Lab 3 report

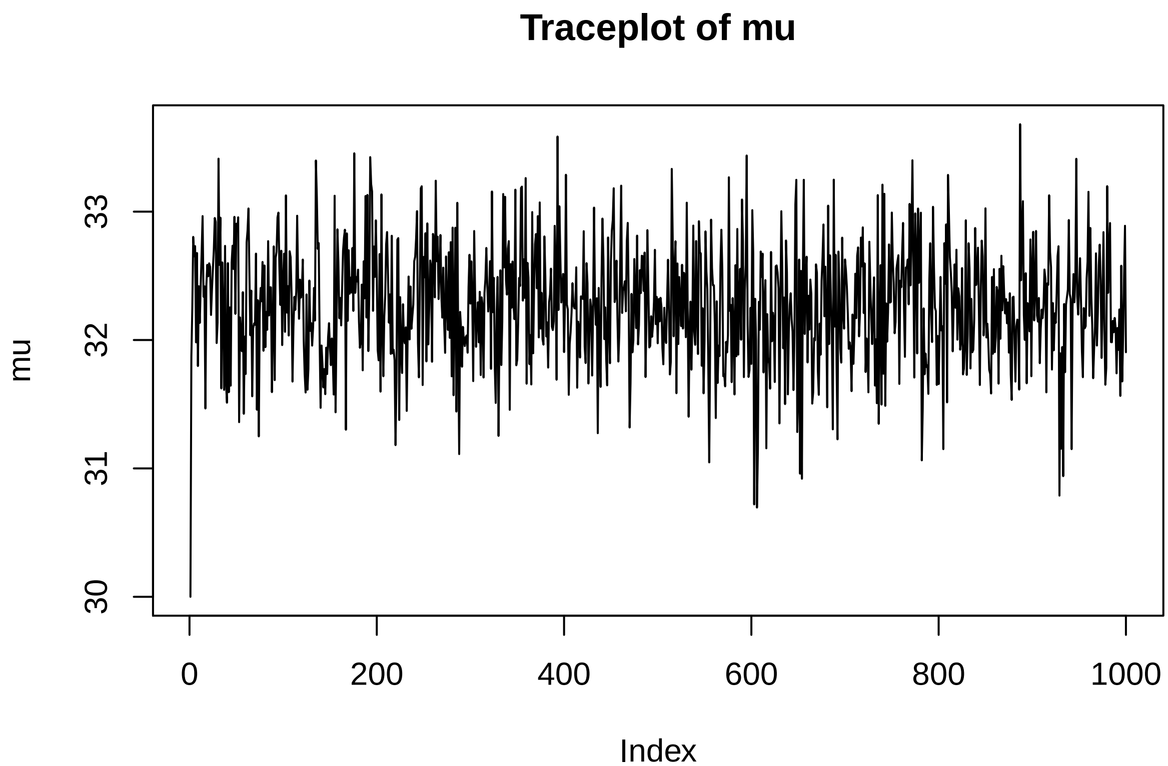
Andrea Bruzzone, Thomas Zhang

Assignment 1

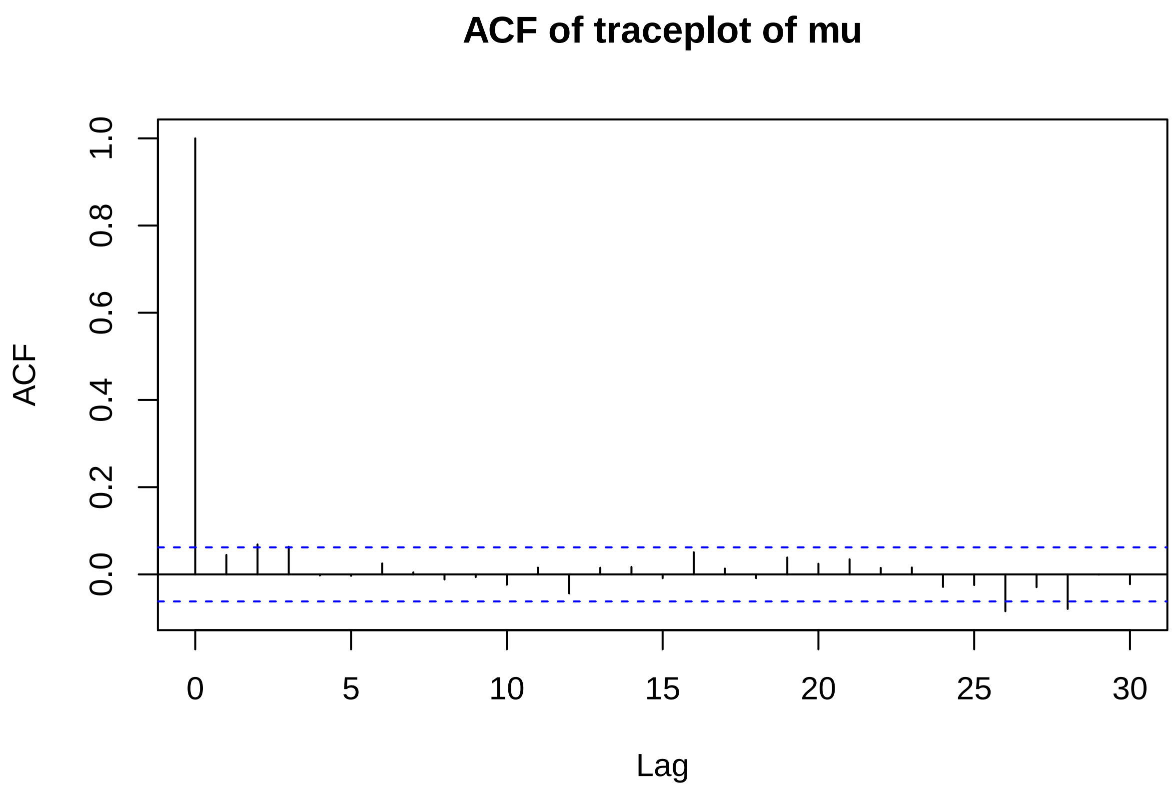
1.a

Assuming that the daily precipitation are independent normally distributed with parameters μ and σ^2 we use a Gibbs sampler to simulate from the joint posterior $p(\mu, \sigma^2 | y_1, .., y_n)$

Let see if the sampler converges, starting with the traceplot of μ:



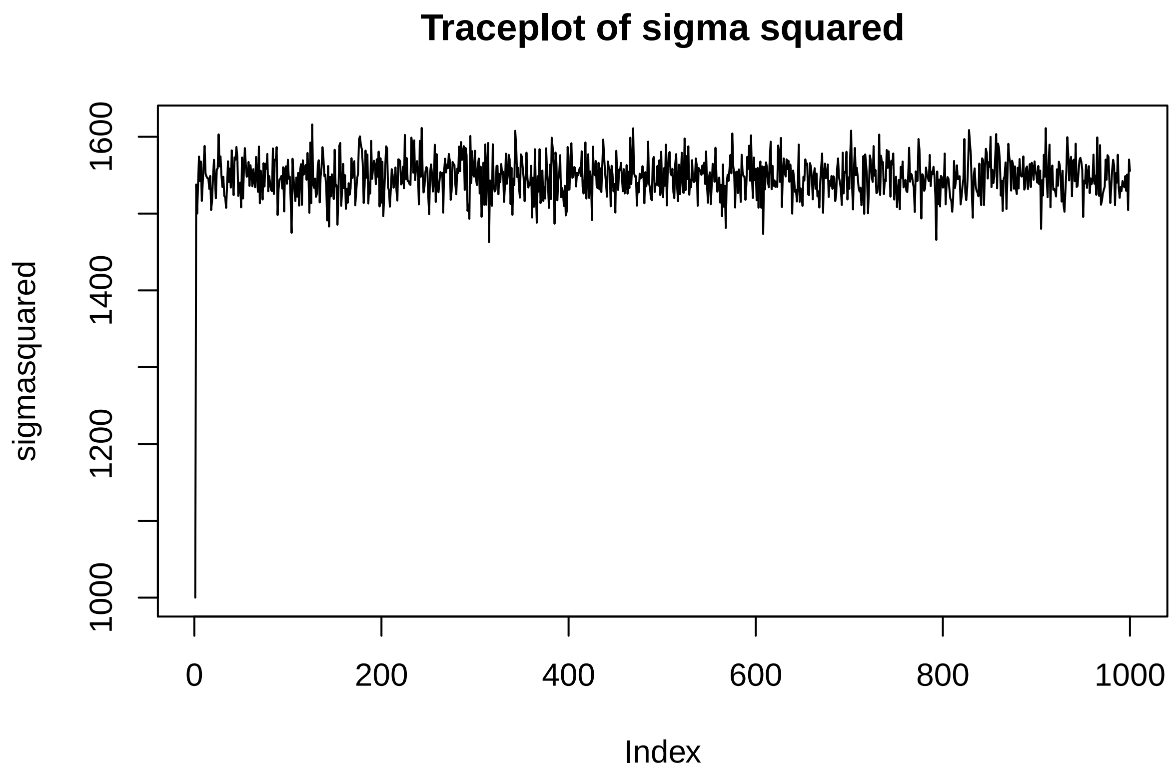
From the traceplot μ seems to converge to a value between 32 and 33. For further analysis we compute the ACF and the inefficiency factor:



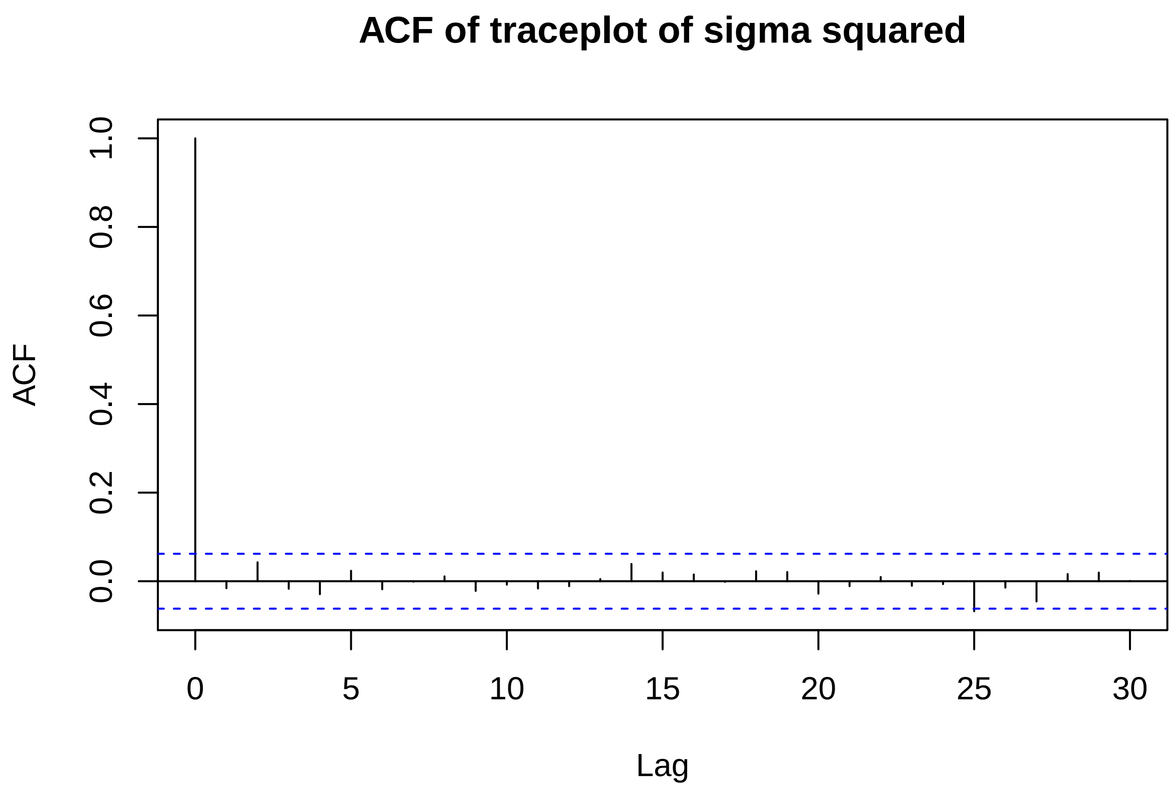

```
## [1] 1.135625
```

They both shows that the sampler is very efficience since the draws seem to be independent.

The traceplot of σ^2:



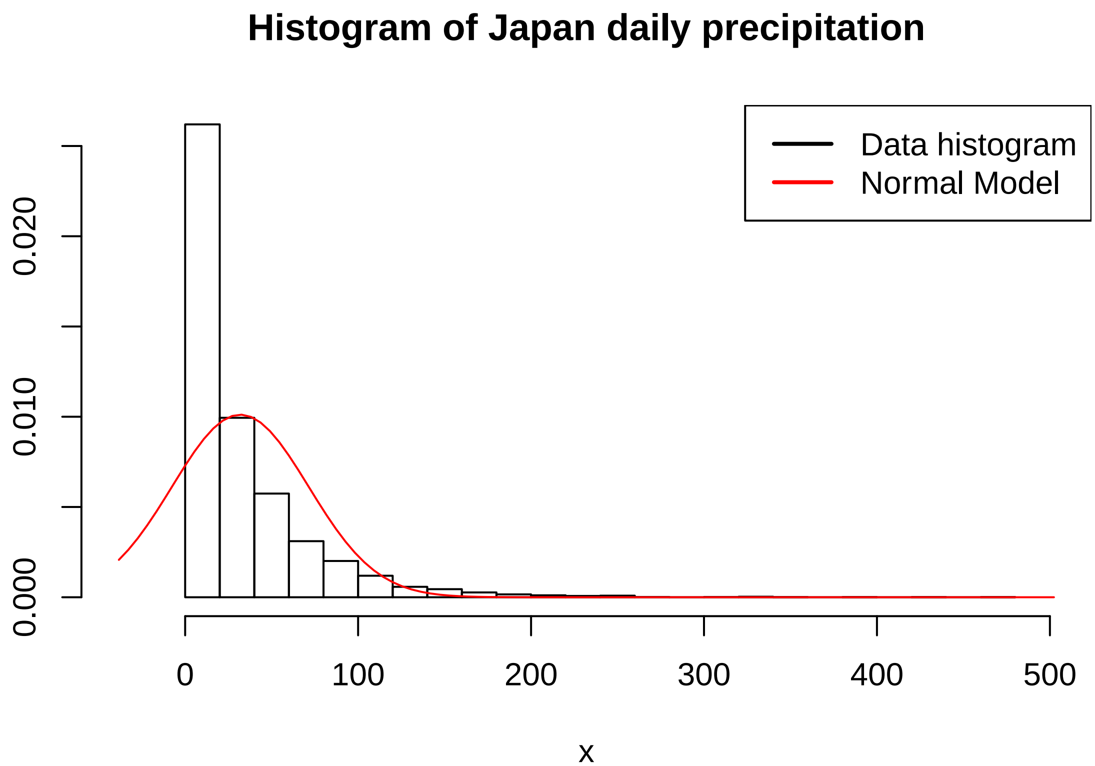
Also σ^2 seems to converge, to a value around 1550. The ACF and inefficiency factor:




```
## [1] 0.8412501
```

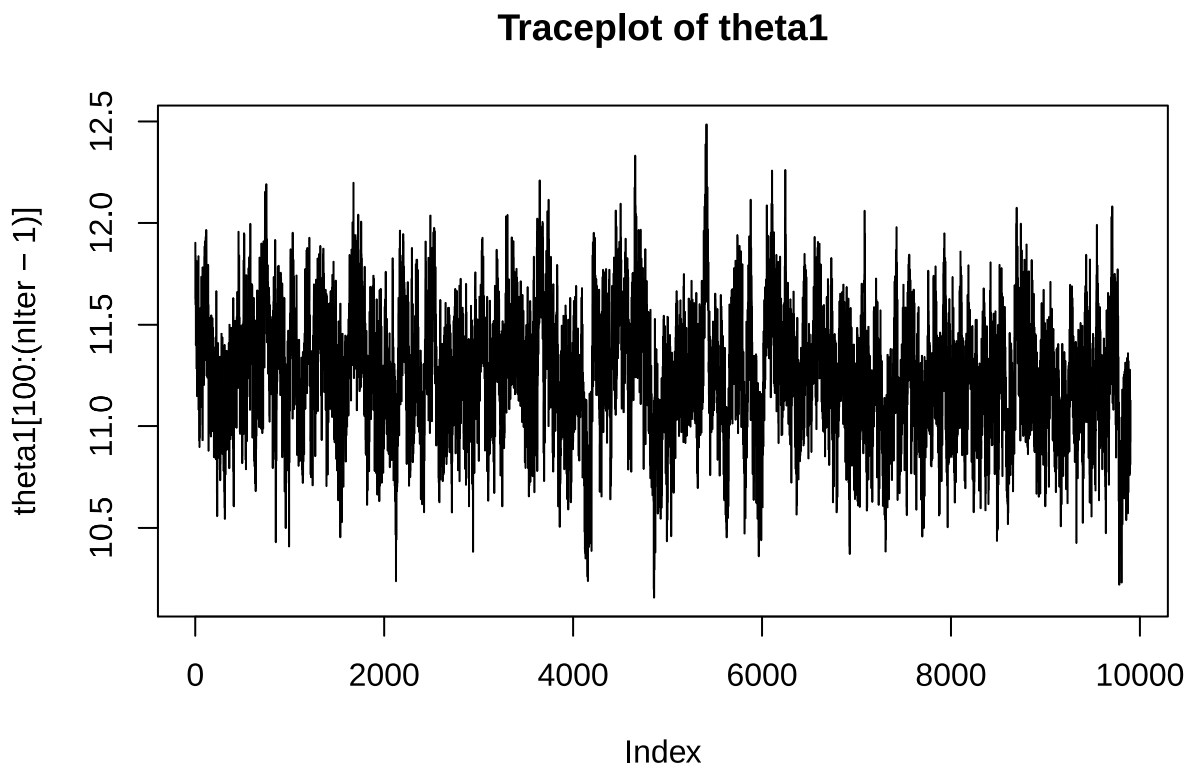
As for μ, we can say that the draws seem to be independent. In general the Gibbs sampler is very good.

In this final plot we report the histogram of the data together with the normal distribution using the posterior mean values for the two parameters:

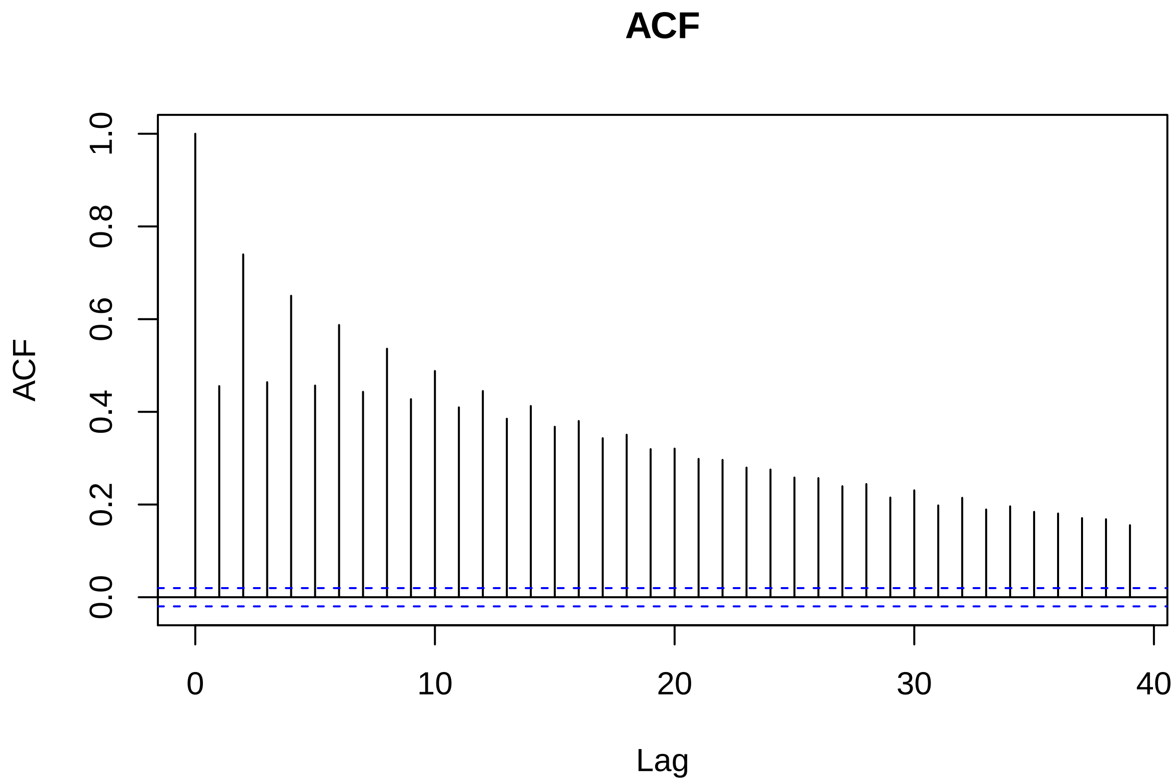


1.b

Using mixture normal model, here we present the plot for the last iteration with the two components. Let check the convergence and the efficiency of all the parameters, we decide to choose 100 as burnin. Traceplot, ACF and inefficiency factor of μ_1:



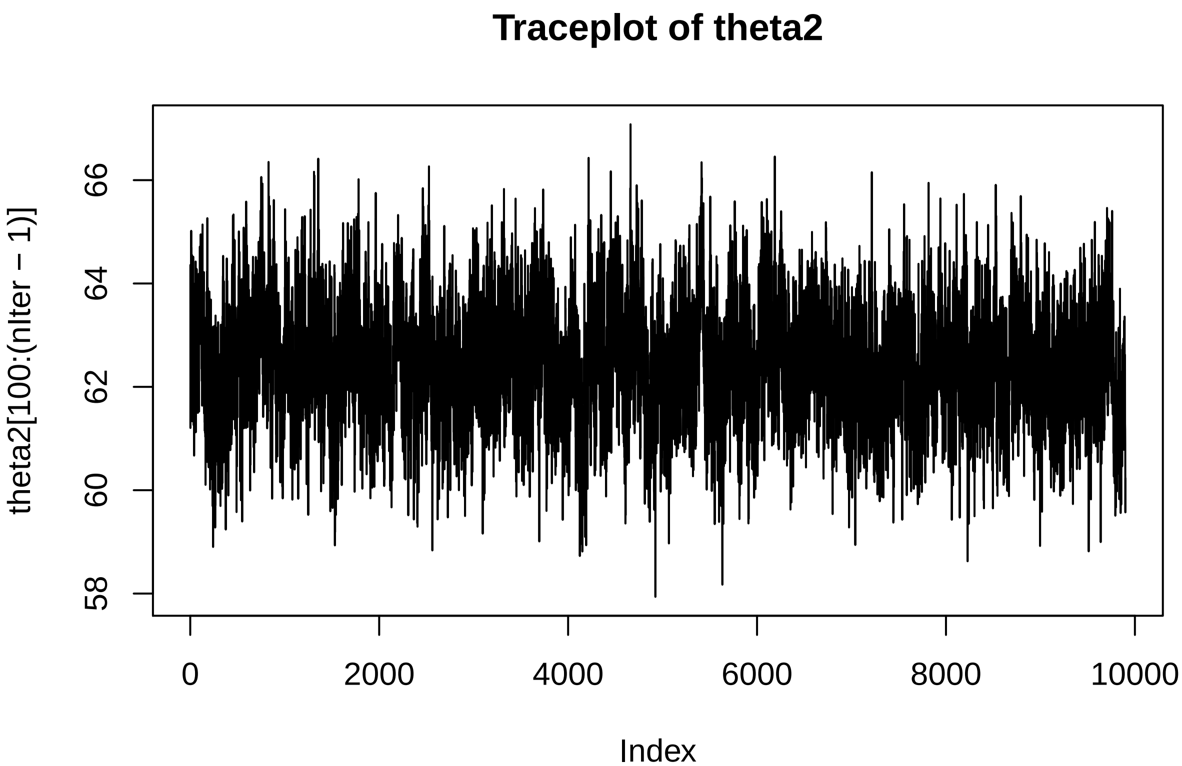
Traceplot of theta1
theta1[100:(nIter − 1)]
12.5
12.0
11.5
11.0
10.5
0
2000
4000
6000
8000
10000
Index

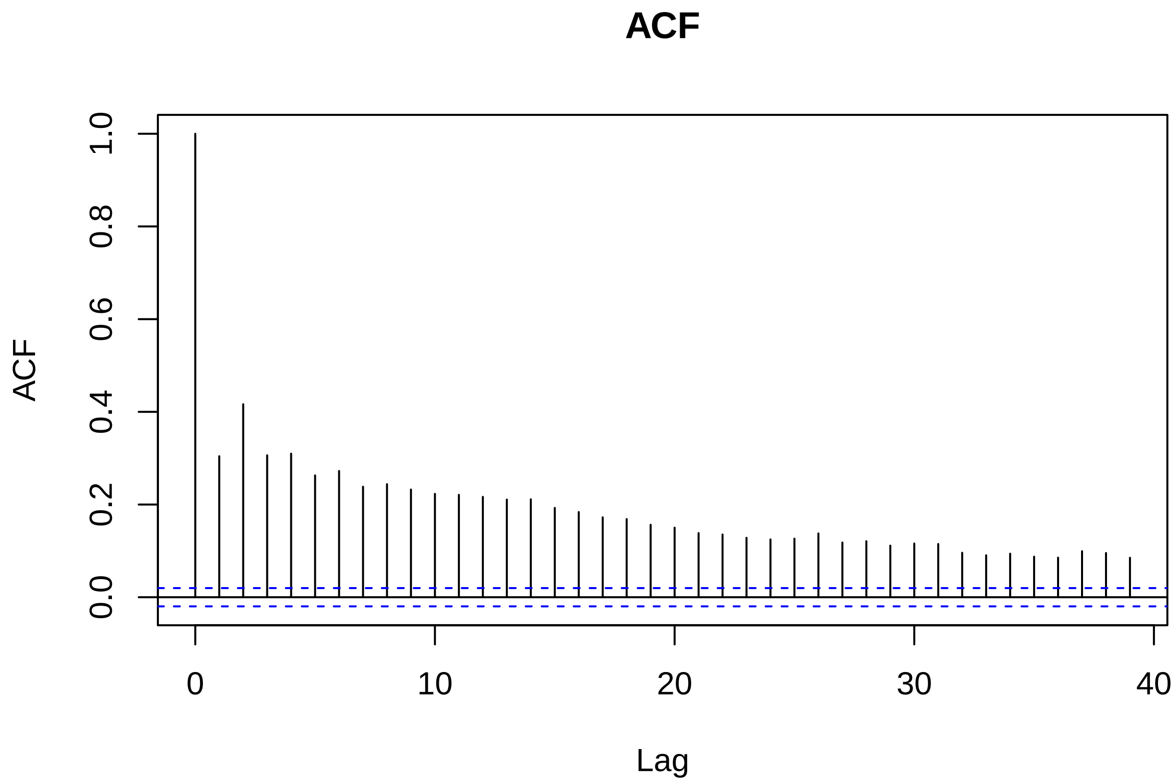



```
## [1] "IF:  27.16"
```

μ_1 seems to converge to a value around 11 but the draws seem to be highly positively correlated, which leads to a high IF, which means that the number of equivalent independent draws is merely a fraction of the actual draws.

Traceplot, ACF and inefficiency factor of μ_2:

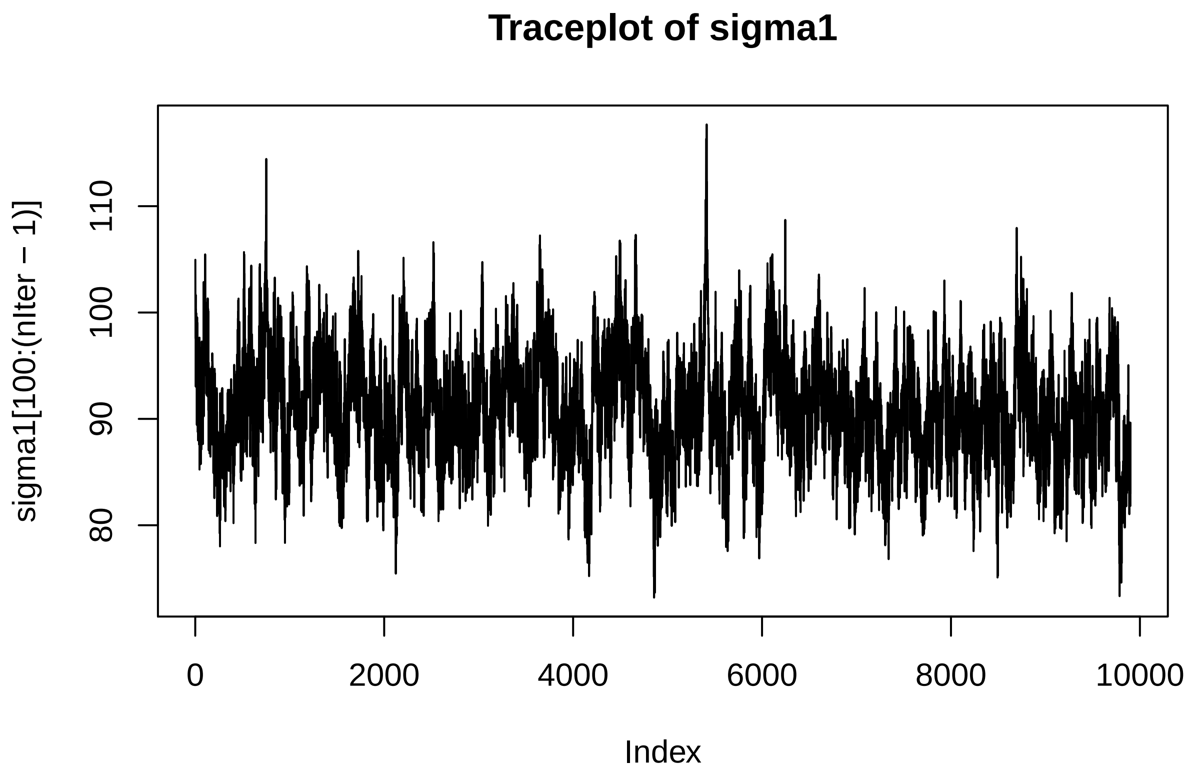

Traceplot of theta2
theta2[100:(nIter − 1)]
58
60
62
64
66
0
2000
4000
6000
8000
10000
Index



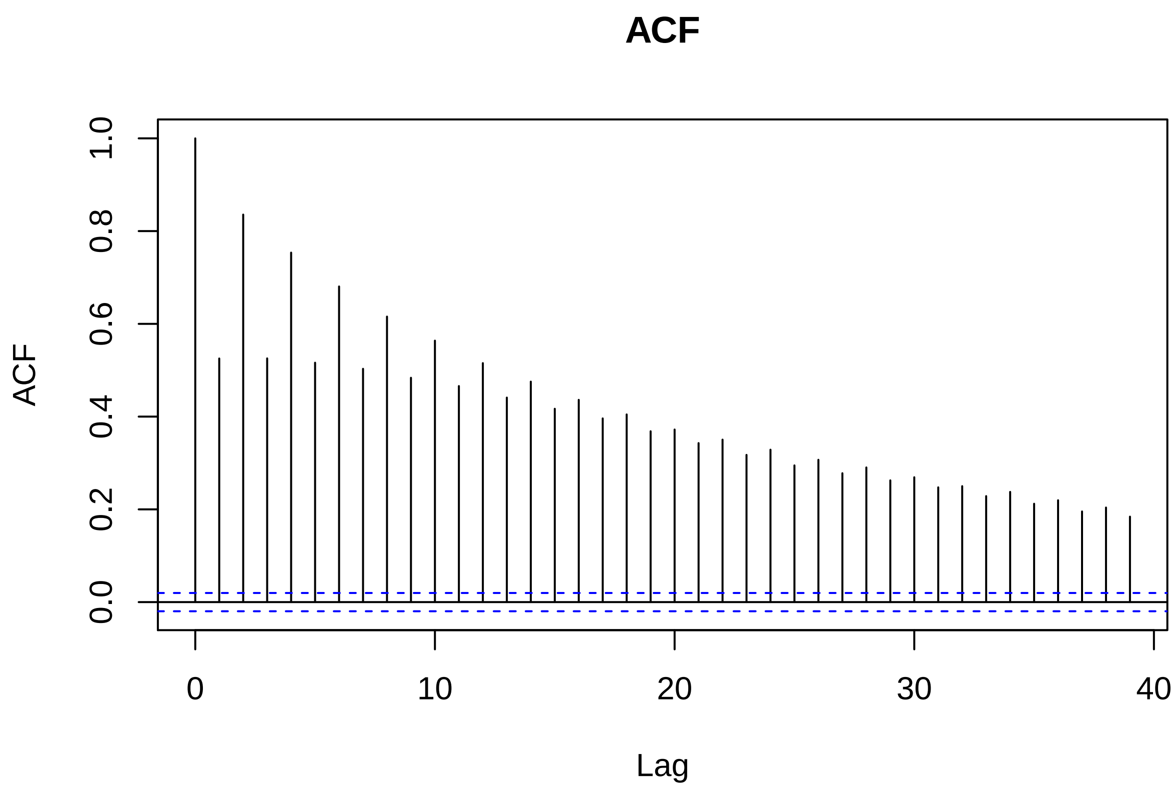

```
## [1] "IF:  14.43"
```

μ_2 seems to converge to a value around 61 but the draws seem to be positively correlated. The IF number is rather large here as well.

Traceplot, ACF and inefficiency factor of σ_1^2:



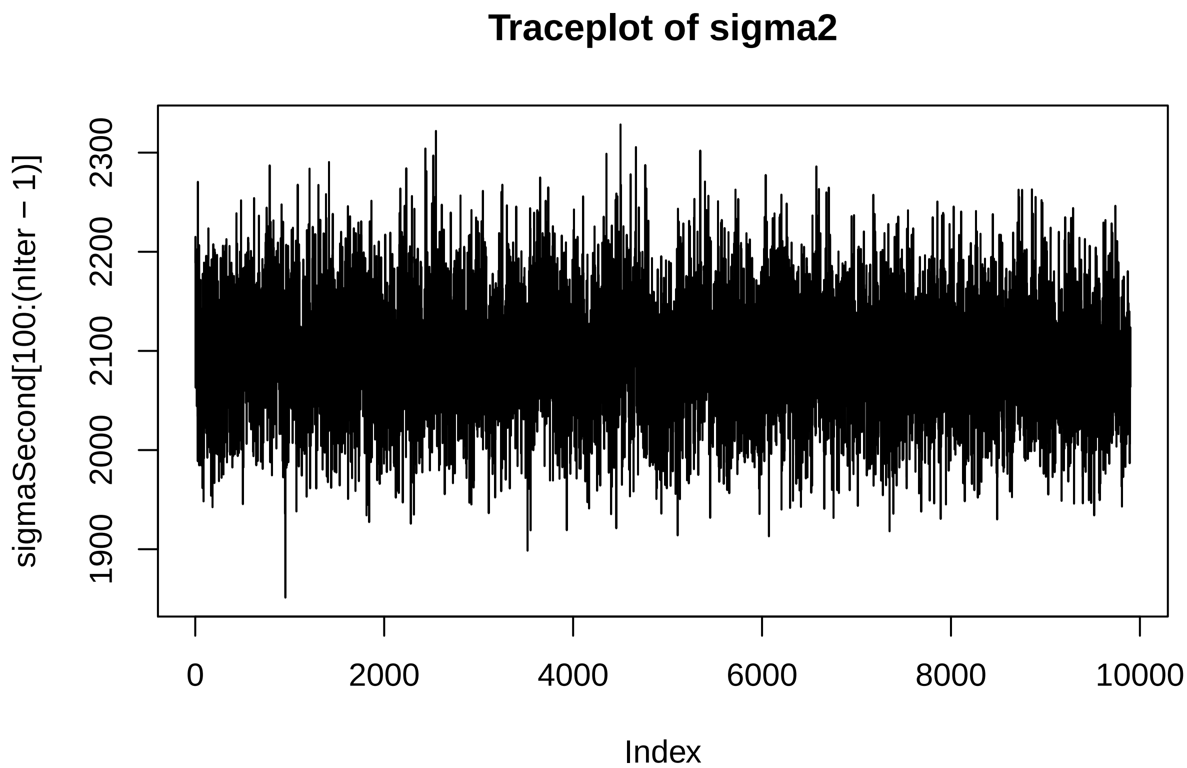
Traceplot of sigma1
sigma1[100:(nIter − 1)]
110
100
90
80
0
2000
4000
6000
8000
10000
Index

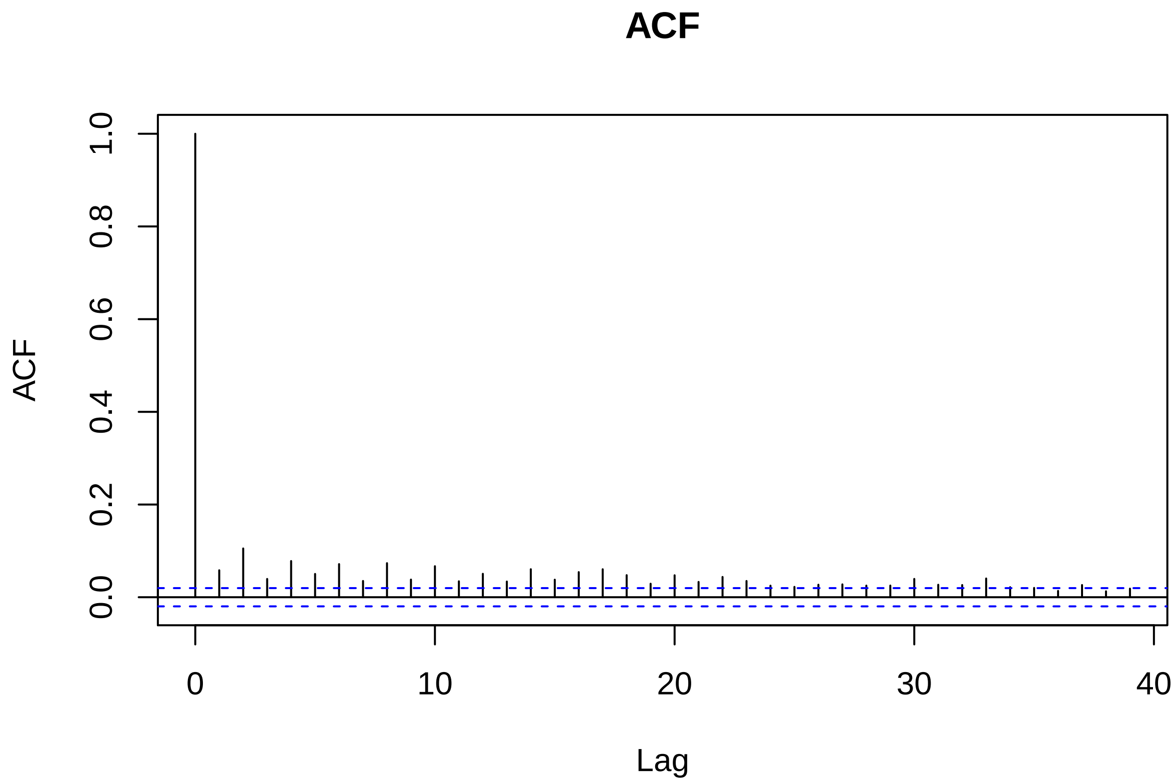



```
## [1] "IF:  31.26"
```

σ_1^2 converges to a value close to 86, the draws seem to be highly positively correlated, which leads to a high IF, which means that the number of equivalent independent draws is merely a fraction of the actual draws.

Traceplot, ACF and inefficiency factor of σ_2^2:

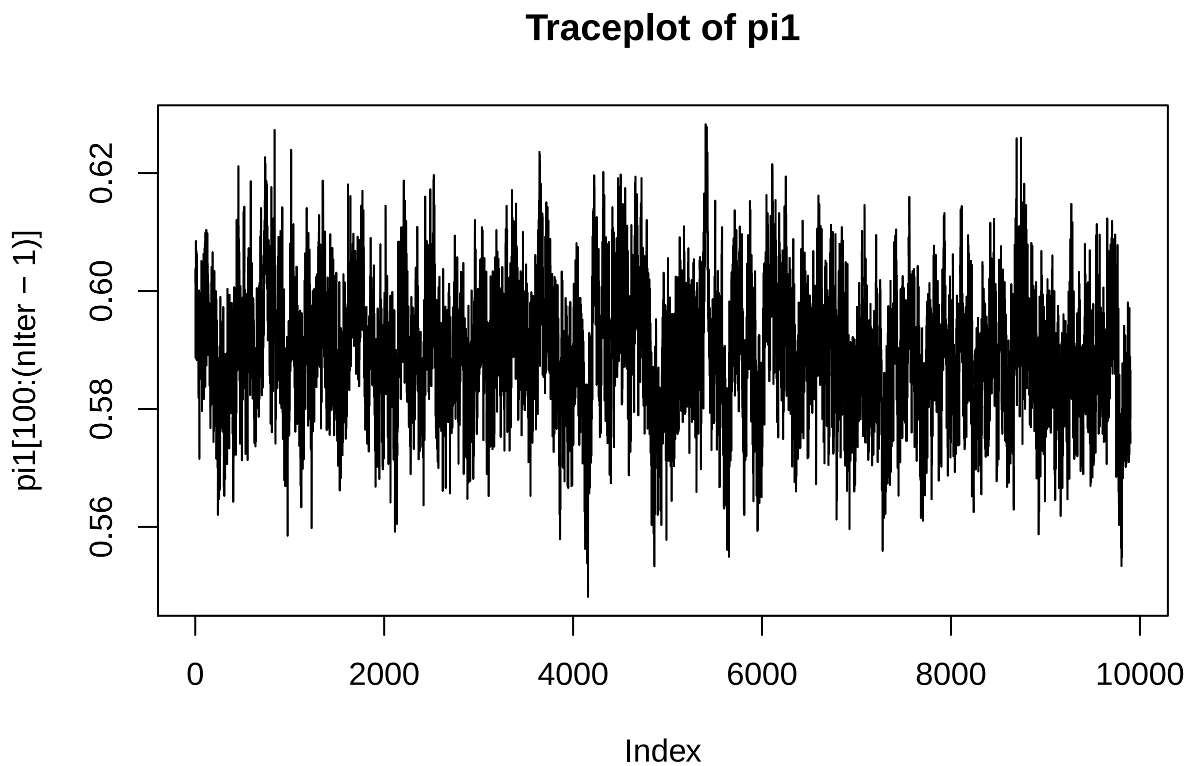

Traceplot of sigma2
sigmaSecond[100:(nIter − 1)]
1900
2000
2100
2200
2300
0
2000
4000
6000
8000
10000
Index

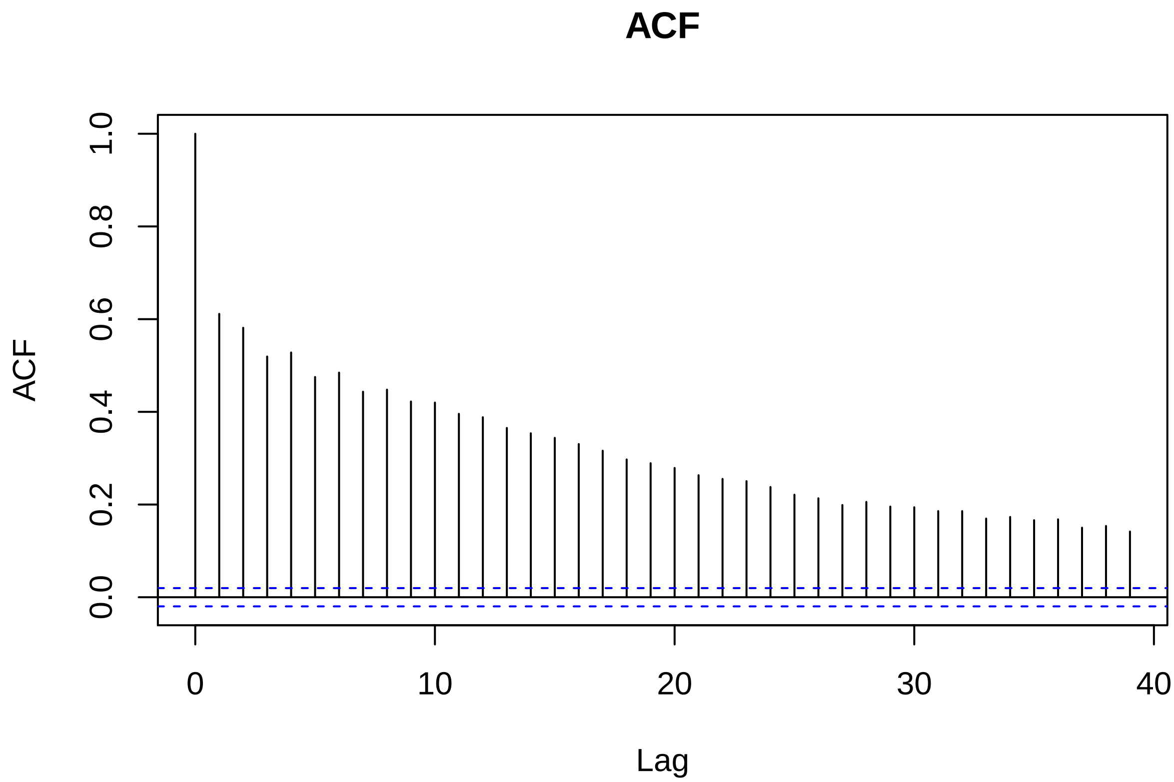



```
## [1] "IF:  4.13"
```

σ_2^2 converges to a value close to 2100. The draws are not really correlated and the IF value is quite small.

Traceplot, ACF and inefficiency factor of π_1:

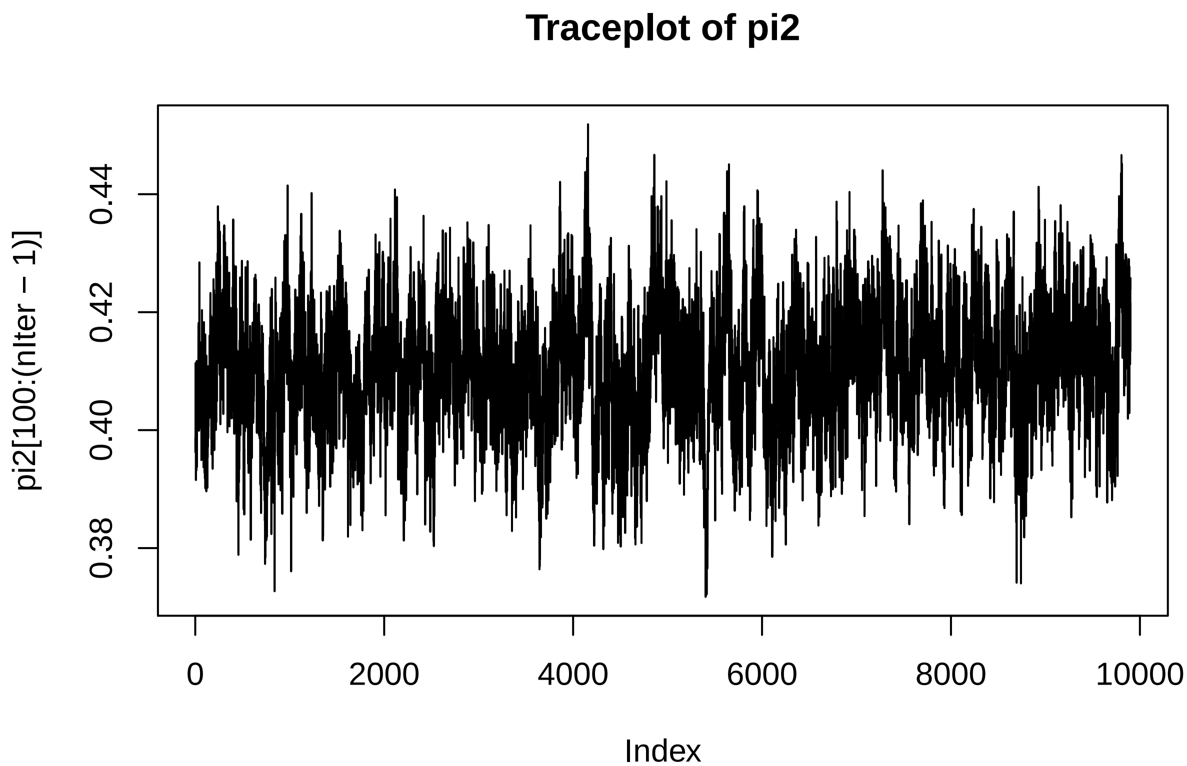

Traceplot of pi1
pi1[100:(nIter − 1)]
0.56
0.58
0.60
0.62
0
2000
4000
6000
8000
10000
Index

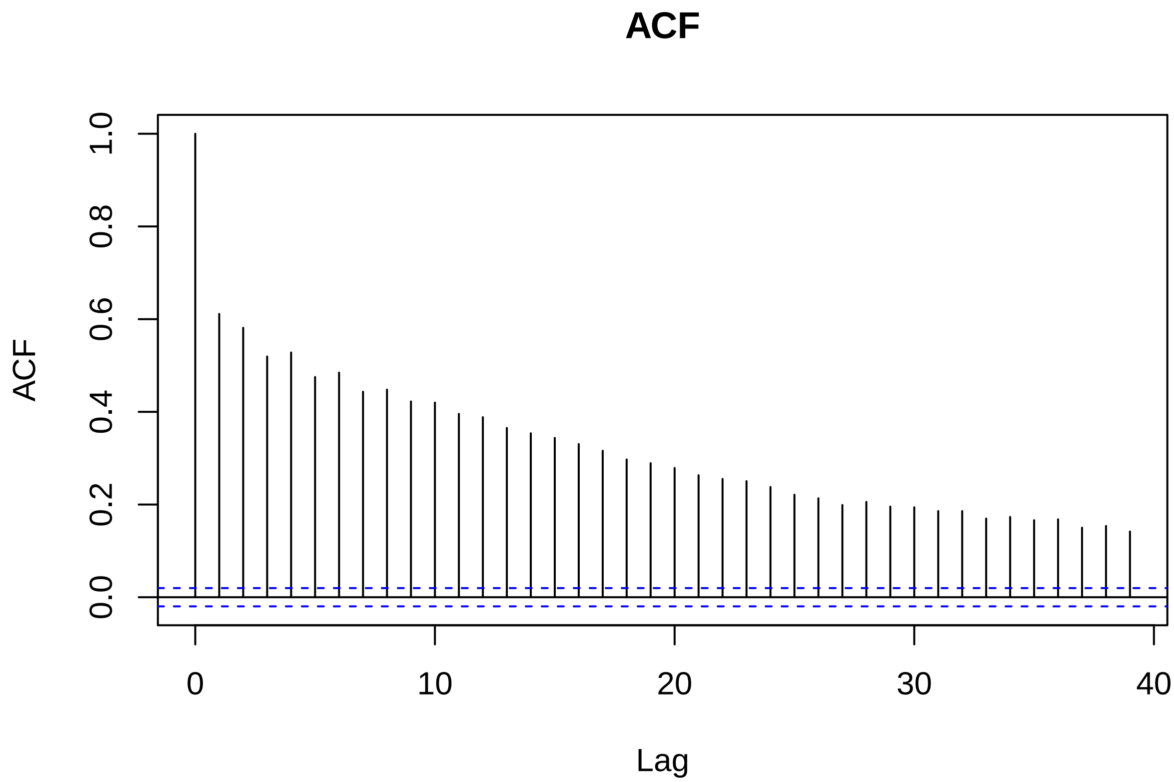



```
## [1] "IF:  24.76"
```

π_1 converges to a value that is around 0.57 and the draws seems to be highly positively correlated, with a high IF value.

Traceplot, ACF and inefficiency factor of π_2:


Traceplot of pi2
pi2[100:(nIter − 1)]
0.38
0.40
0.42
0.44
0
2000
4000
6000
8000
10000
Index

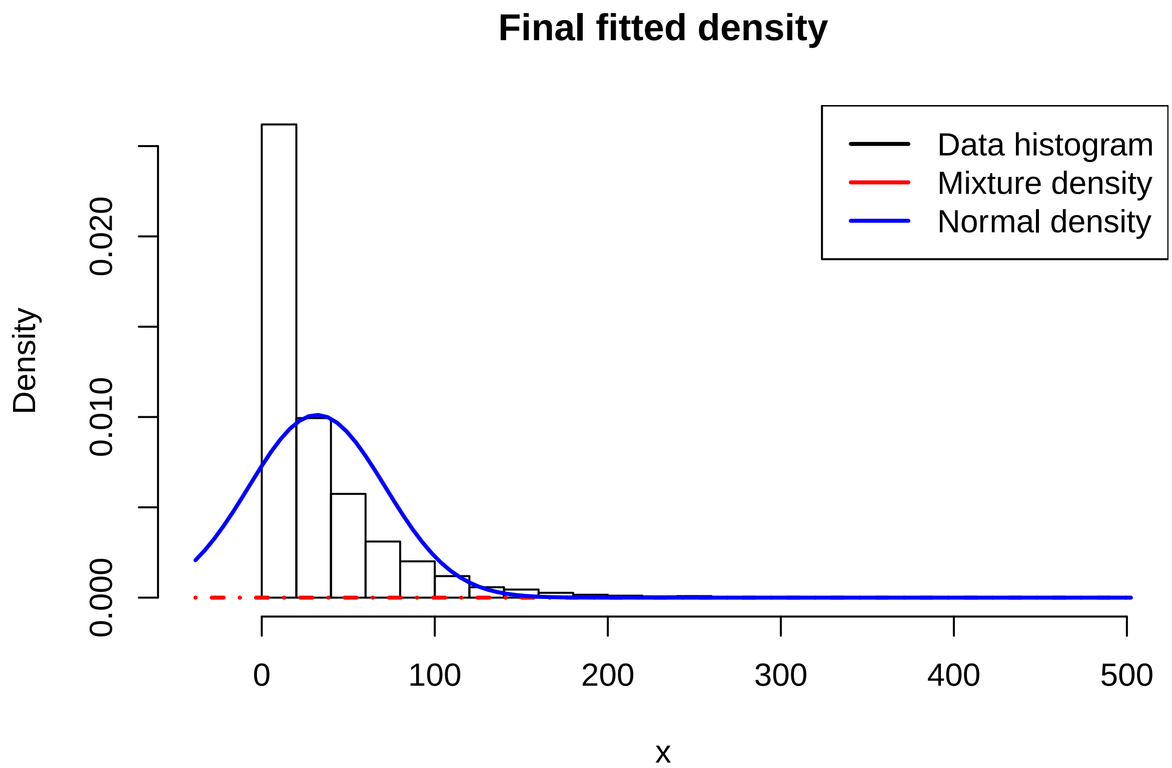



```
## [1] "IF:  24.76"
```

Also π_2 converges to a value around 0.43, but in this case from the ACF and IF we can see that the draws seem to be highly positively correlated.

1.c

Histogram of the data with the model in 1.a and 1.b using the posterior mean values for all the parameters:

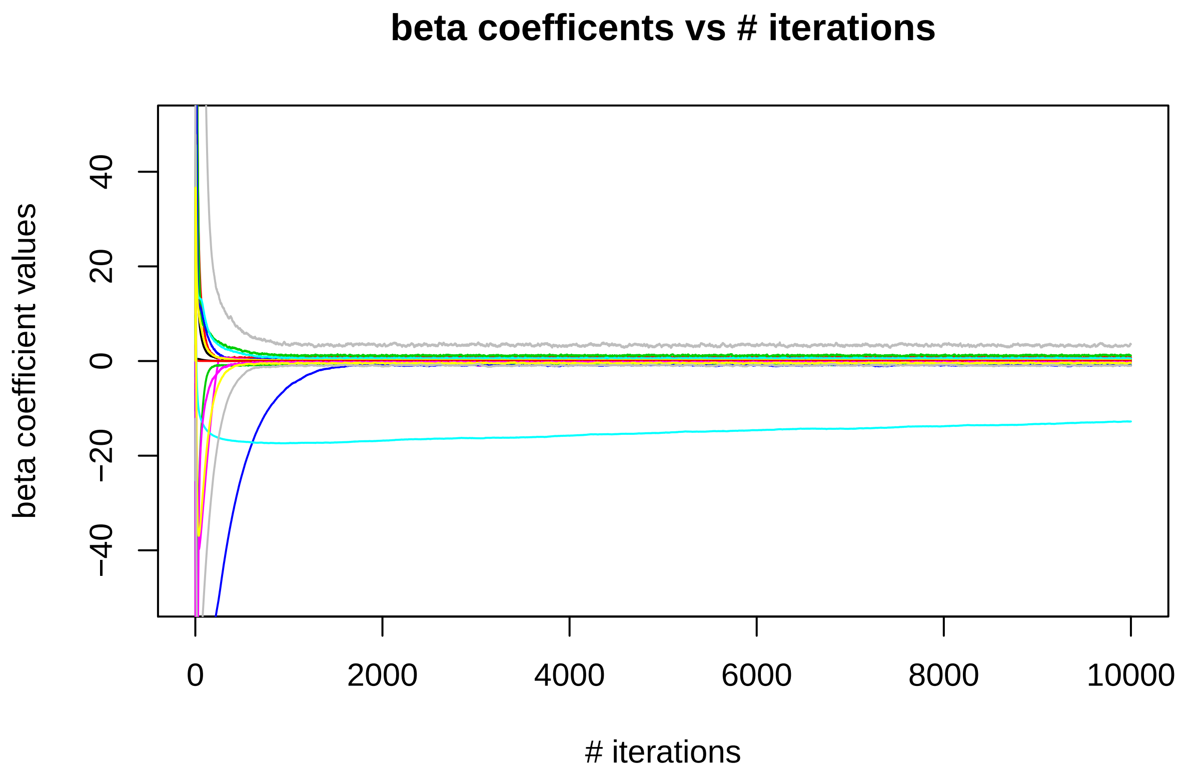


The mixture of normal models seems to be the best in terms of fitting.

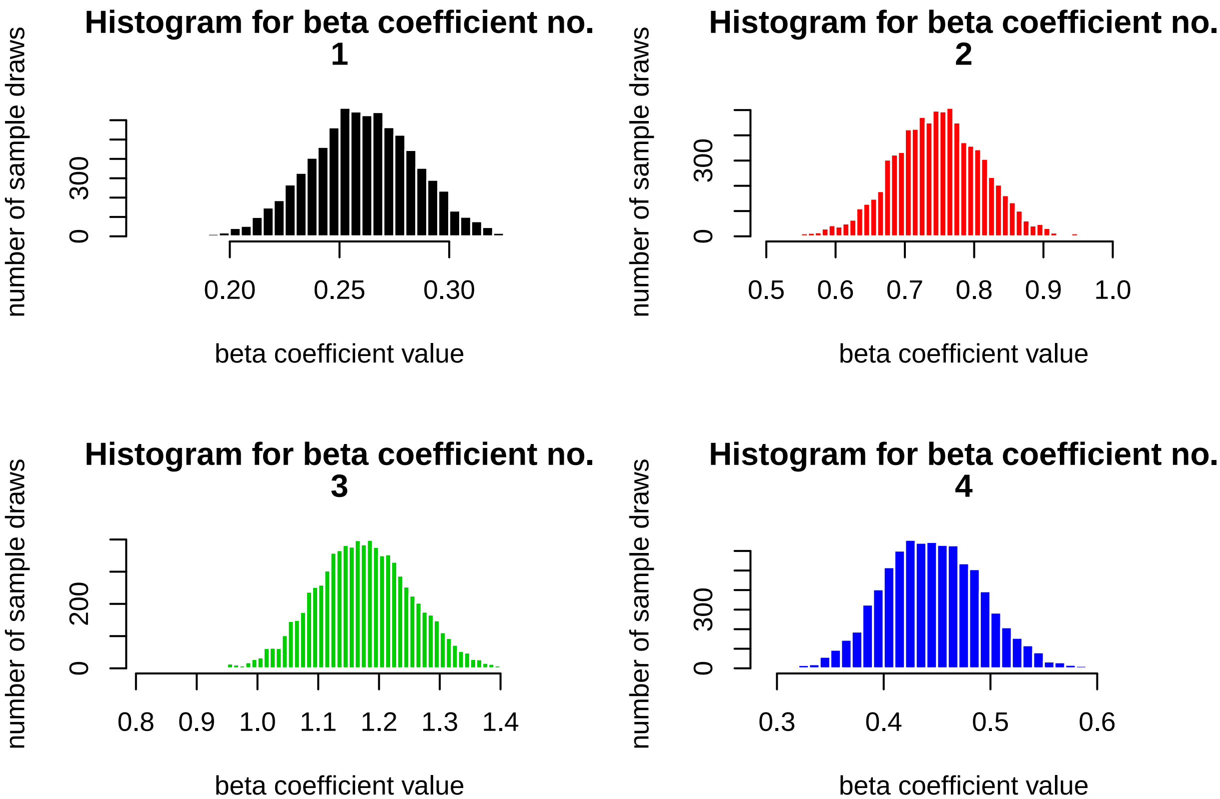
Assignment 2

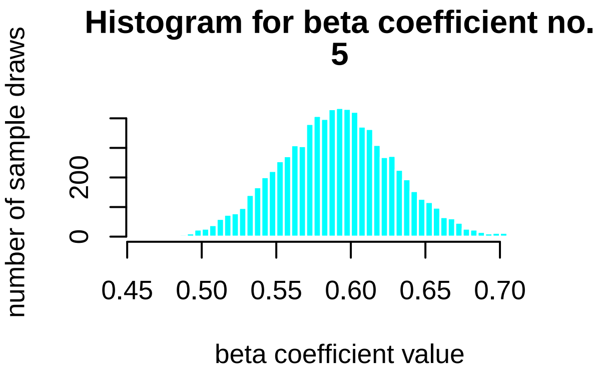
We are going to use a gibbs sampler to find the regression coefficients to be used in the probit regression model and then compare the performance of the regression coefficients β to regression coefficients drawn from $\mathcal{N}(\tilde{\beta}, J^{-1})$.

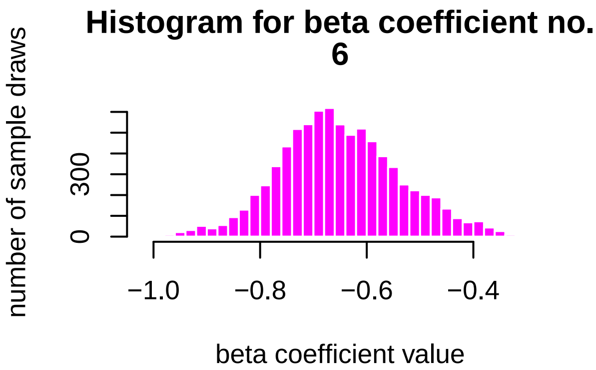
```
## Loading required package: randtoolbox

## Loading required package: rngWELL

## This is randtoolbox. For overview, type 'help("randtoolbox")'.

## [1] 4.059786
```

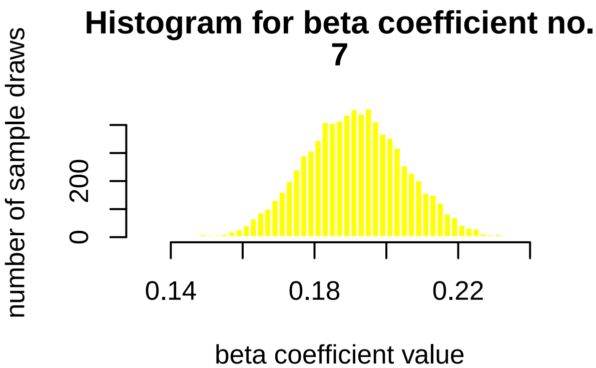


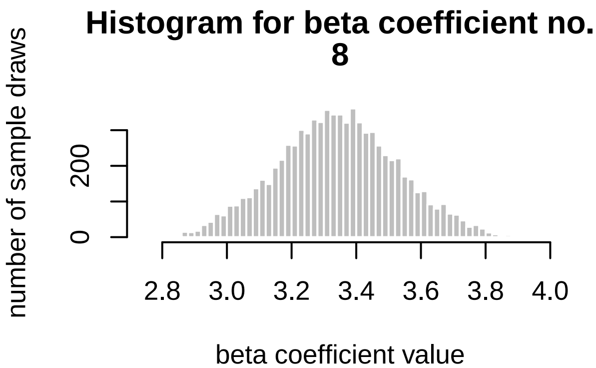
We see that the plot shows that the gibbs sampler make the regression coefficients converge to relatively small magnitudes after initially diverging from zero. We take the burn-in period to be the first 2000 iterations and make marginal posterior distribution histograms of the beta coefficients generated after the burn-in.

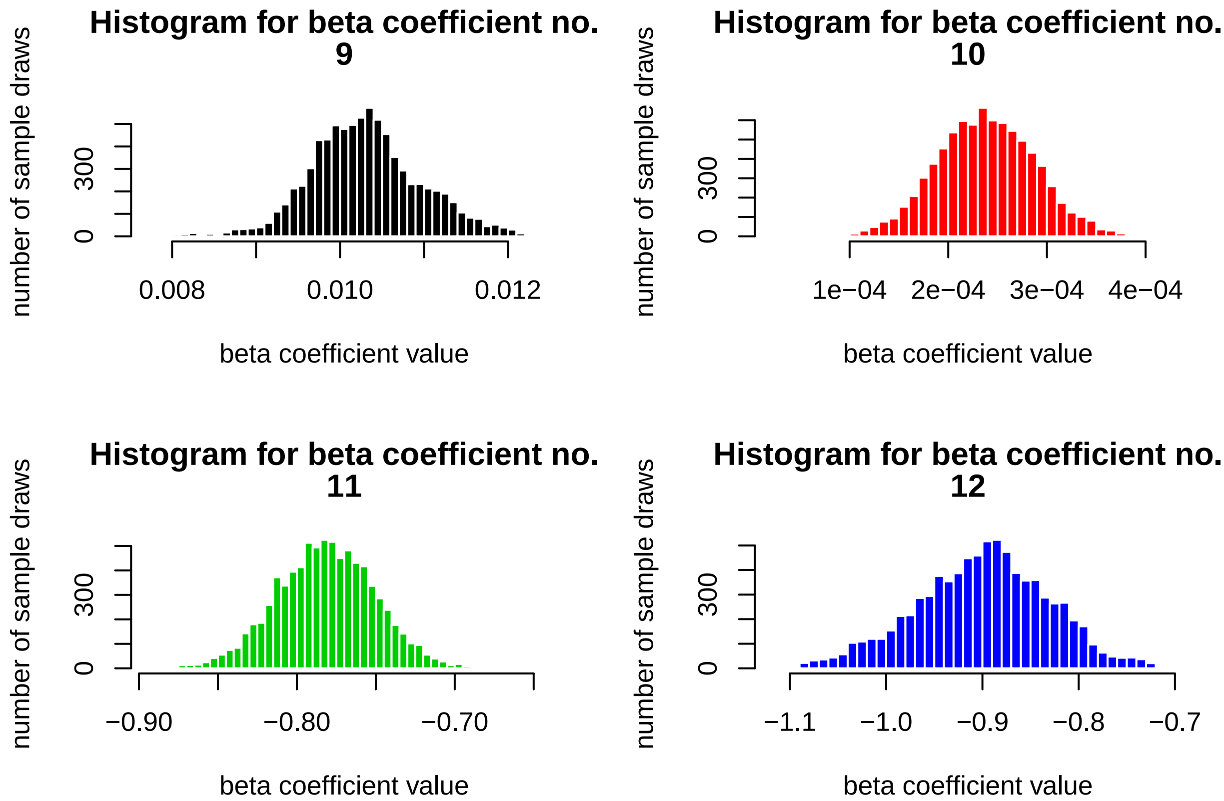

Histogram for beta coefficient no. 1
number of sample draws
0
300
0.20
0.25
0.30
beta coefficient value
Histogram for beta coefficient no. 2
number of sample draws
0
300
0.5
0.6
0.7
0.8
0.9
1.0
beta coefficient value
Histogram for beta coefficient no. 3
number of sample draws
0
200
0.8
0.9
1.0
1.1
1.2
1.3
1.4
beta coefficient value
Histogram for beta coefficient no. 4
number of sample draws
0
300
0.3
0.4
0.5
0.6
beta coefficient value

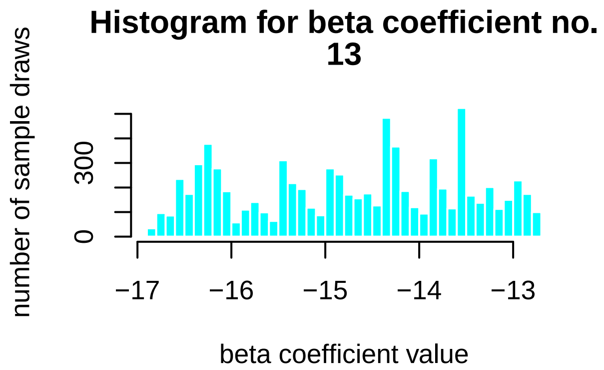

Histogram for beta coefficient no. 5
number of sample draws
0
200
0.45
0.50
0.55
0.60
0.65
0.70
beta coefficient value

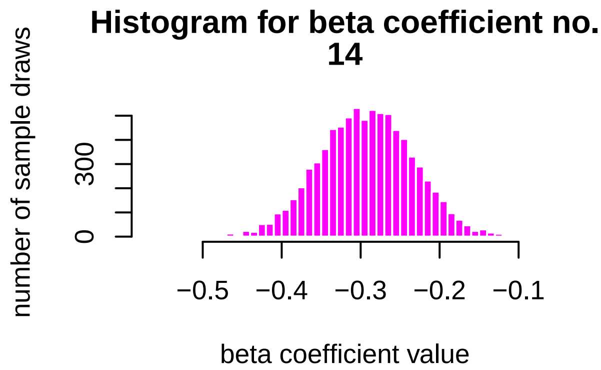

Histogram for beta coefficient no. 6
number of sample draws
0
300
−1.0
−0.8
−0.6
−0.4
beta coefficient value

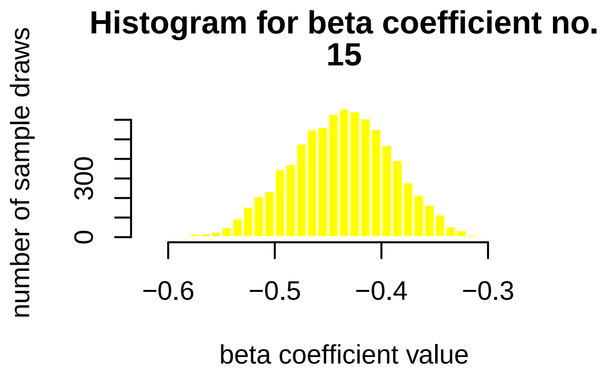

Histogram for beta coefficient no. 7
number of sample draws
0
200
0.14
0.18
0.22
beta coefficient value

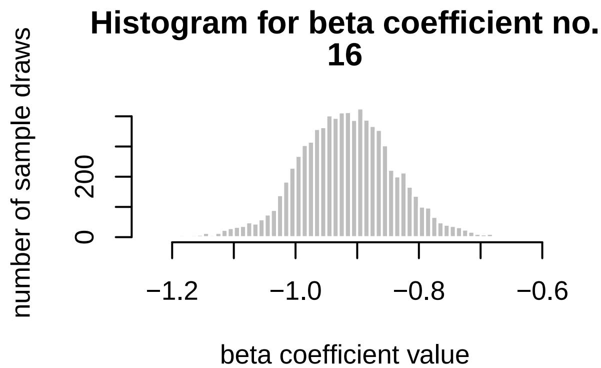

Histogram for beta coefficient no. 8
number of sample draws
0
200
2.8
3.0
3.2
3.4
3.6
3.8
4.0
beta coefficient value


Histogram for beta coefficient no. 9
number of sample draws
0
300
0.008
0.010
0.012
beta coefficient value
Histogram for beta coefficient no. 10
number of sample draws
0
300
1e−04
2e−04
3e−04
4e−04
beta coefficient value
Histogram for beta coefficient no. 11
number of sample draws
0
300
−0.90
−0.80
−0.70
beta coefficient value
Histogram for beta coefficient no. 12
number of sample draws
0
300
−1.1
−1.0
−0.9
−0.8
−0.7
beta coefficient value









It can be seen that most of the beta coefficients have reached a stationary posterior mode value and have normal marginal posterior distributions around their posterior modes. A quick check on trceplot of beta coefficient no. 13 reveals that it is still climbing slowly towards more posivive values by iteration 10000, hence the uniform-distribution looking histogram.

Let us look and compare the point estimates of β from 2.a and 2.c:

```
## [1] "Point estimate from 2.a:"

##  [1]  2.890548e-01  7.769794e-01  1.358184e+00  4.202192e-01  6.121006e-01
##  [6] -1.052757e+00  1.595389e-01  3.271652e+00  6.348322e-03  4.444389e-04
## [11] -7.904525e-01 -7.255792e-01 -4.459422e+00 -4.232624e-01 -3.172078e-01
## [16] -7.237761e-01

## [1] "Point estimate from 2.c:"

##  [1]  2.712491e-01  8.369774e-01  1.167379e+00  4.434609e-01  5.140302e-01
##  [6] -4.142739e-01  1.790807e-01  3.567231e+00  1.113069e-02  2.918796e-04
## [11] -8.499380e-01 -9.631469e-01 -1.277401e+01 -3.130100e-01 -3.378077e-01
## [16] -9.569427e-01
```

It is seen that the two point estimates are very similar.

```
## [1] "Classification rate for coefficients drawn from posterior mode normal dist.:  9.06e+01 %"

## [1] "Classification rate for gibbs sampler coefficients:  9.06e+01 %"
```

We also see that the classification performance of the gibbs sampler coefficients is as good as the one obtained from coefficients found from the posterior mode normal distribution.