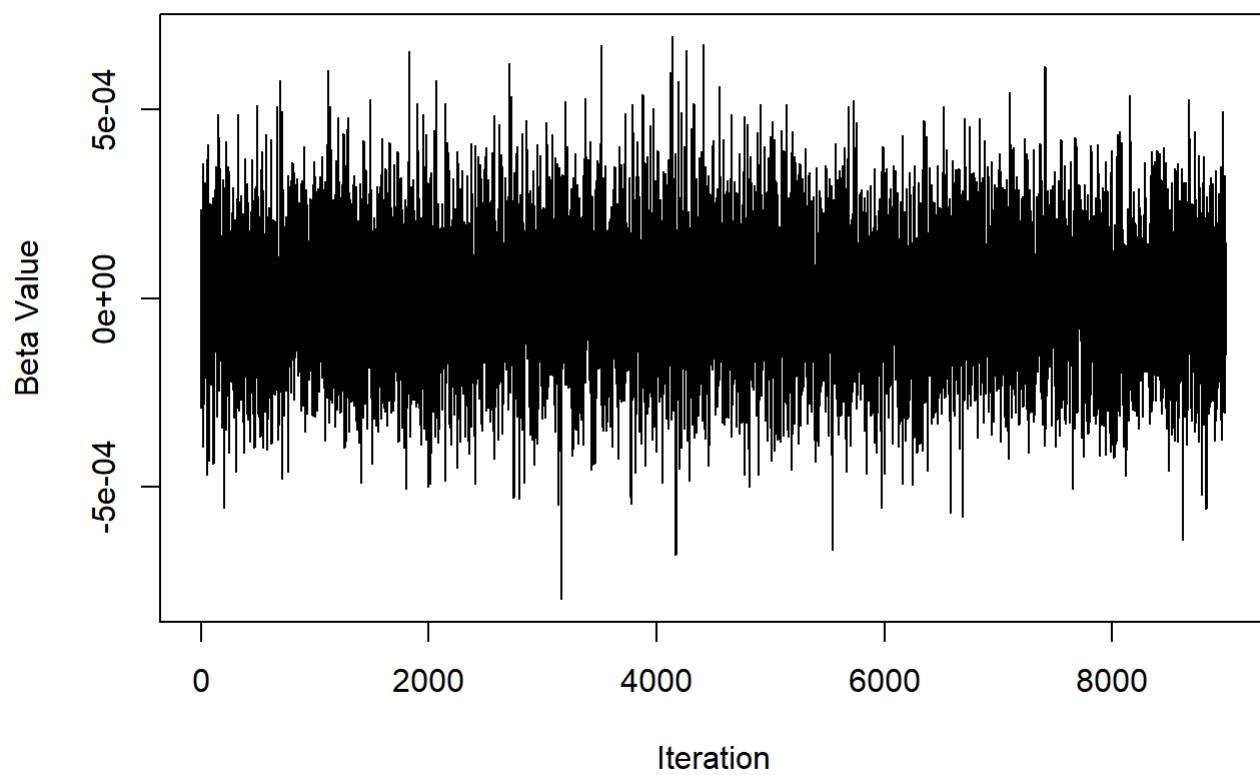
**Trace plot for Beta (Observation 7 )****Trace plot for Beta (Observation 8 )**

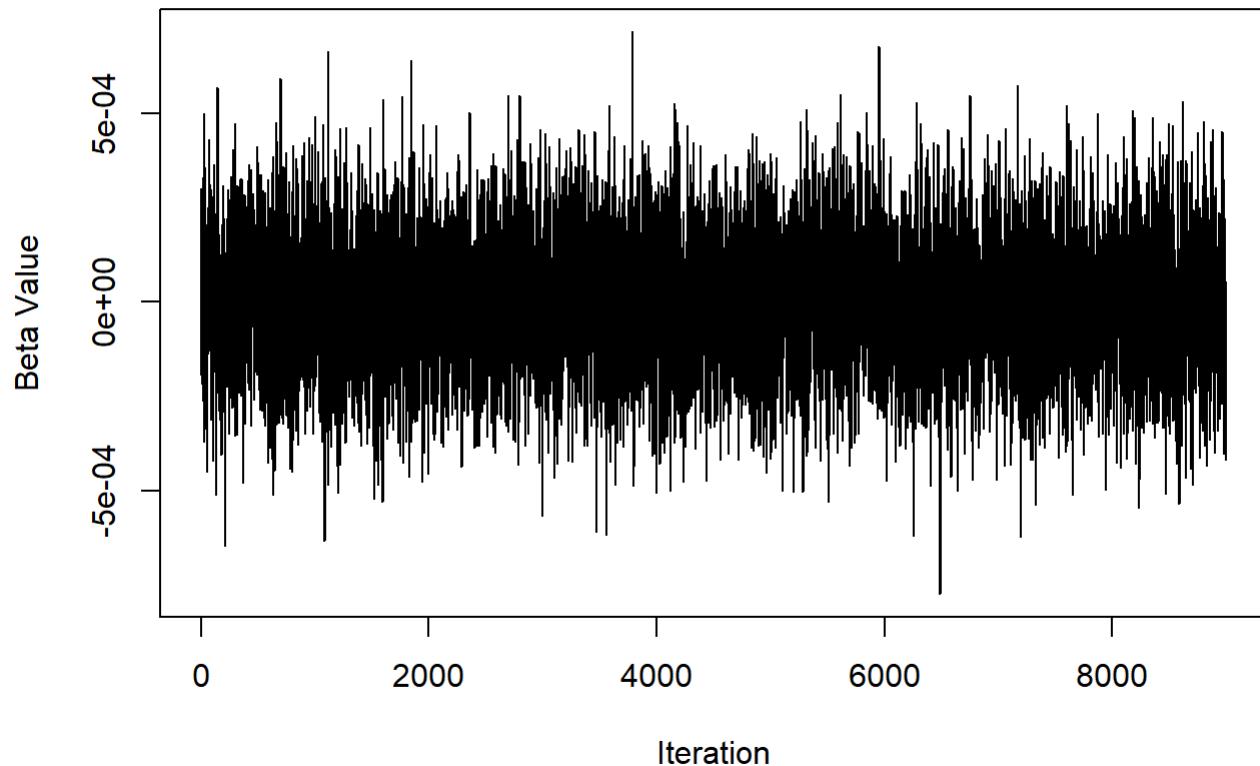
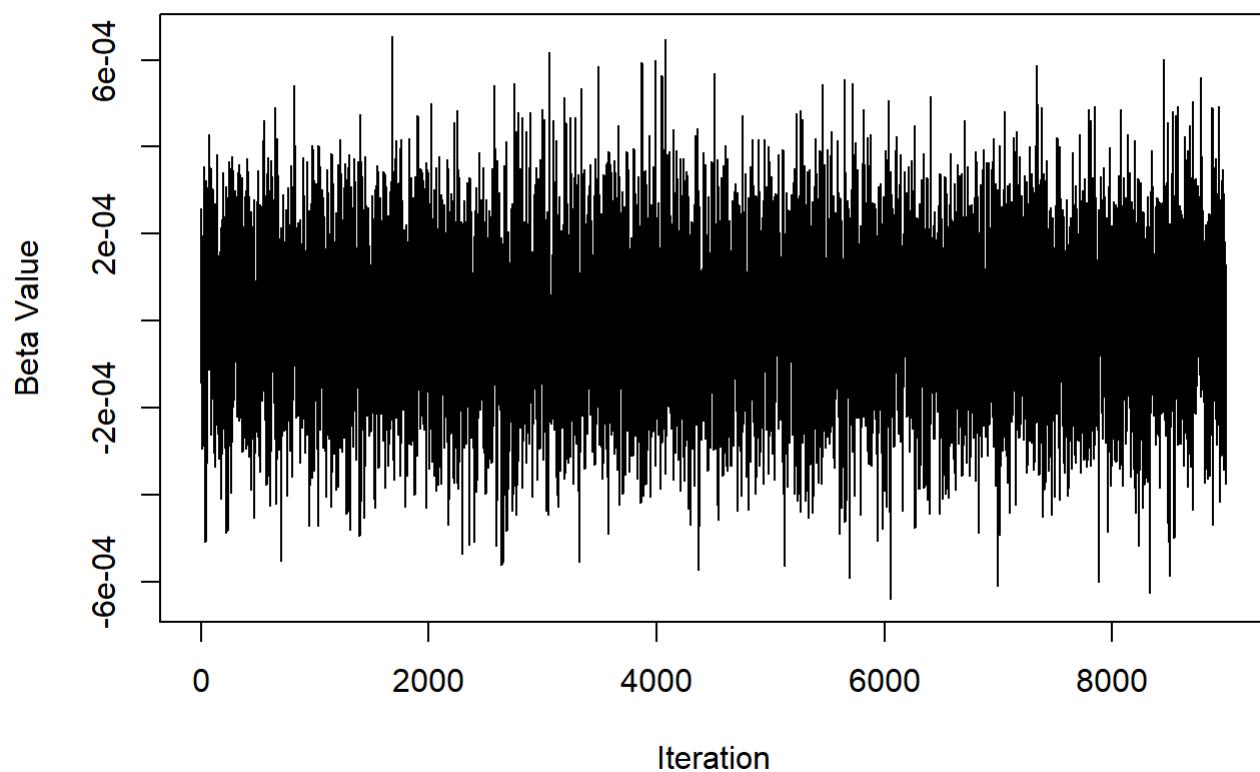
**Trace plot for Beta (Observation 9 )****Trace plot for Beta (Observation 10 )**

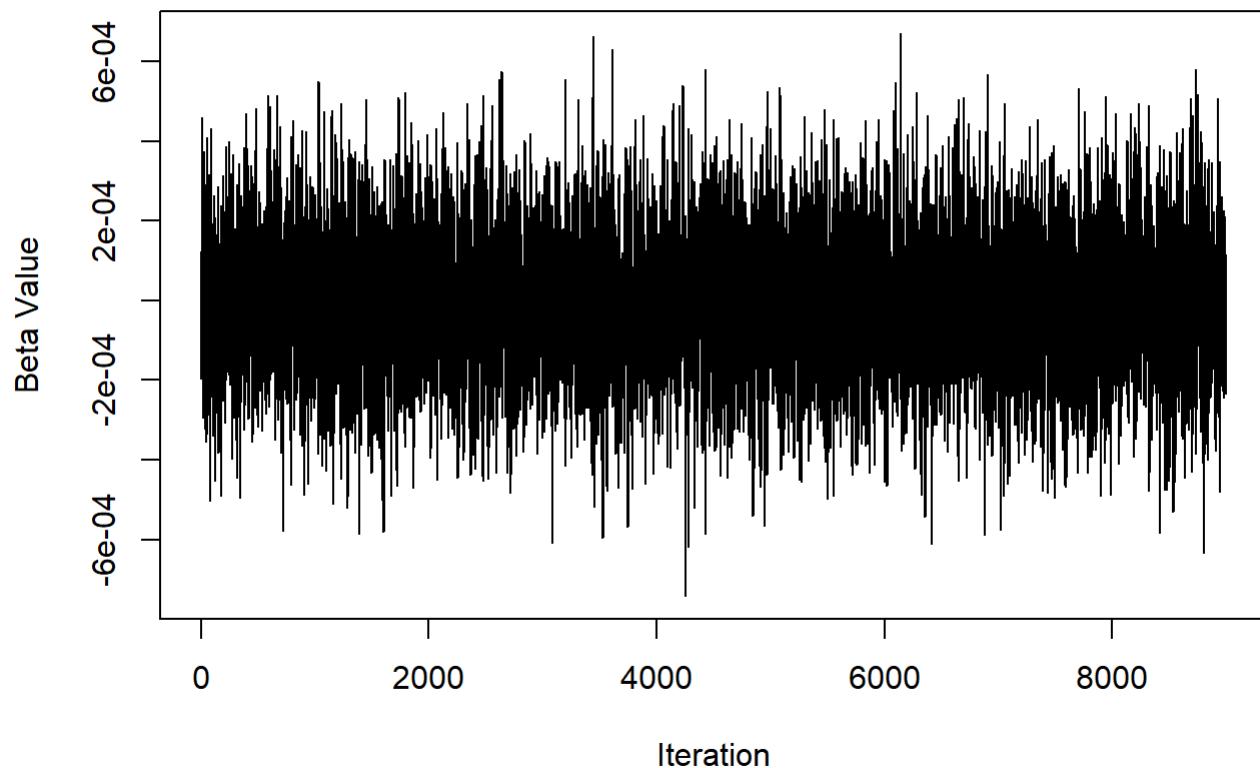
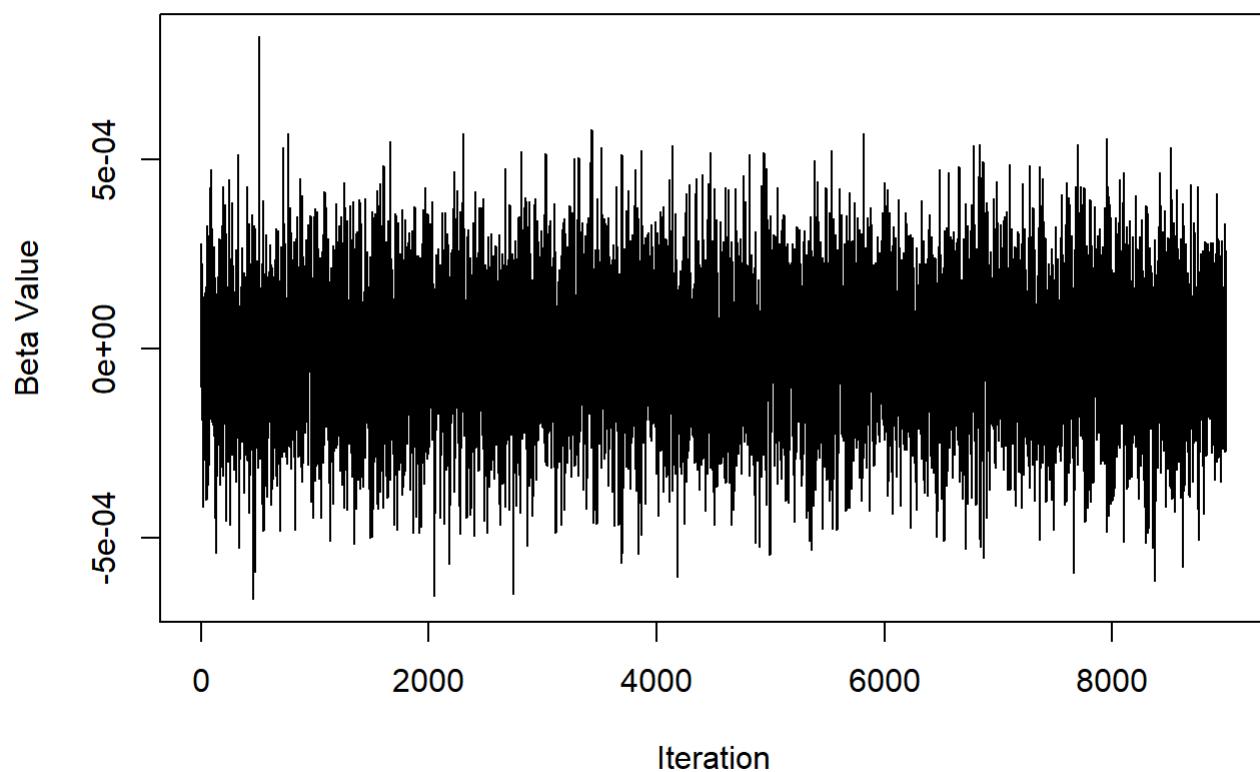
**Trace plot for Beta (Observation 11 )****Trace plot for Beta (Observation 12 )**

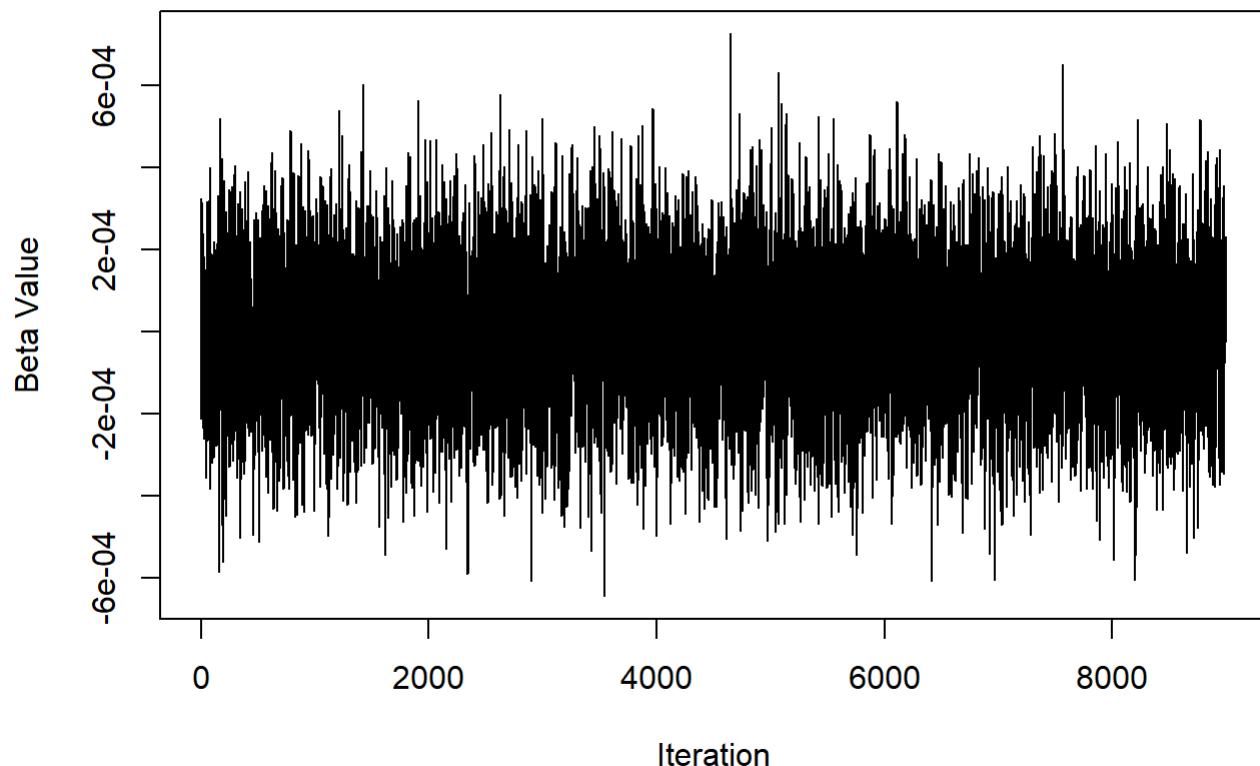
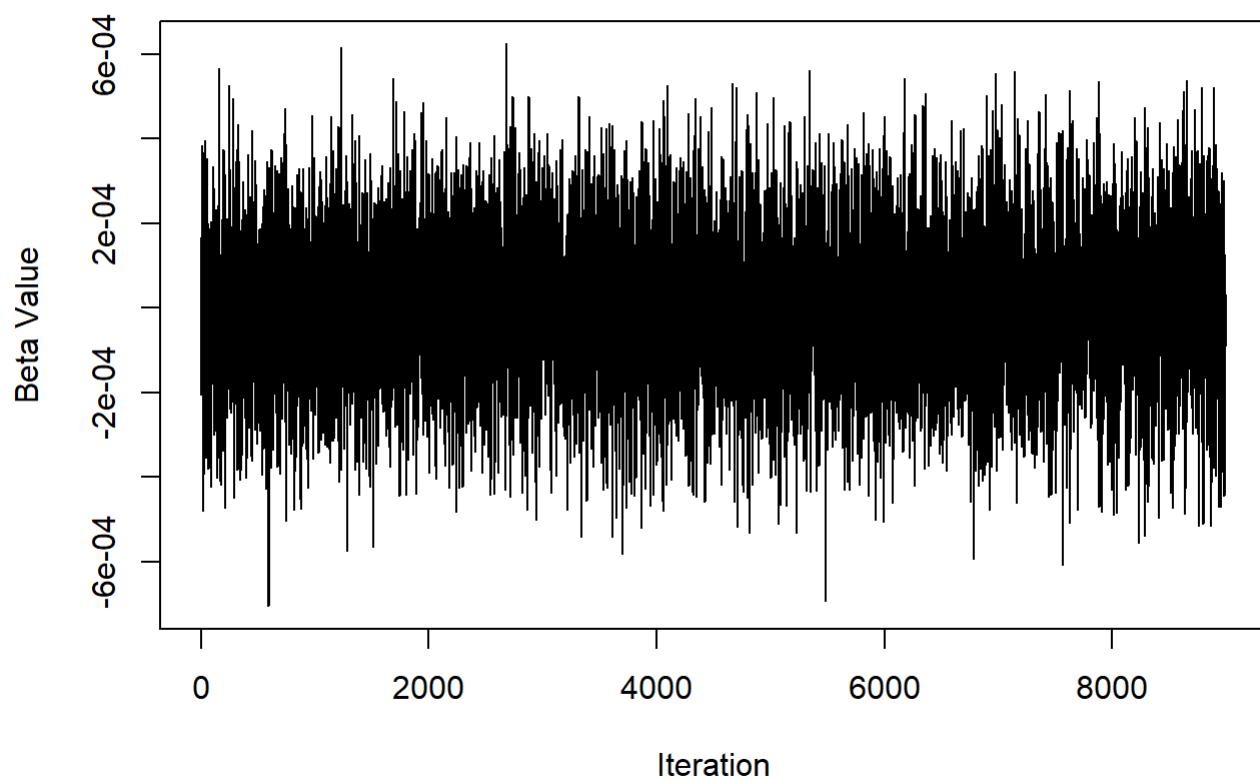
**Trace plot for Beta (Observation 13 )****Trace plot for Beta (Observation 14 )**

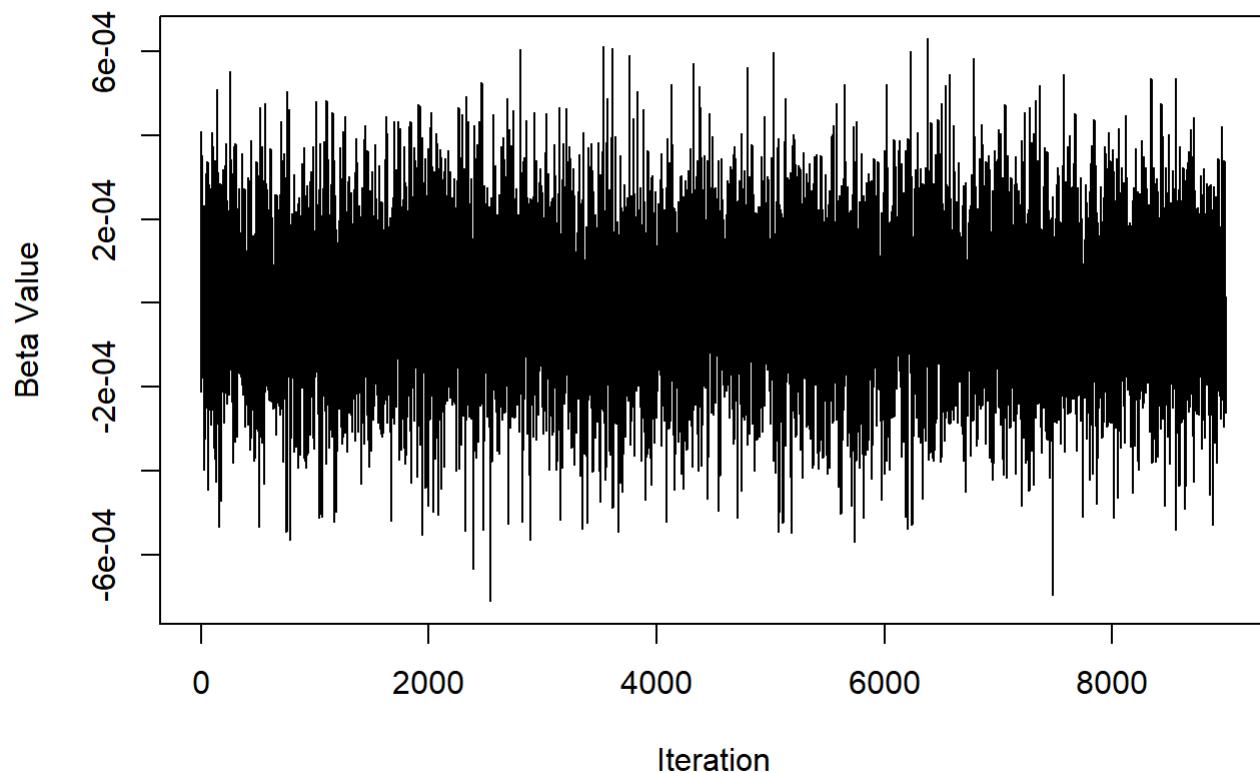
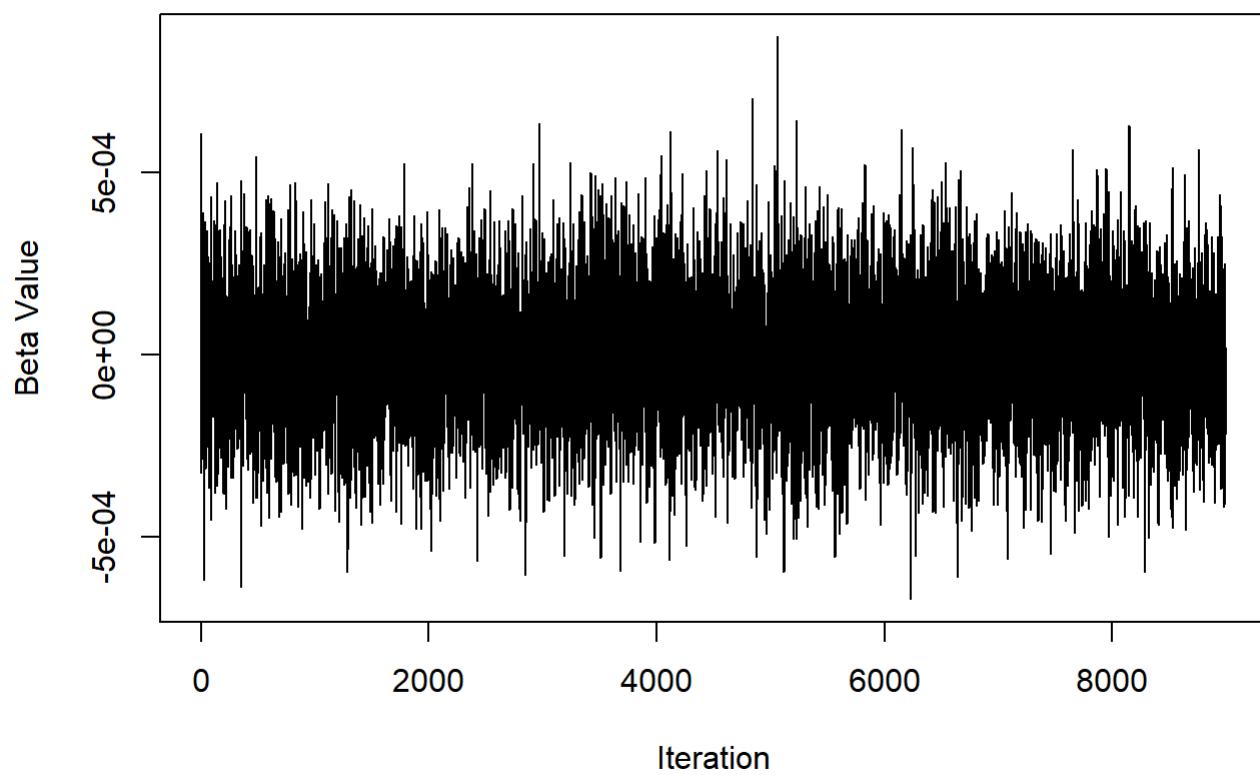
**Trace plot for Beta (Observation 15 )****Trace plot for Beta (Observation 16 )**

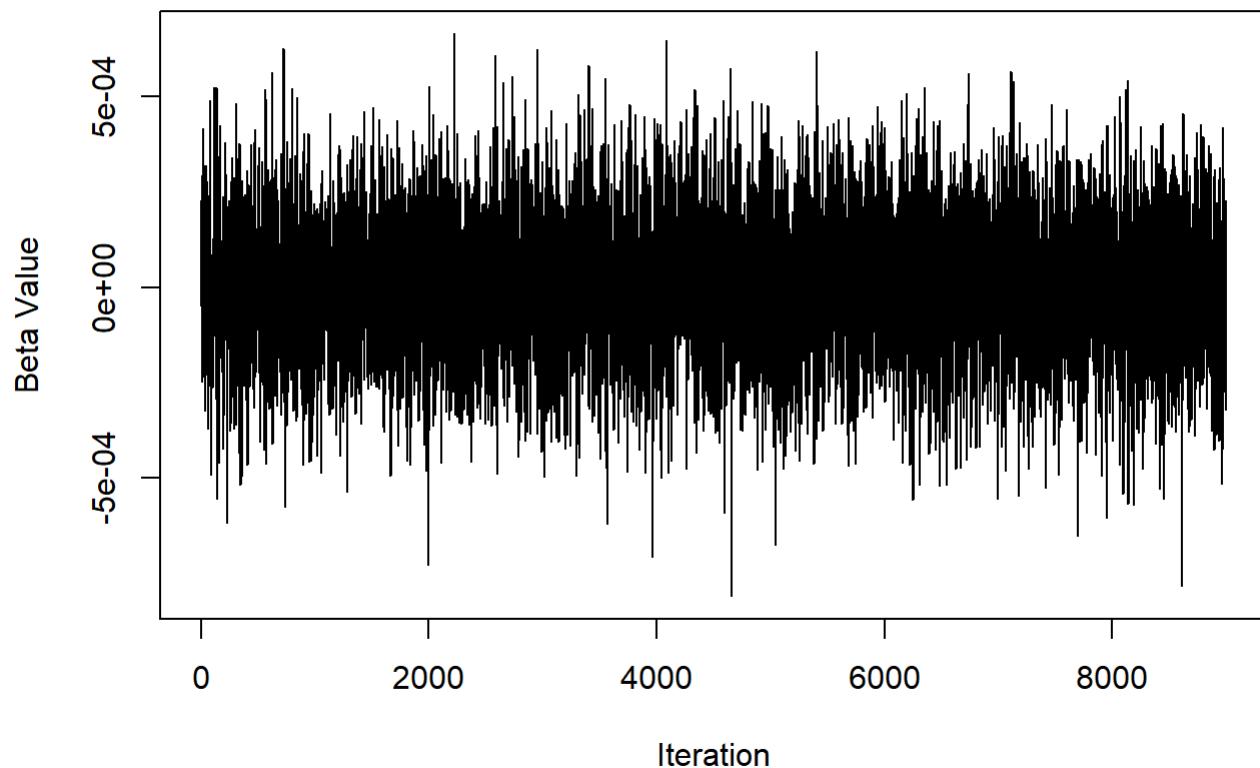
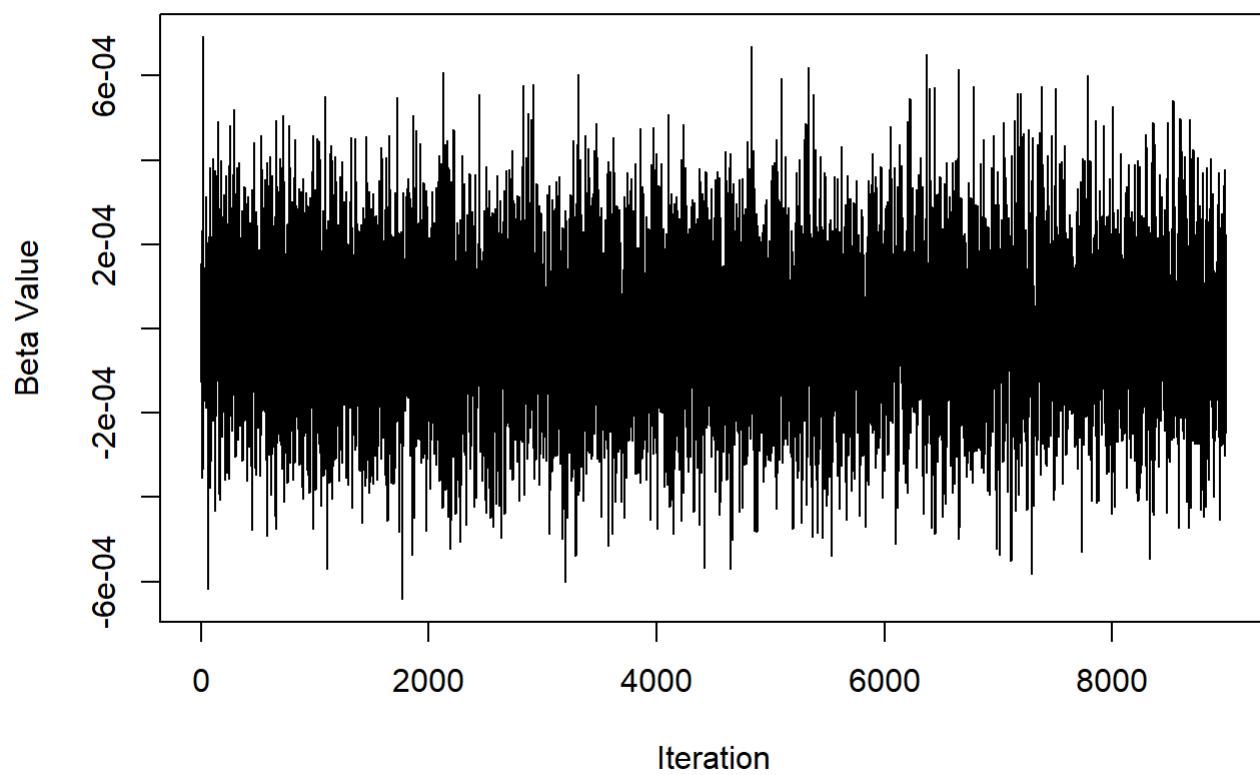
**Trace plot for Beta (Observation 17 )****Trace plot for Beta (Observation 18 )**

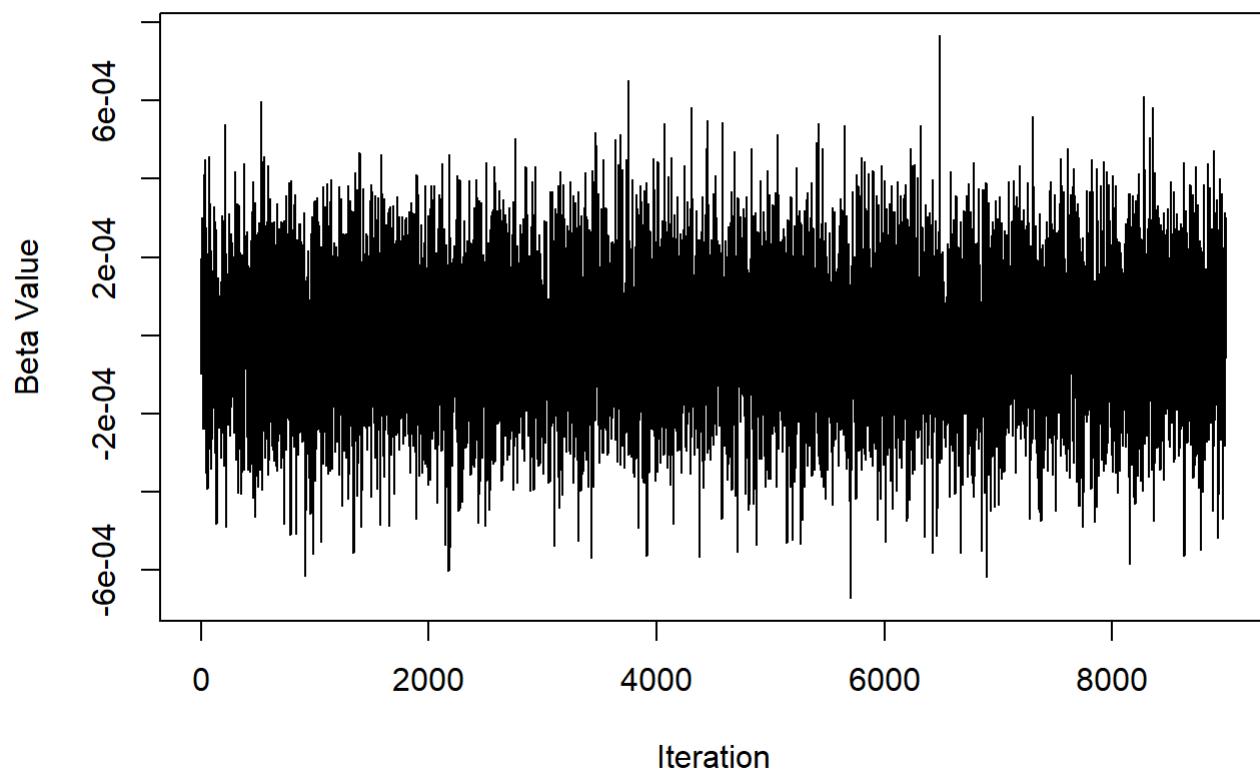
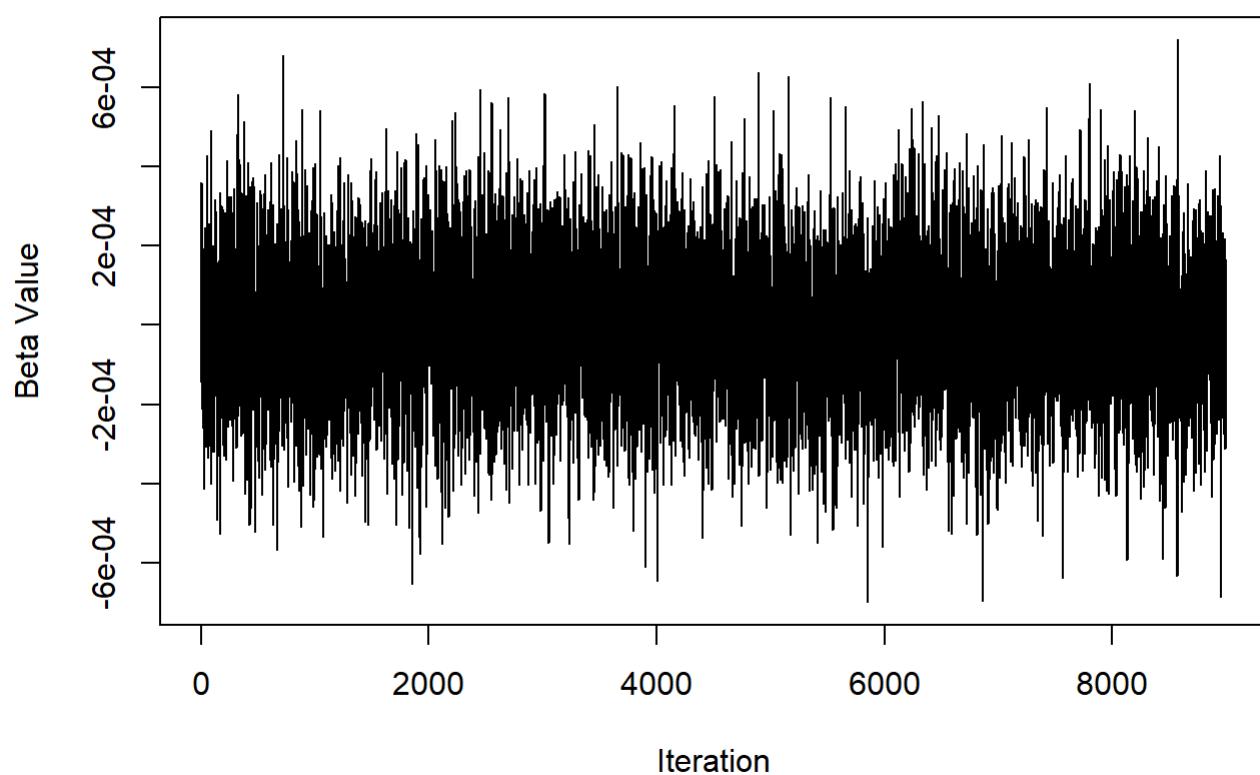
**Trace plot for Beta (Observation 19 )****Trace plot for Beta (Observation 20 )**

**Trace plot for Beta (Observation 21 )****Trace plot for Beta (Observation 22 )**

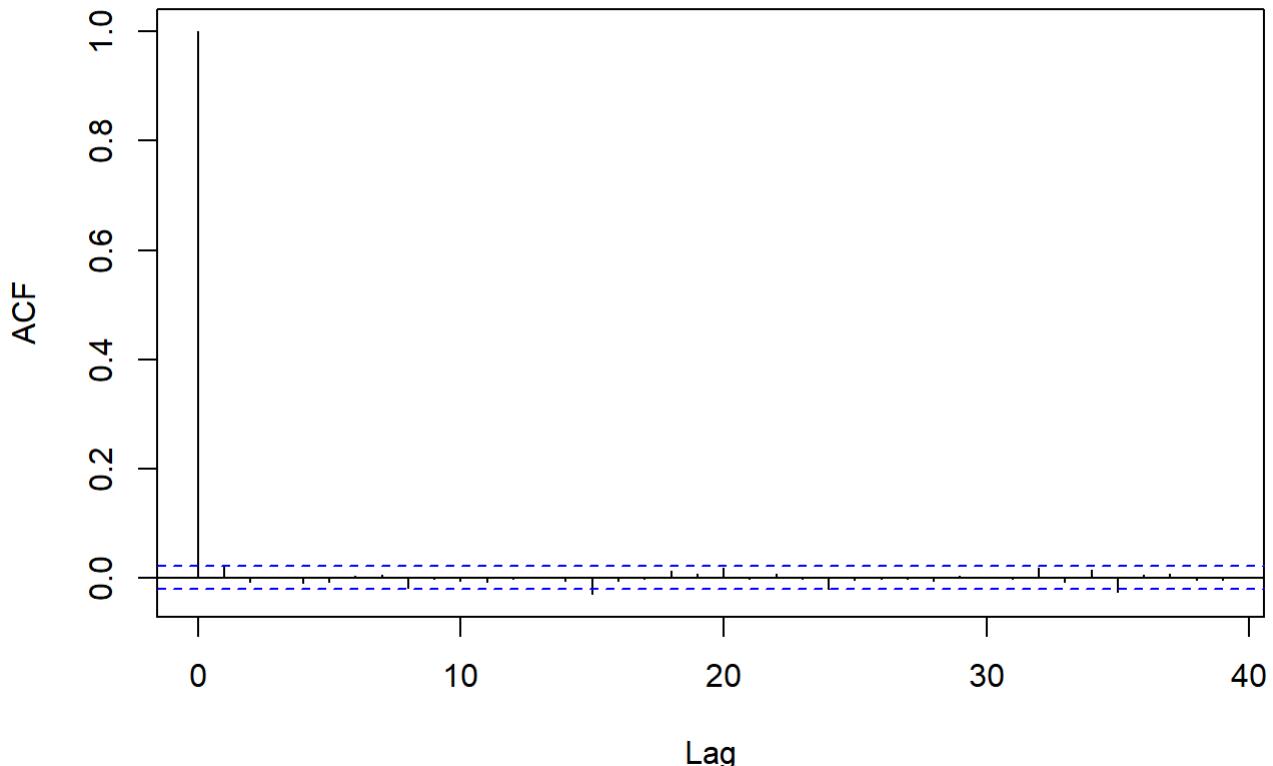
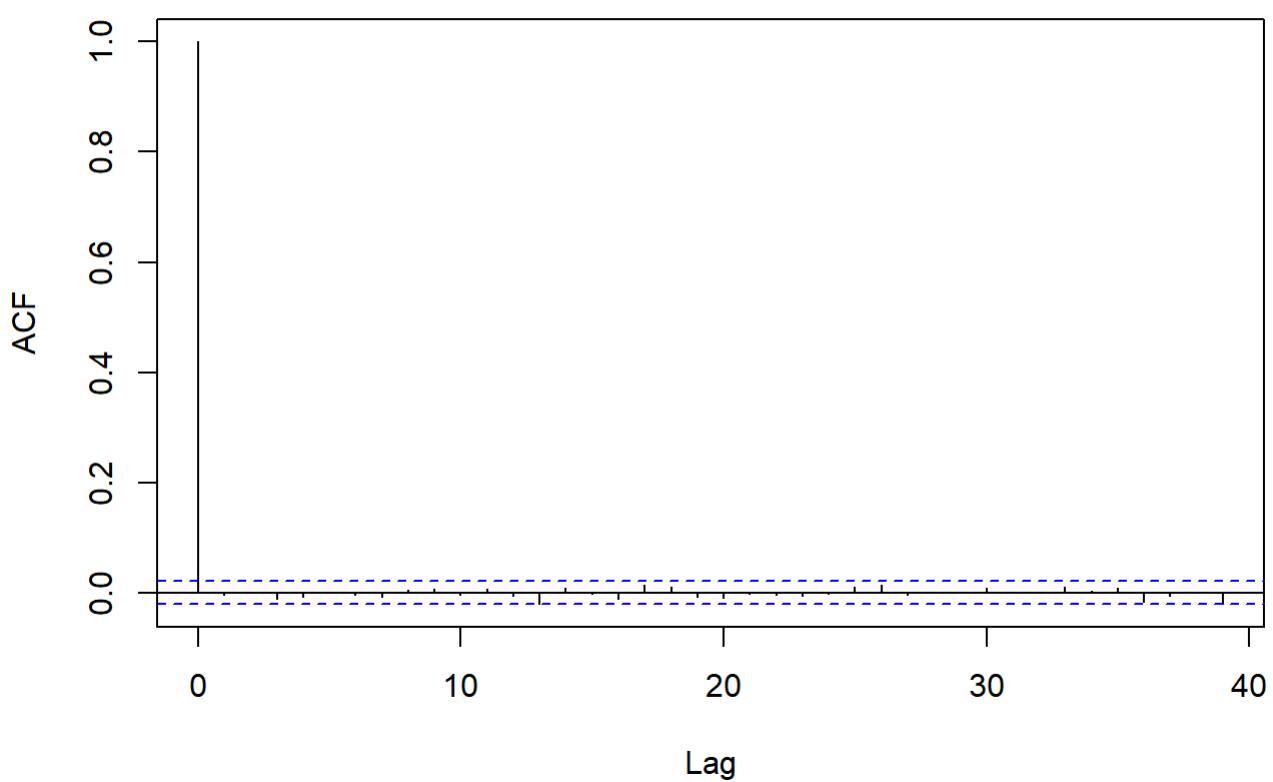
**Trace plot for Beta (Observation 23 )****Trace plot for Beta (Observation 24 )**

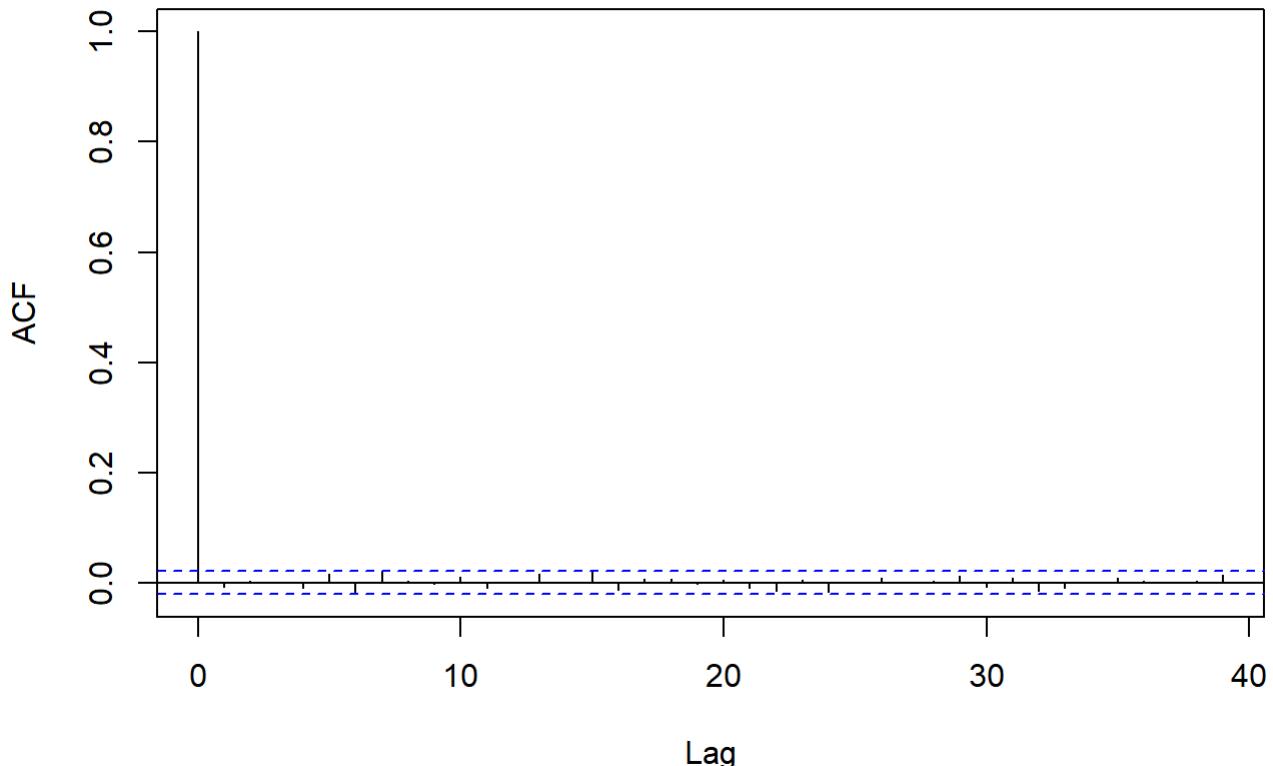
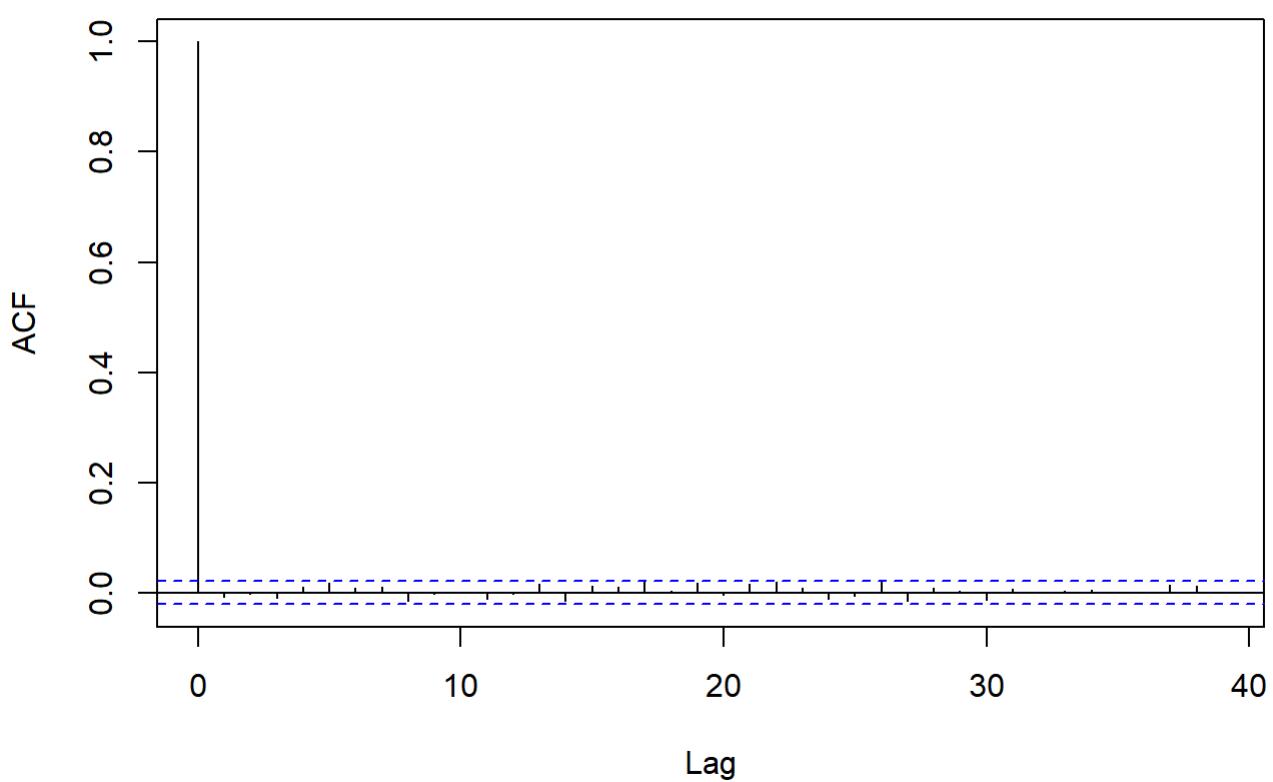
**Trace plot for Beta (Observation 25 )****Trace plot for Beta (Observation 26 )**

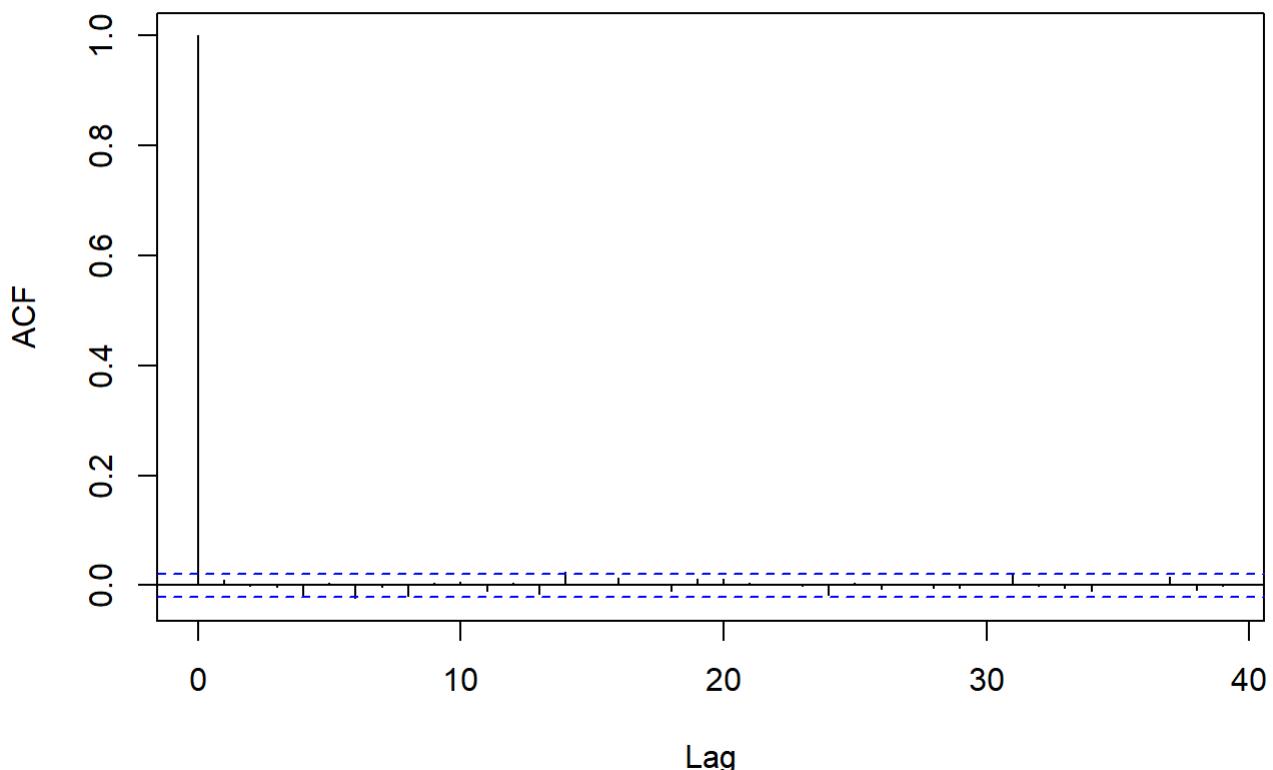
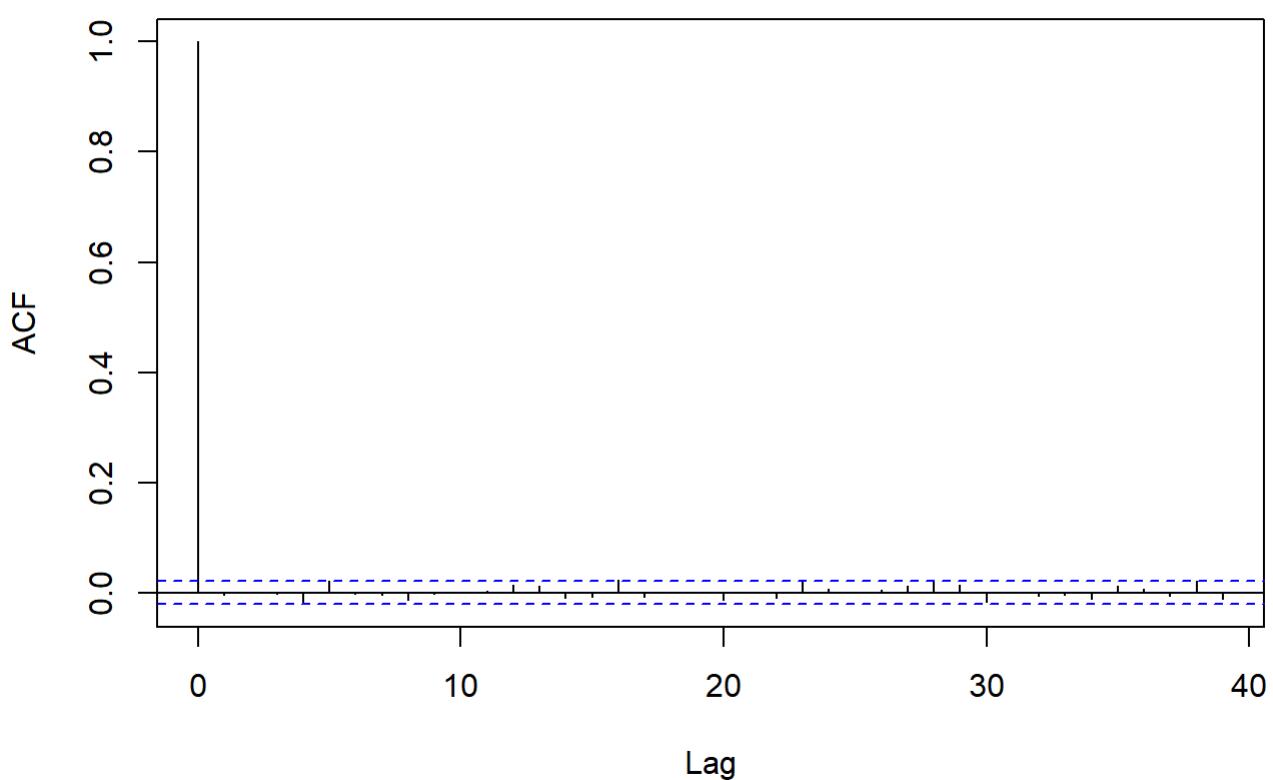
**Trace plot for Beta (Observation 27 )****Trace plot for Beta (Observation 28 )**

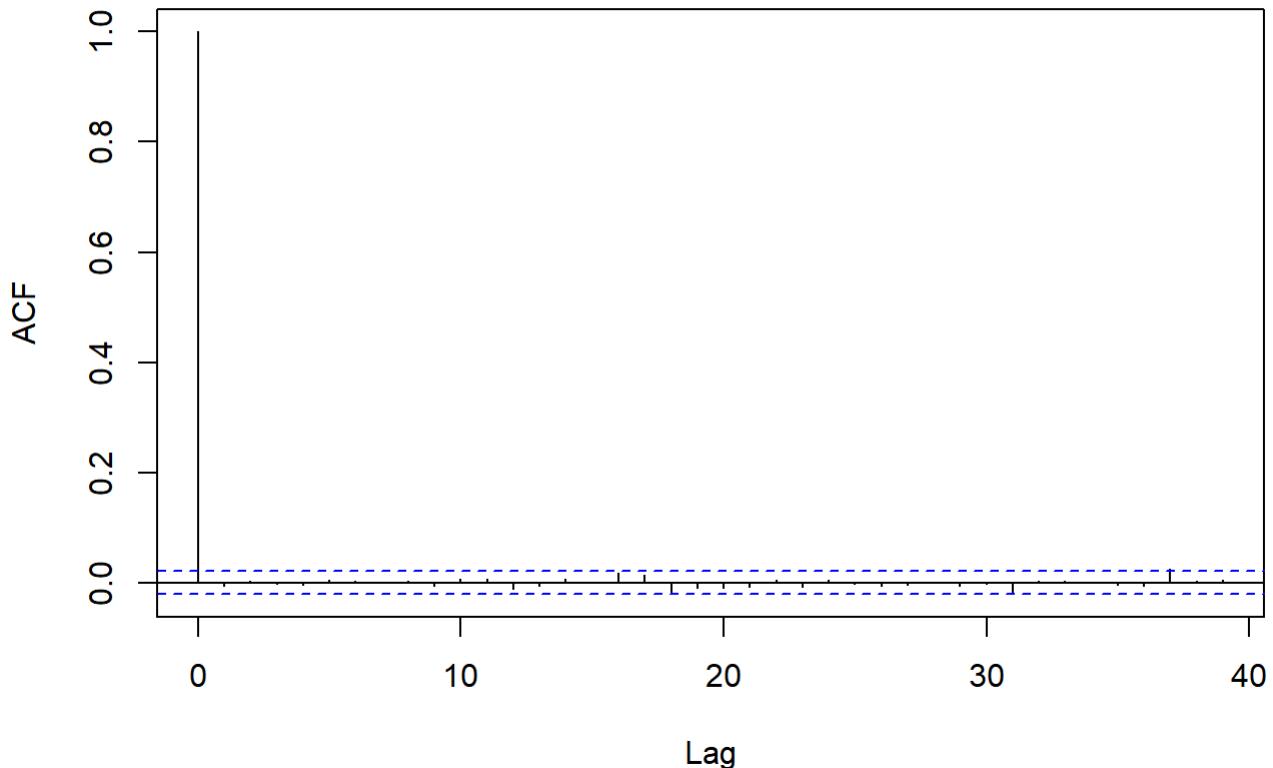
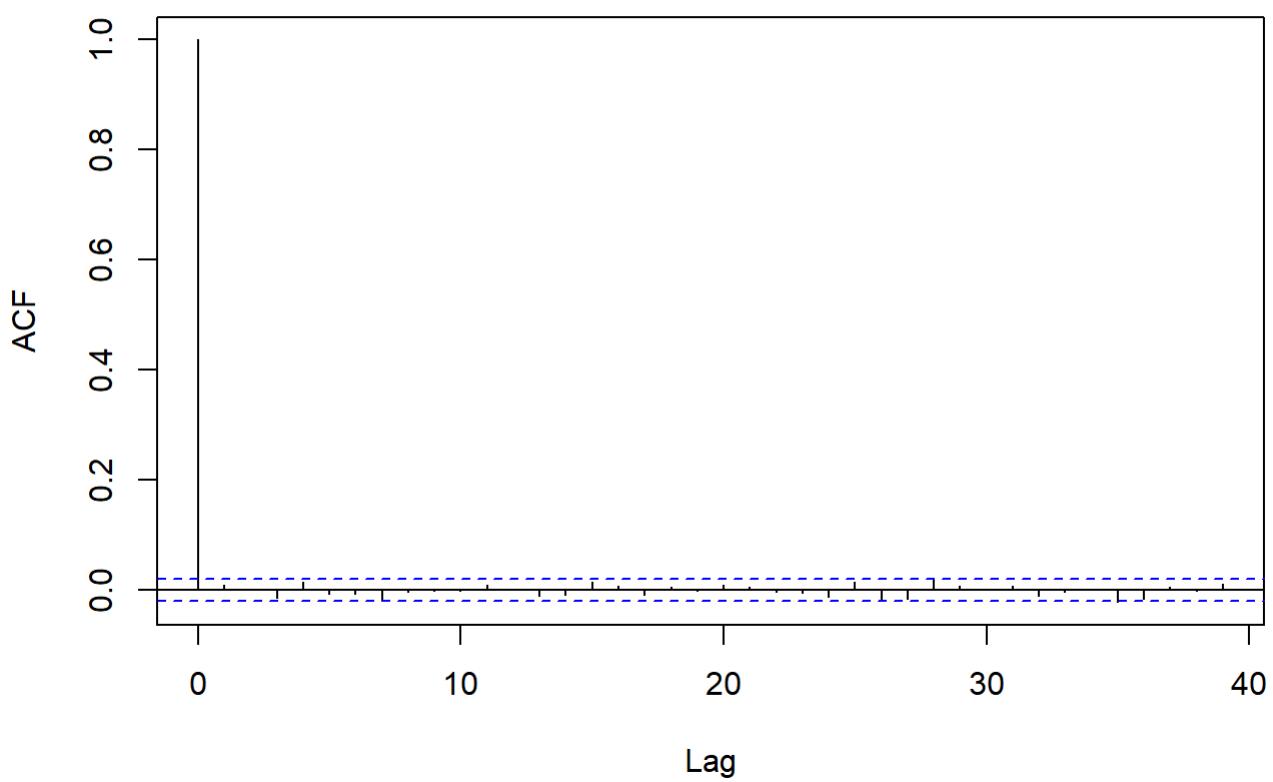
**Trace plot for Beta (Observation 29 )****Trace plot for Beta (Observation 30 )**

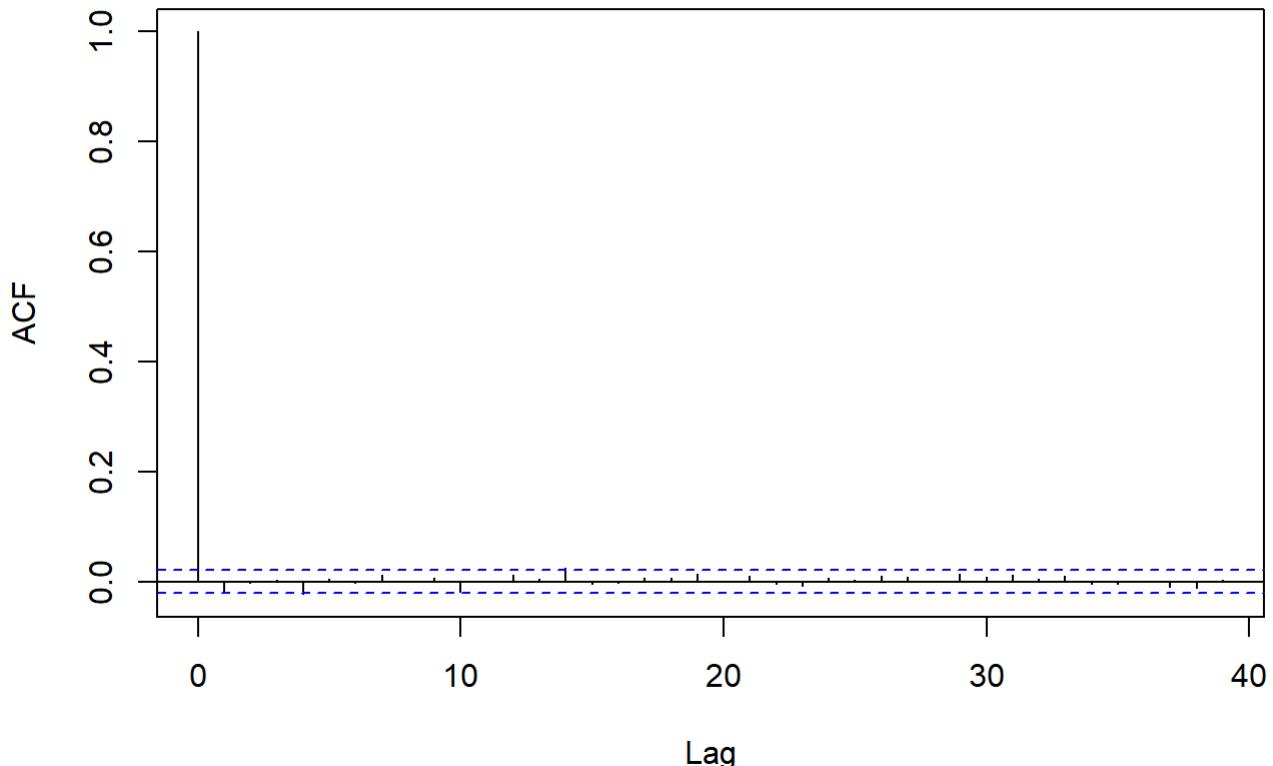
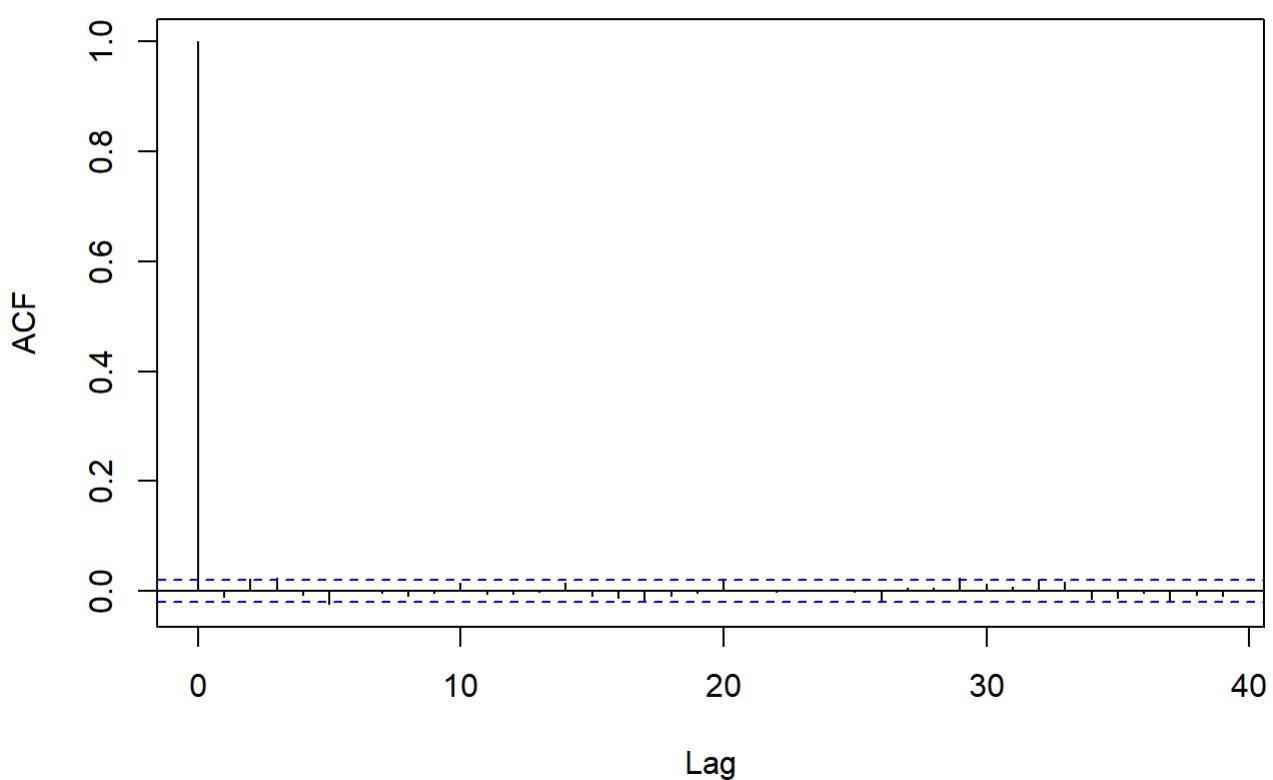
```
# ACF plots for alpha
for (i in 1:30) {
  # ACF for alpha (i-th observation)
  acf(alpha_post[i, ], main = paste("ACF for Alpha (Observation", i, ")"))
}
```

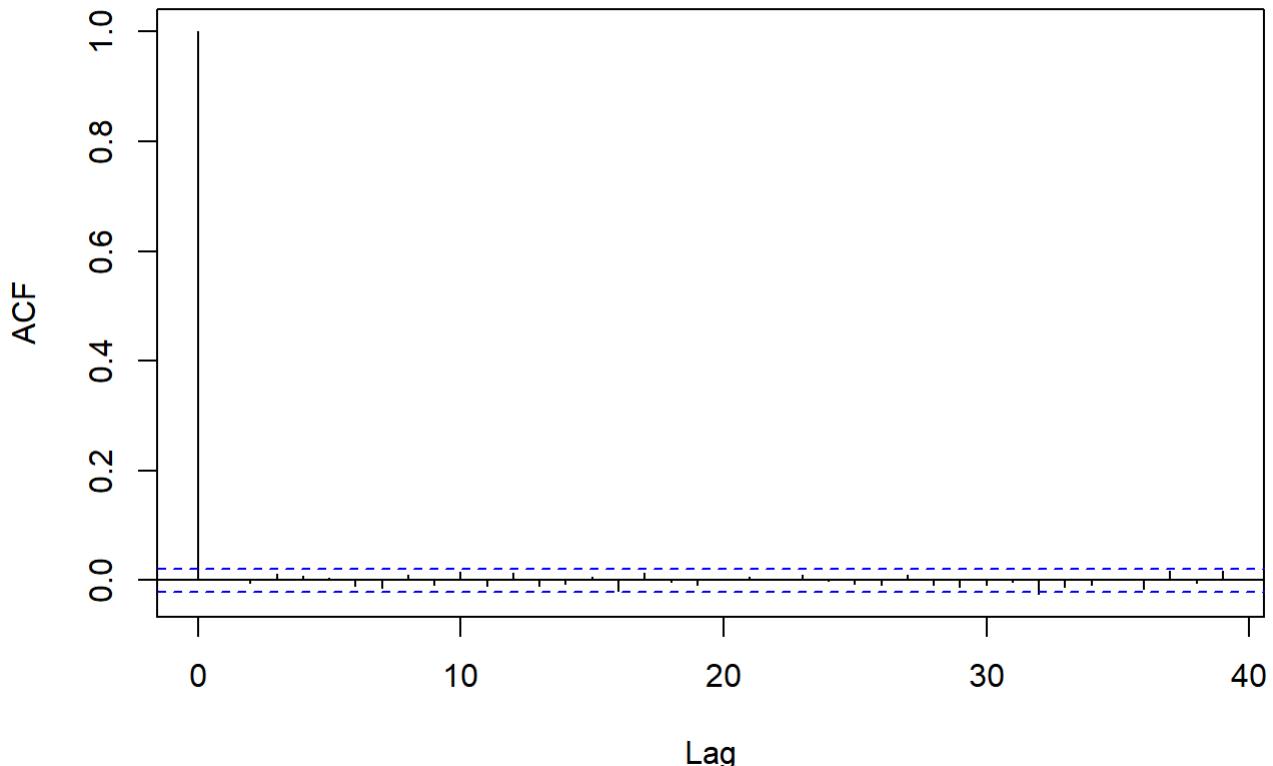
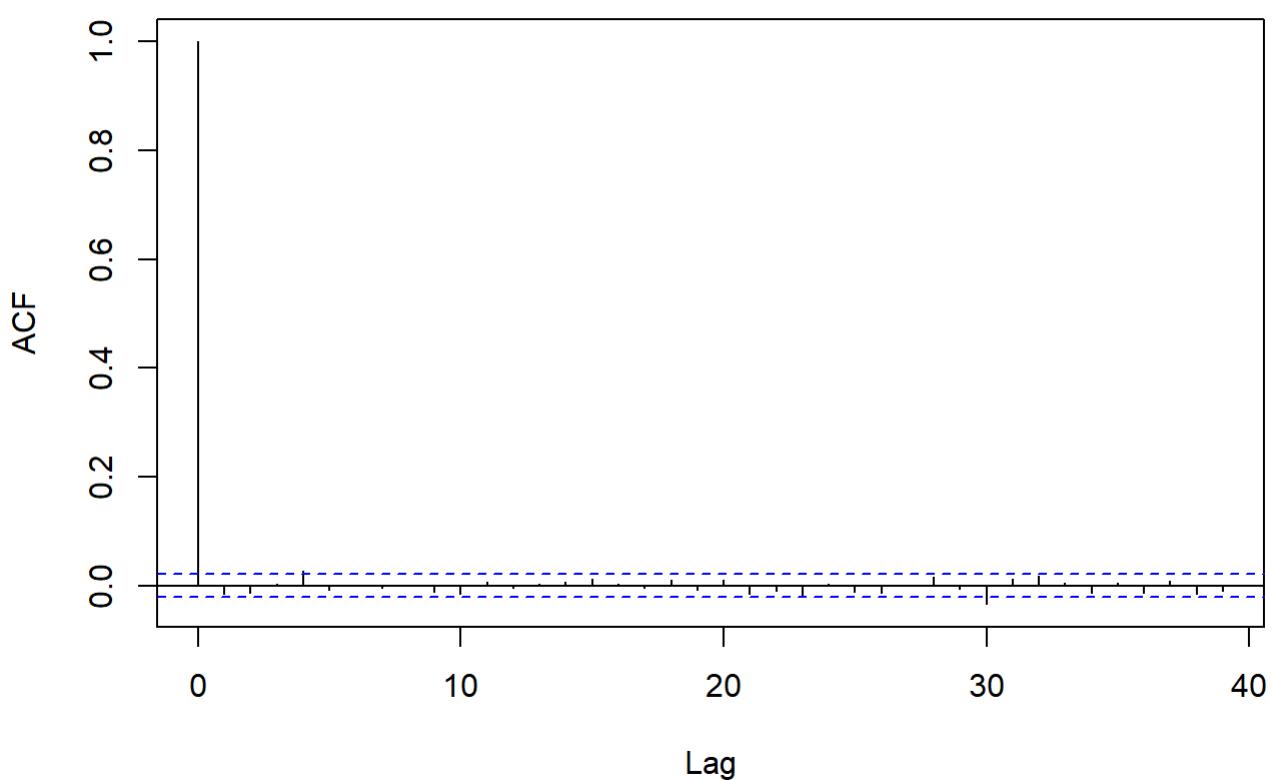
**ACF for Alpha (Observation 1 )****ACF for Alpha (Observation 2 )**

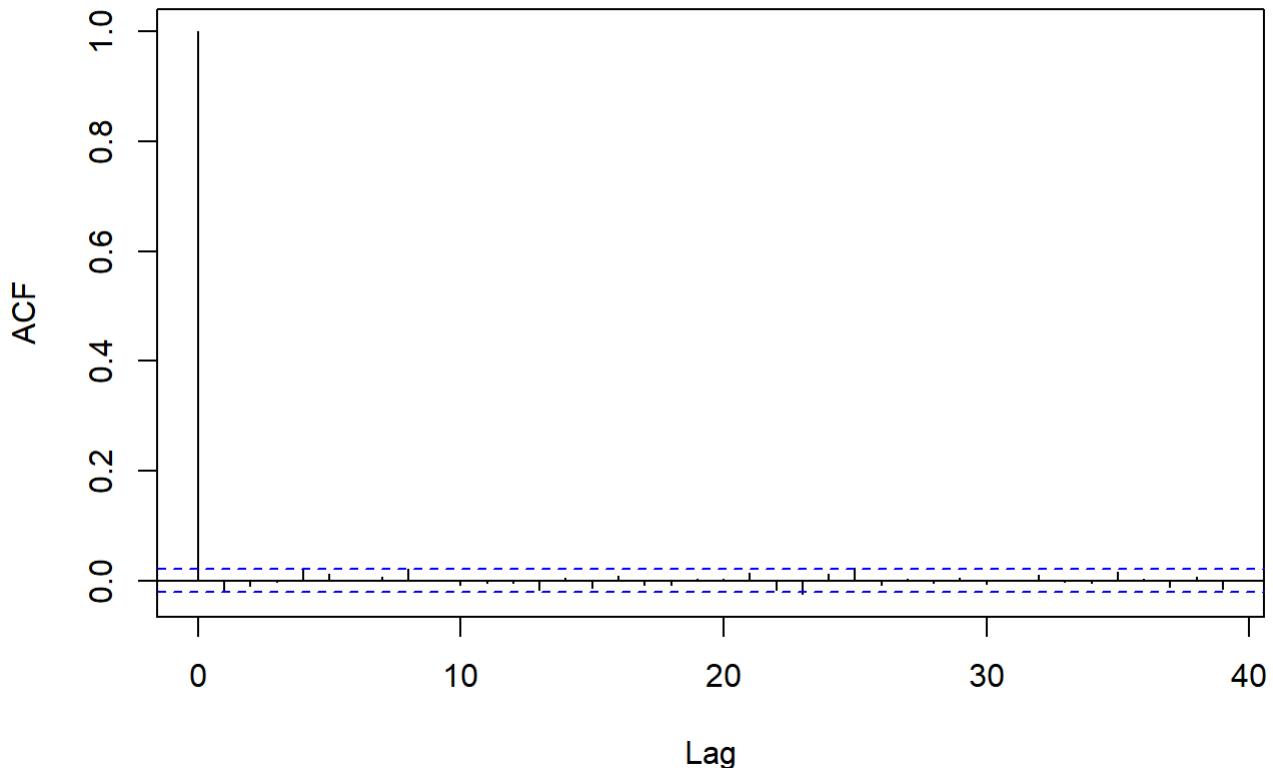
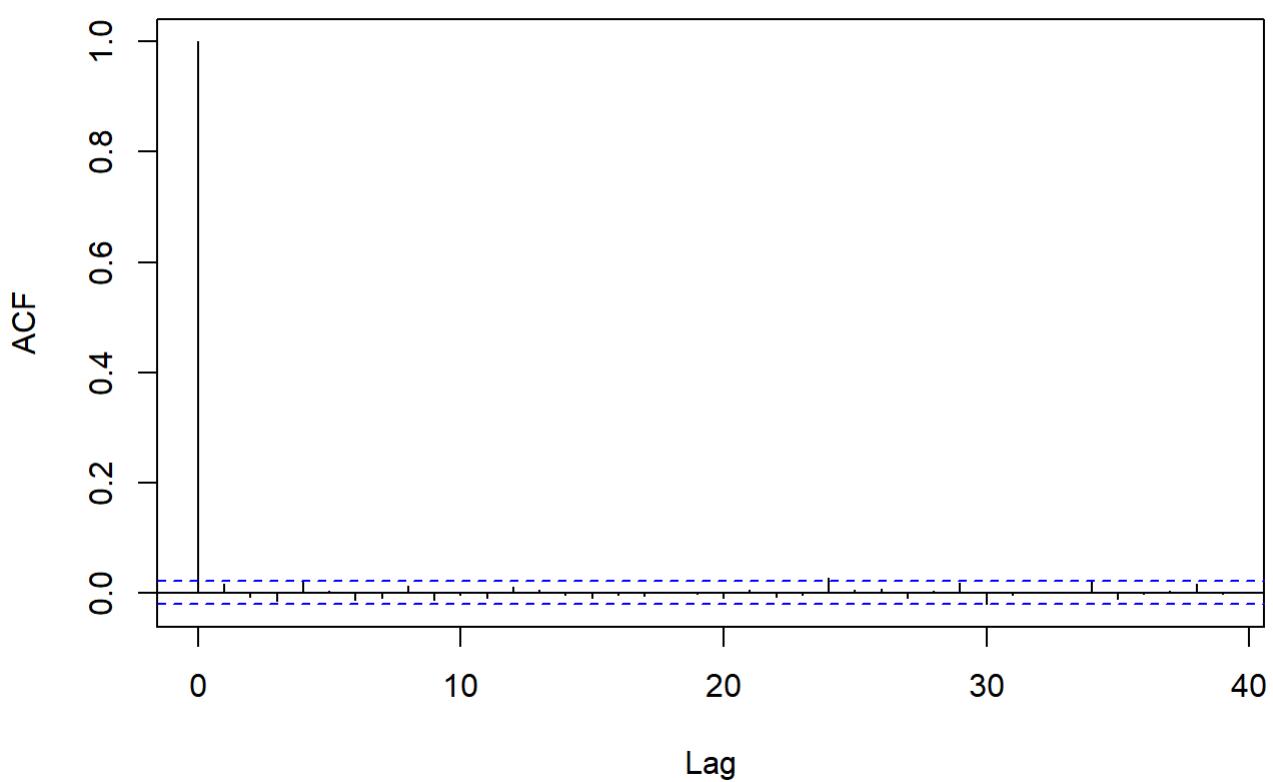
**ACF for Alpha (Observation 3 )****ACF for Alpha (Observation 4 )**

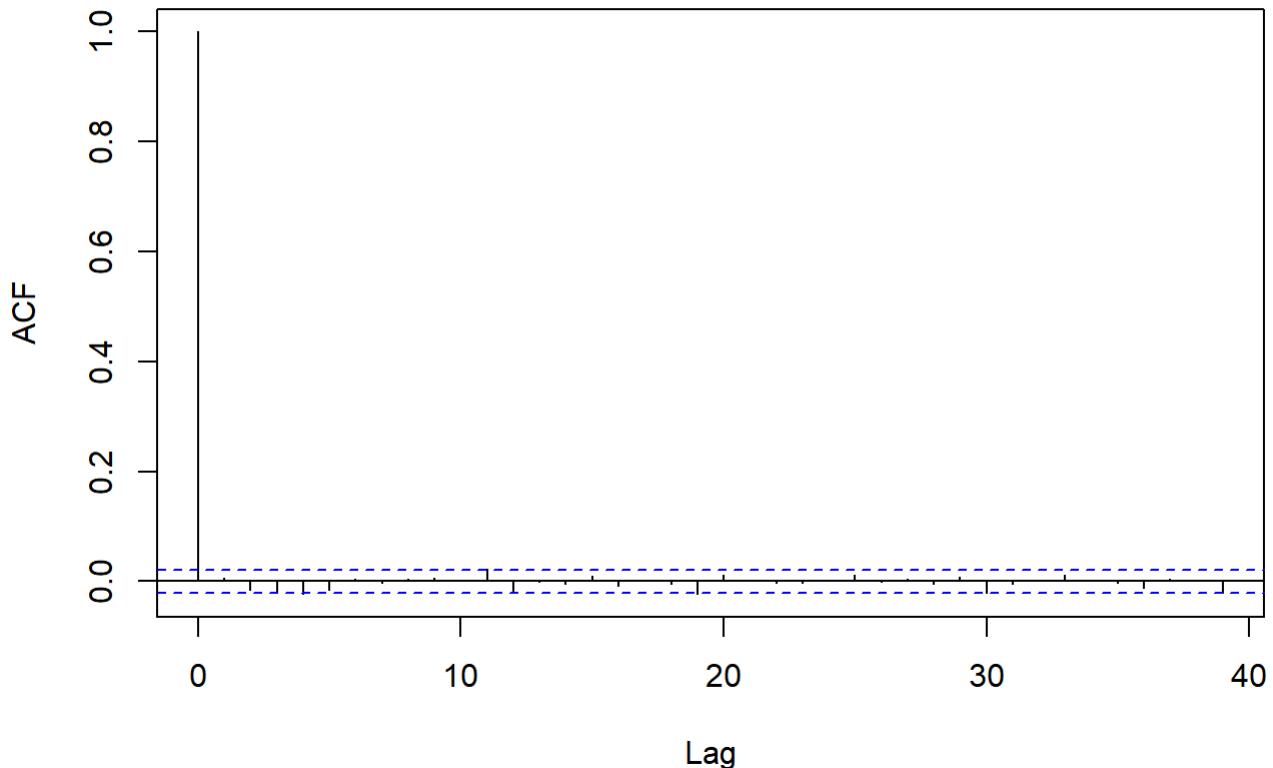
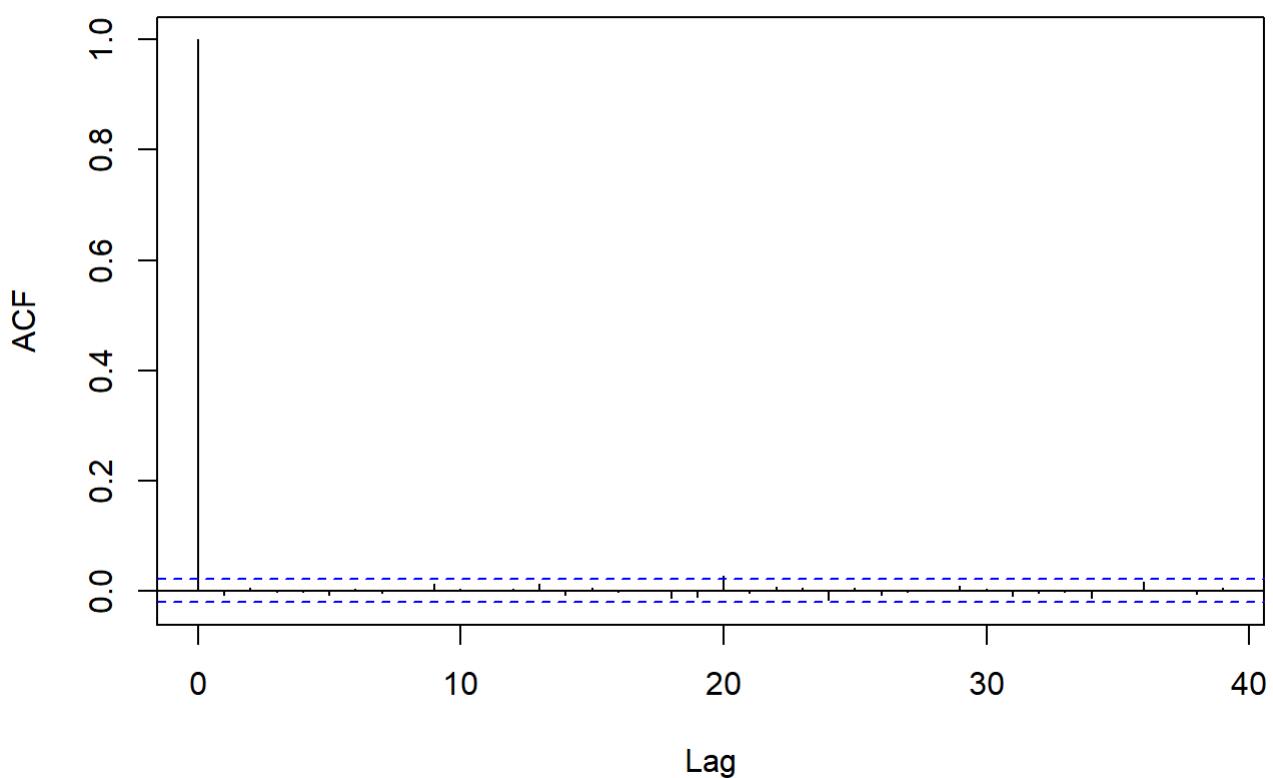
**ACF for Alpha (Observation 5 )****ACF for Alpha (Observation 6 )**

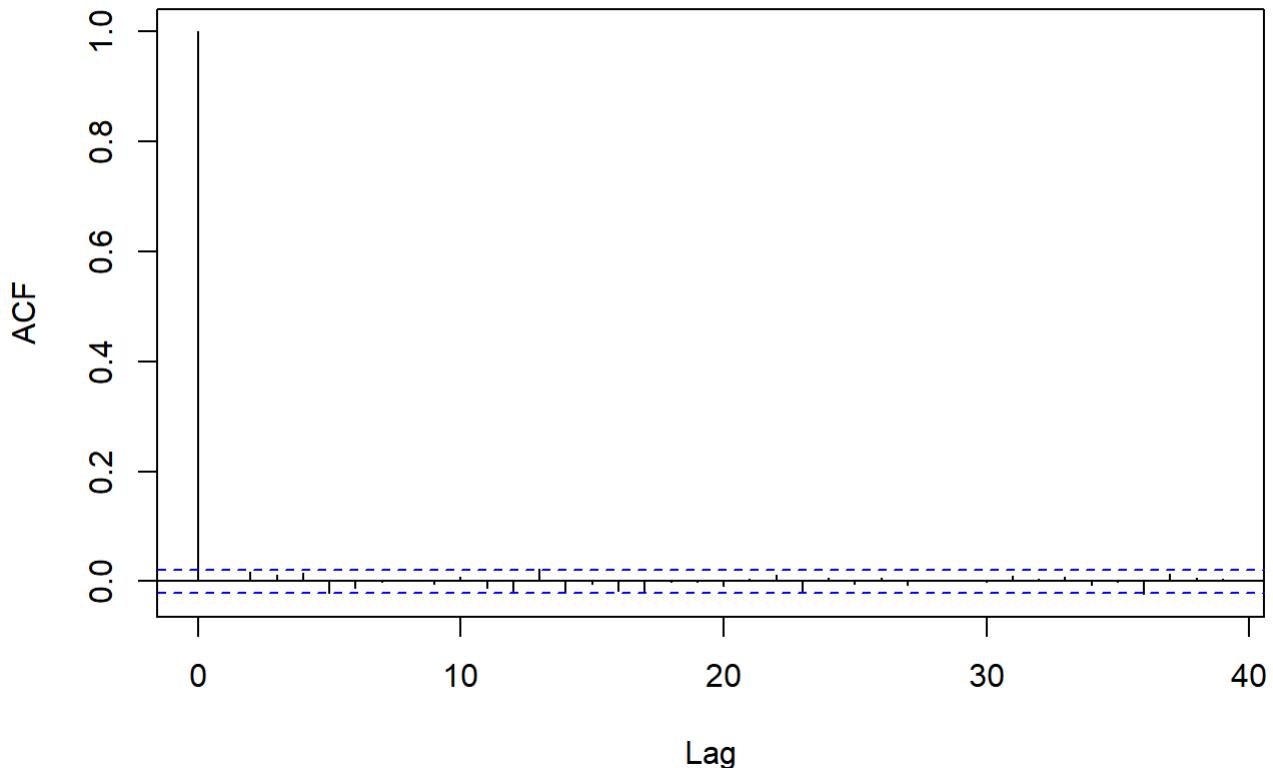
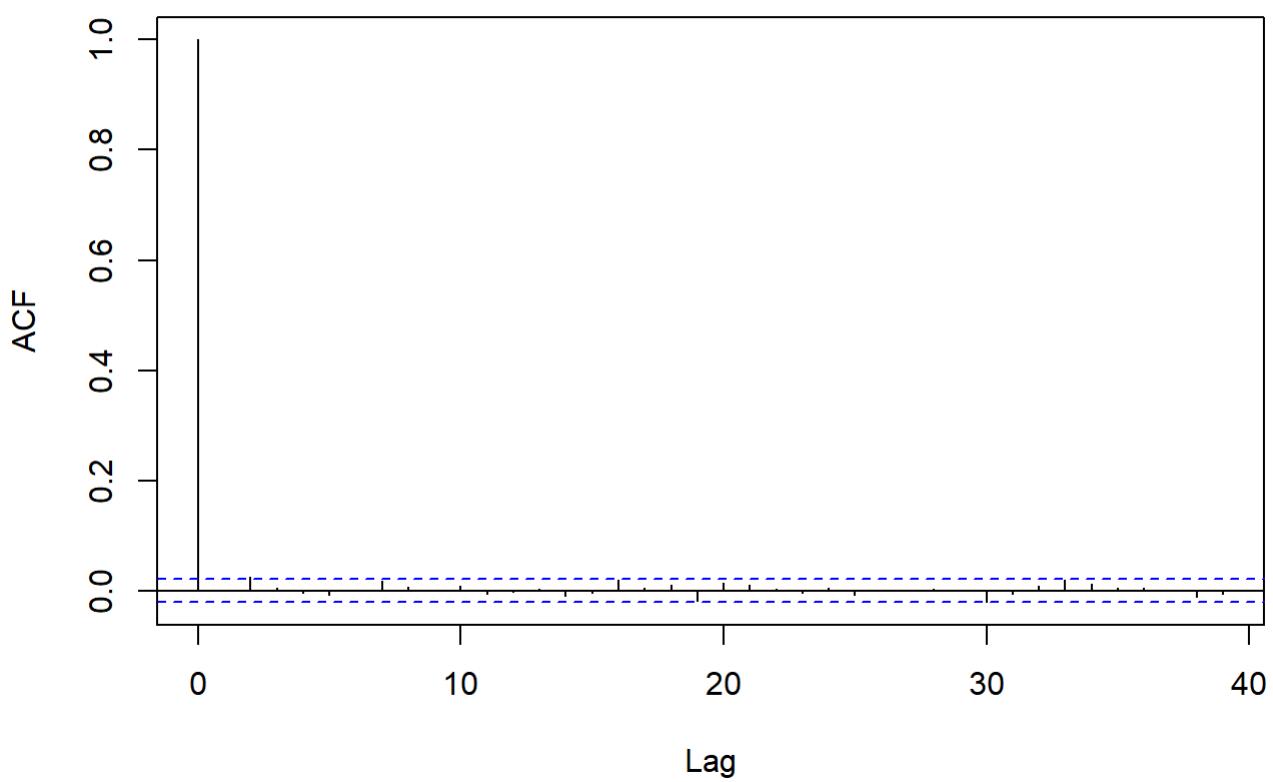
**ACF for Alpha (Observation 7 )****ACF for Alpha (Observation 8 )**

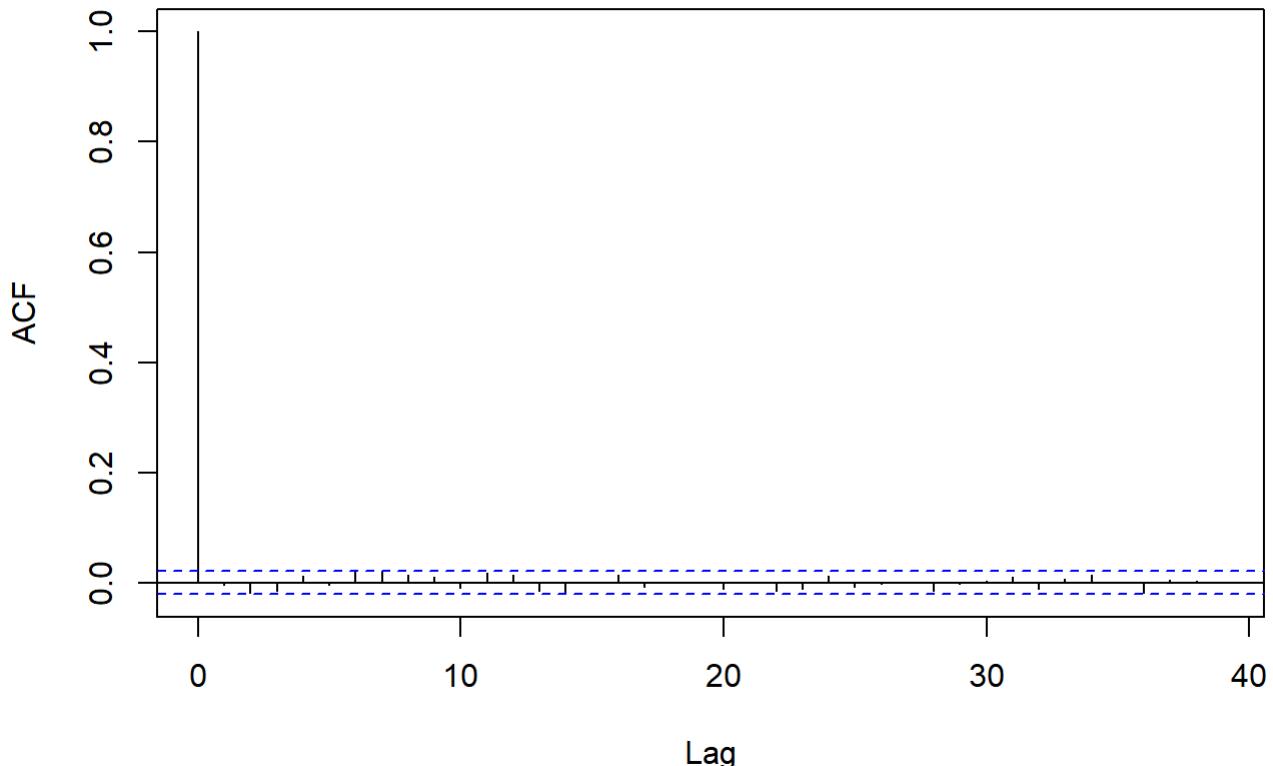
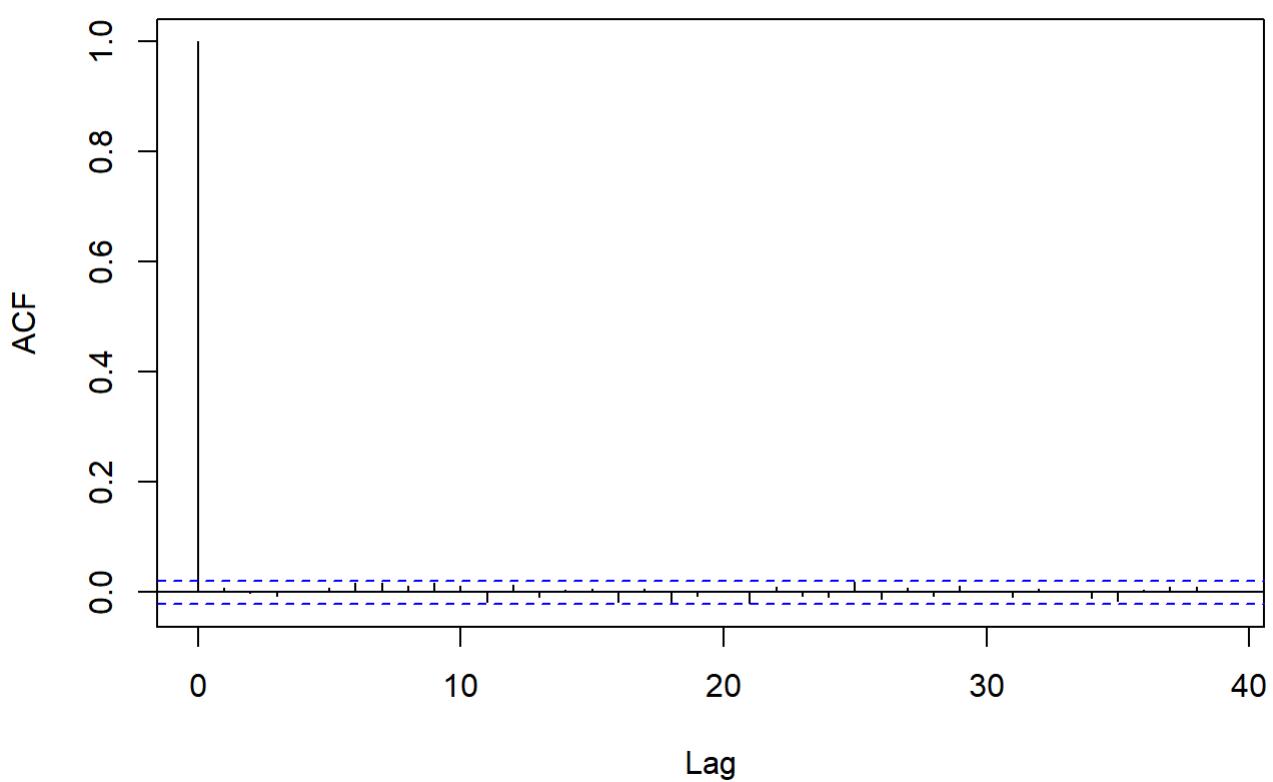
**ACF for Alpha (Observation 9 )****ACF for Alpha (Observation 10 )**

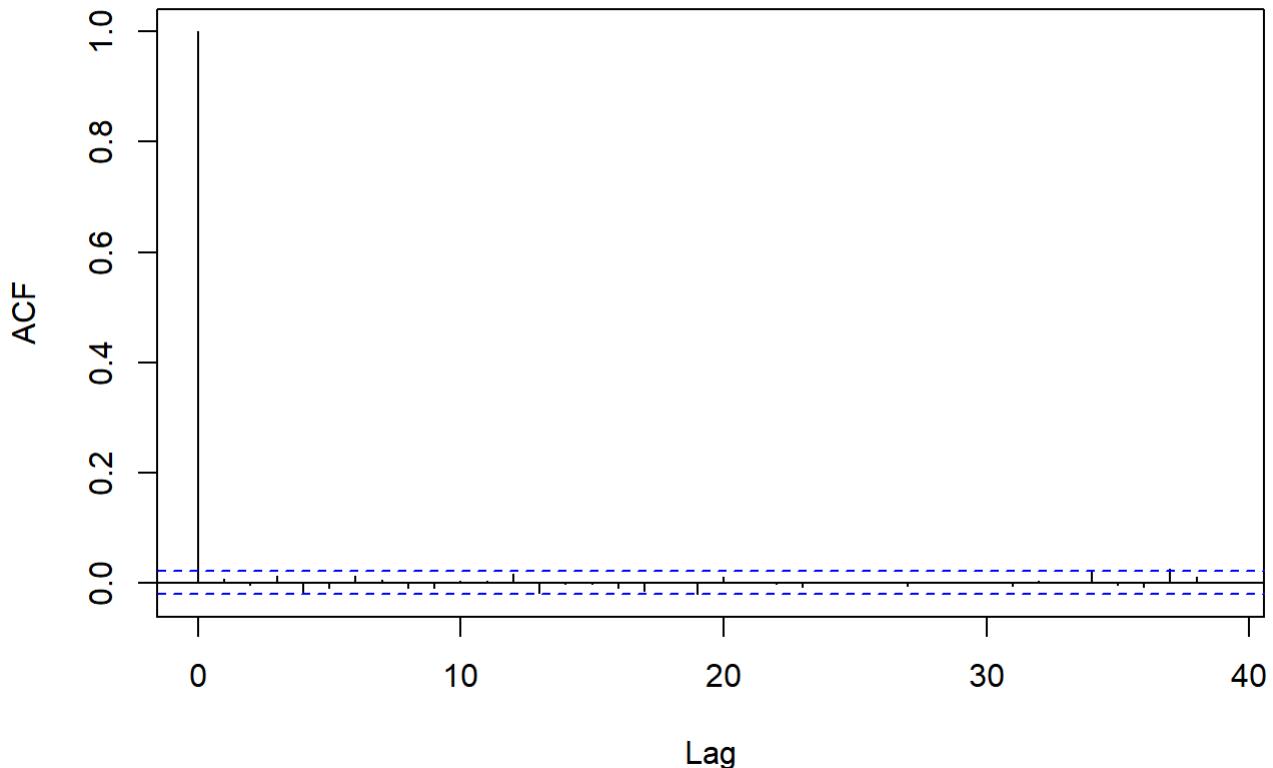
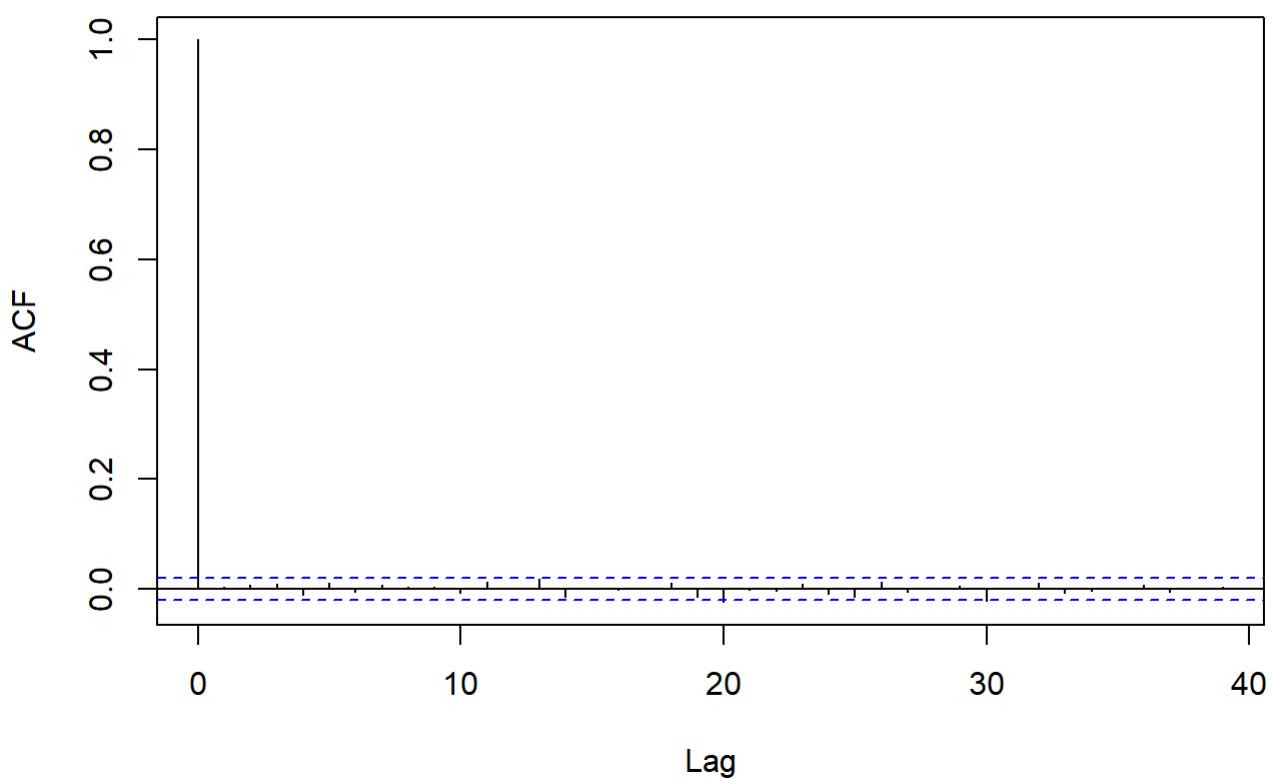
**ACF for Alpha (Observation 11 )****ACF for Alpha (Observation 12 )**

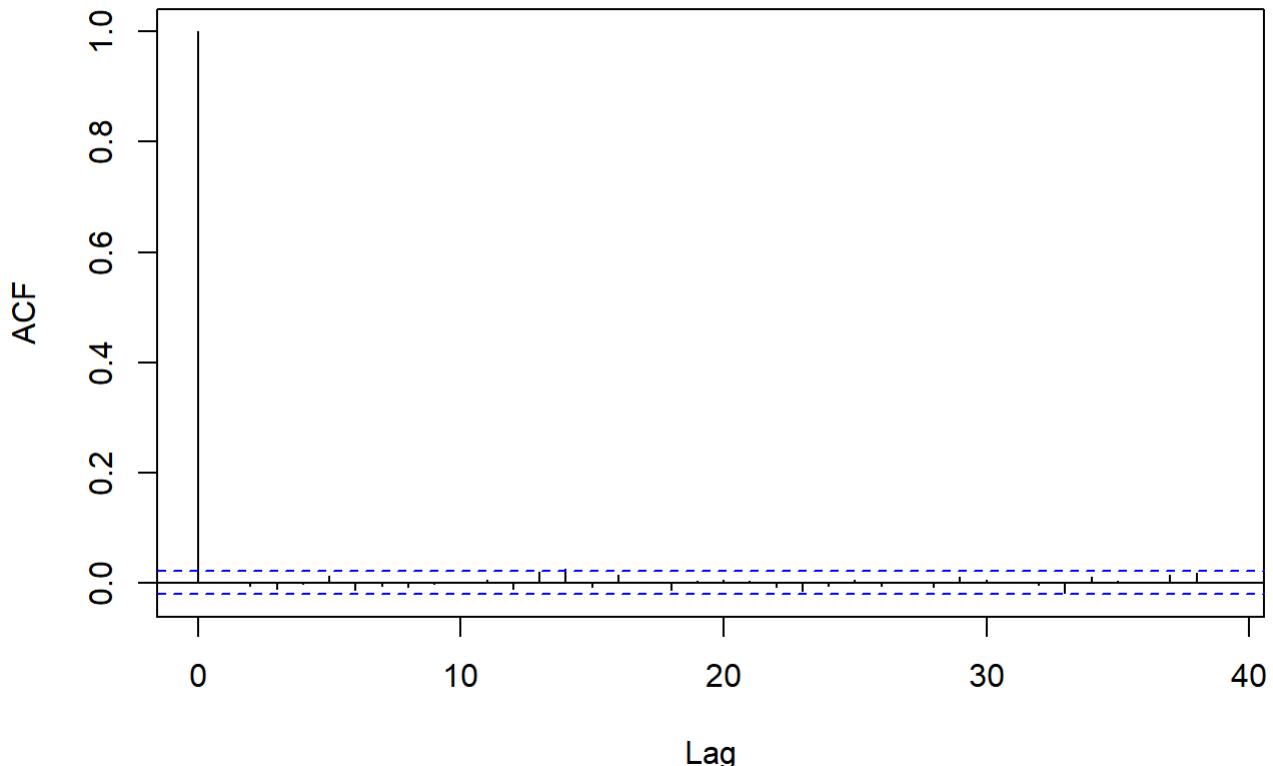
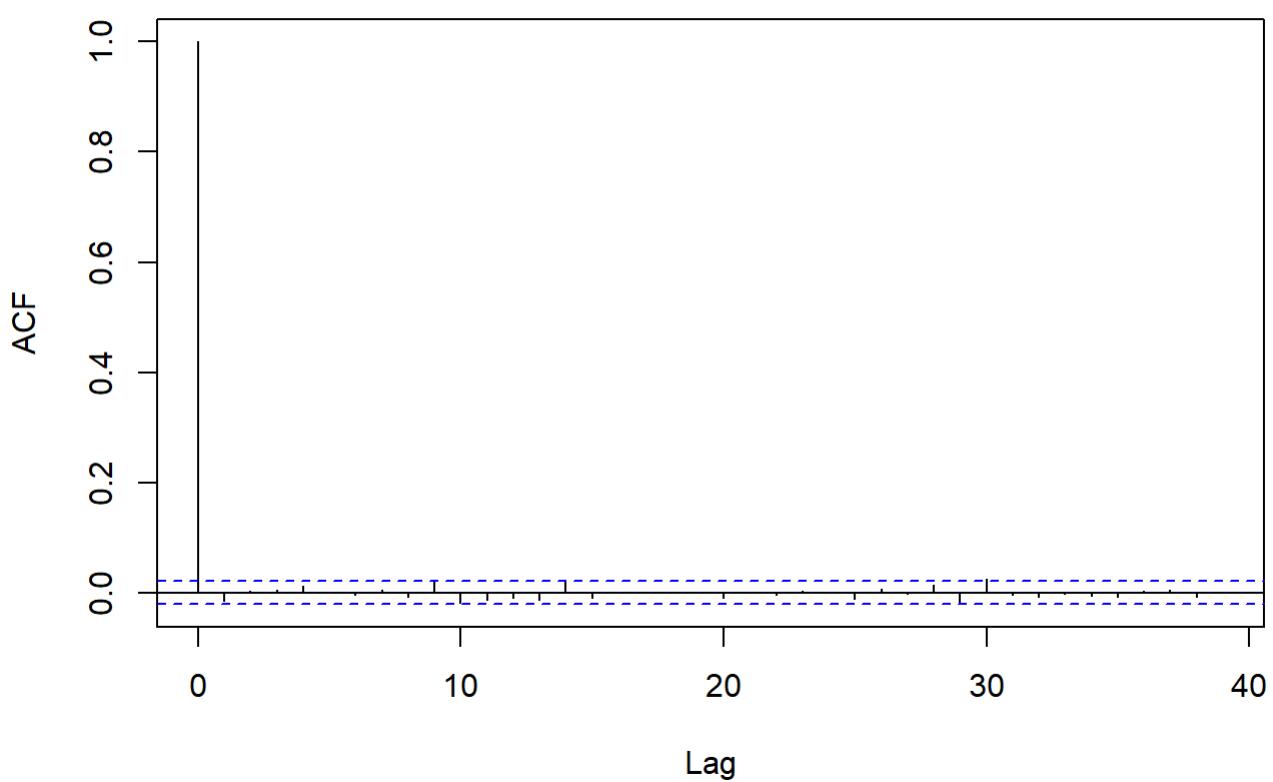
**ACF for Alpha (Observation 13 )****ACF for Alpha (Observation 14 )**

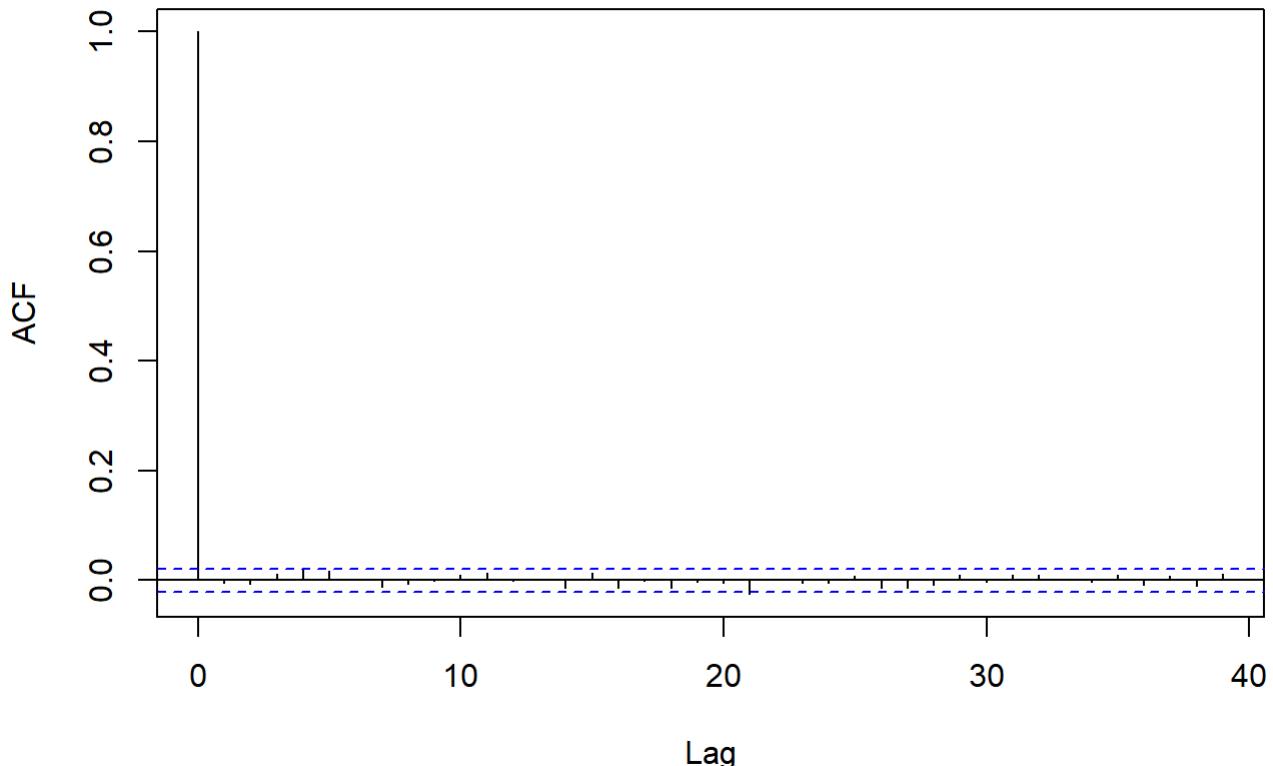
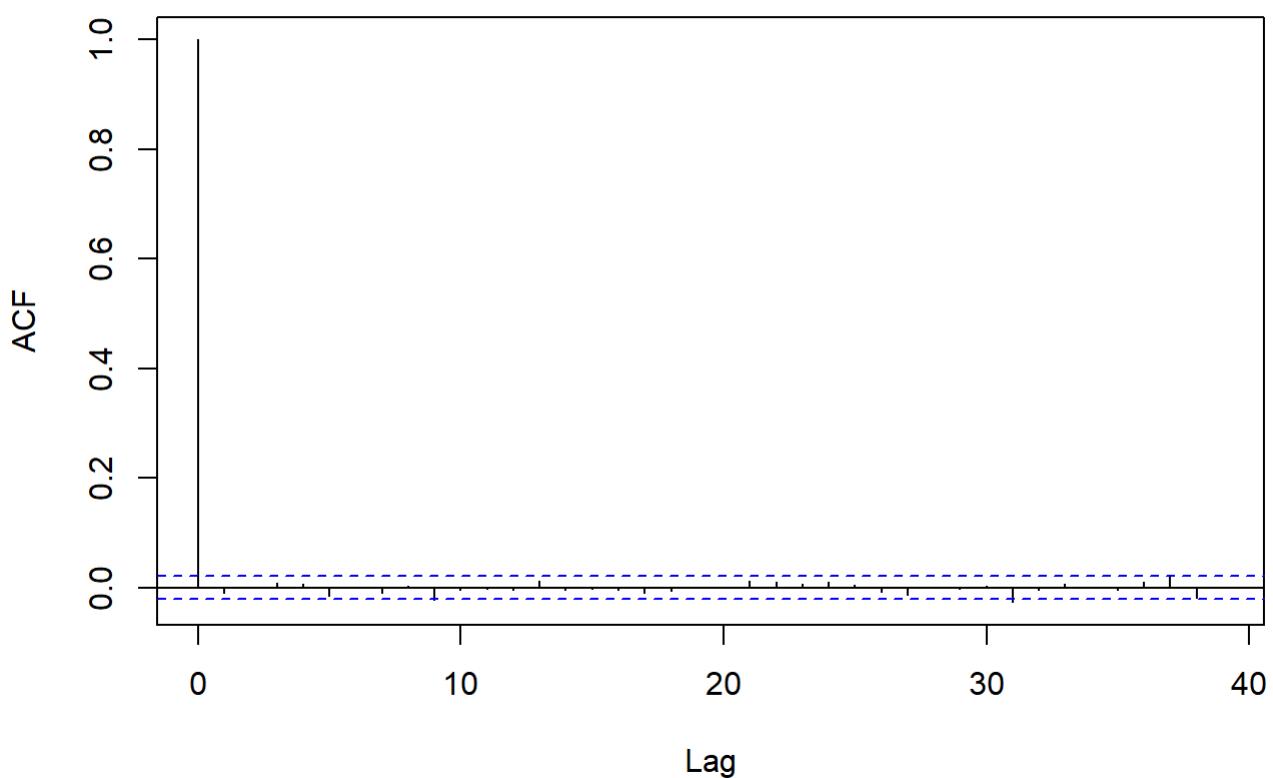
**ACF for Alpha (Observation 15 )****ACF for Alpha (Observation 16 )**

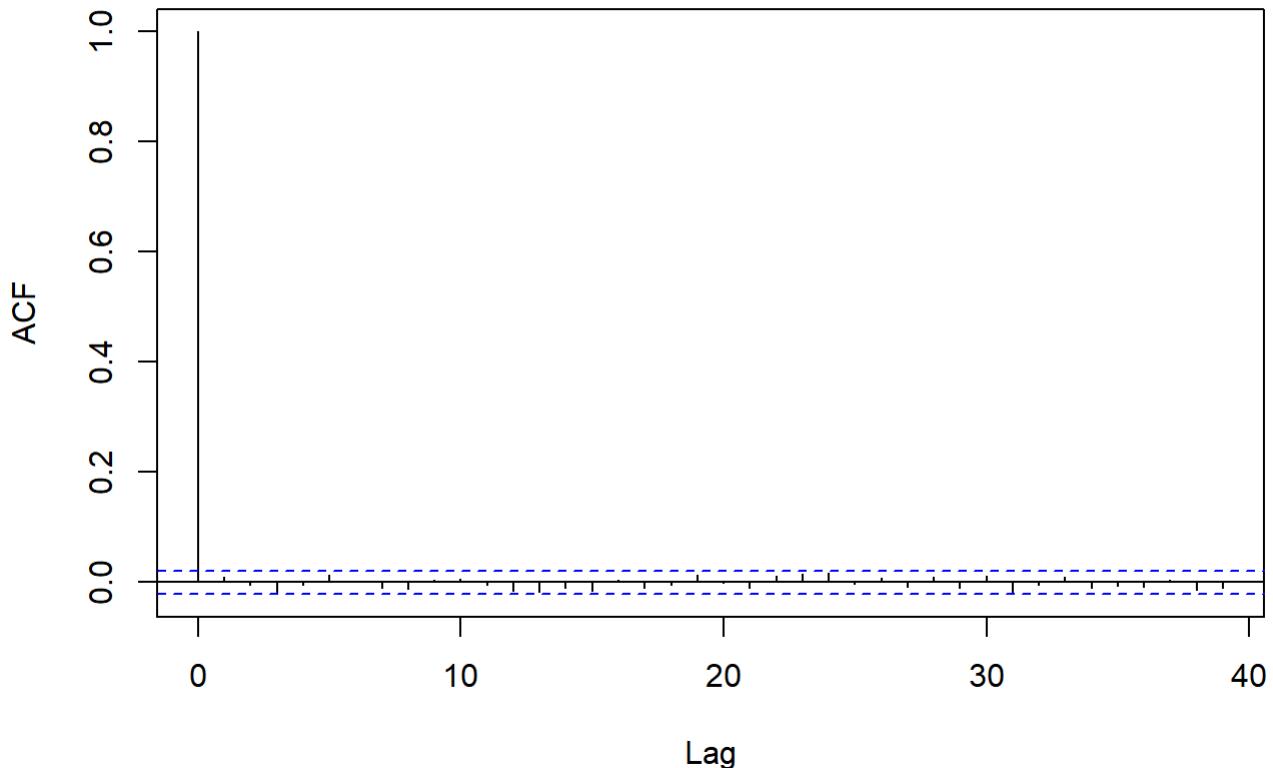
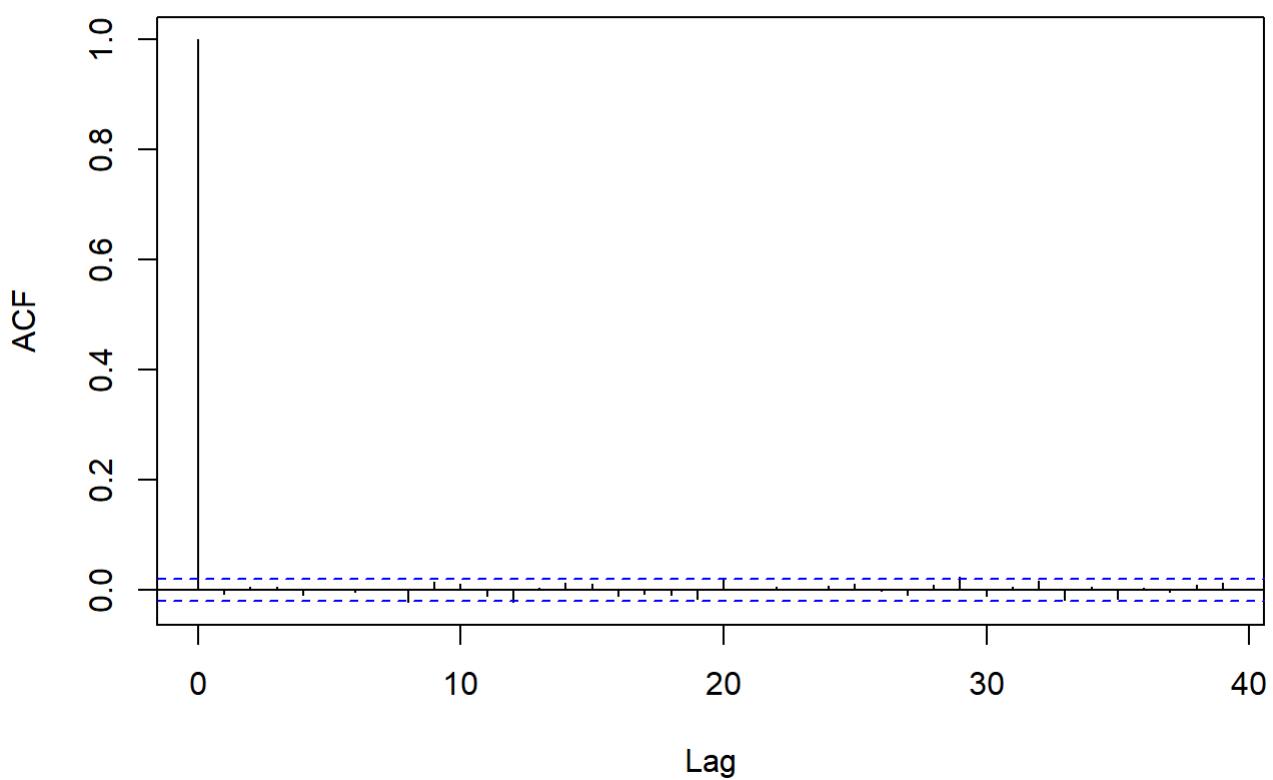
**ACF for Alpha (Observation 17 )****ACF for Alpha (Observation 18 )**

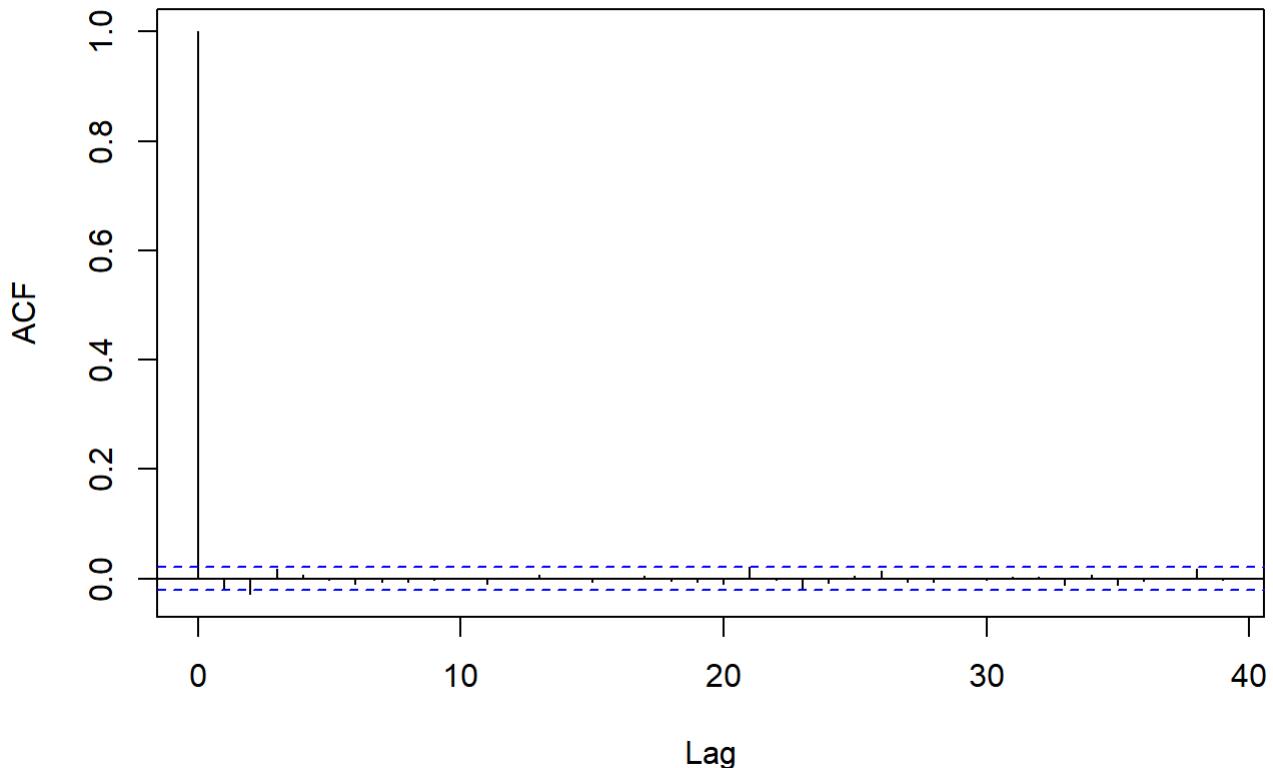
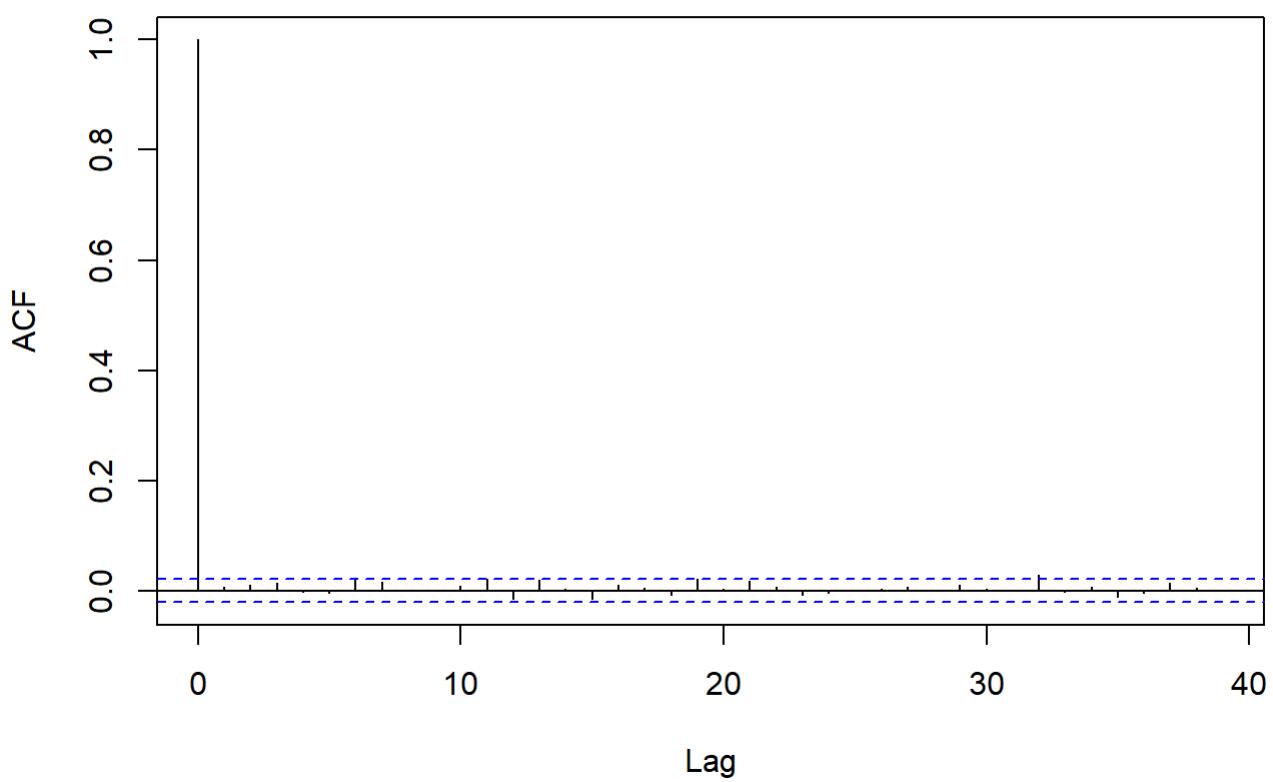
**ACF for Alpha (Observation 19 )****ACF for Alpha (Observation 20 )**

**ACF for Alpha (Observation 21 )****ACF for Alpha (Observation 22 )**

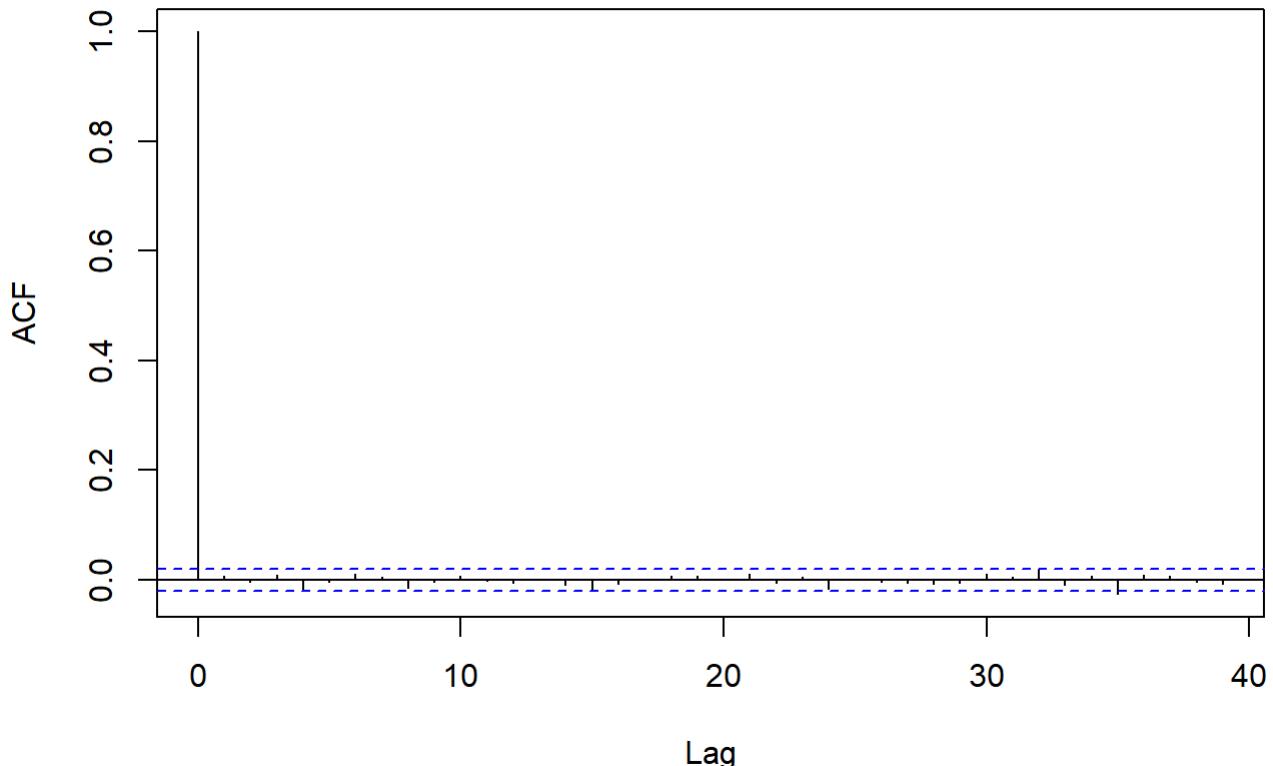
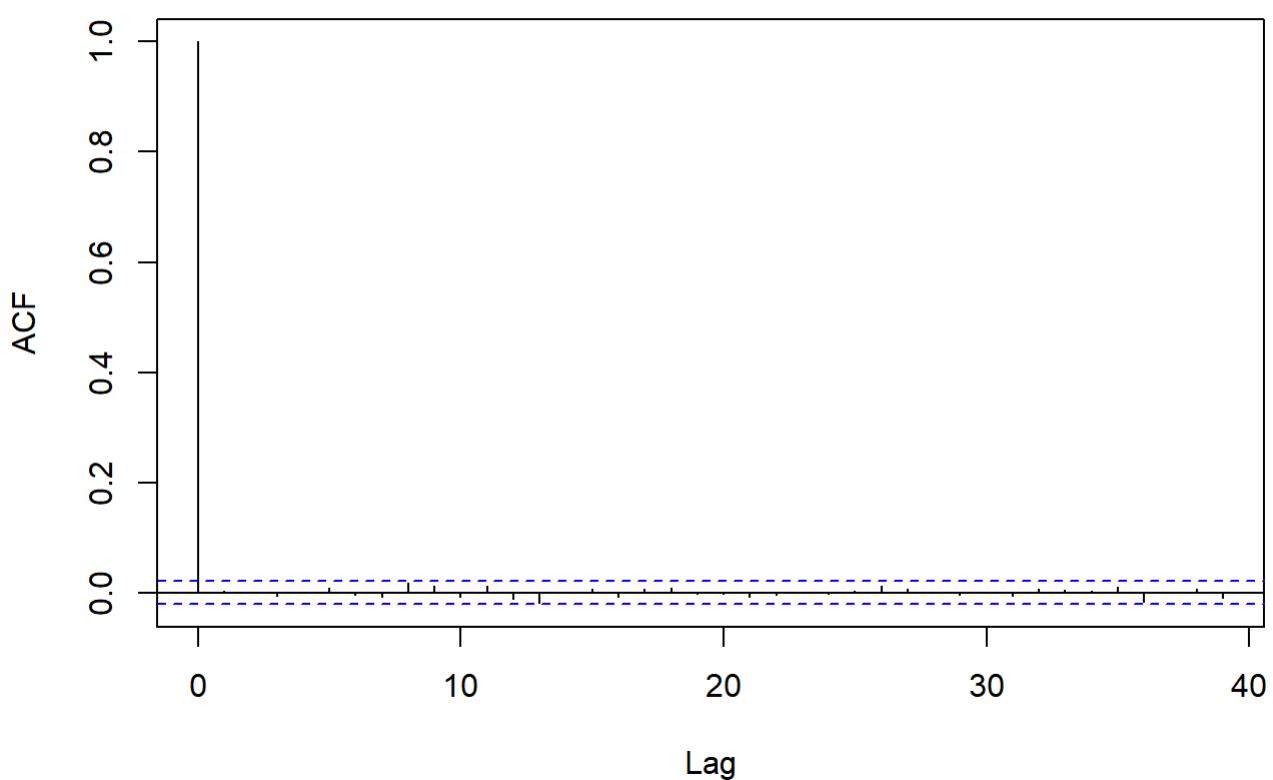
**ACF for Alpha (Observation 23 )****ACF for Alpha (Observation 24 )**

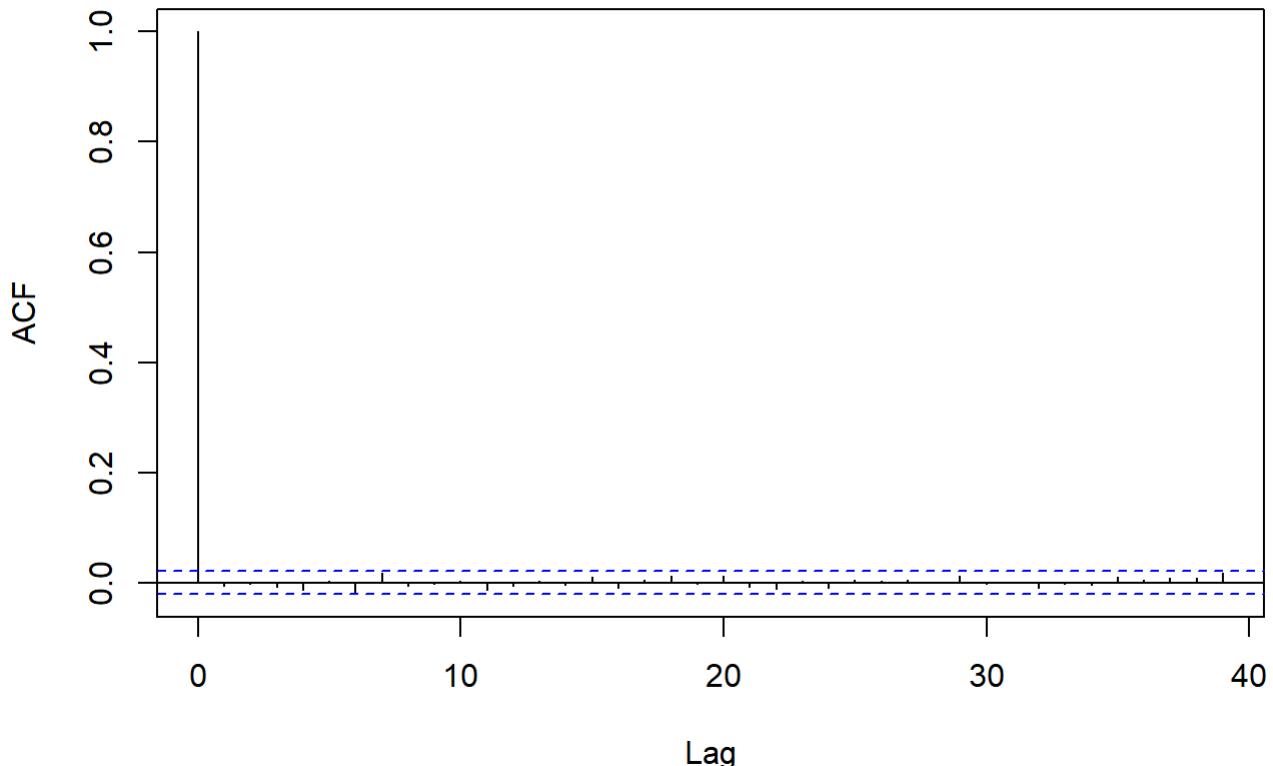
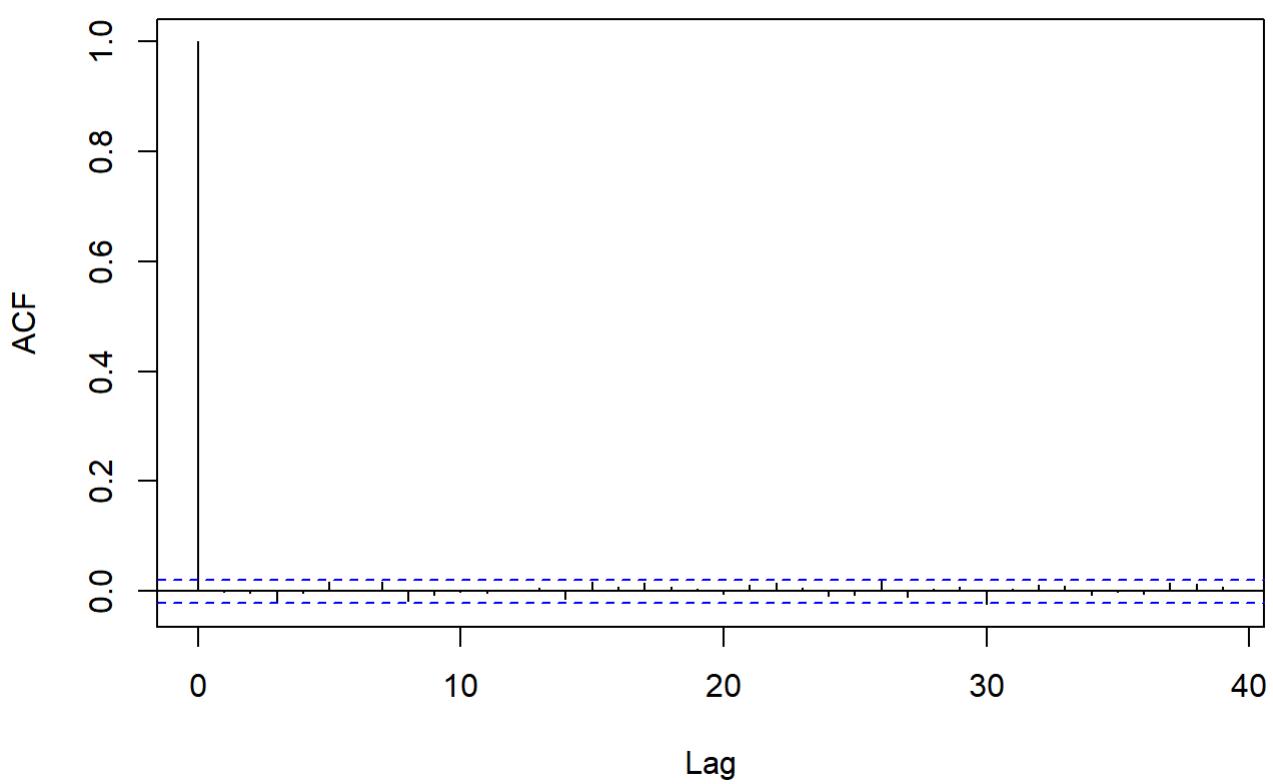
**ACF for Alpha (Observation 25 )****ACF for Alpha (Observation 26 )**

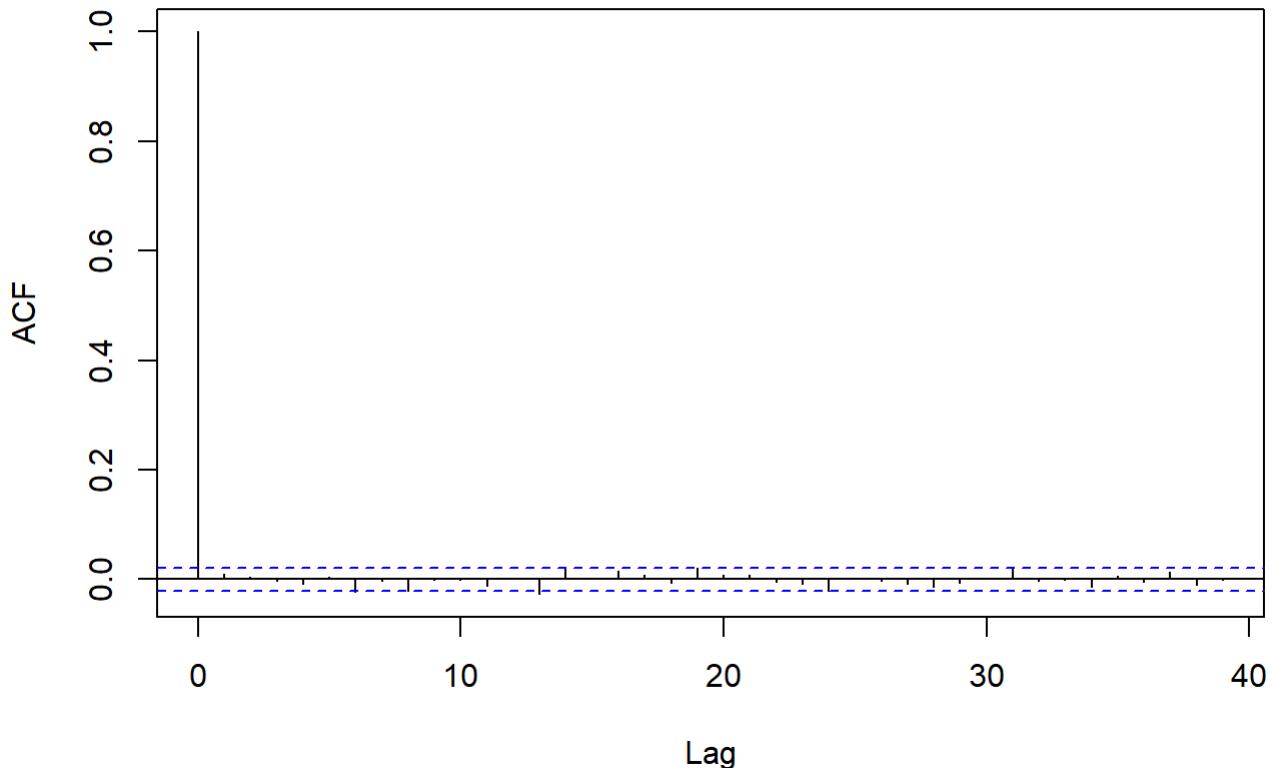
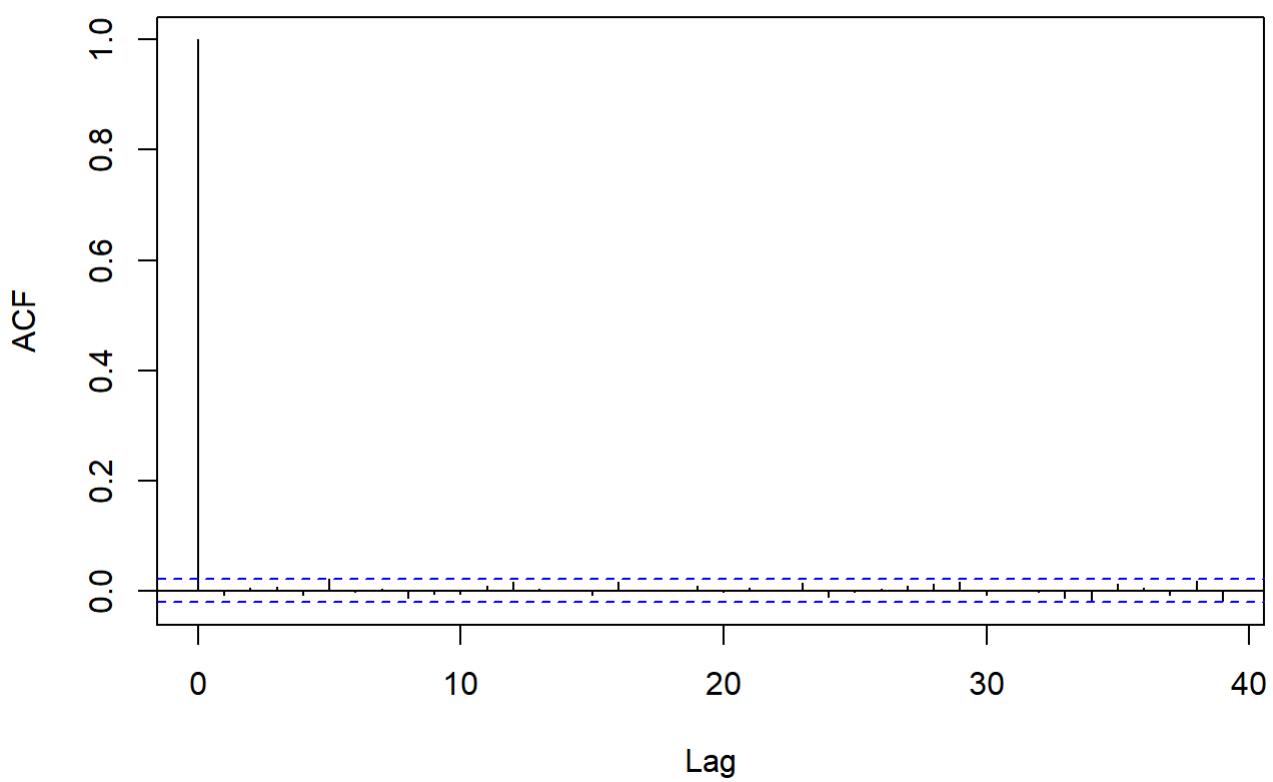
**ACF for Alpha (Observation 27 )****ACF for Alpha (Observation 28 )**

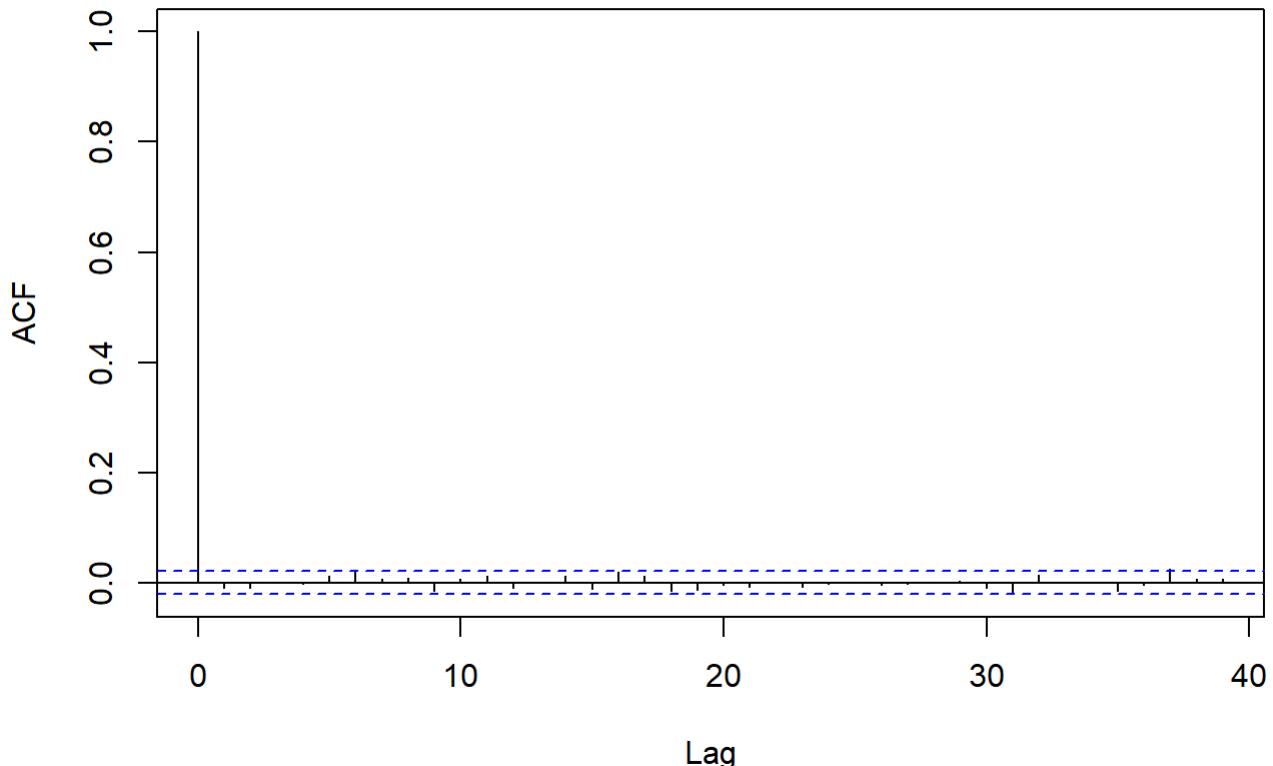
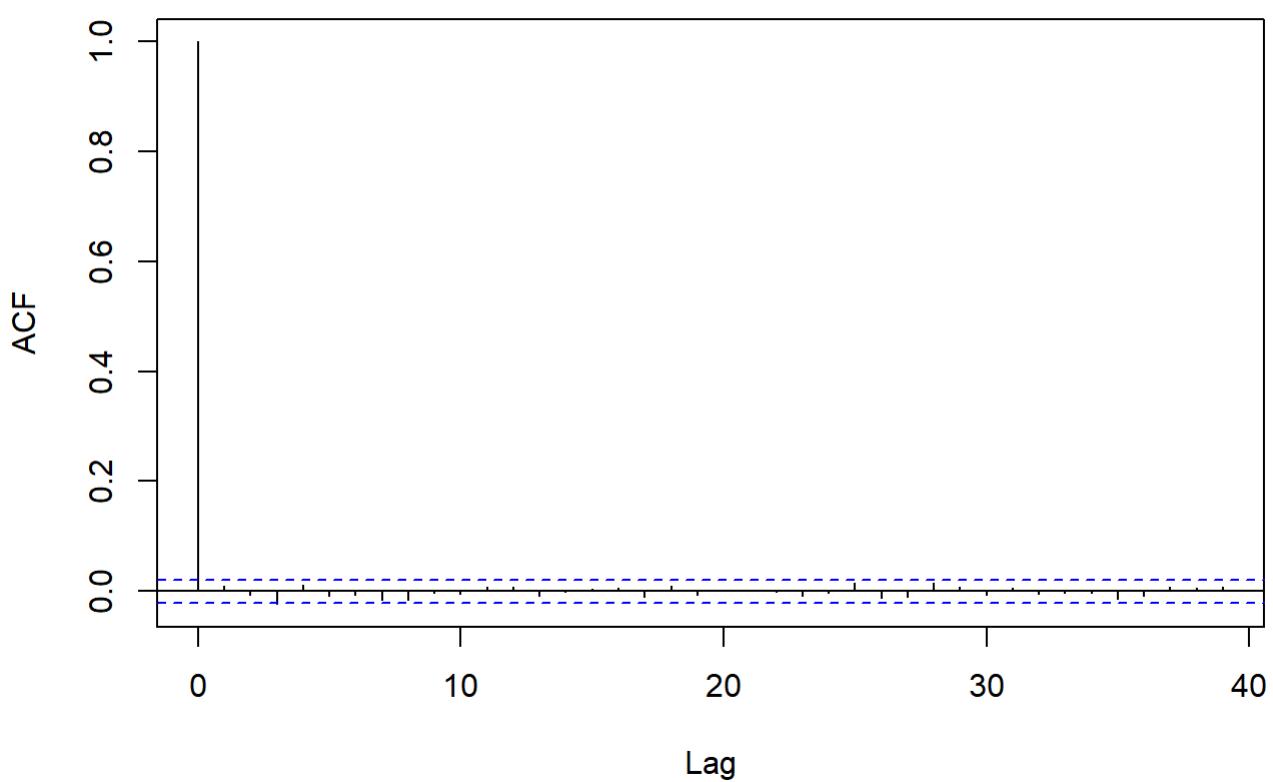
**ACF for Alpha (Observation 29 )****ACF for Alpha (Observation 30 )**

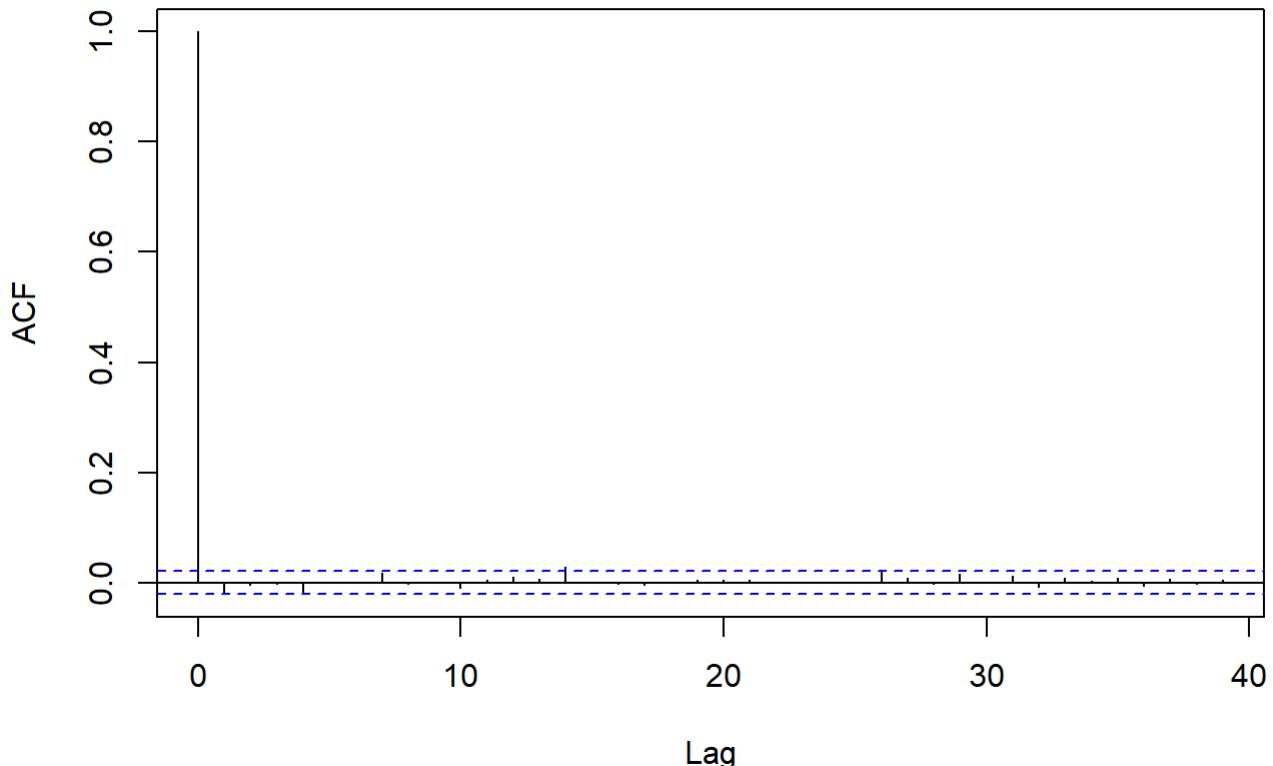
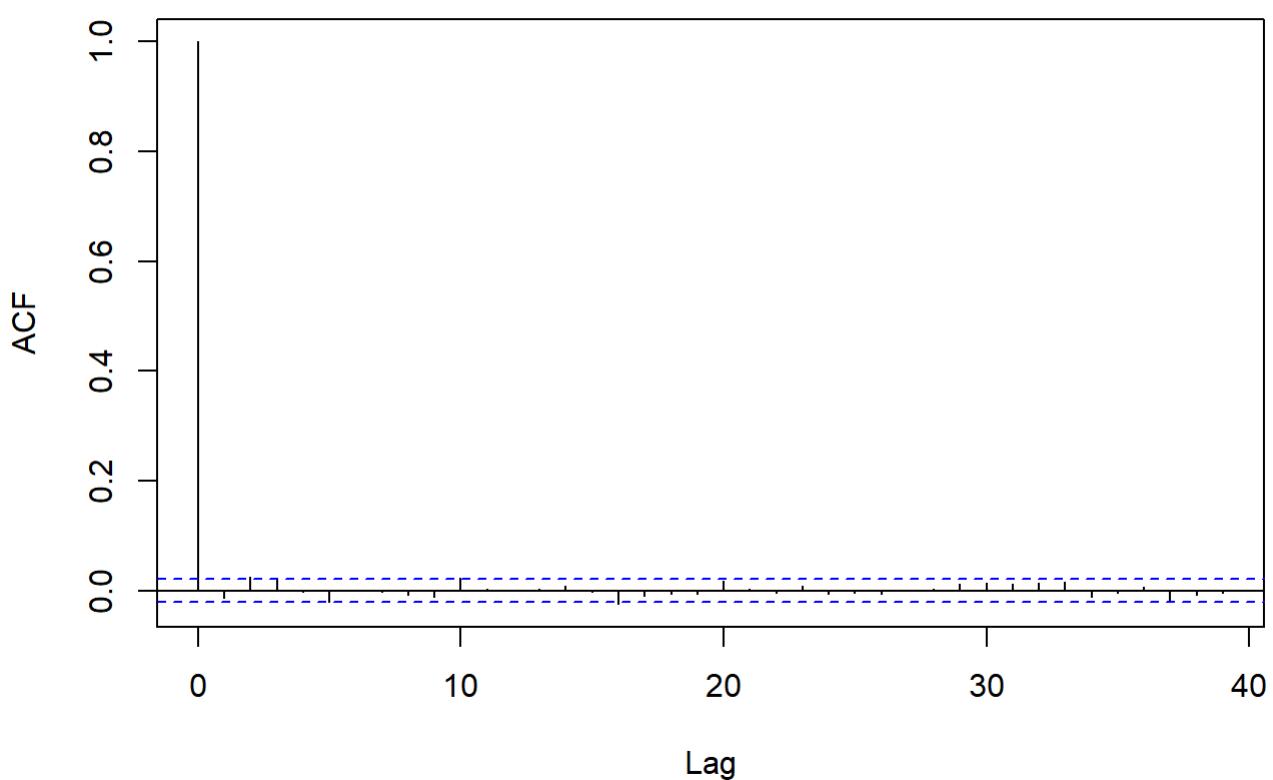
```
# ACF plots for beta
for (i in 1:30) {
  # ACF for beta (i-th observation)
  acf(beta_post[i, ], main = paste("ACF for Beta (Observation", i, ")"))
}
```

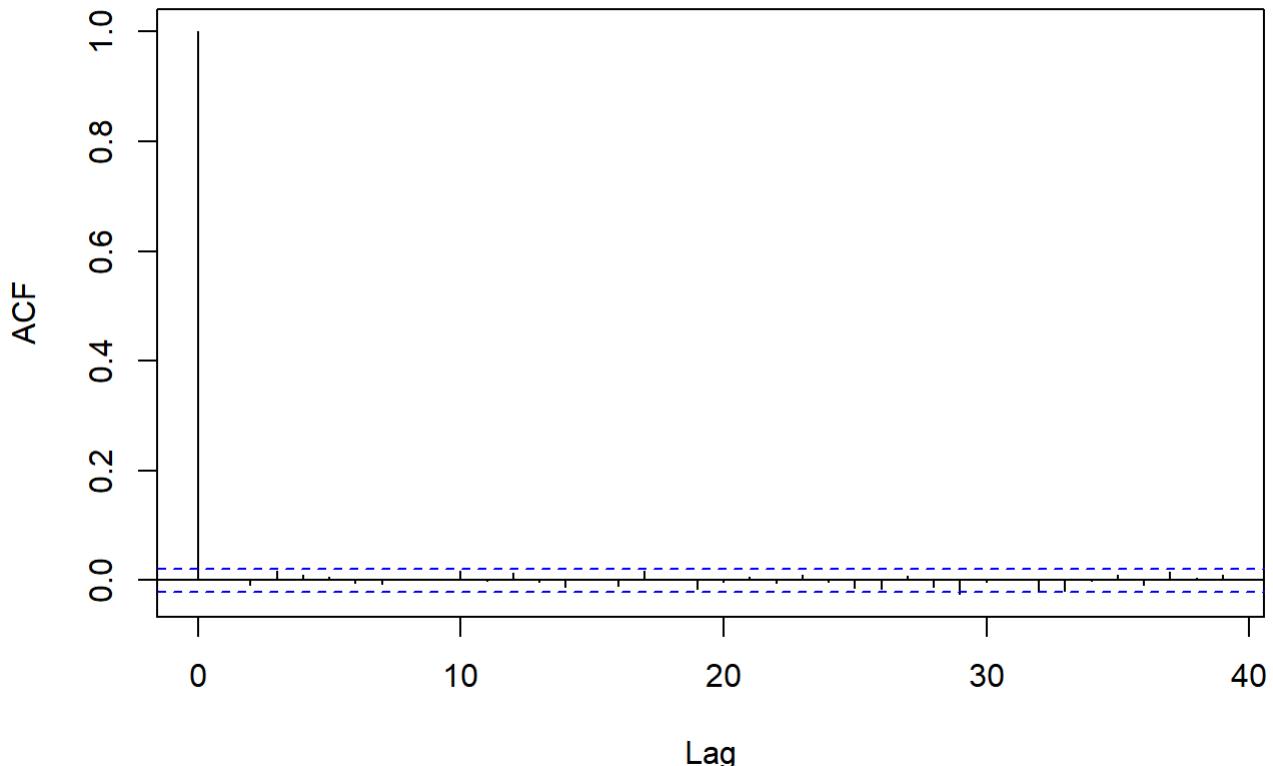
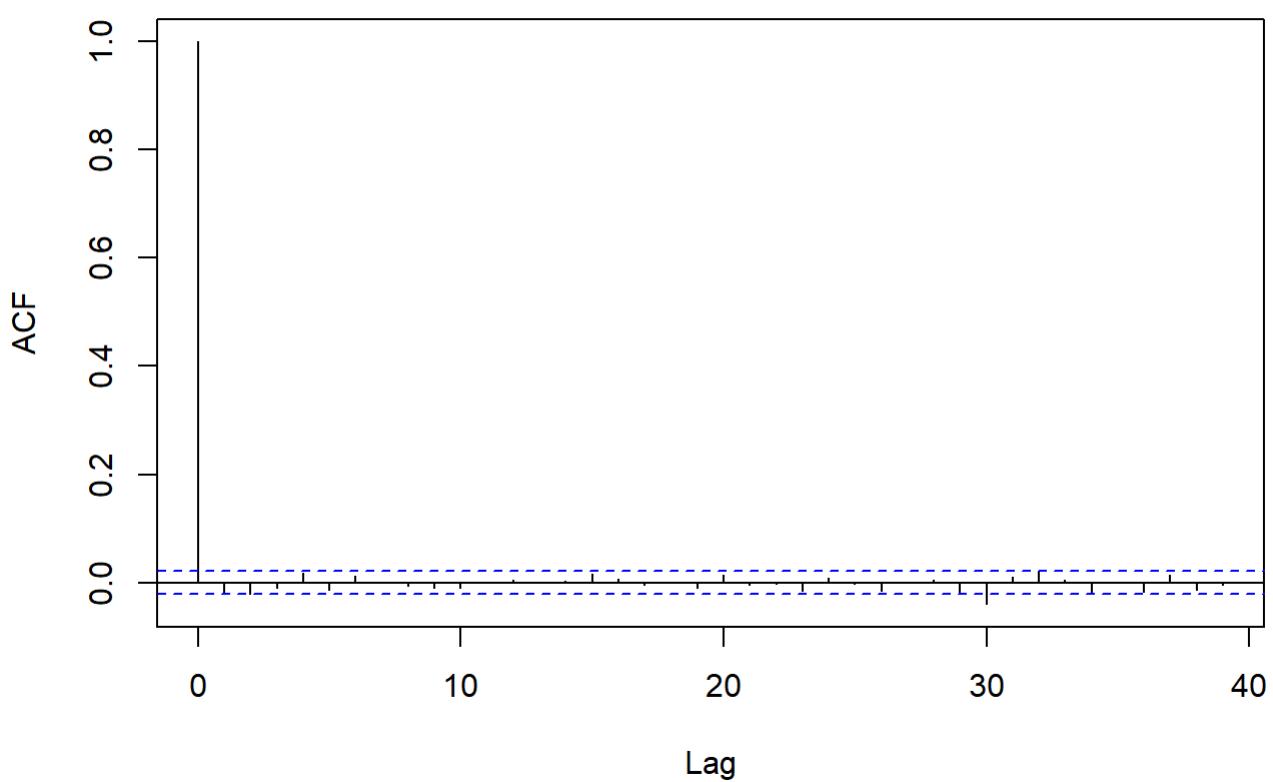
**ACF for Beta (Observation 1 )****ACF for Beta (Observation 2 )**

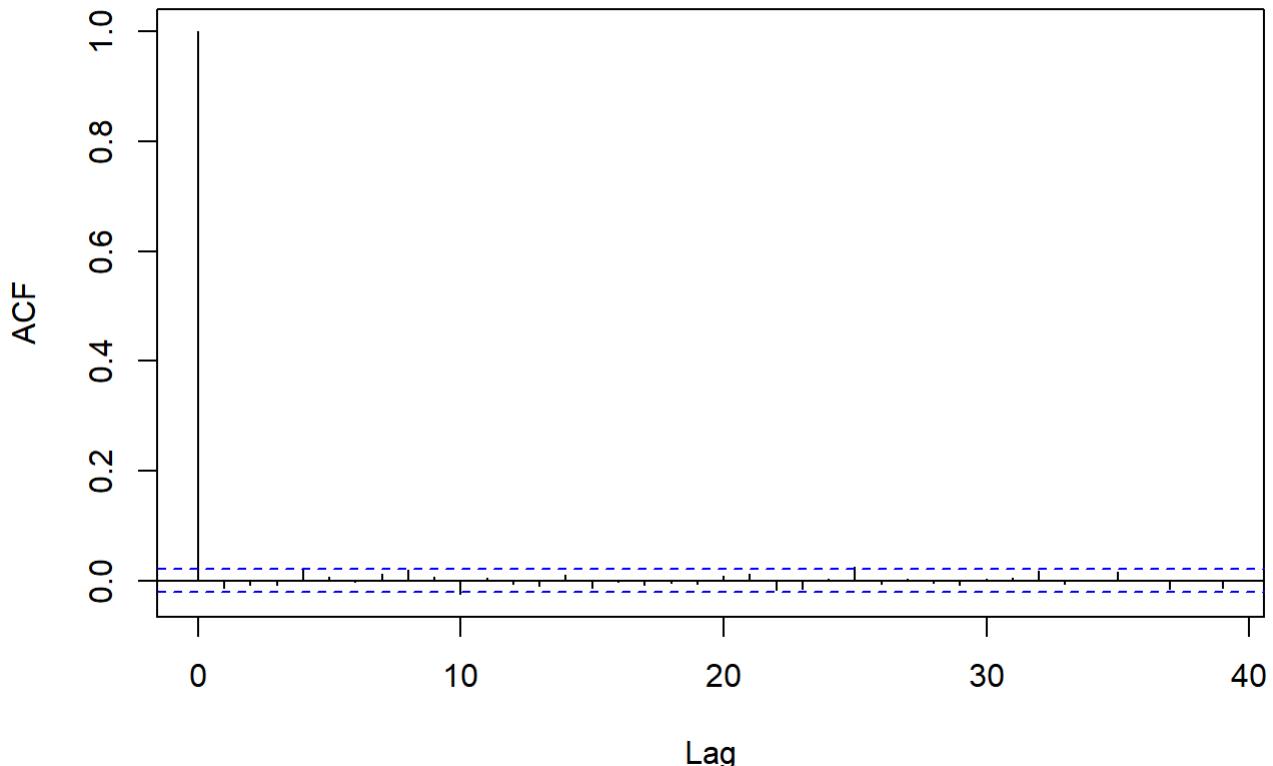
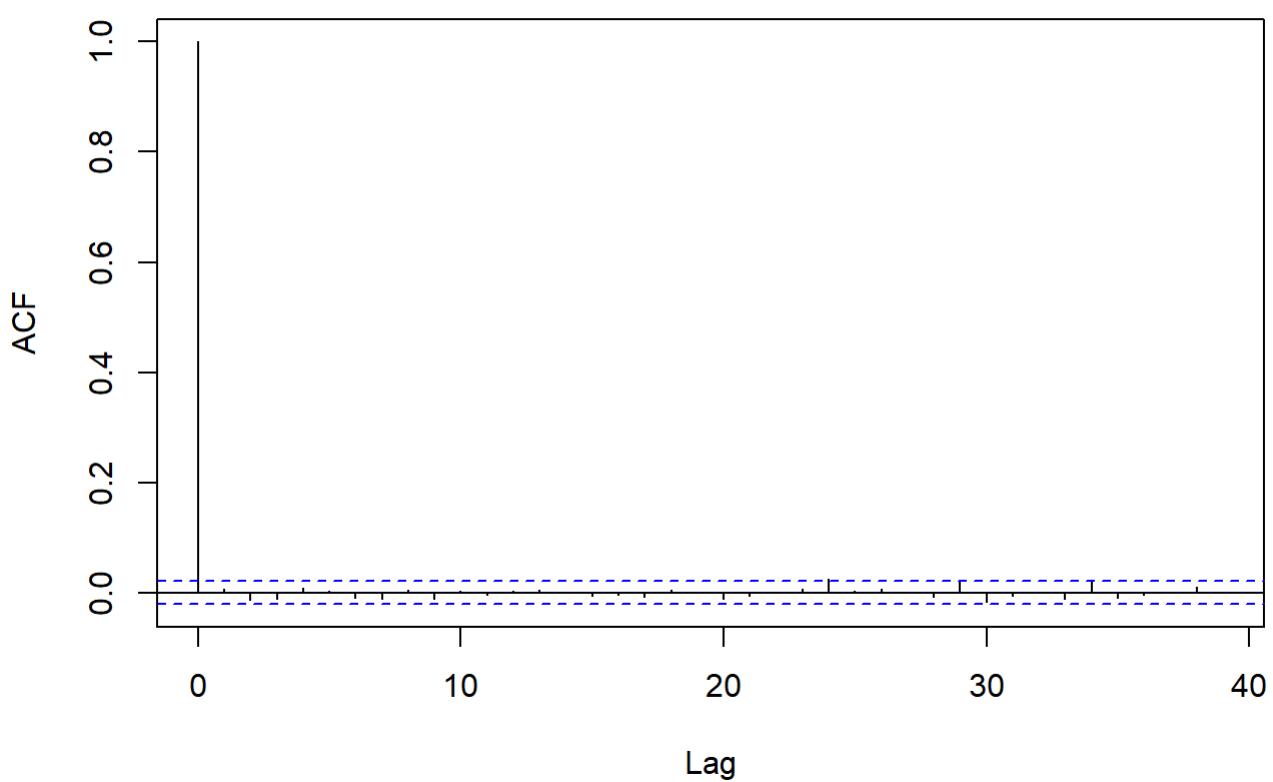
**ACF for Beta (Observation 3 )****ACF for Beta (Observation 4 )**

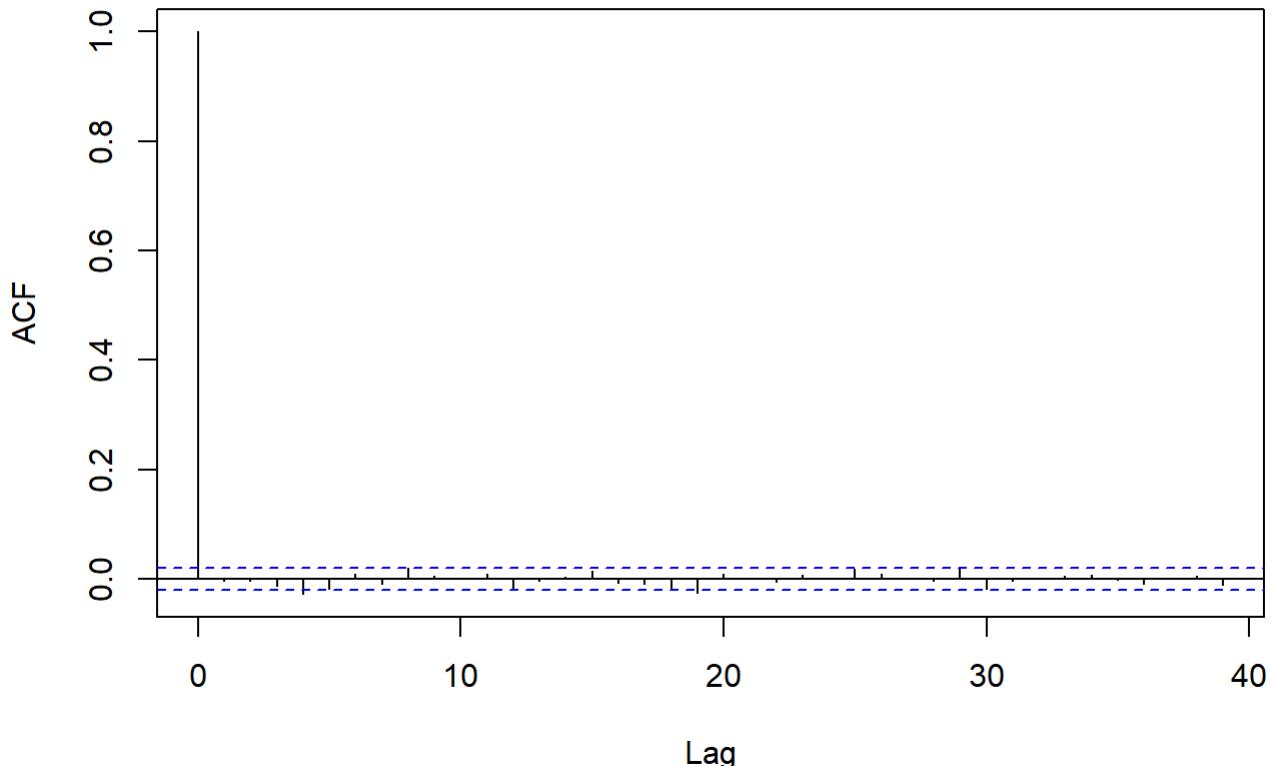
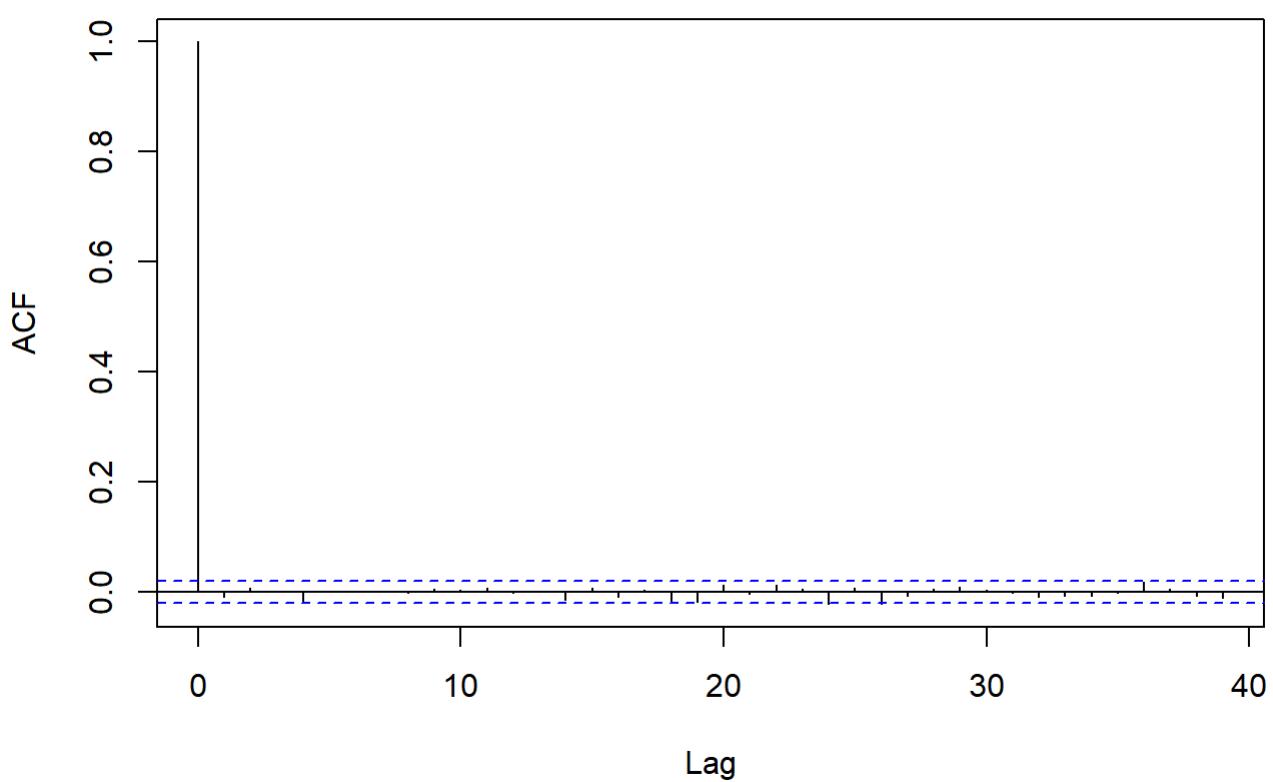
**ACF for Beta (Observation 5 )****ACF for Beta (Observation 6 )**

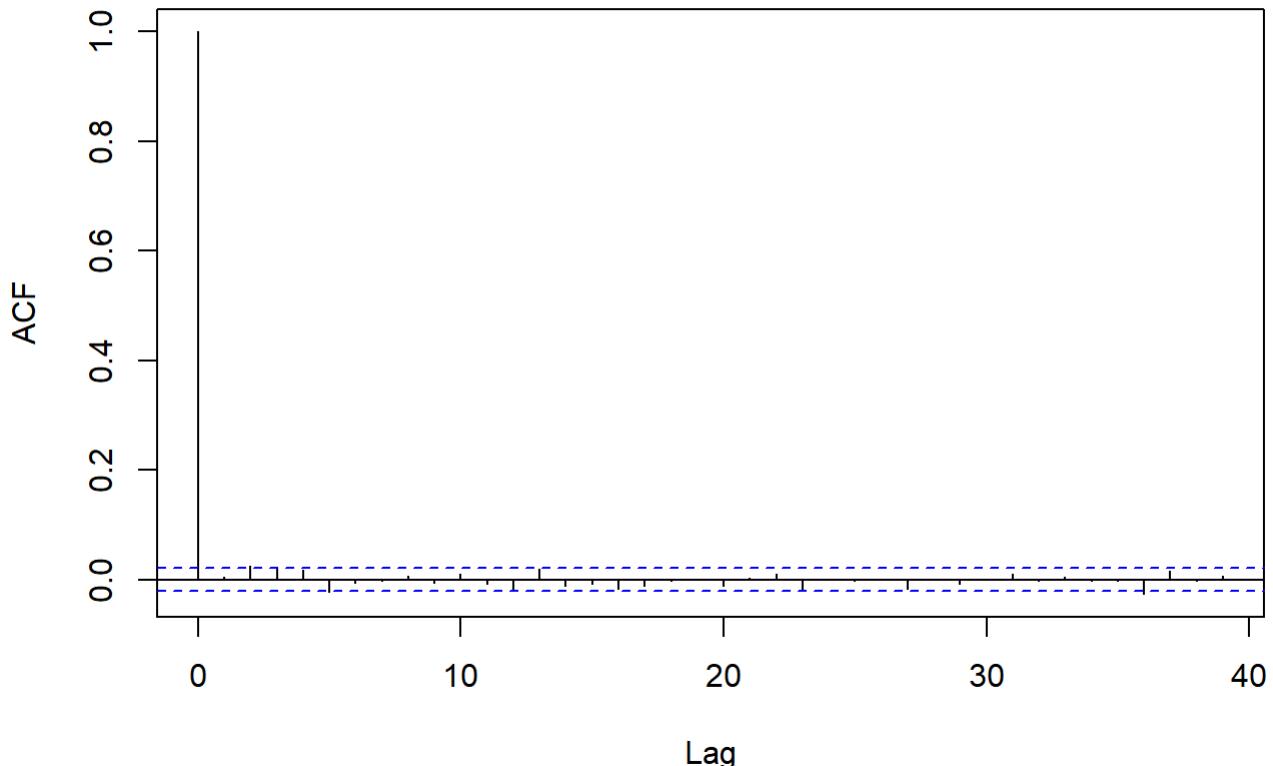
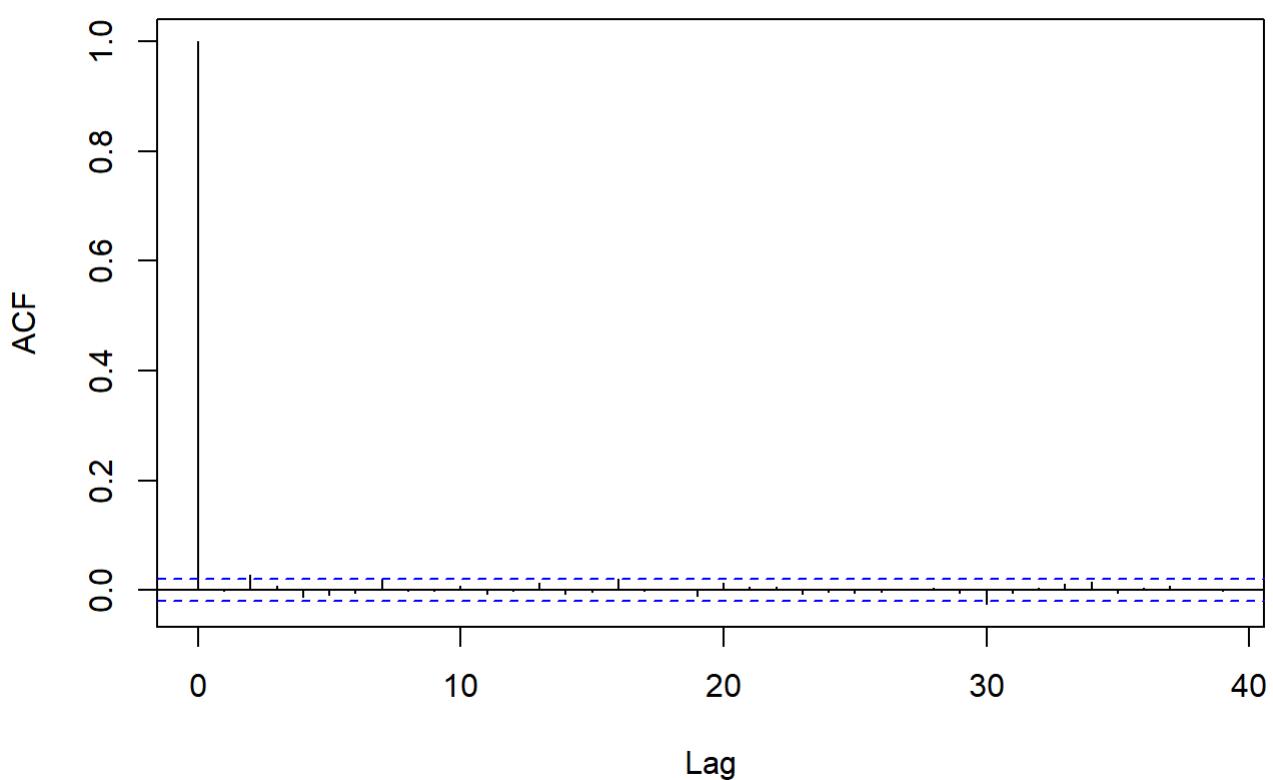
**ACF for Beta (Observation 7 )****ACF for Beta (Observation 8 )**

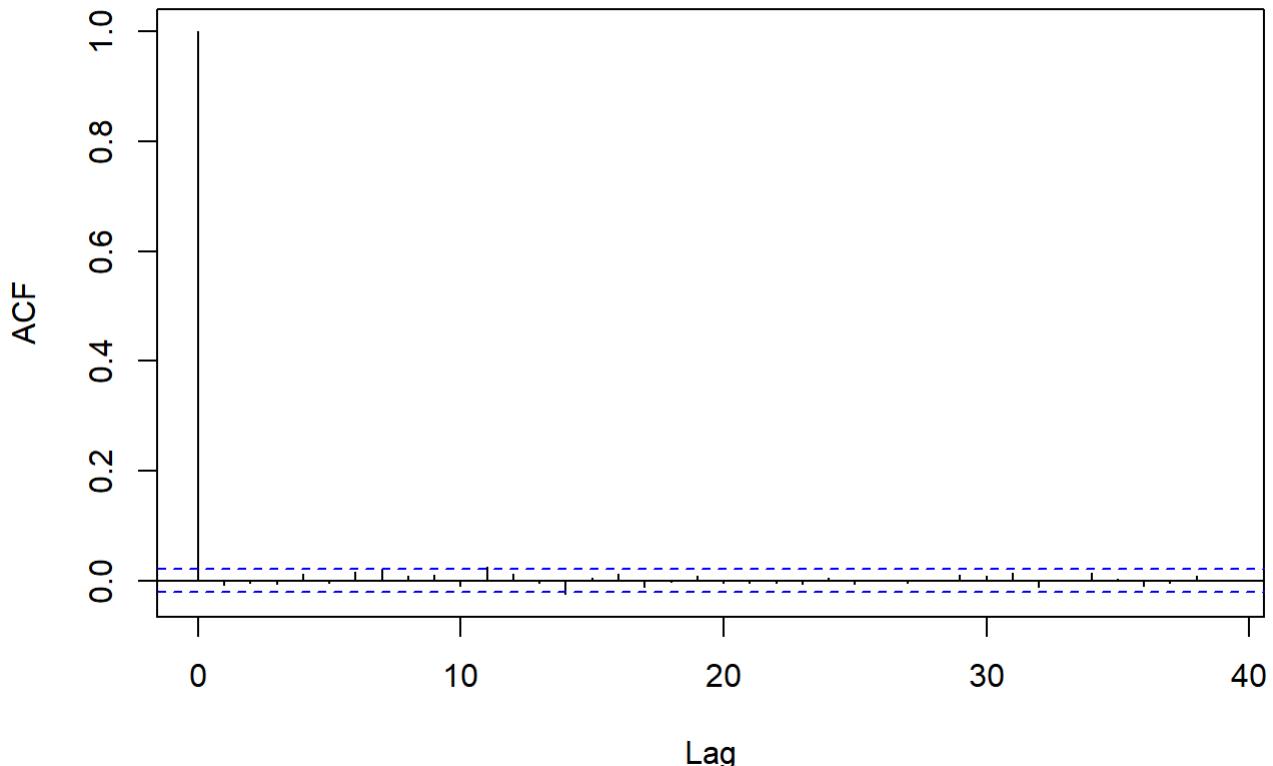
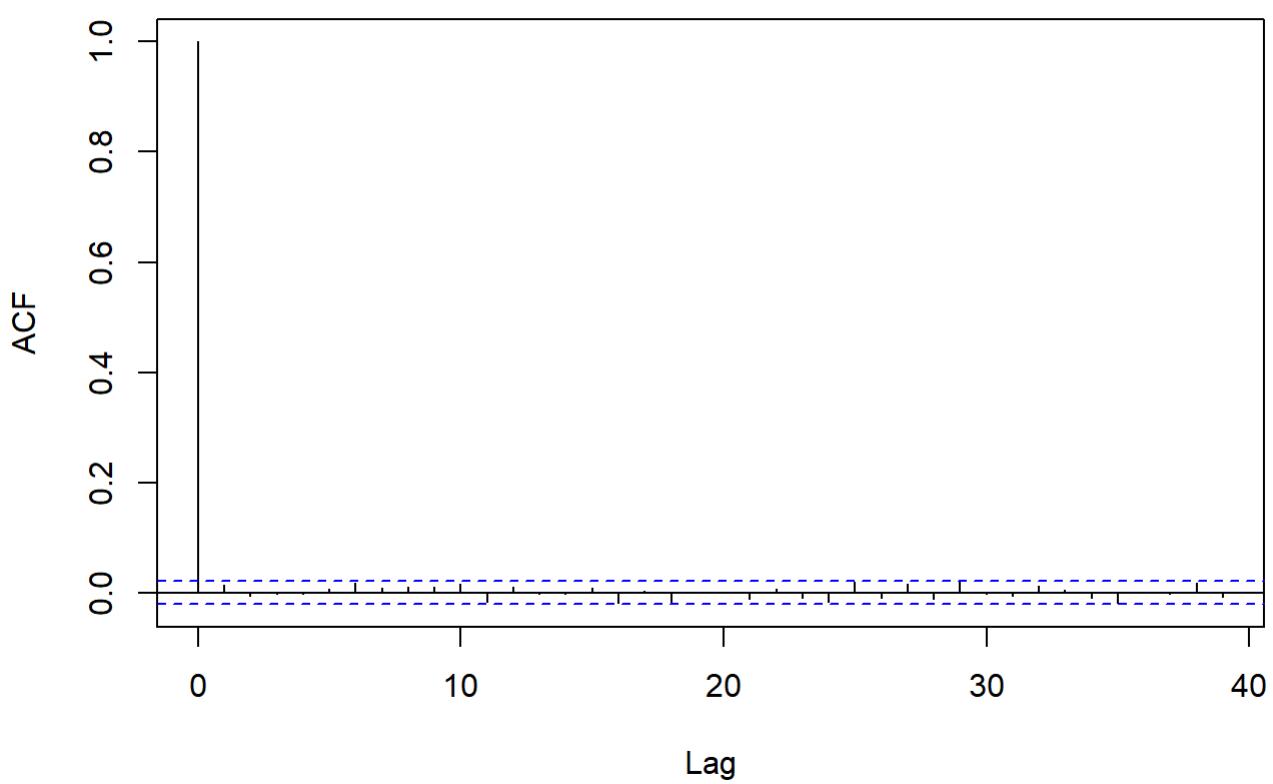
**ACF for Beta (Observation 9 )****ACF for Beta (Observation 10 )**

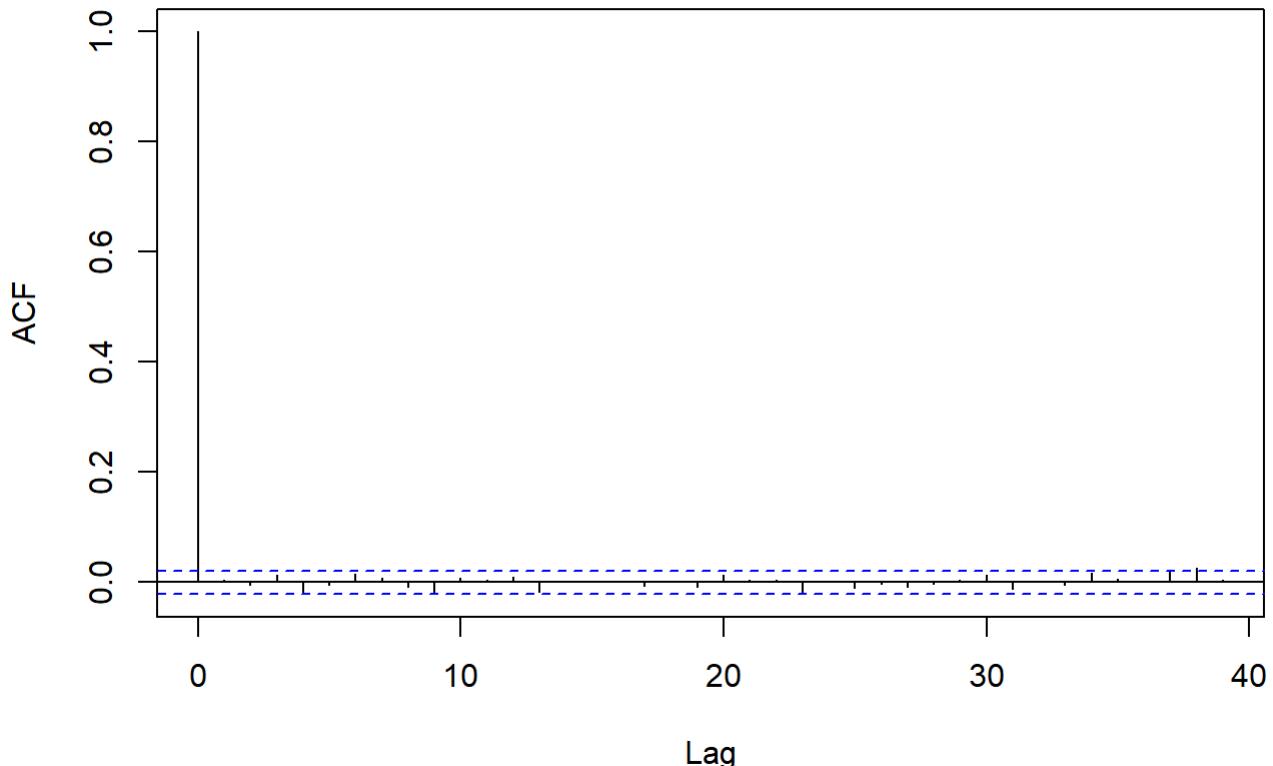
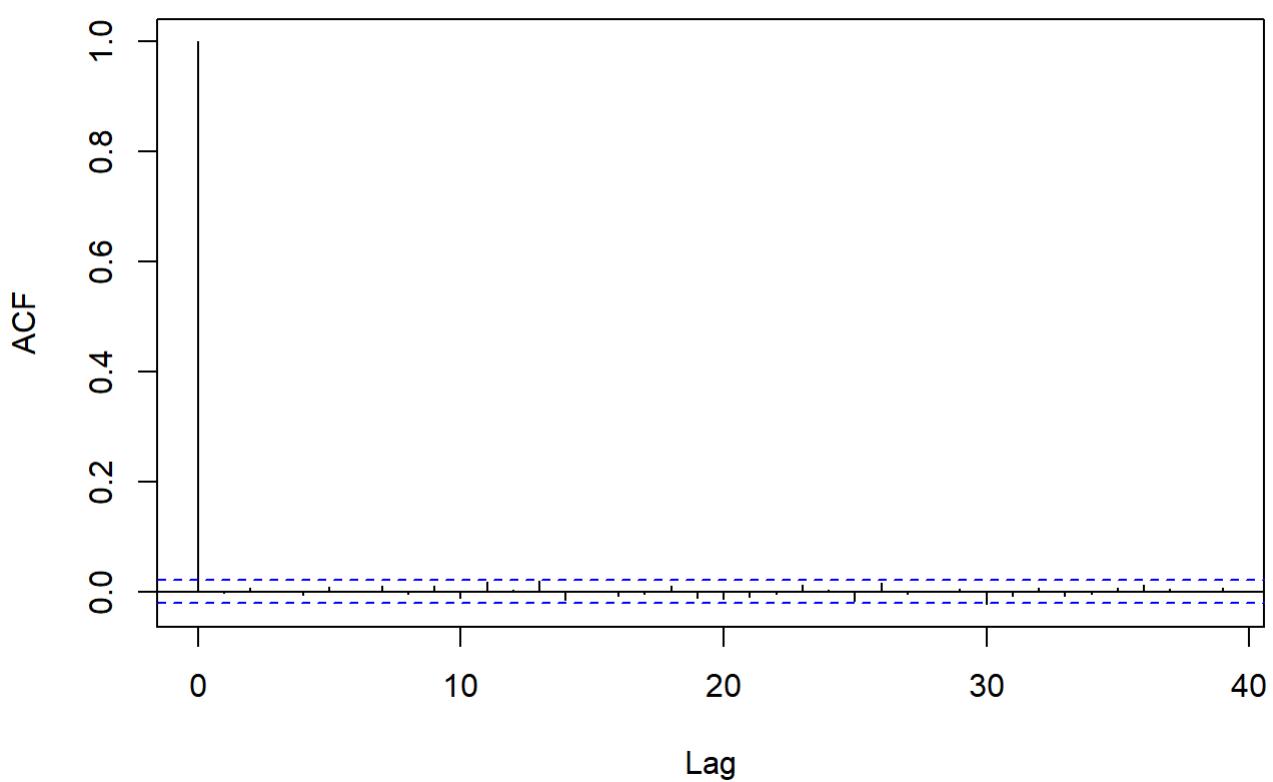
**ACF for Beta (Observation 11 )****ACF for Beta (Observation 12 )**

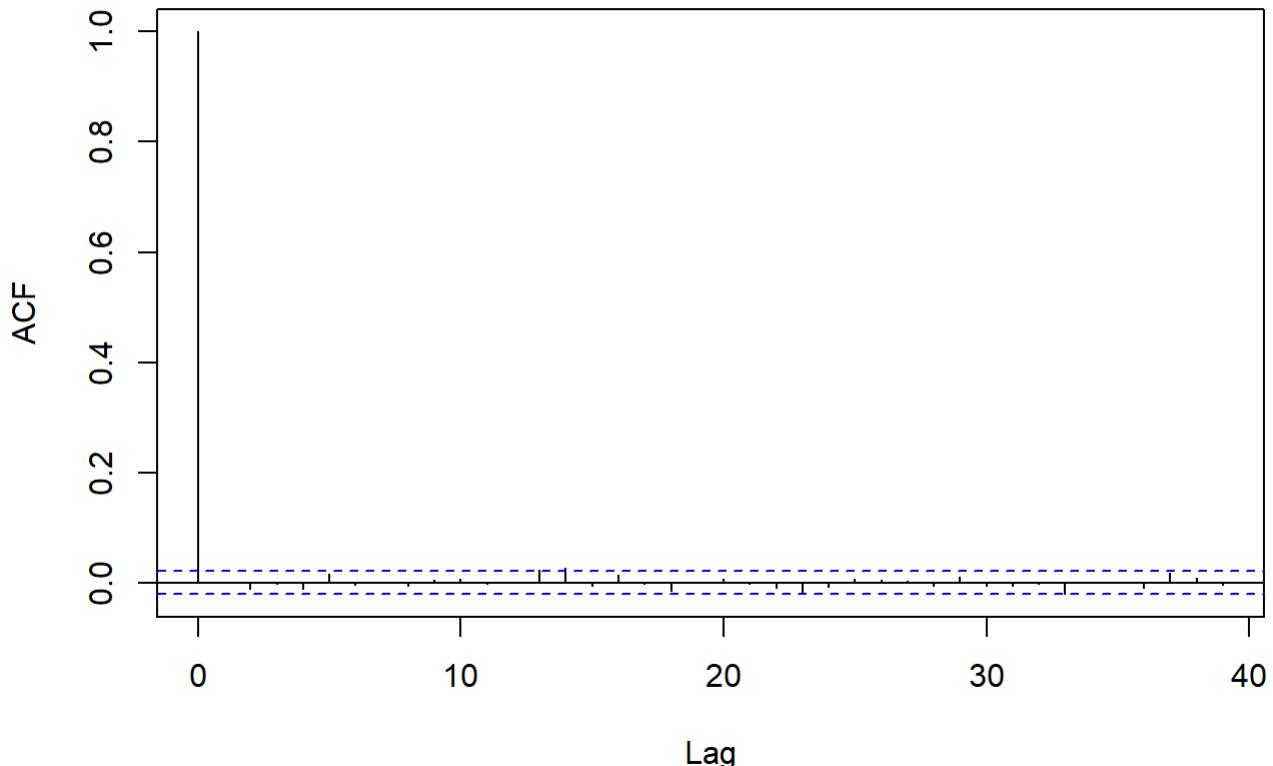
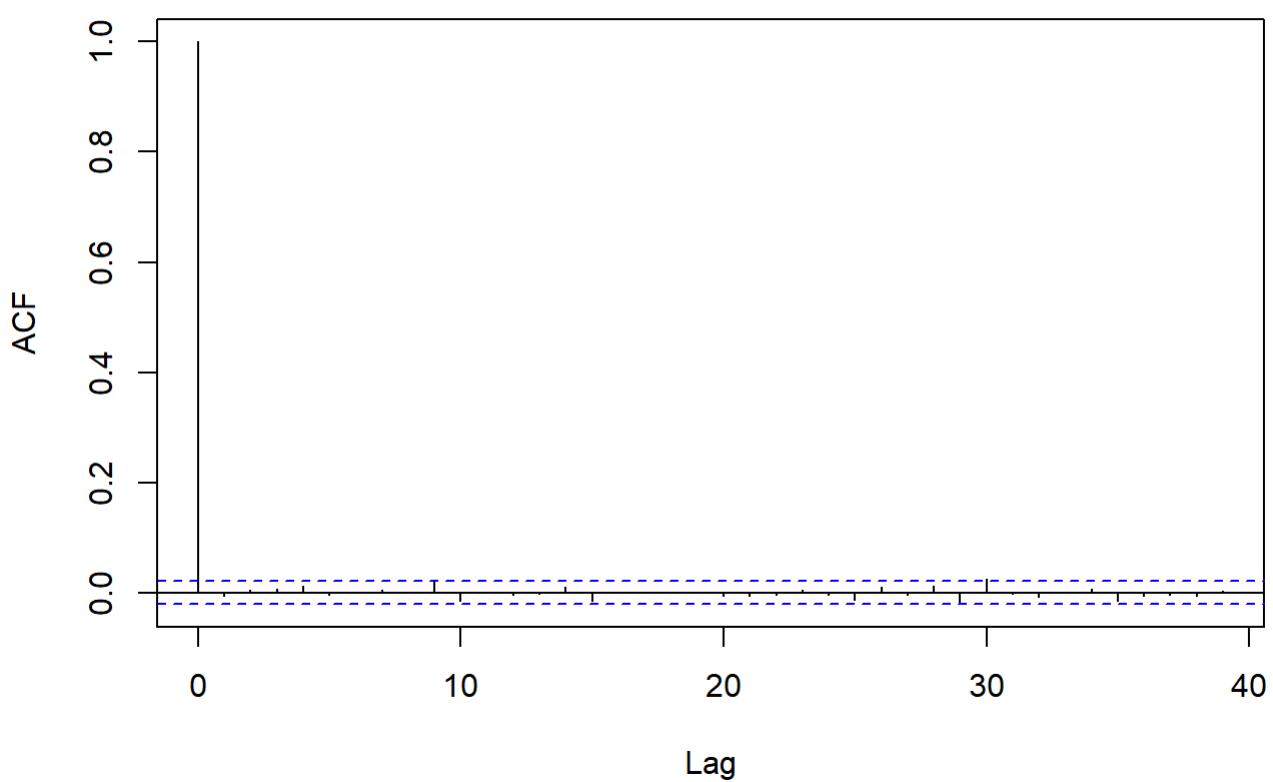
**ACF for Beta (Observation 13 )****ACF for Beta (Observation 14 )**

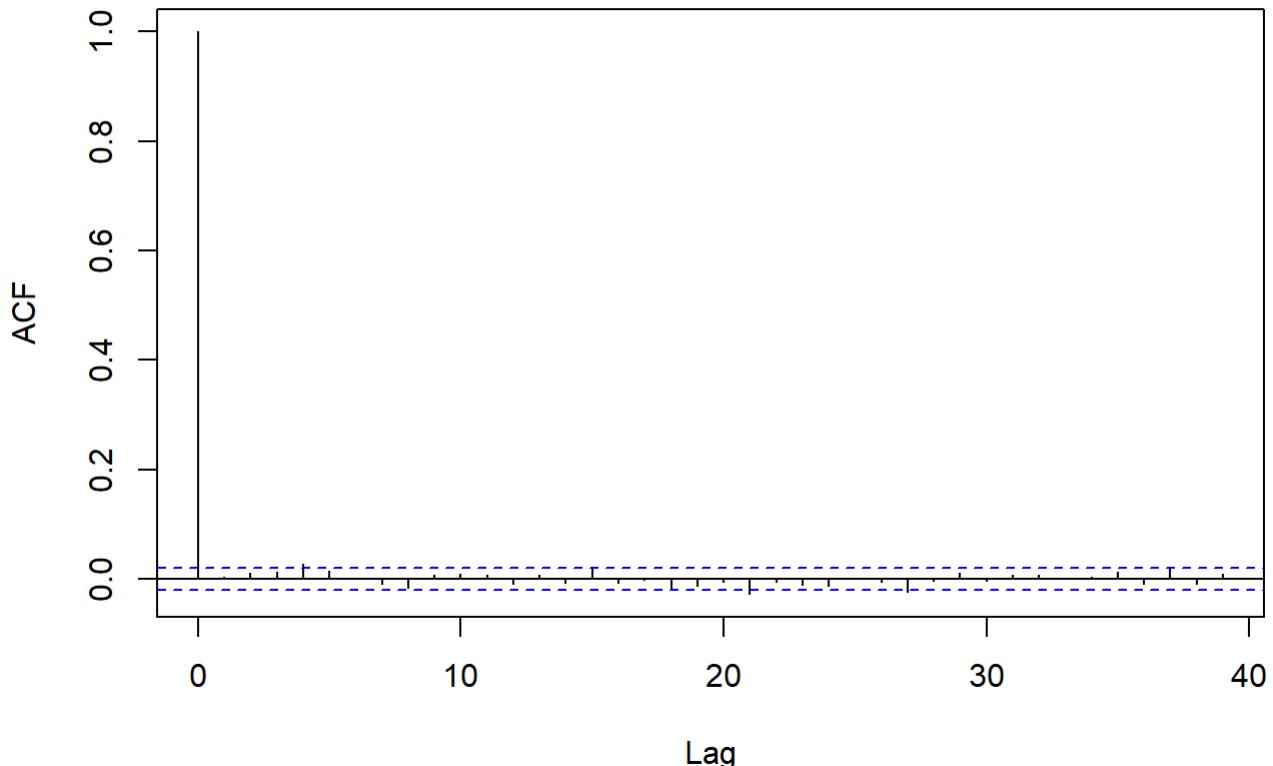
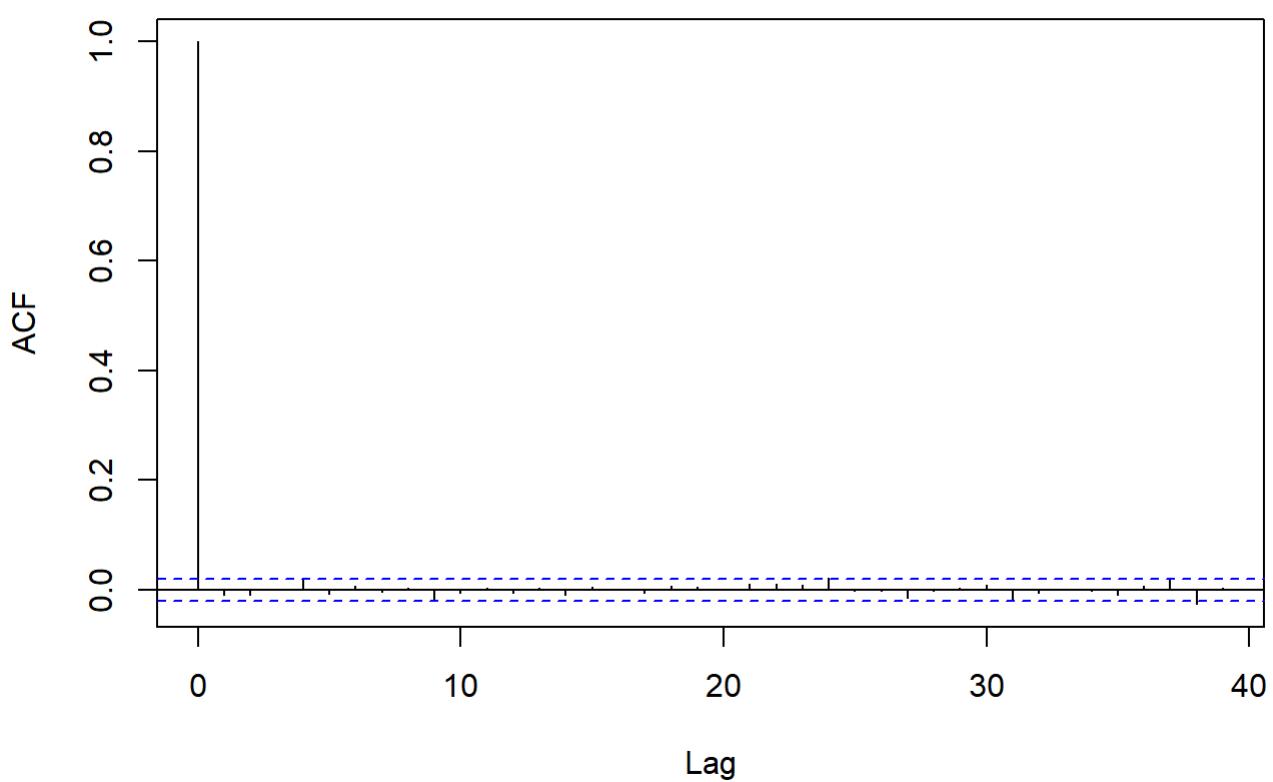
**ACF for Beta (Observation 15 )****ACF for Beta (Observation 16 )**

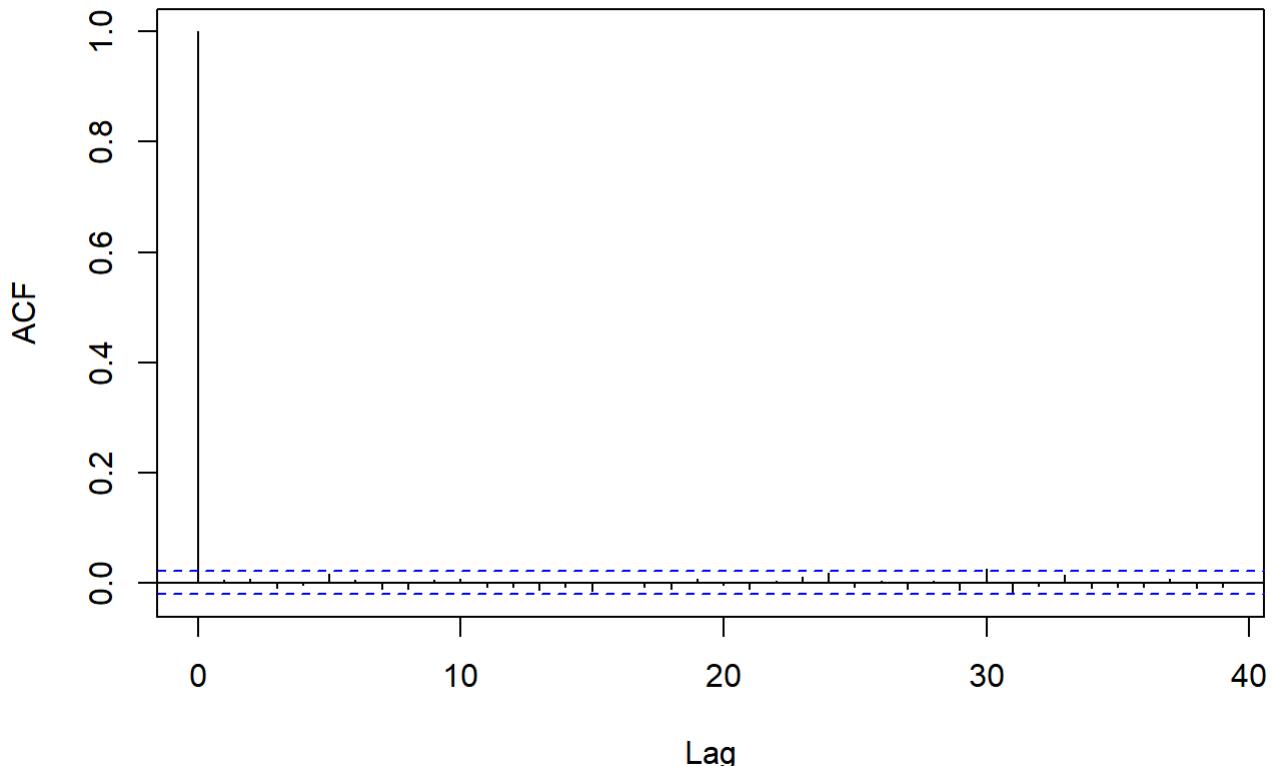
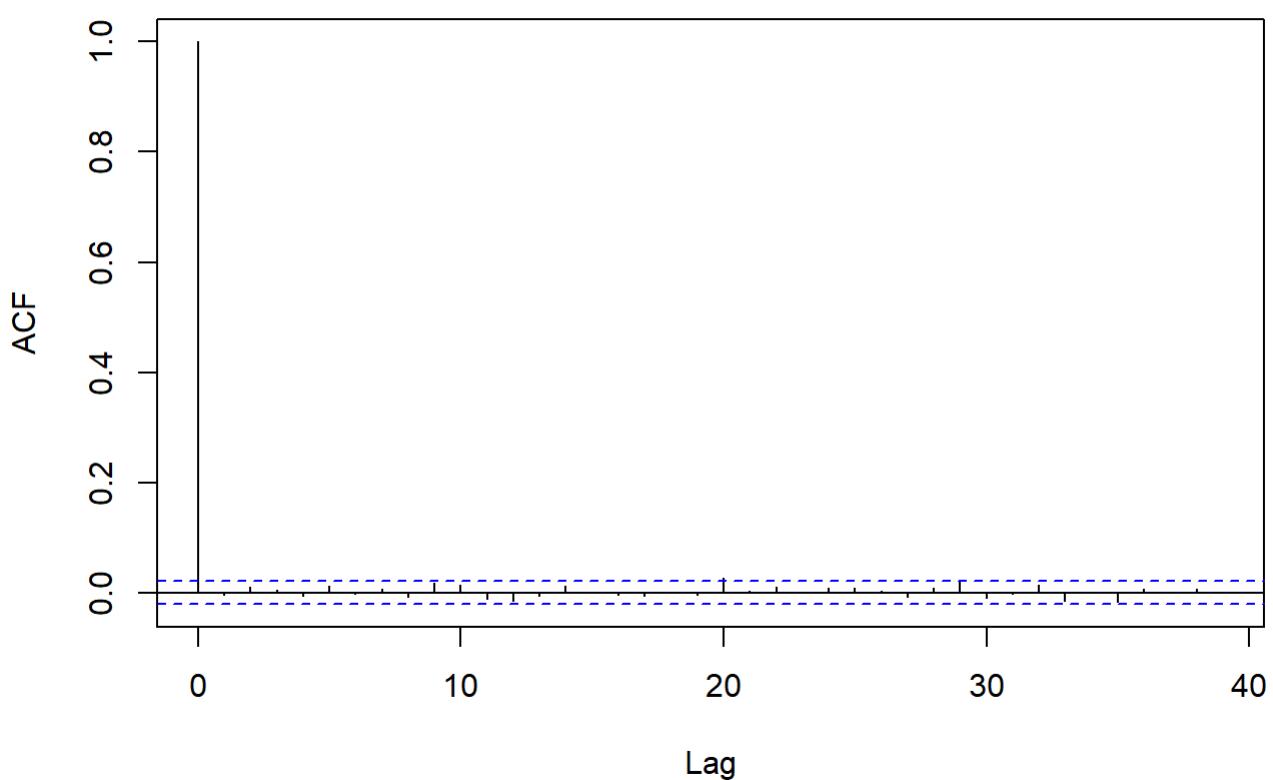
**ACF for Beta (Observation 17 )****ACF for Beta (Observation 18 )**

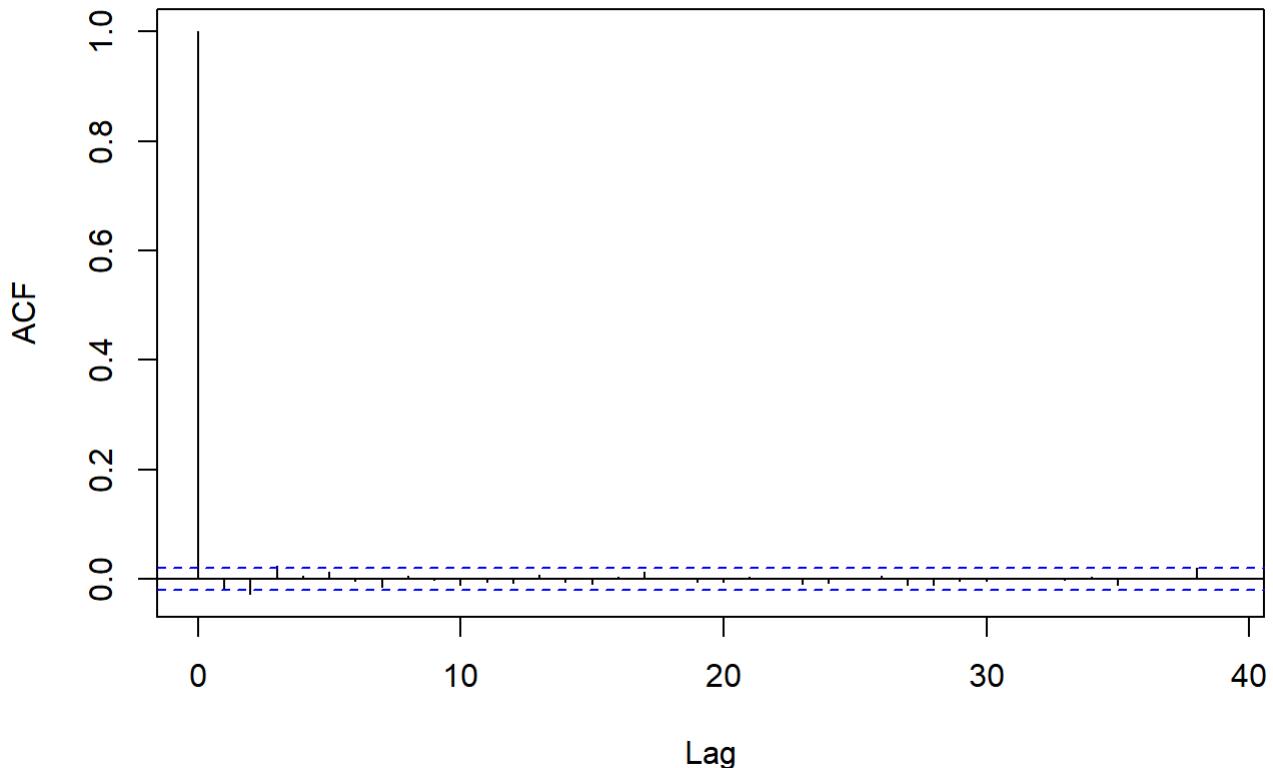
**ACF for Beta (Observation 19 )****ACF for Beta (Observation 20 )**

**ACF for Beta (Observation 21 )****ACF for Beta (Observation 22 )**

**ACF for Beta (Observation 23 )****ACF for Beta (Observation 24 )**

**ACF for Beta (Observation 25 )****ACF for Beta (Observation 26 )**

**ACF for Beta (Observation 27 )****ACF for Beta (Observation 28 )**

**ACF for Beta (Observation 29 )****ACF for Beta (Observation 30 )**