Midterm

Xinyu LIU

2020/5/5

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
source('ama.R')
library(glmnet)
```

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

```
## Loaded glmnet 3.0-2
```

```
source('Lassosim.R')
wear_data=read.table("WEAR.DAT",header=TRUE)

y=cbind(wear_data[,5],wear_data[,6],wear_data[,7])
y1 <- factor(wear_data[,2])
y2 <- factor(wear_data[,3])
y3 <- factor(wear_data[,4])
m2=manova(y~y1+y2+y3+y1*y2+y2*y3+y1*y3+y1*y2*y3)
m2
```

```
## Call:
##    manova(y ~ y1 + y2 + y3 + y1 * y2 + y2 * y3 + y1 * y3 + y1 *
##     y2 * y3)
##
## Terms:
##                        y1       y2       y3    y1:y2    y2:y3    y1:y3
## resp 1           26268.17  6800.67   170.67  3952.67   400.17    10.67
## resp 2            5017.04 70959.38   260.04    57.04   145.04    77.04
## resp 3            1441.50 48240.67     6.00     0.17   294.00   337.50
## Deg. of Freedom         1        1        1        1        1        1
##                  y1:y2:y3 Residuals
## resp 1             121.50  13683.33
## resp 2              45.37  15936.67
## resp 3               4.17   5715.33
## Deg. of Freedom         1        16
##
## Residual standard errors: 29.24395 31.56013 18.89996
## Estimated effects may be unbalanced
```

```
summary(m2,test="Wilks")
```

```
##           Df   Wilks approx F num Df den Df    Pr(>F)
## y1         1 0.23414   15.264      3     14 0.0001081 ***
## y2         1 0.04680   95.038      3     14 1.514e-09 ***
## y3         1 0.89485    0.548      3     14 0.6573716
## y1:y2      1 0.50355    4.601      3     14 0.0192682 *
```

```
## y2:y3      1 0.94904    0.251      3     14 0.8595847
## y1:y3      1 0.87284    0.680      3     14 0.5788014
## y1:y2:y3   1 0.96542    0.167      3     14 0.9167541
## Residuals 16
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

1. The table is shown above.
2. See the table above. According to the Wilks test, we can identify factor1(P) and factor2(S) as well as the interaction term P*S are significant on the 5% level.
3. The p-value of the three-way interaction is 0.9167541, therefore it's not signifiant.
4. The term P*S is significant at 5% level with p-value being 0.019.
5. P and S are significant given there very low p-value.

Including Plots

You can also embed plots, for example:

```
# covariance matrices test
temp_data=read.table("TEMPERATURE.DAT",header=TRUE)
y <- temp_data[,1:3]
x <- temp_data[,4:6]
colnames(x)<-colnames(y)
nv = c(dim(x)[1],dim(y)[1])
data = rbind(x,y)
BoxM(data,nv)
```

```
## [1] "determinant"
## [1] 11891.15
## [1] "determinant"
## [1] 11284.97
## Test result:
##                  [,1]
## Box.M-C 4.349986e+01
## p.value 9.287392e-08
```

1. According to the BoxM test, p value is very small and we can reject the null hypothesis, meaning the covariance matrices of Y1 and Y2 are different.

```
# mean test
Behrens(x,y)
```

```
## Estimate of v:  86.2884
## Test result:
##      T2-stat   p.value
## [1,]   115.4 1.554e-15
```

2. According to the Behrens test, p value is very small and we can reject the null hypothesis, meaning the mean matrices of Y1 and Y2 are different.

```
confreg(y-x)
```

```
## [1] "C.R. based on T^2"
##              [,1]       [,2]
```

```
## [1,]  -6.12281722  -1.529357
## [2,]  -0.03385702   2.425161
## [3,] -21.58964103 -12.366881
## [1] "CR based on individual t"
##             [,1]       [,2]
## [1,]  -5.3802952  -2.271879
## [2,]   0.3636375   2.027667
## [3,] -20.0988036 -13.857718
## [1] "CR based on Bonferroni"
##             [,1]       [,2]
## [1,]  -5.7450424  -1.907131
## [2,]   0.1683773   2.222927
## [3,] -20.8311440 -13.125378
## [1] "Asymp. simu. CR"
##              [,1]       [,2]
## [1,]  -5.98325695  -1.668917
## [2,]   0.04085381   2.350451
## [3,] -21.30943163 -12.647090
```

```
m4 <-mmlr(y,x)
```

```
## Beta-Hat matrix:
##        y1     y2       y3
##     2.531 12.696 -101.769
## y1  0.984  0.438    2.514
## y2 -0.175  0.271   -0.232
## y3  0.038 -0.001    0.333
## LS residual covariance matrix:
##        y1     y2     y3
## y1 29.128  2.091 21.018
## y2  2.091  2.726 10.207
## y3 21.018 10.207 61.653
## Individual LSE of the parameter
##       Estimate stand.Err t-ratio p-value
##  [1,]    2.531    29.527   0.086   0.932
##  [2,]    0.984     0.354   2.782   0.008
##  [3,]   -0.175     0.379  -0.462   0.647
##  [4,]    0.038     0.113   0.342   0.734
##  [5,]   12.696     9.033   1.406   0.167
##  [6,]    0.438     0.108   4.048   0.000
##  [7,]    0.271     0.116   2.341   0.024
##  [8,]   -0.001     0.034  -0.029   0.977
##  [9,] -101.769    42.958  -2.369   0.023
## [10,]    2.514     0.515   4.885   0.000
## [11,]   -0.232     0.551  -0.422   0.675
## [12,]    0.333     0.164   2.032   0.048
## ===================
## Test for overall mmlr:
## Test statistic, df,  and p-value:  98.79071 9 0
## ===================
## [1] "Testing individual regressor"
##      regressor test-stat p-value
## [1,]         1   18.4629   4e-04
##      regressor test-stat p-value
## [1,]         2   15.7809  0.0013
```

```
##      regressor test-stat p-value
## [1,]         3   11.4251  0.0096
```

4. See the detailed regression coefficients above. Note that this regression is significant.

```
names(m4)
```

```
## [1] "beta"      "residuals" "sigma"     "ZtZinv"    "y"         "z"
## [7] "intercept"
```

```
m4$beta%*%c(90.7,70.1,109.5)
```

```
##            [,1]
##    -10024.20901
## y1    395.30995
## y2    -22.30073
## y3     39.85636
```

```
mmlrInt(m4,c(90.7,70.1,190.5))
```

```
## at predictors:  1 90.7 70.1 190.5
## Point prediction:
##      y1      y2      y3
##  86.870  71.253 173.386
## Simultaneous C.I. with prob 0.95
##          [,1]     [,2]
## [1,]  84.3793  89.3609
## [2,]  70.4912  72.0151
## [3,] 169.7624 177.0101
## Simultaneous P.I. with prob 0.95
##          [,1]     [,2]
## [1,]  69.7976 103.9426
## [2,]  66.0305  76.4757
## [3,] 148.5479 198.2246
```

```
z<-temp_data[,7:9]

m7<-mmlr(z,y,constant=T)
```

```
## Beta-Hat matrix:
##         y7      y8     y9
##    100.611 -58.653 88.020
## y1  -0.025  -0.264  0.045
## y2  -0.074   3.647 10.602
## y3   0.009  -0.747 -2.599
## LS residual covariance matrix:
##       y7      y8      y9
## y7 1.499   1.078   7.475
## y8 1.078  58.451 118.004
## y9 7.475 118.004 379.548
## Individual LSE of the parameter
##       Estimate stand.Err t-ratio p-value
##  [1,]  100.611     7.829  12.851   0.000
##  [2,]   -0.025     0.041  -0.619   0.539
##  [3,]   -0.074     0.159  -0.464   0.645
##  [4,]    0.009     0.031   0.292   0.771
##  [5,]  -58.653    48.892  -1.200   0.237
##  [6,]   -0.264     0.256  -1.028   0.310
```

```
##  [7,]    3.647     0.995   3.665   0.001
##  [8,]   -0.747     0.196  -3.816   0.000
##  [9,]   88.020   124.587   0.706   0.484
## [10,]    0.045     0.653   0.068   0.946
## [11,]   10.602     2.536   4.181   0.000
## [12,]   -2.599     0.499  -5.211   0.000
## ===================
## Test for overall mmlr:
## Test statistic, df,  and p-value:  45.85019 9 6.420202e-07
## ===================
## [1] "Testing individual regressor"
##      regressor test-stat p-value
## [1,]         1    4.0228   0.259
##      regressor test-stat p-value
## [1,]         2    16.418   9e-04
##      regressor test-stat p-value
## [1,]         3   23.0869       0
```

Again this is a significant regression

```
m9<-mmlr(z,x+y,constant =T)
```

```
## Beta-Hat matrix:
##         y7      y8      y9
##    105.921 -46.136 160.692
## y1  -0.040  -0.503  -0.567
## y2  -0.074   1.957   4.956
## y3   0.018  -0.254  -1.000
## LS residual covariance matrix:
##        y7     y8      y9
## y7  1.458  2.035  10.461
## y8  2.035 36.125  82.746
## y9 10.461 82.746 321.234
## Individual LSE of the parameter
##       Estimate stand.Err t-ratio p-value
##  [1,]  105.921     7.007  15.117   0.000
##  [2,]   -0.040     0.033  -1.222   0.229
##  [3,]   -0.074     0.061  -1.220   0.229
##  [4,]    0.018     0.014   1.227   0.227
##  [5,] -46.136    34.871  -1.323   0.193
##  [6,]   -0.503     0.163  -3.091   0.004
##  [7,]    1.957     0.303   6.464   0.000
##  [8,]   -0.254     0.071  -3.563   0.001
##  [9,]  160.692   103.983   1.545   0.130
## [10,]   -0.567     0.486  -1.167   0.250
## [11,]    4.956     0.903   5.489   0.000
## [12,]   -1.000     0.212  -4.712   0.000
## ===================
## Test for overall mmlr:
## Test statistic, df,  and p-value:  75.47319 9 1.274314e-12
## ===================
## [1] "Testing individual regressor"
##      regressor test-stat p-value
## [1,]         1   12.2684  0.0065
##      regressor test-stat p-value
```

```
## [1,]         2   35.0188       0
##      regressor test-stat p-value
## [1,]         3   26.4917       0
```

Test statistics and pvalue show that this contribution is significant.

```
library(leaps)
temp_data=read.table("TEMPERATURE.DAT",header=TRUE)
y <- temp_data[,11]
x <- temp_data[,1:10]

x1=data.frame(x)
nn=lm(y~.,data=x1)
step(nn)
```

```
## Start:  AIC=187.98
## y ~ y1 + y2 + y3 + y4 + y5 + y6 + y7 + y8 + y9 + y10
##
##        Df Sum of Sq    RSS    AIC
## - y7    1      0.00 1697.3 185.98
## - y8    1      0.16 1697.5 185.98
## - y2    1      3.19 1700.5 186.06
## - y1    1      5.06 1702.4 186.11
## - y5    1     17.94 1715.3 186.46
## - y3    1     21.43 1718.8 186.55
## - y4    1     47.20 1744.5 187.24
## <none>              1697.3 187.98
## - y10   1    151.33 1848.7 189.91
## - y6    1    195.56 1892.9 190.99
## - y9    1    413.06 2110.4 196.00
##
## Step:  AIC=185.98
## y ~ y1 + y2 + y3 + y4 + y5 + y6 + y8 + y9 + y10
##
##        Df Sum of Sq    RSS    AIC
## - y8    1      0.17 1697.5 183.98
## - y2    1      3.19 1700.5 184.06
## - y1    1      5.12 1702.5 184.12
## - y5    1     19.13 1716.5 184.49
## - y3    1     21.44 1718.8 184.55
## - y4    1     47.55 1744.9 185.25
## <none>              1697.3 185.98
## - y10   1    154.45 1851.8 187.98
## - y6    1    237.12 1934.5 189.99
## - y9    1    497.33 2194.7 195.80
##
## Step:  AIC=183.98
## y ~ y1 + y2 + y3 + y4 + y5 + y6 + y9 + y10
##
##        Df Sum of Sq    RSS    AIC
## - y2    1      3.24 1700.8 182.07
## - y1    1      5.00 1702.5 182.12
## - y3    1     21.54 1719.0 182.56
## - y5    1     21.84 1719.3 182.57
## - y4    1     49.71 1747.2 183.31
```

```
## <none>              1697.5 183.98
## - y10   1    154.70 1852.2 185.99
## - y6    1    243.52 1941.0 188.15
## - y9    1   1151.67 2849.2 205.80
##
## Step:  AIC=182.07
## y ~ y1 + y3 + y4 + y5 + y6 + y9 + y10
##
##        Df Sum of Sq    RSS    AIC
## - y1    1      7.81 1708.6 180.28
## - y5    1     28.15 1728.9 180.82
## - y4    1     47.78 1748.5 181.34
## <none>              1700.8 182.07
## - y3    1    117.17 1817.9 183.13
## - y10   1    151.93 1852.7 184.00
## - y6    1    295.00 1995.8 187.43
## - y9    1   1353.40 3054.2 207.00
##
## Step:  AIC=180.28
## y ~ y3 + y4 + y5 + y6 + y9 + y10
##
##        Df Sum of Sq    RSS    AIC
## - y5    1     26.23 1734.8 178.98
## - y4    1     49.18 1757.7 179.59
## <none>              1708.6 180.28
## - y3    1    113.09 1821.7 181.23
## - y10   1    146.15 1854.7 182.06
## - y6    1    287.45 1996.0 185.43
## - y9    1   1398.05 3106.6 205.78
##
## Step:  AIC=178.98
## y ~ y3 + y4 + y6 + y9 + y10
##
##        Df Sum of Sq    RSS    AIC
## - y4    1     58.69 1793.5 178.51
## <none>              1734.8 178.98
## - y10   1    119.94 1854.7 180.06
## - y3    1    150.17 1885.0 180.80
## - y6    1    308.25 2043.0 184.50
## - y9    1   2596.38 4331.2 219.07
##
## Step:  AIC=178.51
## y ~ y3 + y6 + y9 + y10
##
##        Df Sum of Sq    RSS    AIC
## <none>              1793.5 178.51
## - y10   1     94.38 1887.9 178.87
## - y3    1     95.01 1888.5 178.88
## - y6    1    559.70 2353.2 189.00
## - y9    1   2797.26 4590.7 219.75
##
## Call:
## lm(formula = y ~ y3 + y6 + y9 + y10, data = x1)
```

```
##
## Coefficients:
## (Intercept)           y3           y6           y9          y10
##   131.98178     -0.17784      0.37840     -0.35718      0.01092
```

1. The result shows that: Step: AIC=178.51 y ~ y3 + y6 + y9 + y10

```
lm(formula = y ~ y3 + y6 + y9 + y10, data = x1)
```

```
##
## Call:
## lm(formula = y ~ y3 + y6 + y9 + y10, data = x1)
##
## Coefficients:
## (Intercept)           y3           y6           y9          y10
##   131.98178     -0.17784      0.37840     -0.35718      0.01092
```

The coefficients above match the influence

```
leaps(x,y,nbest=1)
```

```
## $which
##        1     2     3     4     5     6     7     8    9     A
## 1  FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE TRUE FALSE
## 2  FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE TRUE FALSE
## 3  FALSE FALSE  TRUE FALSE FALSE  TRUE FALSE FALSE TRUE FALSE
## 4  FALSE  TRUE FALSE FALSE FALSE  TRUE FALSE FALSE TRUE  TRUE
## 5  FALSE FALSE  TRUE  TRUE FALSE  TRUE FALSE FALSE TRUE  TRUE
## 6  FALSE FALSE  TRUE  TRUE  TRUE  TRUE FALSE FALSE TRUE  TRUE
## 7   TRUE FALSE  TRUE  TRUE  TRUE  TRUE FALSE FALSE TRUE  TRUE
## 8   TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE FALSE TRUE  TRUE
## 9   TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE TRUE  TRUE
## 10  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE TRUE  TRUE
##
## $label
##  [1] "(Intercept)" "1"           "2"           "3"           "4"
##  [6] "5"           "6"           "7"           "8"           "9"
## [11] "A"
##
## $size
##  [1]  2  3  4  5  6  7  8  9 10 11
##
## $Cp
##  [1] 24.2969638  2.6800571  0.9285976  0.8687476  1.7722073  3.2314283
##  [7]  5.0703038  7.0034959  9.0000226 11.0000000
```

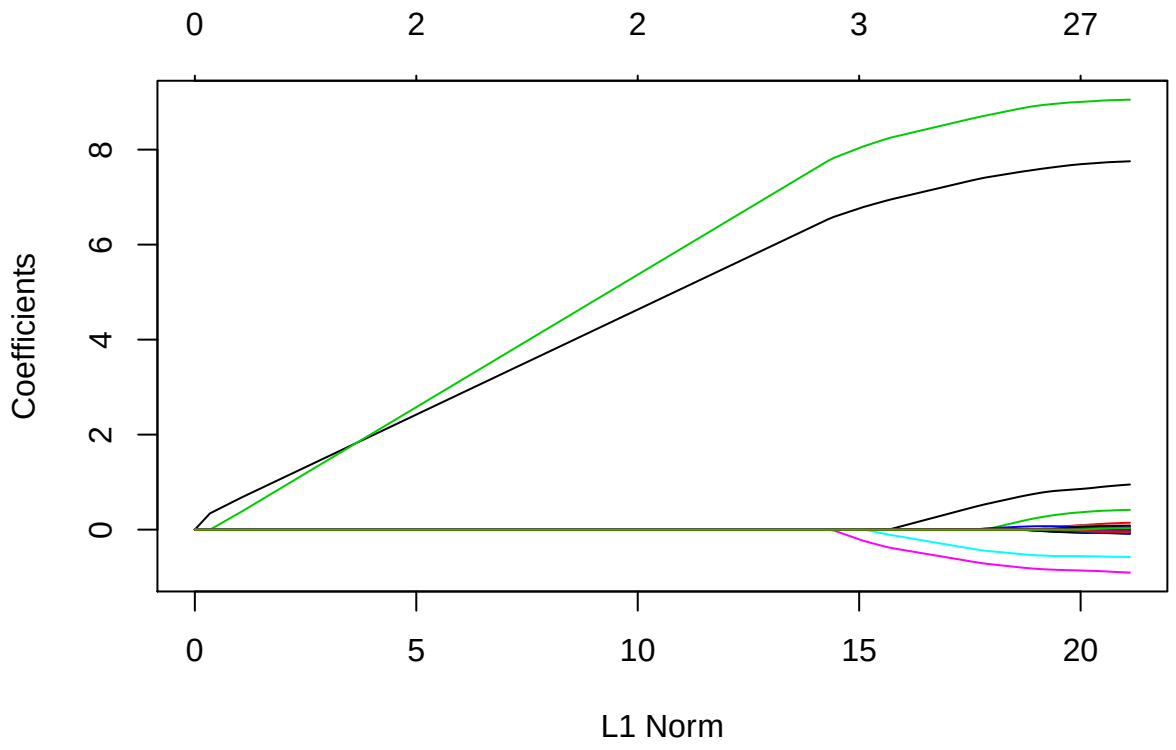
3. According to Cp value, closest but smaller, we conclude the best model is y3, y6, y9
4. from leaps result, two predictor are y6, y9
5. from leaps result, two predictor are y3, y6, y9

```
# Lasso regression
library(glmnet)
source('Lassosim.R')
da = read.csv(file = 'ProblemI.csv')
y1 <- as.numeric(da[,1])
x1 <- matrix(unlist(da[,2:501]), ncol = 500, byrow = FALSE)
```

```
require(glmnet)
m2 <- glmnet(x1,y1,alpha=1,nfolds = 10)
cv.m2 <- cv.glmnet(x1,y1,alpha=1,nfolds = 10)
cv.m2$lambda.min
```

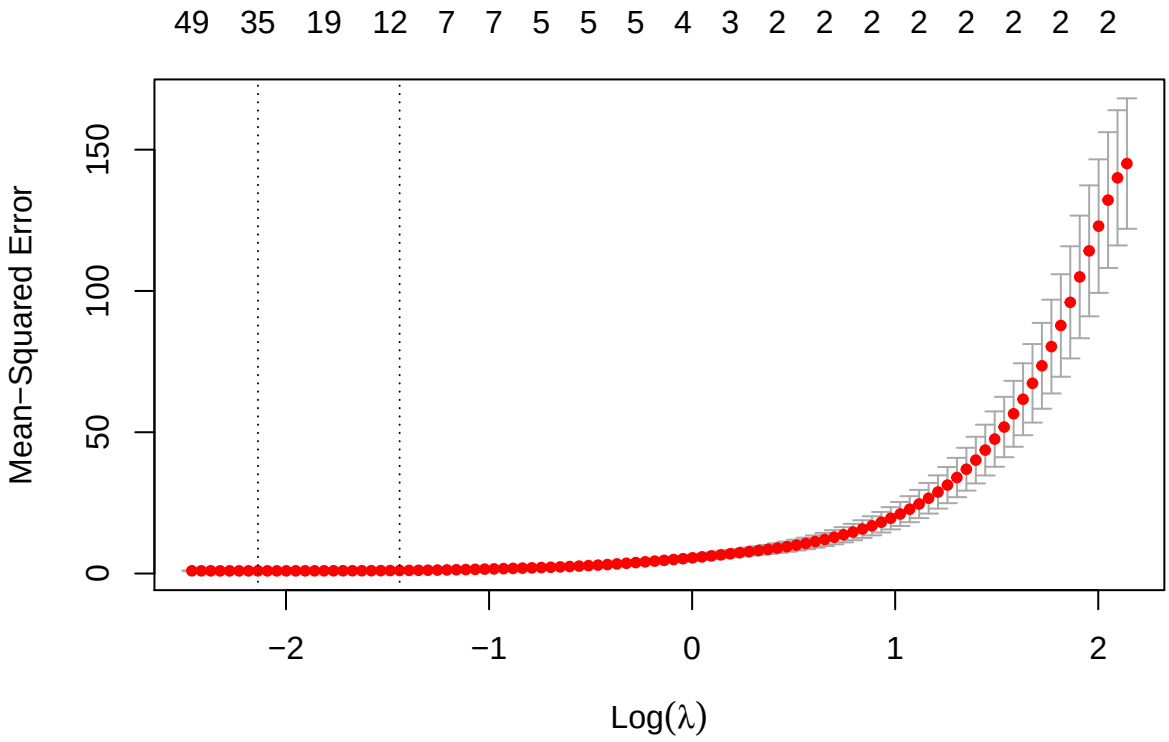
```
## [1] 0.1178664
```

```
plot(m2)
```

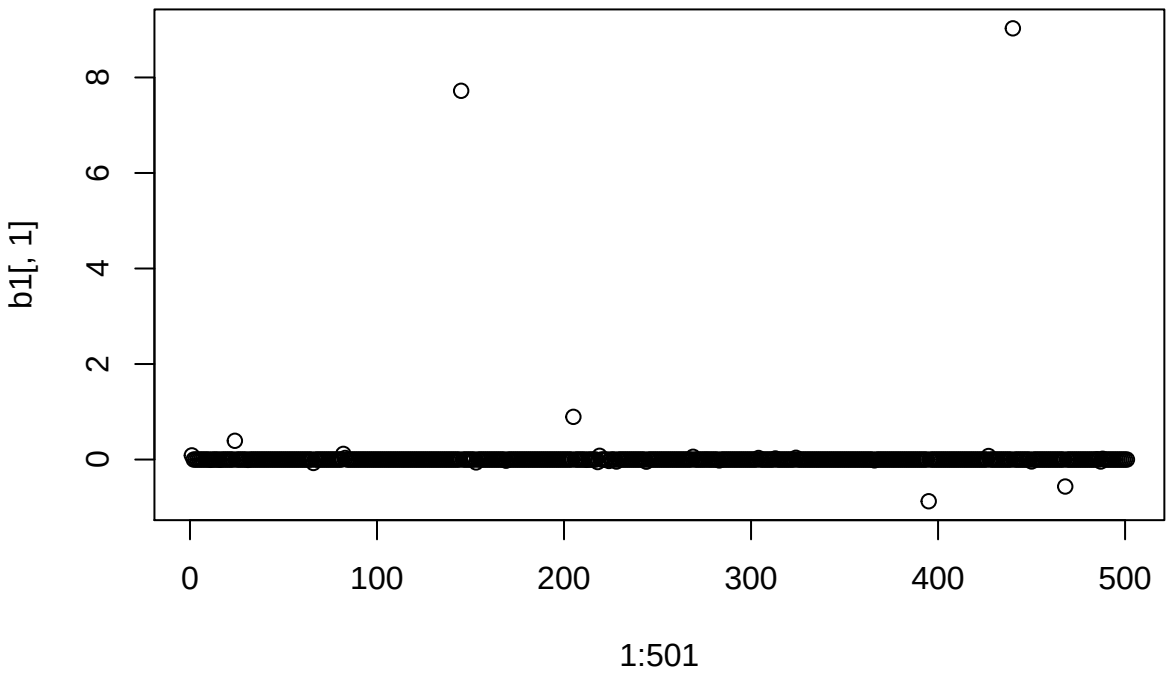


index are : 23 144 204 394 439 467

```
plot(cv.m2)
```




```
b1 <-coef(m2,s=cv.m2$lambda.min)
plot(1:501,b1[,1])
```




```
idx <- c(1:500)[abs(b1[2:501,1])>0.2]
```