R Notebook

This is an R Markdown (http://rmarkdown.rstudio.com) Notebook. When you execute code within the notebook, the results appear beneath the code.

Try executing this chunk by clicking the *Run* button within the chunk or by placing your cursor inside it and pressing *Ctrl+Shift+Enter*.

```
dat=read.csv("CreditCard_SOW_data.csv")
library("MCMCpack")
```

```
## Warning: package 'MCMCpack' was built under R version 4.0.0
```

```
## Loading required package: coda
```

```
## Warning: package 'coda' was built under R version 4.0.0
```

```
## Loading required package: MASS
```

```
## ##
## ## Markov Chain Monte Carlo Package (MCMCpack)
```

```
## ## Copyright (C) 2003-2020 Andrew D. Martin, Kevin M. Quinn, and Jong Hee Park
```

```
## ##
## ## Support provided by the U.S. National Science Foundation
```

```
## ## (Grants SES-0350646 and SES-0350613)
## ##
```
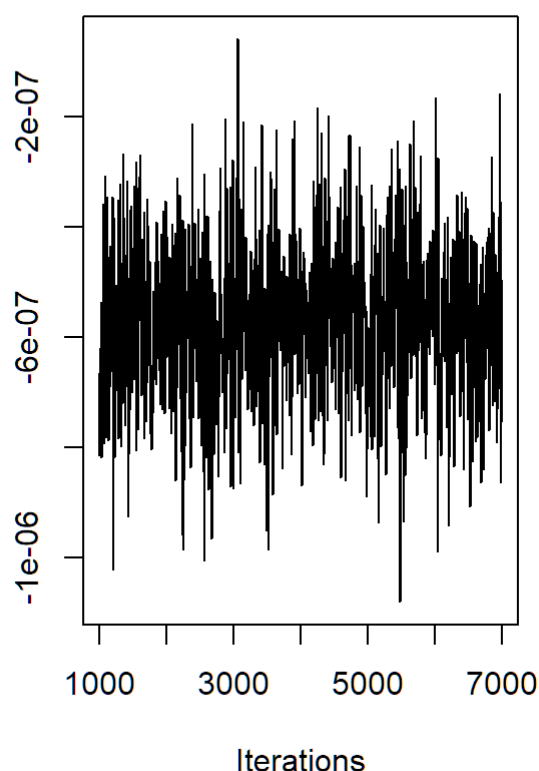
```
dat$ConsumerID=as.factor(dat$ConsumerID)
dat$logSowRatio = log(dat$WalletShare/(1-dat$WalletShare))
```

```
l1=MCMCregress(logSowRatio~History+Income+Balance+Promotion, mcmc=6000, thin=6, data=dat)
summary(l1)
```
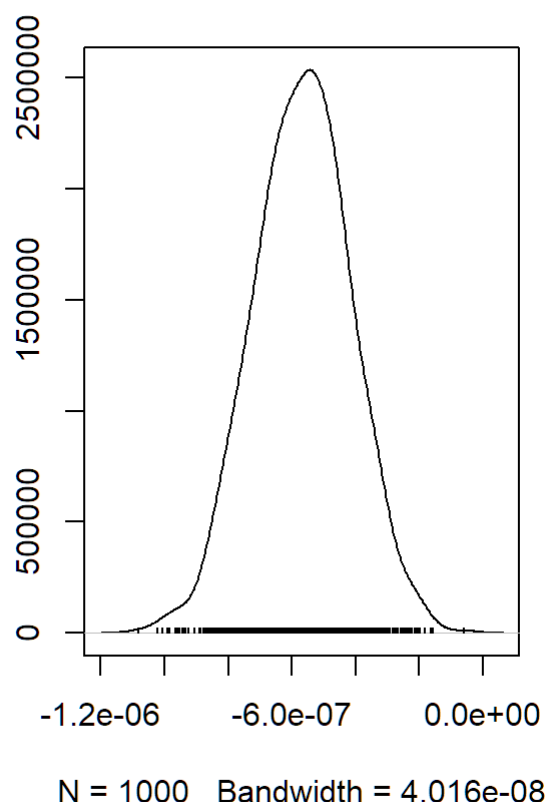
```
## 
## Iterations = 1001:6995
## Thinning interval = 6
## Number of chains = 1
## Sample size per chain = 1000
## 
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
## 
##                  Mean        SD  Naive SE Time-series SE
## (Intercept)  1.915e-01 1.699e-02 5.372e-04      5.372e-04
## History      8.765e-03 2.233e-04 7.063e-06      7.063e-06
## Income      -5.682e-07 1.508e-07 4.770e-09      5.041e-09
## Balance     -4.960e-04 2.776e-06 8.780e-08      8.780e-08
## Promotion    1.757e-01 9.001e-03 2.846e-04      2.998e-04
## sigma2       4.332e-02 1.031e-03 3.259e-05      3.259e-05
## 
## 2. Quantiles for each variable:
## 
##                  2.5%       25%       50%       75%     97.5%
## (Intercept)  1.572e-01 1.812e-01 1.919e-01 2.030e-01 2.250e-01
## History      8.293e-03 8.614e-03 8.769e-03 8.921e-03 9.189e-03
## Income      -8.602e-07 -6.685e-07 -5.645e-07 -4.636e-07 -2.739e-07
## Balance     -5.014e-04 -4.979e-04 -4.961e-04 -4.942e-04 -4.902e-04
## Promotion    1.587e-01 1.697e-01 1.754e-01 1.819e-01 1.938e-01
## sigma2       4.141e-02 4.256e-02 4.333e-02 4.400e-02 4.531e-02
```

```
plot(ll[,3],type="l")
```
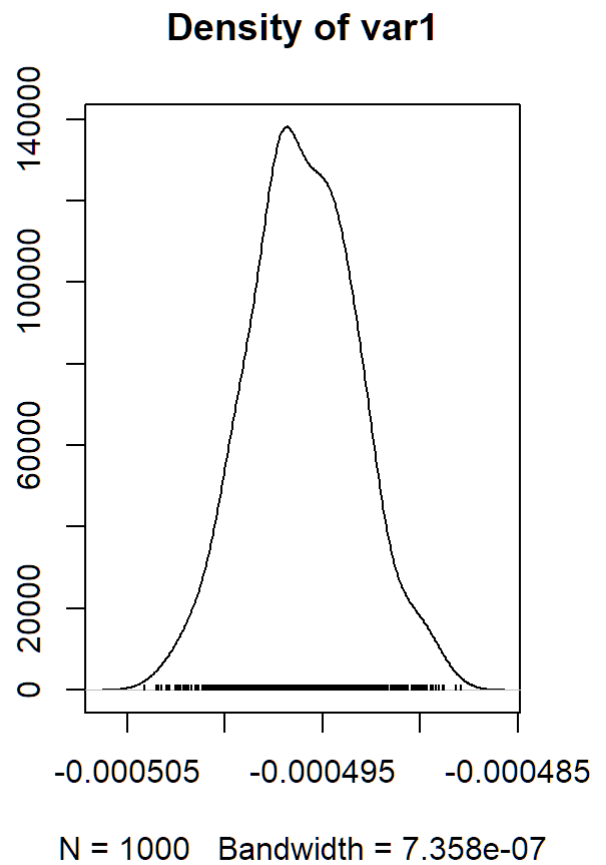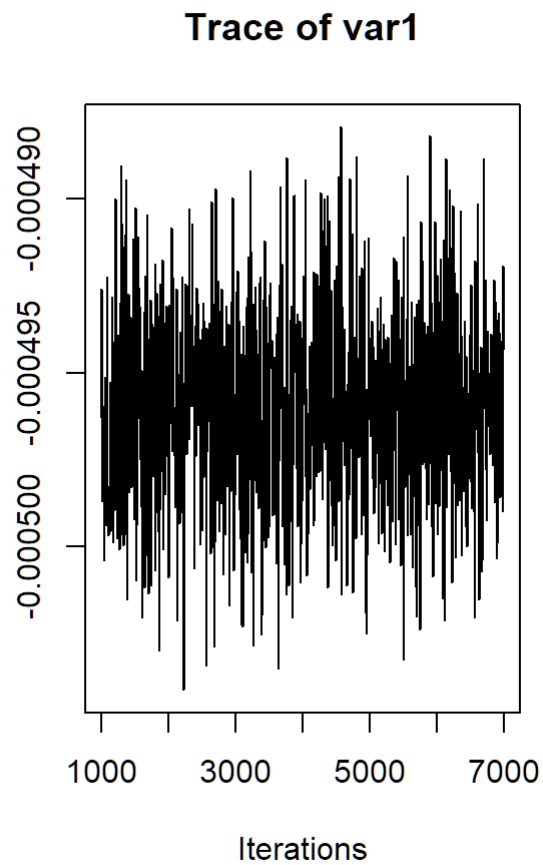

Trace of var1

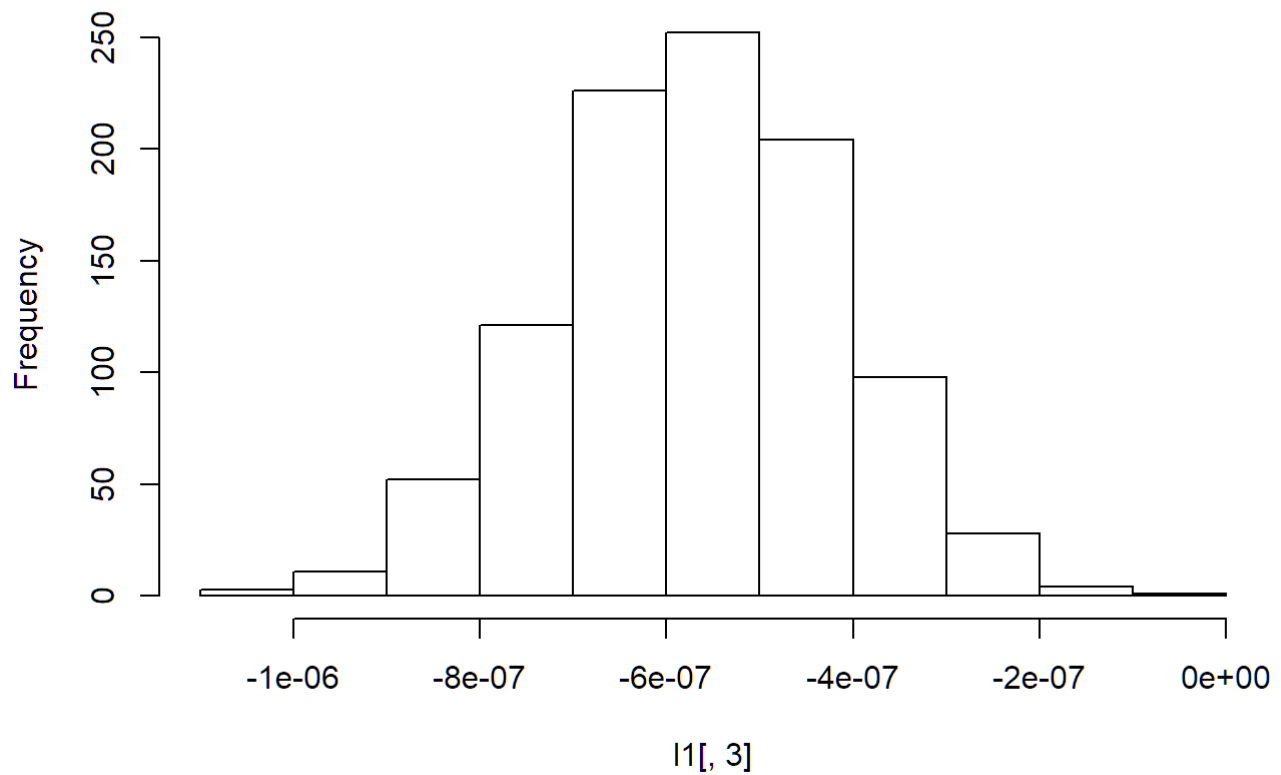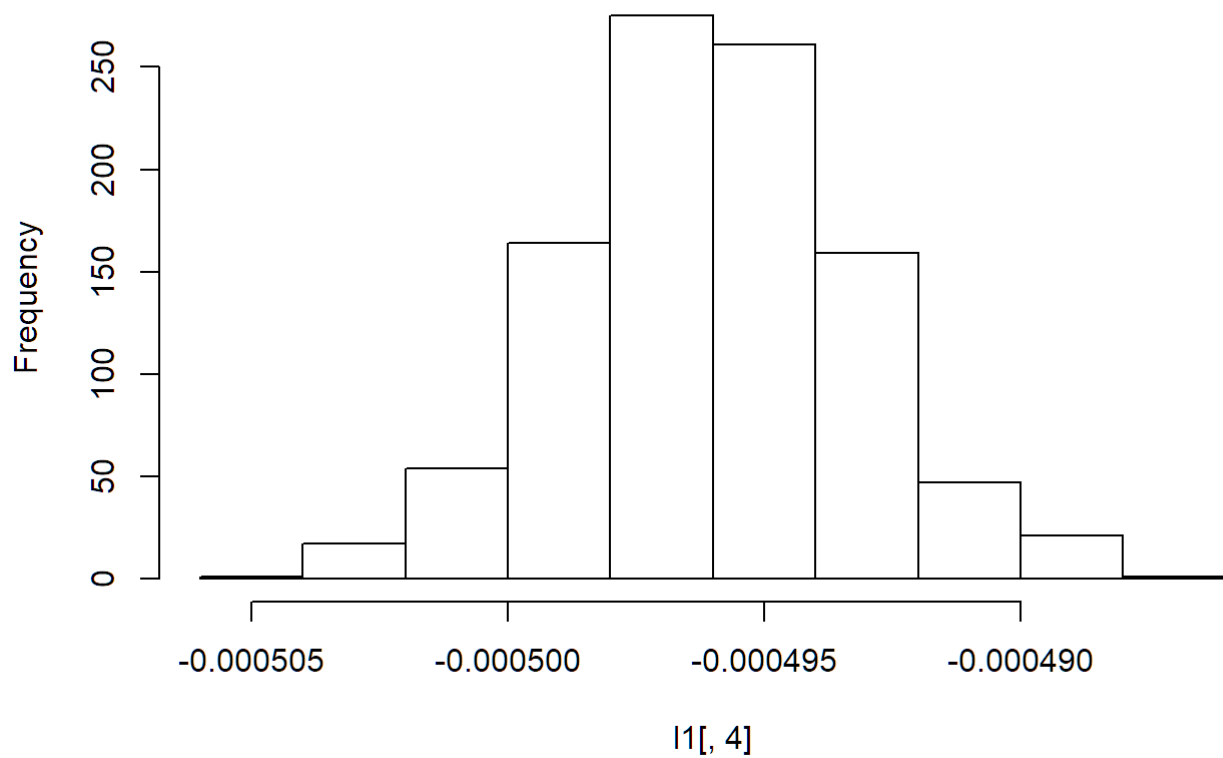**Density of var1**

Iterations

N = 1000   Bandwidth = 4.016e-08

```
plot(ll[,4],type="l")
```

Trace of var1

Density of var1


```
hist(ll[,3])
```

Histogram of l1[, 3]

l1[, 3]

```
hist(l1[,4])
```
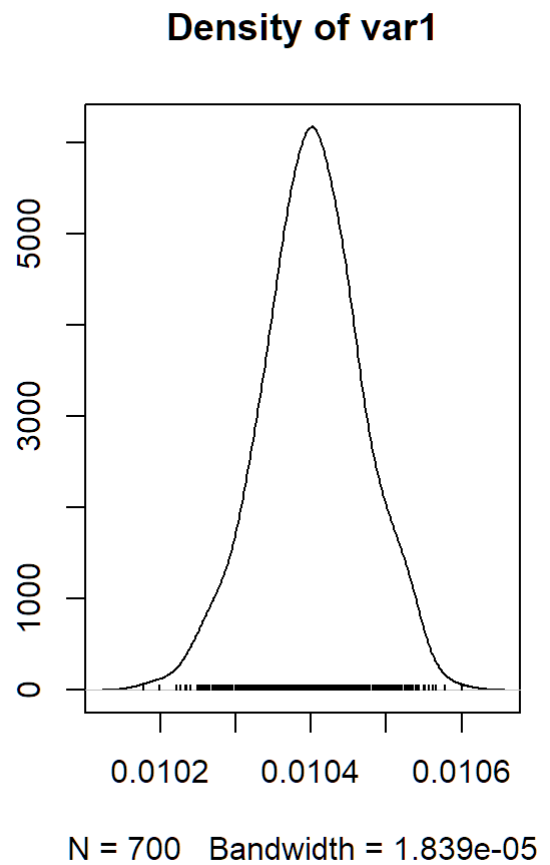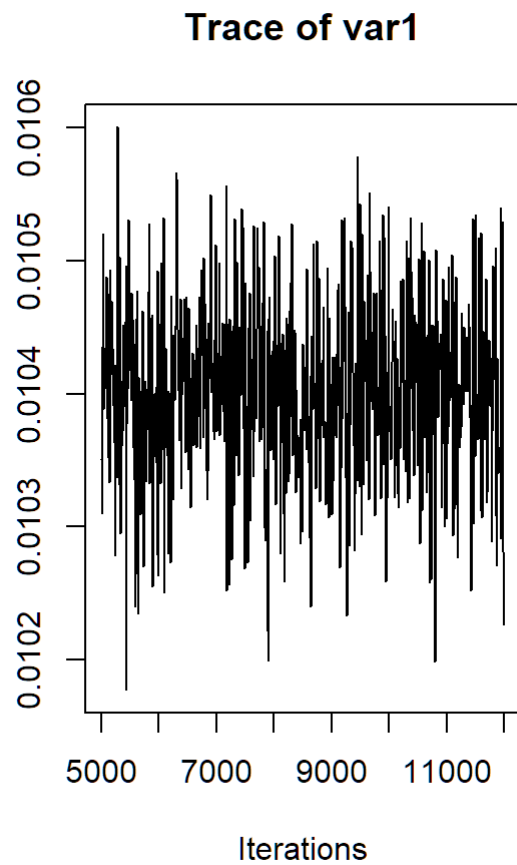
Histogram of l1[, 4]

l1[, 4]

```
rp1 = MCMChregress(fixed=logSowRatio~History+Balance+Promotion+History:Promotion+Income:Promoti
on, random=~Promotion, group="ConsumerID", mcmc=7000, thin=10, data=dat, r=3, R=diag(2),burnin=
5000)
```

```
##
## Running the Gibbs sampler. It may be long, keep cool :)
##
## **********:10.0%
## **********:20.0%
## **********:30.0%
## **********:40.0%
## **********:50.0%
## **********:60.0%
## **********:70.0%
## **********:80.0%
## **********:90.0%
## **********:100.0%
```
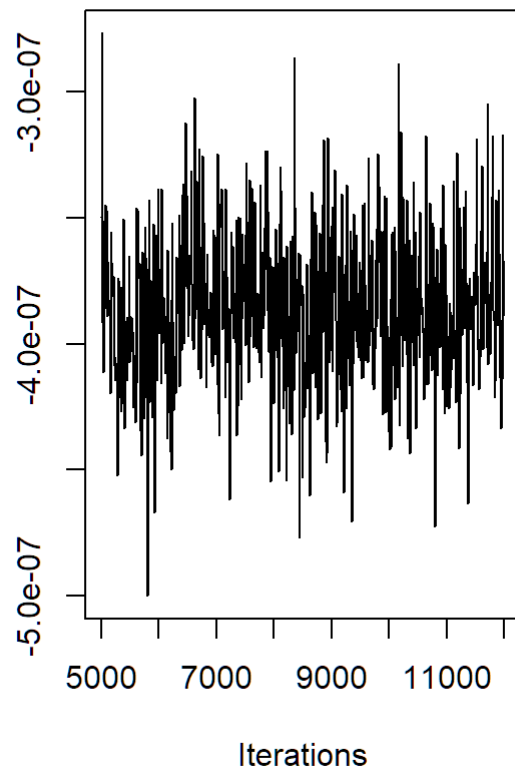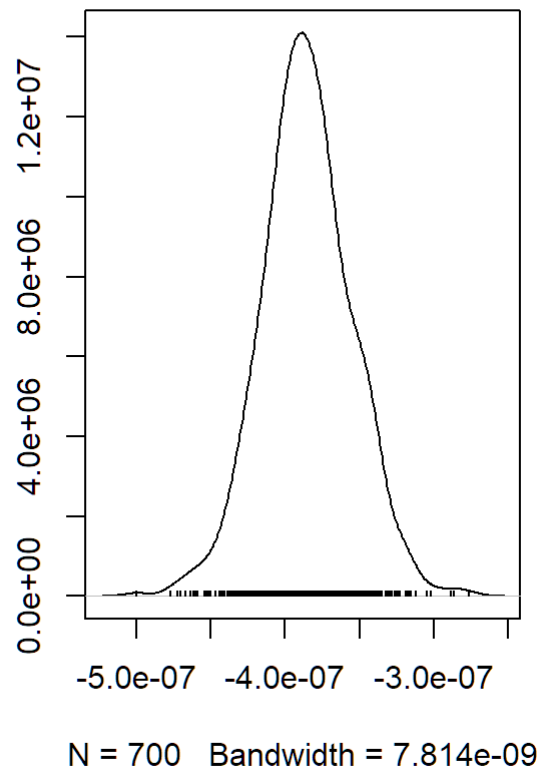
```
summary(rp1$mcmc[,1:6])
```

```
##
## Iterations = 5001:11991
## Thinning interval = 10
## Number of chains = 1
## Sample size per chain = 700
##
## 1. Empirical mean and standard deviation for each variable,
##     plus standard error of the mean:
##
##                            Mean        SD  Naive SE Time-series SE
## beta.(Intercept)       9.659e-02 2.425e-03 9.164e-05      9.164e-05
## beta.History           1.040e-02 6.699e-05 2.532e-06      2.532e-06
## beta.Balance          -5.008e-04 1.473e-07 5.569e-09      5.569e-09
## beta.Promotion         2.940e-01 3.083e-03 1.165e-04      1.380e-04
## beta.History:Promotion -2.574e-03 4.256e-05 1.609e-06      1.818e-06
## beta.Promotion:Income  -3.855e-07 3.006e-08 1.136e-09      1.136e-09
##
## 2. Quantiles for each variable:
##
##                            2.5%       25%       50%       75%
## beta.(Intercept)       9.198e-02  9.500e-02  9.646e-02  9.811e-02
## beta.History           1.026e-02  1.036e-02  1.040e-02  1.045e-02
## beta.Balance          -5.011e-04 -5.009e-04 -5.008e-04 -5.007e-04
## beta.Promotion         2.873e-01  2.921e-01  2.940e-01  2.961e-01
## beta.History:Promotion -2.654e-03 -2.604e-03 -2.574e-03 -2.545e-03
## beta.Promotion:Income  -4.433e-07 -4.039e-07 -3.864e-07 -3.673e-07
##                           97.5%
## beta.(Intercept)        1.017e-01
## beta.History            1.053e-02
## beta.Balance           -5.005e-04
## beta.Promotion          2.998e-01
## beta.History:Promotion -2.492e-03
## beta.Promotion:Income  -3.244e-07
```

```
plot(rp1$mcmc[,2],type="l")
```

Trace of var1

Density of var1

N = 700   Bandwidth = 1.839e-05

```
plot(rp1$mcmc[,6],type="l")
```

Trace of var1

Iterations

Density of var1

N = 700   Bandwidth = 7.814e-09

```
hist(rp1$mcmc[,2])
```

Histogram of rp1$mcmc[, 2]
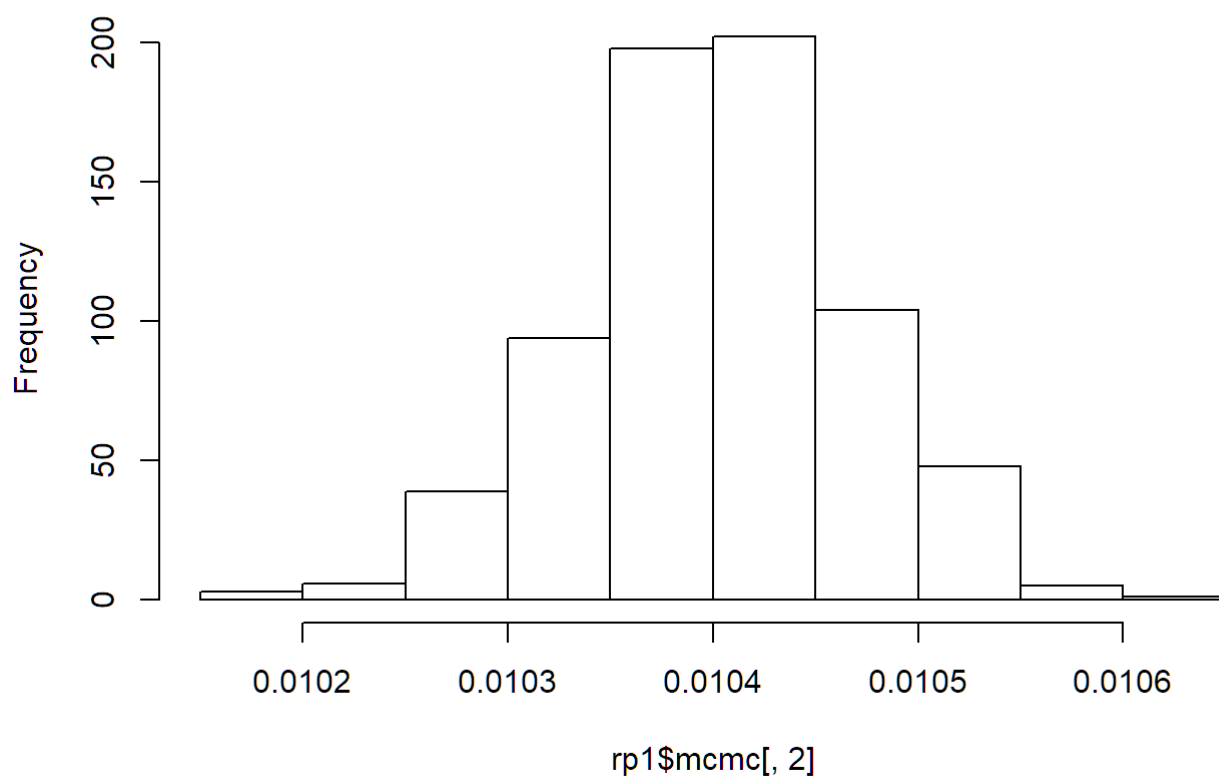
rp1$mcmc[, 2]

```
hist(rp1$mcmc[,6])
```

Histogram of rp1$mcmc[, 6]
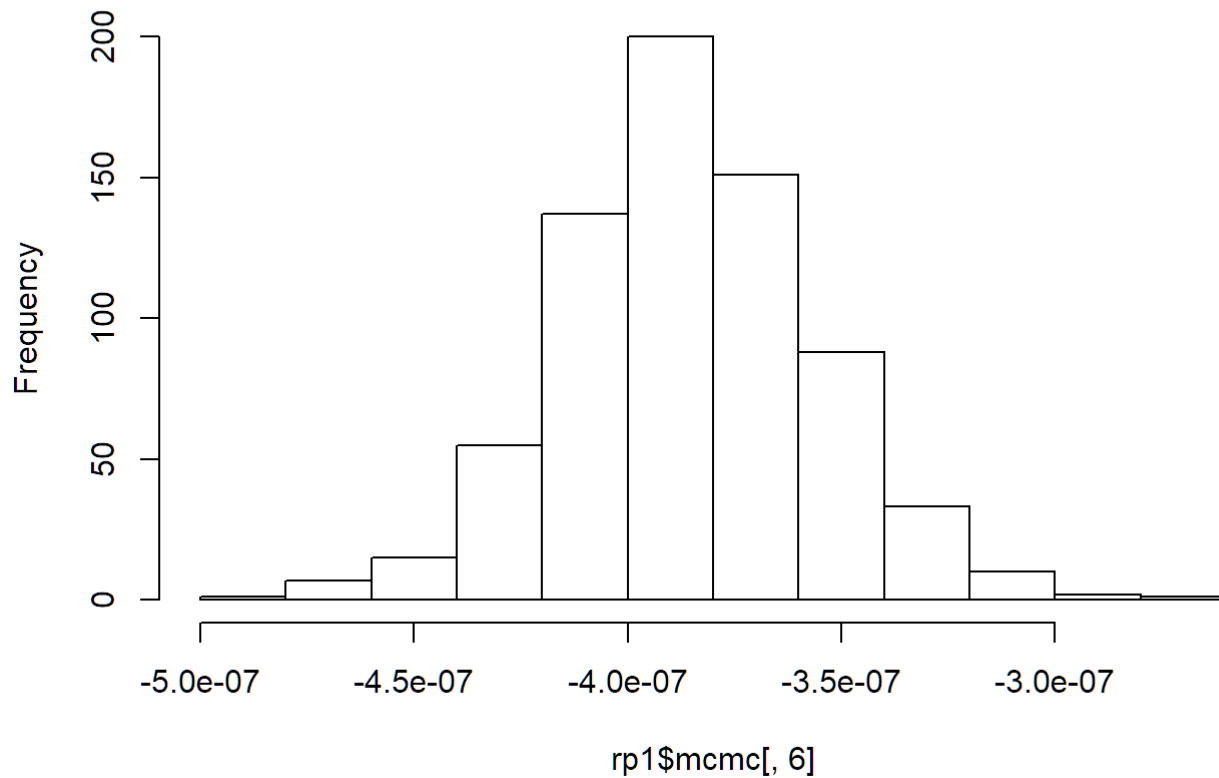

```
bank=read.csv("Bank_Retention_Data.csv")
bank$TractID=as.factor(bank$TractID)
```

```
glm1=glm(Churn~Age+Income+HomeVal+Tenure+DirectDeposit+Loan+Dist+MktShare,data=bank,family=bino
mial(link="logit"))
summary(glm1)
```

```
##
## Call:
## glm(formula = Churn ~ Age + Income + HomeVal + Tenure + DirectDeposit +
##     Loan + Dist + MktShare, family = binomial(link = "logit"),
##     data = bank)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.2054  -0.6823  -0.5328  -0.3401   2.6266
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -0.606224   0.296596  -2.044 0.040960 *
## Age           -0.016103   0.004150  -3.881 0.000104 ***
## Income         0.107067   0.015985   6.698 2.11e-11 ***
## HomeVal       -0.026059   0.005477  -4.758 1.95e-06 ***
## Tenure        -0.029709   0.006549  -4.536 5.73e-06 ***
## DirectDeposit -0.465836   0.110617  -4.211 2.54e-05 ***
## Loan           0.099376   0.124380   0.799 0.424310
## Dist           0.267618   0.061958   4.319 1.57e-05 ***
## MktShare      -0.082440   0.325551  -0.253 0.800089
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2355.9  on 2504  degrees of freedom
## Residual deviance: 2189.4  on 2496  degrees of freedom
## AIC: 2207.4
##
## Number of Fisher Scoring iterations: 5
```

```
glm2=glm(Churn~Age+Income+HomeVal+Tenure+DirectDeposit+Loan+Dist+MktShare,data=bank,family=bino
mial(link="probit"))
summary(glm2)
```

```
##
## Call:
## glm(formula = Churn ~ Age + Income + HomeVal + Tenure + DirectDeposit +
##     Loan + Dist + MktShare, family = binomial(link = "probit"),
##     data = bank)
##
## Deviance Residuals:
##     Min      1Q   Median      3Q      Max
## -1.1714  -0.6886  -0.5374  -0.3252   2.7140
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -0.397967   0.168825  -2.357   0.0184 *
## Age           -0.009050   0.002314  -3.910 9.22e-05 ***
## Income         0.059194   0.008871   6.673 2.51e-11 ***
## HomeVal       -0.014360   0.002922  -4.914 8.90e-07 ***
## Tenure        -0.016430   0.003550  -4.628 3.69e-06 ***
## DirectDeposit -0.263070   0.062851  -4.186 2.84e-05 ***
## Loan           0.057756   0.070224   0.822   0.4108
## Dist           0.154712   0.036313   4.261 2.04e-05 ***
## MktShare      -0.045443   0.184547  -0.246   0.8055
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2355.9  on 2504  degrees of freedom
## Residual deviance: 2188.6  on 2496  degrees of freedom
## AIC: 2206.6
##
## Number of Fisher Scoring iterations: 6
```

```
AIC(glm1)
```

```
## [1] 2207.358
```

```
AIC(glm2)
```

```
## [1] 2206.626
```

```
BIC(glm1)
```

```
## [1] 2259.793
```

```
BIC(glm2)
```

```
## [1] 2259.06
```

```
library(lme4)
```

```
## Warning: package 'lme4' was built under R version 4.0.0
```

```
## Loading required package: Matrix
```

```
glmer1=glmer(Churn~Age+Income+HomeVal+Tenure+DirectDeposit+Loan+Dist+MktShare+(1|TractID), data
=bank, family=binomial, glmerControl(optimizer="bobyqa",optCtrl=list(maxfun=100000)))
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl =
## control$checkConv, : Model failed to converge with max|grad| = 0.0156544
## (tol = 0.001, component 1)
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, : Model is near
ly unidentifiable: very large eigenvalue
##  - Rescale variables?
```

```
summary(glmer1)
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
##   Approximation) [glmerMod]
##  Family: binomial  ( logit )
## Formula: Churn ~ Age + Income + HomeVal + Tenure + DirectDeposit + Loan +
##     Dist + MktShare + (1 | TractID)
##    Data: bank
## Control:
## glmerControl(optimizer = "bobyqa", optCtrl = list(maxfun = 1e+05))
##
##      AIC      BIC   logLik deviance df.resid
##   2208.7   2266.9  -1094.3   2188.7     2495
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.0913 -0.5118 -0.3894 -0.2447  5.3463
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  TractID (Intercept) 0.01988  0.141
## Number of obs: 2505, groups:  TractID, 26
##
## Fixed effects:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -0.561878   0.305951  -1.836   0.0663 .
## Age           -0.016503   0.004178  -3.950 7.81e-05 ***
## Income         0.106973   0.016078   6.653 2.87e-11 ***
## HomeVal       -0.026715   0.005692  -4.693 2.69e-06 ***
## Tenure        -0.029232   0.006564  -4.453 8.46e-06 ***
## DirectDeposit -0.461198   0.111002  -4.155 3.25e-05 ***
## Loan           0.099832   0.124633   0.801   0.4231
## Dist           0.266895   0.063377   4.211 2.54e-05 ***
## MktShare       0.006009   0.373151   0.016   0.9872
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) Age    Income HomeVl Tenure DrctDp Loan   Dist
## Age         -0.647
## Income      -0.221  0.055
## HomeVal     -0.207 -0.060 -0.534
## Tenure       0.014 -0.285 -0.075  0.077
## DirectDepst -0.176  0.012 -0.050  0.081 -0.115
## Loan        -0.073  0.073 -0.007 -0.059 -0.105 -0.083
## Dist        -0.324  0.000 -0.012 -0.150 -0.013 -0.008 -0.012
## MktShare    -0.359 -0.006 -0.031  0.060 -0.140  0.005 -0.008  0.260
## convergence code: 0
## Model failed to converge with max|grad| = 0.0156544 (tol = 0.001, component 1)
## Model is nearly unidentifiable: very large eigenvalue
##  - Rescale variables?
```

```
AIC(glmer1)
```

```
## [1] 2208.686
```

```
BIC(glmer1)
```

```
## [1] 2266.947
```

```
a=MCMChlogit(Churn~Age+Income+HomeVal+Tenure+DirectDeposit+Loan+Dist+MktShare,random=~1,group=
"TractID", data=bank,r=2,R=1, burin=10000, mcmc=20000, thin=20)
```

```
##
## Running the Gibbs sampler. It may be long, keep cool :)
##
## **********:10.0%, mean accept. rate=0.374
## **********:20.0%, mean accept. rate=0.416
## **********:30.0%, mean accept. rate=0.459
## **********:40.0%, mean accept. rate=0.445
## **********:50.0%, mean accept. rate=0.565
## **********:60.0%, mean accept. rate=0.485
## **********:70.0%, mean accept. rate=0.544
## **********:80.0%, mean accept. rate=0.536
## **********:90.0%, mean accept. rate=0.473
## **********:100.0%, mean accept. rate=0.545
```

```
summary(a$mcmc[,1:9])
```

```
##
## Iterations = 5001:24981
## Thinning interval = 20
## Number of chains = 1
## Sample size per chain = 1000
##
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
##
##                         Mean        SD  Naive SE Time-series SE
## beta.(Intercept)    -0.45298 0.1126221 3.561e-03      0.0650863
## beta.Age            -0.01592 0.0007637 2.415e-05      0.0003775
## beta.Income          0.12871 0.0030566 9.666e-05      0.0018888
## beta.HomeVal        -0.03239 0.0004787 1.514e-05      0.0002394
## beta.Tenure         -0.04029 0.0008128 2.570e-05      0.0002731
## beta.DirectDeposit  -0.57058 0.0285845 9.039e-04      0.0156295
## beta.Loan            0.22673 0.0285611 9.032e-04      0.0136874
## beta.Dist            0.26031 0.0212924 6.733e-04      0.0138720
## beta.MktShare       -0.19183 0.1132437 3.581e-03      0.0603667
##
## 2. Quantiles for each variable:
##
##                        2.5%      25%      50%      75%    97.5%
## beta.(Intercept)    -0.61111 -0.56889 -0.46363 -0.35185 -0.24824
## beta.Age            -0.01728 -0.01649 -0.01606 -0.01521 -0.01451
## beta.Income          0.12343  0.12636  0.12886  0.13110  0.13415
## beta.HomeVal        -0.03347 -0.03271 -0.03239 -0.03200 -0.03149
## beta.Tenure         -0.04162 -0.04101 -0.04017 -0.03963 -0.03902
## beta.DirectDeposit  -0.60220 -0.59035 -0.58099 -0.56050 -0.50647
## beta.Loan            0.18521  0.20597  0.22662  0.24088  0.30619
## beta.Dist            0.23376  0.24633  0.25331  0.26551  0.30882
## beta.MktShare       -0.35181 -0.30624 -0.19467 -0.08703 -0.01197
```
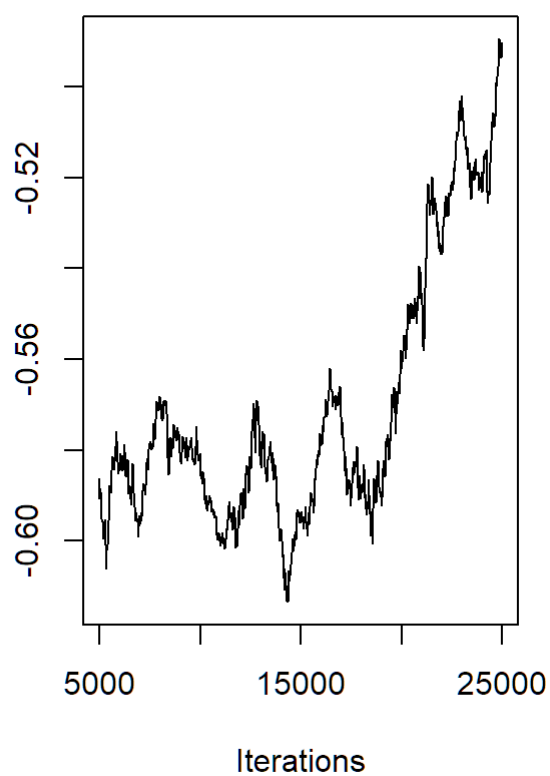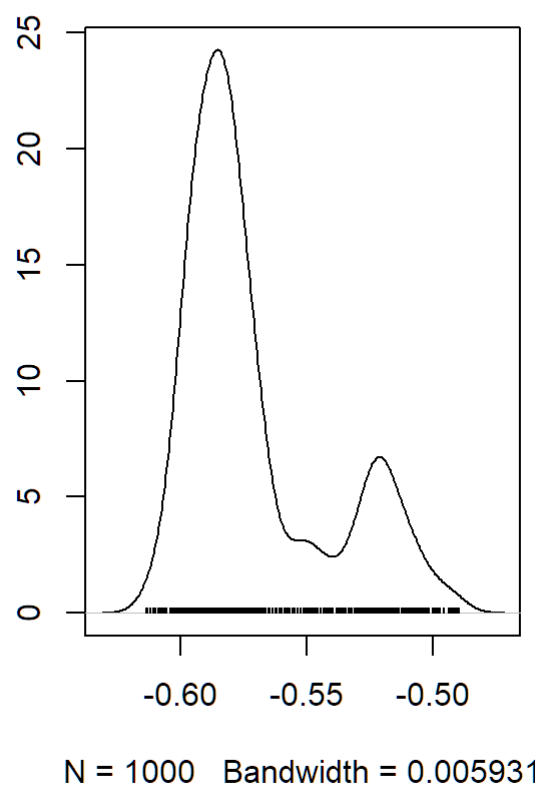
```
plot(a$mcmc[,3])
```

Trace of var1

Iterations

Density of var1

N = 1000   Bandwidth = 0.0008138
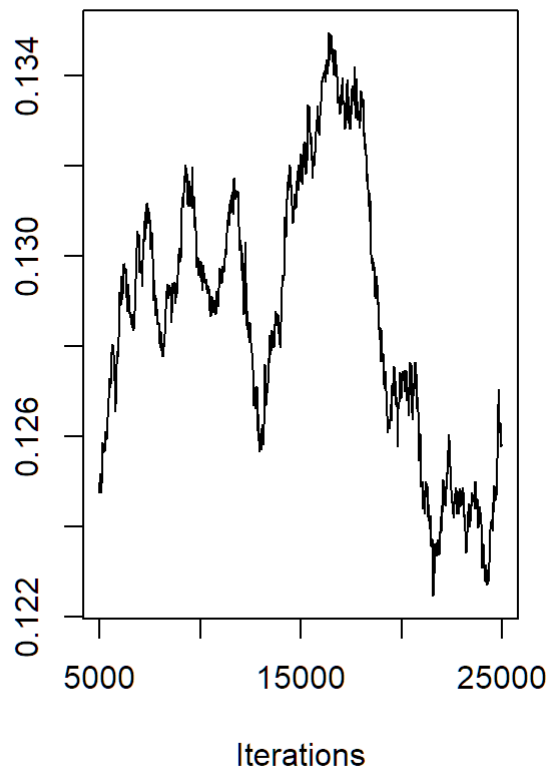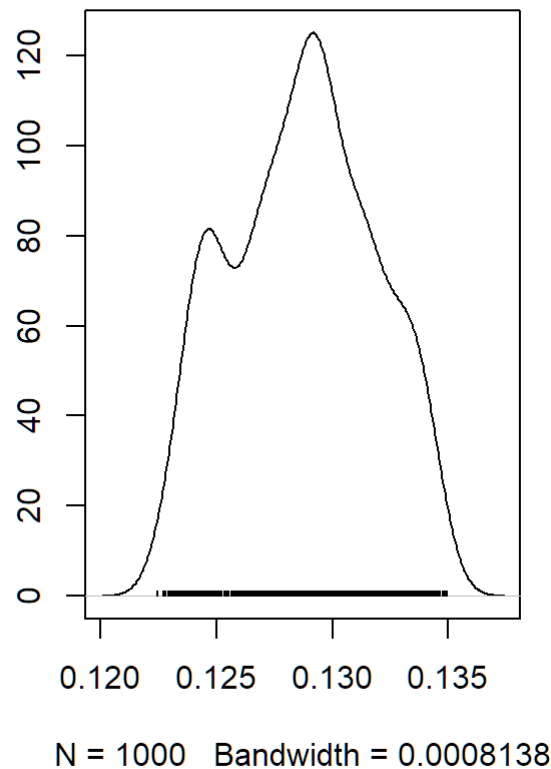
```
plot(a$mcmc[,6])
```

Trace of var1

Iterations

Density of var1

N = 1000   Bandwidth = 0.005931

Add a new chunk by clicking the *Insert Chunk* button on the toolbar or by pressing *Ctrl+Alt+I*.

When you save the notebook, an HTML file containing the code and output will be saved alongside it (click the *Preview* button or press *Ctrl+Shift+K* to preview the HTML file).

The preview shows you a rendered HTML copy of the contents of the editor. Consequently, unlike *Knit*, *Preview* does not run any R code chunks. Instead, the output of the chunk when it was last run in the editor is displayed.