Assn2

Shivam

2023-02-12

```r
#if(!is.null(dev.list())) dev.off()
#cat("\014")
#rm(list=ls())
options(scipen=9)
```

```r
if(!require(pastecs)){install.packages("pastecs")}
```

```
## Loading required package: pastecs
```

```r
library("pastecs")

if(!require(lattice)){install.packages("lattice")}
```

```
## Loading required package: lattice
```

```r
library("lattice")
```

1.Data Transformation and Preparation

1. Initial Transformation a. Rename all variables with your initials appended (just as was done in assignment 1 and 2)

```r
setwd("C:/Users/holys/OneDrive/Desktop/Data Analytics,Mathamatics,Algor/Assign02")

test_SJ <- read.table("PROG8430-23W-Assign02.txt", sep=",", header=TRUE)

    test_SJ <- as.data.frame(test_SJ)
    head(test_SJ)
```

```
##   Index Manufacturer Server Conn RC  ST  SMBR   SMBT BR BT   UC      FA
## 1     1         Lled MB5755 5571 10 253 39806  91685 11 17 2000 1526223
## 2     2         Lled MB3406 6684 12 282 56410 115100 15 22 2000 1799882
## 3     3       Ovonel RQ8547 4790 10  83 55891  98534 15 18 2000 1361793
## 4     4         Lled MB3406 6163 10 247 49546 116361 14 22 2000 2365969
## 5     5         Lled MB5755 8939 12 252 61578 104176 17 20 2000 2222282
## 6     6       Ovonel RP6354 7446  8 263 47692 102983 13 19 2000 2006826
```

```r
colnames(test_SJ) <- paste(colnames(test_SJ), "SJ", sep = "_")
 head(test_SJ)
```

```
##   Index_SJ Manufacturer_SJ Server_SJ Conn_SJ RC_SJ ST_SJ SMBR_SJ SMBT_SJ BR_SJ
## 1        1            Lled    MB5755    5571    10   253   39806   91685    11
## 2        2            Lled    MB3406    6684    12   282   56410  115100    15
## 3        3          Ovonel    RQ8547    4790    10    83   55891   98534    15
## 4        4            Lled    MB3406    6163    10   247   49546  116361    14
## 5        5            Lled    MB5755    8939    12   252   61578  104176    17
## 6        6          Ovonel    RP6354    7446     8   263   47692  102983    13
##   BT_SJ UC_SJ   FA_SJ
## 1    17  2000 1526223
## 2    22  2000 1799882
## 3    18  2000 1361793
## 4    22  2000 2365969
## 5    20  2000 2222282
## 6    19  2000 2006826
```

b. Transform character variables to factor variables.

```r
test_SJ$Manufacturer_SJ <- factor(test_SJ$Manufacturer_SJ)
test_SJ$Server_SJ       <- factor(test_SJ$Server_SJ)
```

2. Reduce Dimensionality

a. Apply the Missing Value Filter to remove appropriate columns
  of data.

```r
summary(test_SJ)
```

```
##     Index_SJ     Manufacturer_SJ  Server_SJ        Conn_SJ          RC_SJ
##  Min.   :    1   Lled   :41078    MB3406:15610   Min.   : 1133   Min.   :-7
##  1st Qu.:20540   Ovonel:41078    MB5755:17663   1st Qu.: 5914   1st Qu.: 8
##  Median :41079                   MG9696: 7805   Median : 6792   Median :10
##  Mean   :41079                   RL3777:11913   Mean   : 6793   Mean   :10
##  3rd Qu.:61617                   RP6354:16431   3rd Qu.: 7668   3rd Qu.:12
##  Max.   :82156                   RQ8547: 6162   Max.   :12321   Max.   :27
##                                  RX8838: 6572
##      ST_SJ          SMBR_SJ         SMBT_SJ           BR_SJ
##  Min.   :  9.0   Min.   : 8455   Min.   : 30139   Min.   : 2.00
##  1st Qu.:215.0   1st Qu.:43180   1st Qu.: 90356   1st Qu.:12.00
##  Median :242.0   Median :49952   Median : 99940   Median :14.00
##  Mean   :237.4   Mean   :49969   Mean   : 99975   Mean   :13.63
##  3rd Qu.:264.0   3rd Qu.:56723   3rd Qu.:109513   3rd Qu.:15.00
##  Max.   :433.0   Max.   :93437   Max.   :158247   Max.   :25.00
##
##      BT_SJ           UC_SJ          FA_SJ
##  Min.   : 6.00   Min.   :2000   Min.   :   4412
##  1st Qu.:17.00   1st Qu.:2000   1st Qu.:1399343
##  Median :19.00   Median :2000   Median :1671136
```

```
##  Mean   :18.75   Mean   :2000   Mean   :1685073
##  3rd Qu.:21.00   3rd Qu.:2000   3rd Qu.:1954341
##  Max.   :30.00   Max.   :2001   Max.   :3656283
##
```

-> As we can see from the summary of the data set, there is no such record which is missing, so there is no change or action to be taken after missing value filter.

b. Apply the Low Variance Filter to remove appropriate columns of data.

```
stat.desc(test_SJ)  #Consider coef of var
```

```
##                       Index_SJ Manufacturer_SJ Server_SJ           Conn_SJ
## nbr.val           82156.0000000              NA        NA     82156.0000000
## nbr.null              0.0000000              NA        NA         0.0000000
## nbr.na                0.0000000              NA        NA         0.0000000
## min                   1.0000000              NA        NA      1133.0000000
## max               82156.0000000              NA        NA     12321.0000000
## range             82155.0000000              NA        NA     11188.0000000
## sum          3374845246.0000000              NA        NA 558117390.0000000
## median            41078.5000000              NA        NA      6792.0000000
## mean              41078.5000000              NA        NA      6793.3856322
## SE.mean              82.7430762              NA        NA         4.5418248
## CI.mean.0.95        162.1758387              NA        NA         8.9019441
## var           562474207.6666666              NA        NA   1694728.1198515
## std.dev           23716.5386949              NA        NA      1301.8172375
## coef.var              0.5773468              NA        NA         0.1916301
##                      RC_SJ            ST_SJ           SMBR_SJ
## nbr.val       82156.00000000    82156.0000000     82156.0000000
## nbr.null          3.00000000        0.0000000         0.0000000
## nbr.na            0.00000000        0.0000000         0.0000000
## min              -7.00000000        9.0000000      8455.0000000
## max              27.00000000      433.0000000     93437.0000000
## range            34.00000000      424.0000000     84982.0000000
## sum          821839.00000000 19506663.0000000 4105257983.0000000
## median           10.00000000      242.0000000     49952.0000000
## mean             10.00339598      237.4344296     49969.0586567
## SE.mean           0.01106041        0.1447164        34.8499149
## CI.mean.0.95      0.02167833        0.2836431        68.3055844
## var              10.05036885     1720.5789760   99779823.2606699
## std.dev           3.17023167       41.4798623      9988.9850966
## coef.var          0.31691554        0.1747003         0.1999034
##                        SMBT_SJ             BR_SJ             BT_SJ
## nbr.val          82156.0000000   82156.000000000   82156.000000000
## nbr.null             0.0000000       0.000000000       0.000000000
## nbr.na               0.0000000       0.000000000       0.000000000
## min              30139.0000000       2.000000000       6.000000000
## max             158247.0000000      25.000000000      30.000000000
## range           128108.0000000      23.000000000      24.000000000
## sum         8213533053.0000000 1119635.000000000 1540088.000000000
```

```
## median           99940.0000000      14.000000000      19.000000000
## mean             99974.8411924      13.628158625      18.745898048
## SE.mean             49.3149953       0.009557366       0.009305629
## CI.mean.0.95        96.6570387       0.018732369       0.018238966
## var          199800825.6032747       7.504395567       7.114276278
## std.dev          14135.0919913       2.739415187       2.667260069
## coef.var             0.1413865       0.201011396       0.142284998
##                              UC_SJ                 FA_SJ
## nbr.val          82156.00000000000       82156.0000000
## nbr.null             0.00000000000           0.0000000
## nbr.na               0.00000000000           0.0000000
## min               2000.00000000000        4412.0000000
## max               2001.00000000000     3656283.0000000
## range                1.00000000000     3651871.0000000
## sum          164312167.00000000000 138438884066.0000000
## median            2000.00000000000     1671136.0000000
## mean              2000.00203271824     1685073.3247237
## SE.mean              0.00015713747        1452.4037204
## CI.mean.0.95         0.00030798831        2846.7009225
## var                  0.00202861099 173306156834.8027344
## std.dev              0.04504010426      416300.5606948
## coef.var             0.00002252003           0.2470519
```

```
summary(test_SJ)
```

```
##     Index_SJ     Manufacturer_SJ  Server_SJ        Conn_SJ          RC_SJ
##  Min.   :    1   Lled  :41078    MB3406:15610   Min.   : 1133   Min.   :-7
##  1st Qu.:20540   Ovonel:41078    MB5755:17663   1st Qu.: 5914   1st Qu.: 8
##  Median :41079                   MG9696: 7805   Median : 6792   Median :10
##  Mean   :41079                   RL3777:11913   Mean   : 6793   Mean   :10
##  3rd Qu.:61617                   RP6354:16431   3rd Qu.: 7668   3rd Qu.:12
##  Max.   :82156                   RQ8547: 6162   Max.   :12321   Max.   :27
##                                  RX8838: 6572
##      ST_SJ          SMBR_SJ         SMBT_SJ           BR_SJ
##  Min.   :  9.0   Min.   : 8455   Min.   : 30139   Min.   : 2.00
##  1st Qu.:215.0   1st Qu.:43180   1st Qu.: 90356   1st Qu.:12.00
##  Median :242.0   Median :49952   Median : 99940   Median :14.00
##  Mean   :237.4   Mean   :49969   Mean   : 99975   Mean   :13.63
##  3rd Qu.:264.0   3rd Qu.:56723   3rd Qu.:109513   3rd Qu.:15.00
##  Max.   :433.0   Max.   :93437   Max.   :158247   Max.   :25.00
##
##      BT_SJ           UC_SJ          FA_SJ
##  Min.   : 6.00   Min.   :2000   Min.   :   4412
##  1st Qu.:17.00   1st Qu.:2000   1st Qu.:1399343
##  Median :19.00   Median :2000   Median :1671136
##  Mean   :18.75   Mean   :2000   Mean   :1685073
##  3rd Qu.:21.00   3rd Qu.:2000   3rd Qu.:1954341
##  Max.   :30.00   Max.   :2001   Max.   :3656283
##
```

-> It is observed that the UC_SJ has a a negligible coefficient of variance value so we can eliminate it because it will not make a big difference in analyzing and we reduce the dimensions for the ease to study it and make observations on it.

```r
test_SJ <- test_SJ[-c(11)]

head(test_SJ,3)
```

```
##   Index_SJ Manufacturer_SJ Server_SJ Conn_SJ RC_SJ ST_SJ SMBR_SJ SMBT_SJ BR_SJ
## 1        1            Lled    MB5755    5571    10   253   39806   91685    11
## 2        2            Lled    MB3406    6684    12   282   56410  115100    15
## 3        3          Ovonel    RQ8547    4790    10    83   55891   98534    15
##   BT_SJ   FA_SJ
## 1    17 1526223
## 2    22 1799882
## 3    18 1361793
```

```
c. Apply the High Correlation Filter to remove appropriate
  columns of data.
```

```r
test2_SJ <- test_SJ[-c(2:3)]
    head(test2_SJ)
```

```
##   Index_SJ Conn_SJ RC_SJ ST_SJ SMBR_SJ SMBT_SJ BR_SJ BT_SJ   FA_SJ
## 1        1    5571    10   253   39806   91685    11    17 1526223
## 2        2    6684    12   282   56410  115100    15    22 1799882
## 3        3    4790    10    83   55891   98534    15    18 1361793
## 4        4    6163    10   247   49546  116361    14    22 2365969
## 5        5    8939    12   252   61578  104176    17    20 2222282
## 6        6    7446     8   263   47692  102983    13    19 2006826
```

```r
cor(test2_SJ,method="spearman")
```

```
##               Index_SJ       Conn_SJ        RC_SJ         ST_SJ       SMBR_SJ
## Index_SJ  1.0000000000  0.0042298416  0.002637313  0.0039510097 -0.0072618256
## Conn_SJ   0.0042298416  1.0000000000  0.002553713 -0.0051813266 -0.0007441048
## RC_SJ     0.0026373131  0.0025537135  1.000000000  0.0040373249  0.0048533195
## ST_SJ     0.0039510097 -0.0051813266  0.004037325  1.0000000000 -0.0000386051
## SMBR_SJ  -0.0072618256 -0.0007441048  0.004853319 -0.0000386051  1.0000000000
## SMBT_SJ  -0.0069800768 -0.0009864149  0.003980611 -0.0005576250  0.7476258171
## BR_SJ    -0.0079967459 -0.0005187586  0.004454767 -0.0001081305  0.9938996733
## BT_SJ    -0.0063522475 -0.0016882428  0.004523766 -0.0008330024  0.7430656692
## FA_SJ     0.0005723258  0.0074740748 -0.002028123  0.0025137913  0.0016462715
##                SMBT_SJ         BR_SJ         BT_SJ         FA_SJ
## Index_SJ -0.0069800768 -0.0079967459 -0.0063522475  0.0005723258
## Conn_SJ  -0.0009864149 -0.0005187586 -0.0016882428  0.0074740748
## RC_SJ     0.0039806108  0.0044547674  0.0045237655 -0.0020281233
## ST_SJ    -0.0005576250 -0.0001081305 -0.0008330024  0.0025137913
## SMBR_SJ   0.7476258171  0.9938996733  0.7430656692  0.0016462715
## SMBT_SJ   1.0000000000  0.7438861976  0.9935613424  0.0014519063
## BR_SJ     0.7438861976  1.0000000000  0.7393055744  0.0015287512
## BT_SJ     0.9935613424  0.7393055744  1.0000000000  0.0019115184
## FA_SJ     0.0014519063  0.0015287512  0.0019115184  1.0000000000
```

-> It is clearly observed that the correlation of SMBR_SJ and SMBT_SJ is very high with BR_SJ and BT_SJ respectively,so we can eliminate two column maintaining their relativity in the data set.

```
test_SJ <- test_SJ[-c(9:10)]
head(test_SJ,3)
```

```
##   Index_SJ Manufacturer_SJ Server_SJ Conn_SJ RC_SJ ST_SJ SMBR_SJ SMBT_SJ
## 1        1            Lled    MB5755    5571    10   253   39806   91685
## 2        2            Lled    MB3406    6684    12   282   56410  115100
## 3        3          Ovonel    RQ8547    4790    10    83   55891   98534
##     FA_SJ
## 1 1526223
## 2 1799882
## 3 1361793
```

```
d. Drop any variables that do not contribute any useful analytical
  information at all.

-> As there is already a counter for rows, there is no need of a seperate column
 Index_SJ for indexing the table. So, lets remove it.
```

```
test_SJ <- test_SJ[-c(1)]
head(test_SJ,3)
```

```
##   Manufacturer_SJ Server_SJ Conn_SJ RC_SJ ST_SJ SMBR_SJ SMBT_SJ   FA_SJ
## 1            Lled    MB5755    5571    10   253   39806   91685 1526223
## 2            Lled    MB3406    6684    12   282   56410  115100 1799882
## 3          Ovonel    RQ8547    4790    10    83   55891   98534 1361793
```

```
3. Outliers

  a. Use an appropriate technique demonstrated in class to identify
   outliers.
```

```
if(!require(readxl)){install.packages("readxl")}
```

```
## Loading required package: readxl
```

```
library("readxl")

if(!require(pastecs)){install.packages("pastecs")}
library("pastecs")

if(!require(ggplot2)){install.packages("ggplot2")}
```

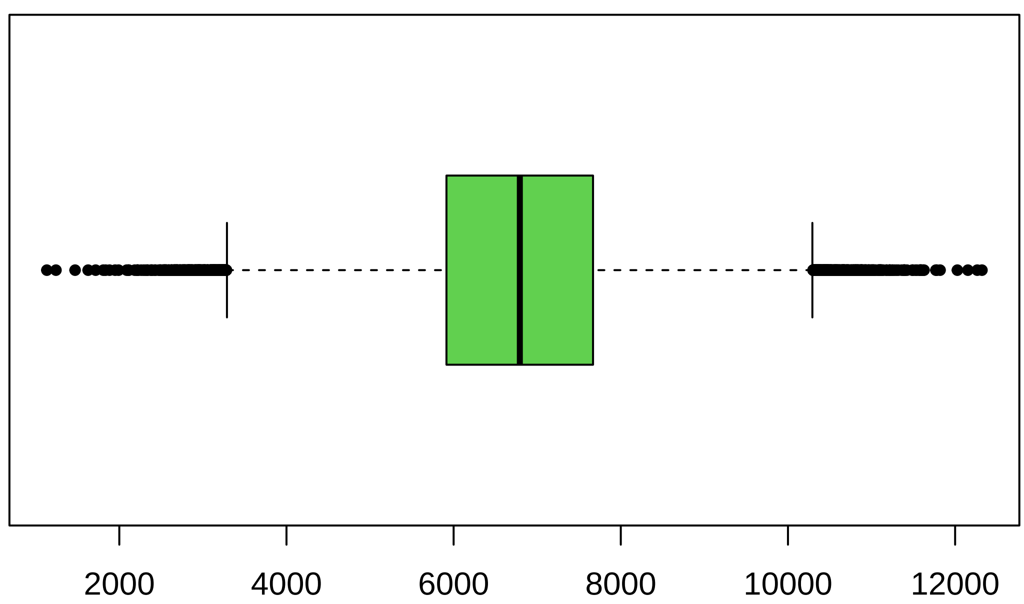
```
## Loading required package: ggplot2
```

```r
library("ggplot2")

if(!require(lattice)){install.packages("lattice")}
library("lattice")

if(!require(tinytex)){install.packages("tinytex")}
```

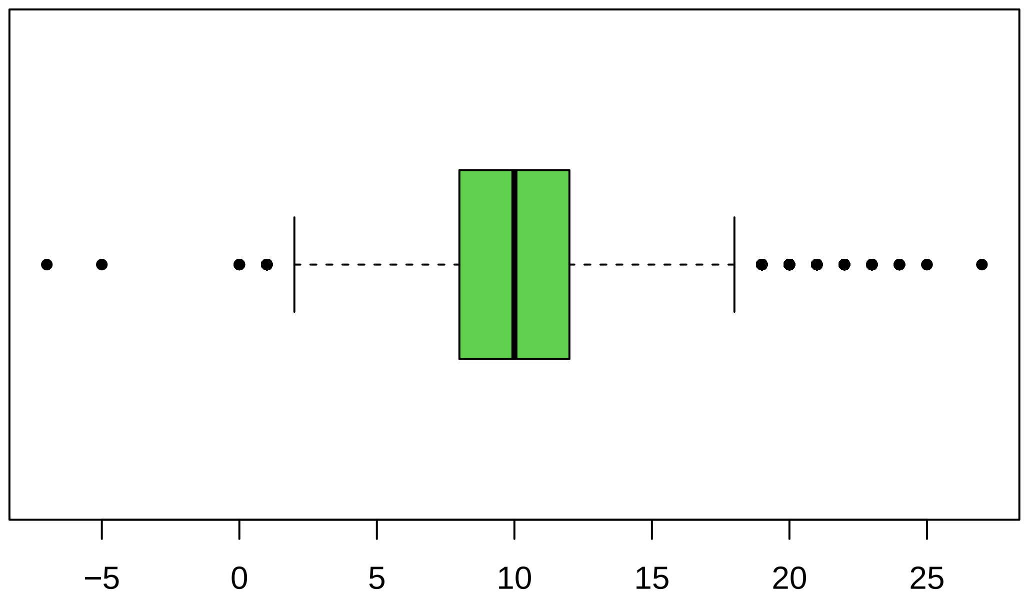
```
## Loading required package: tinytex
```

```r
library("tinytex")
```

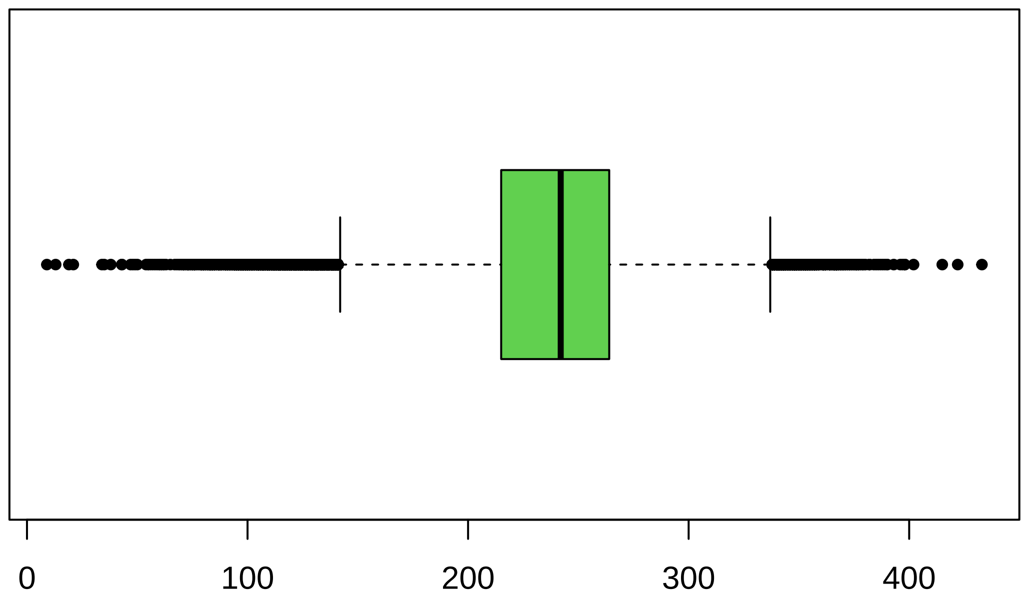
```r
boxplot(test_SJ$Conn_SJ, horizontal=TRUE, col=67, pch=20)
```



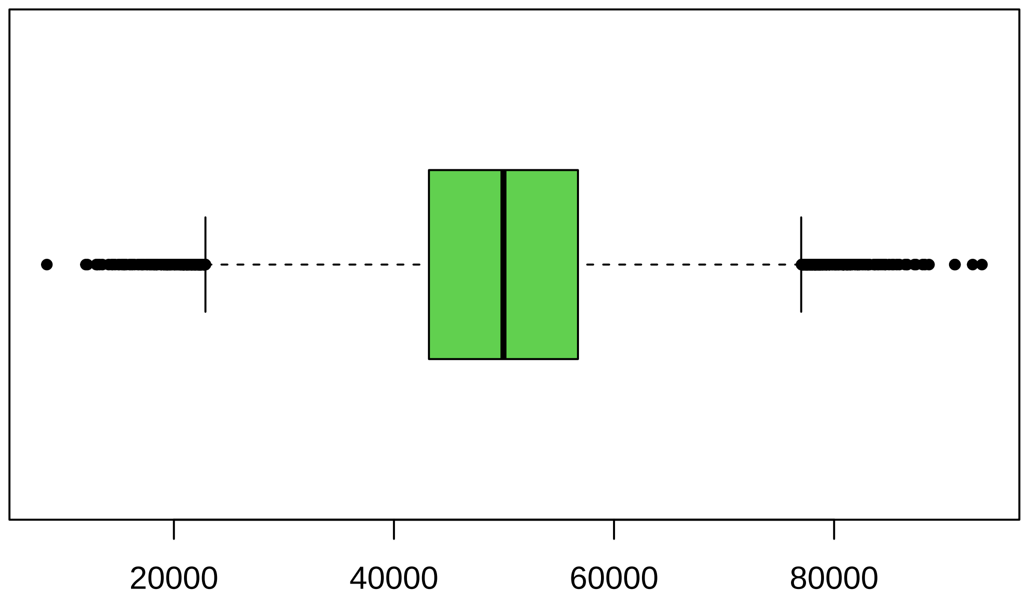

```r
boxplot(test_SJ$RC_SJ, horizontal=TRUE, col=67, pch=20)
```



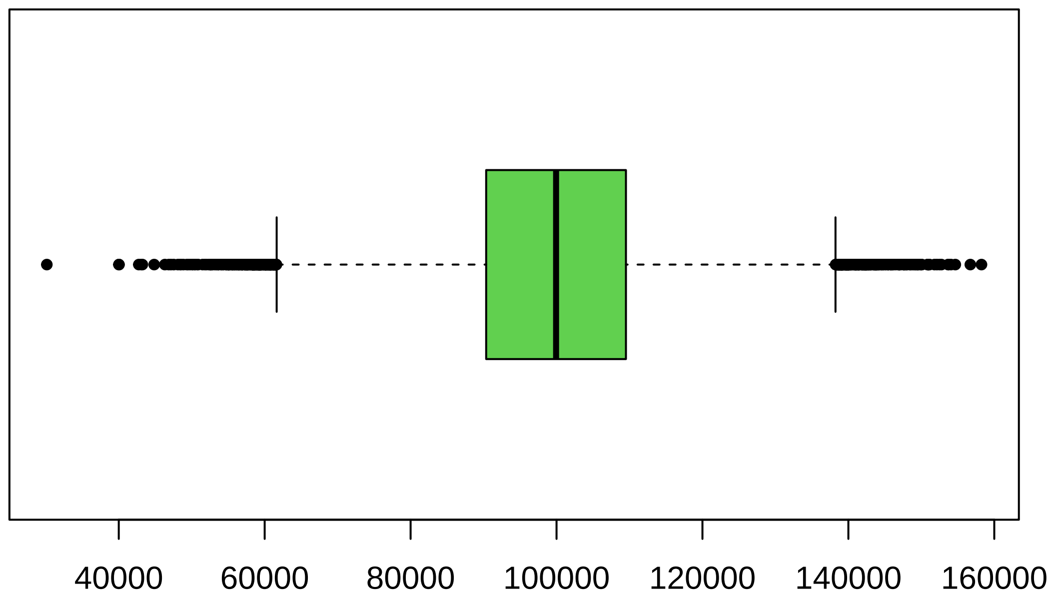

```r
boxplot(test_SJ$ST_SJ, horizontal=TRUE, col=67, pch=20)
```



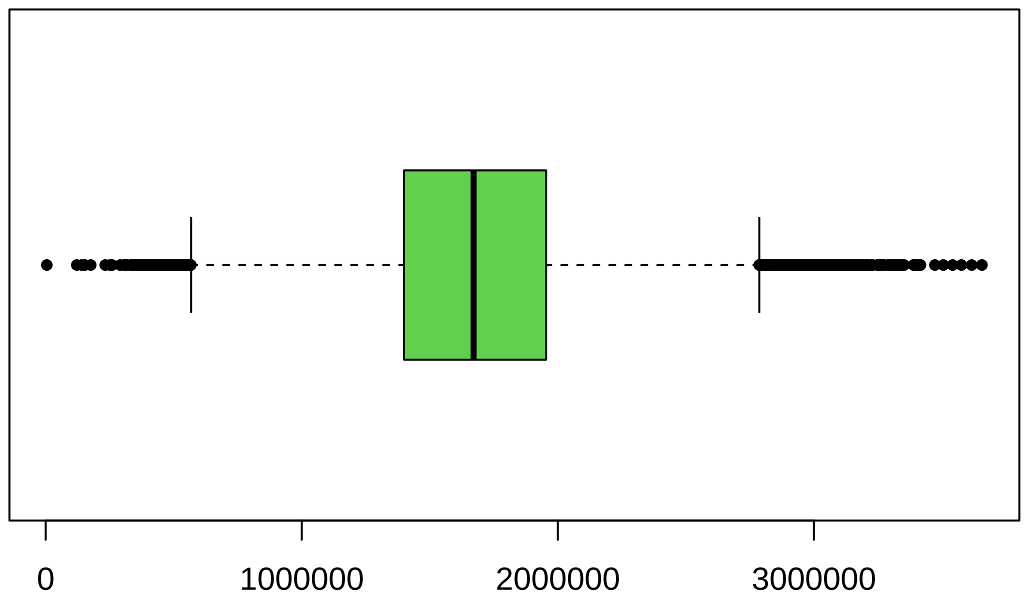

```
boxplot(test_SJ$SMBR_SJ, horizontal=TRUE, col=67, pch=20)
```




```
boxplot(test_SJ$SMBT_SJ, horizontal=TRUE, col=67, pch=20)
```

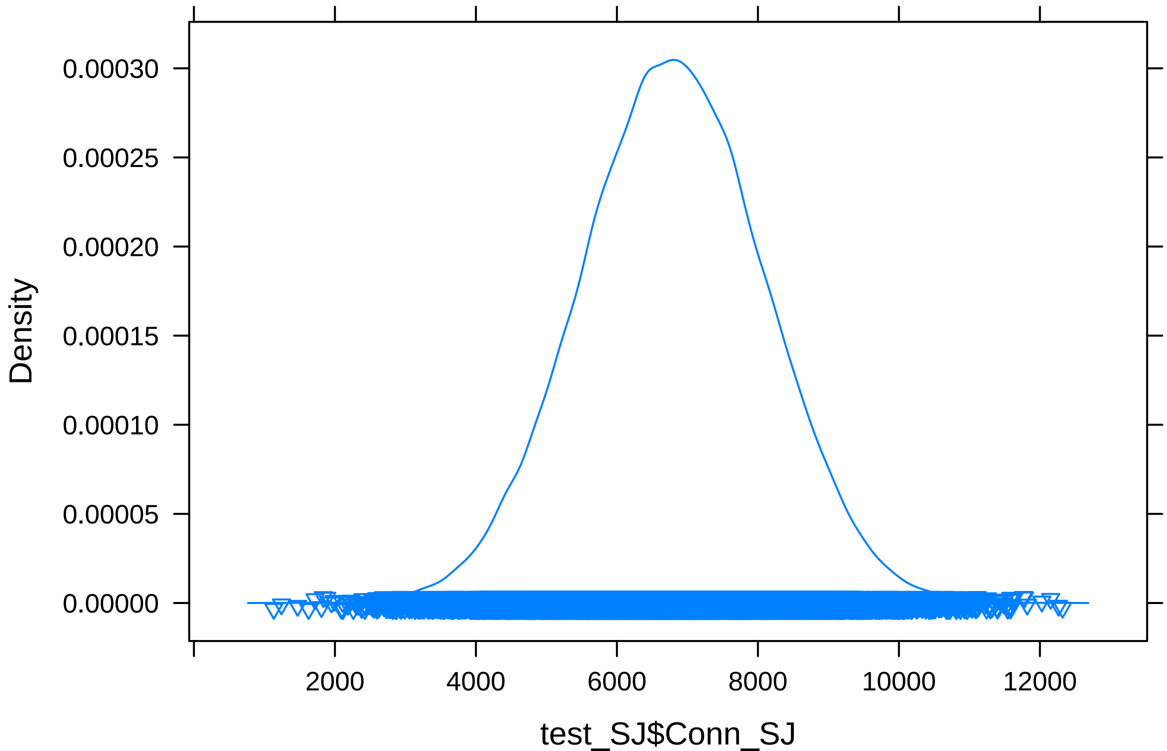



```r
boxplot(test_SJ$FA_SJ, horizontal=TRUE, col=67, pch=20)
```

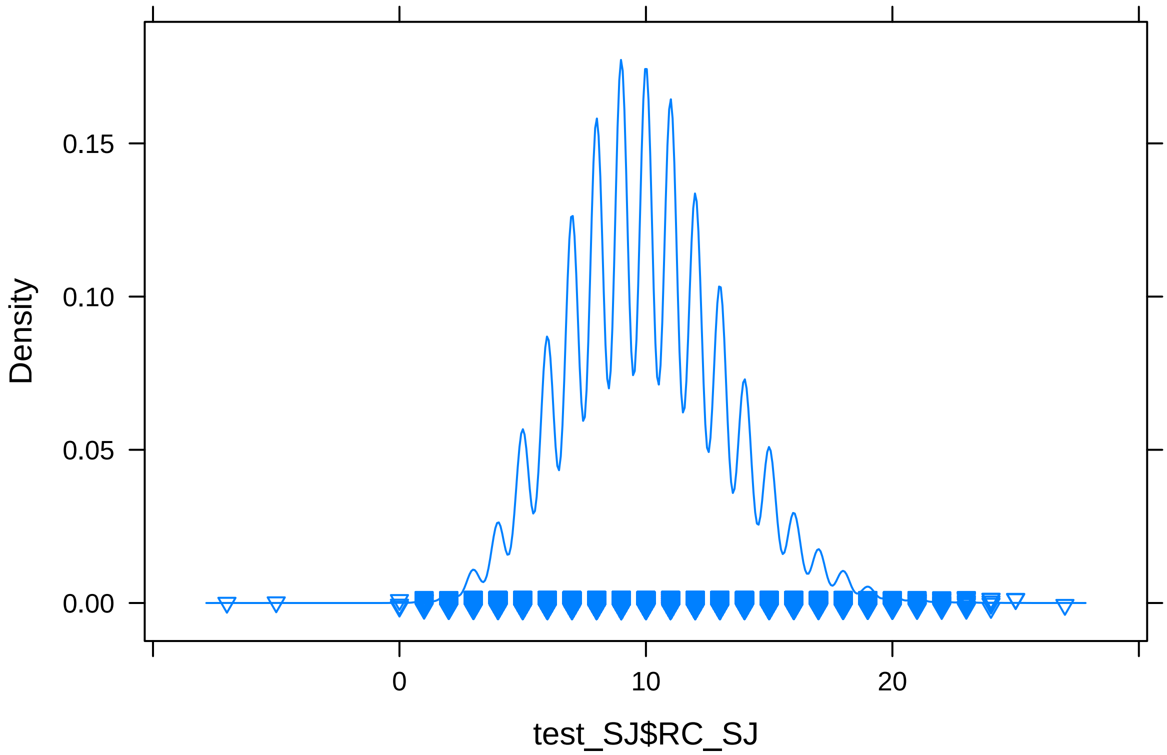


Density Plots for further details

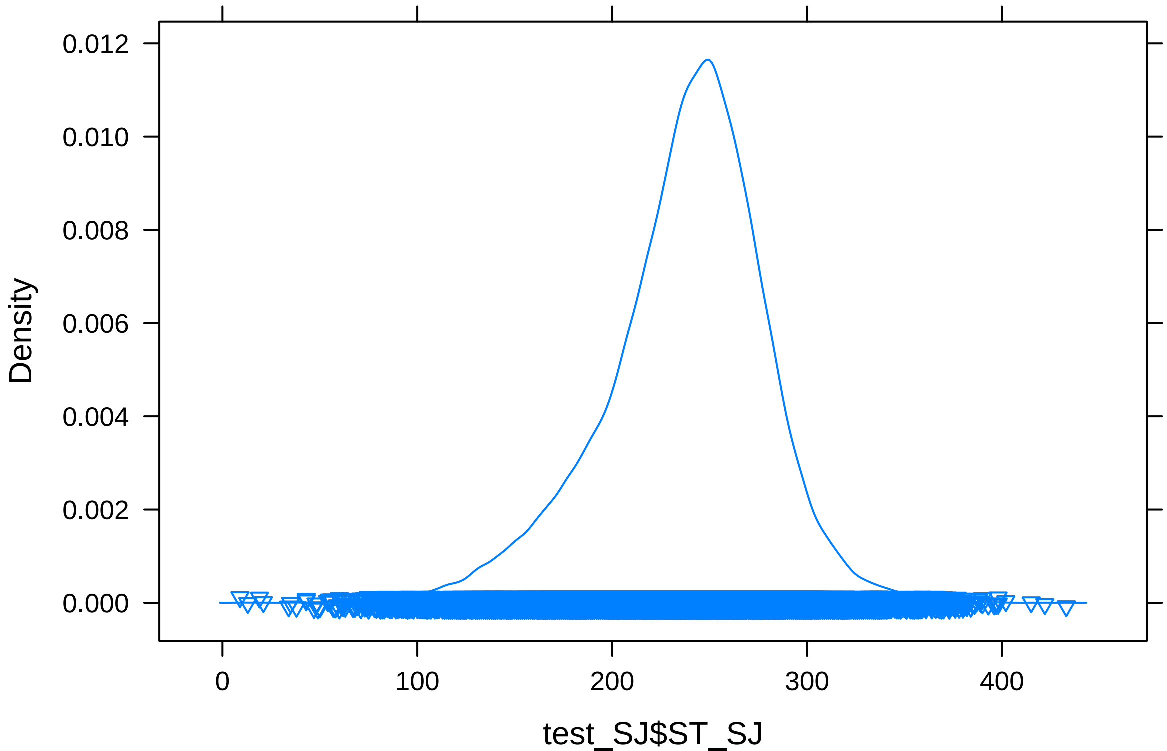
```
densityplot( ~ test_SJ$Conn_SJ, pch=6)
```



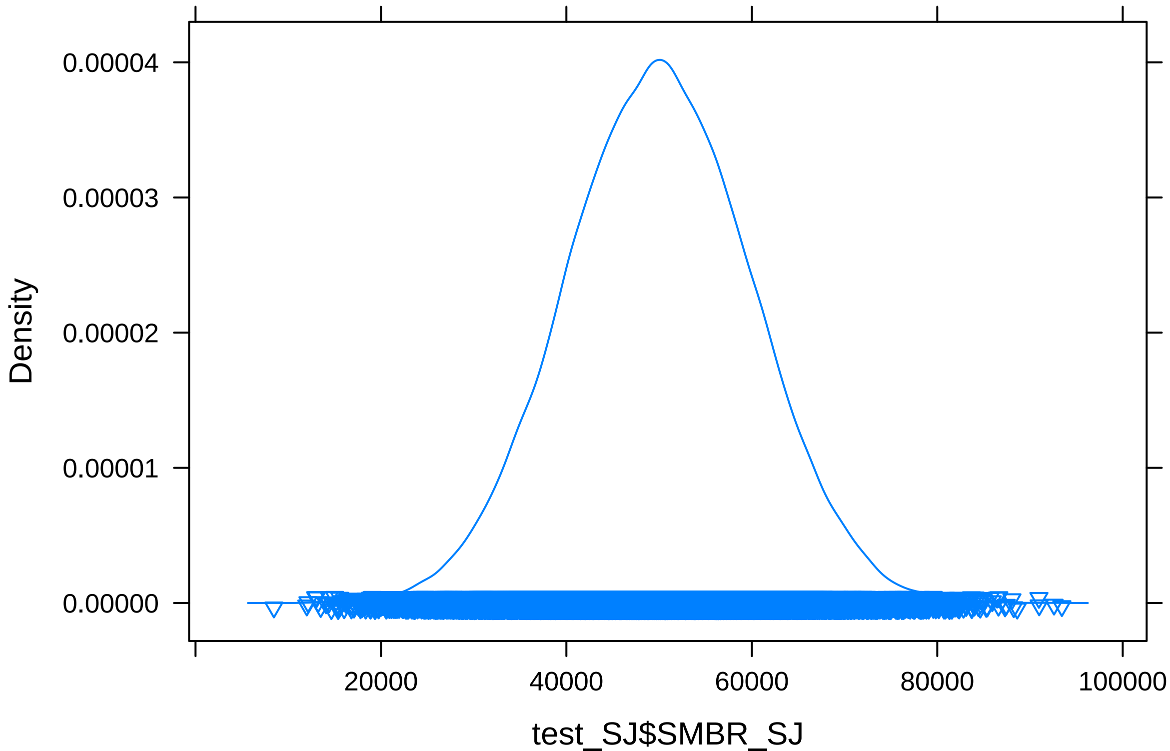

```
densityplot( ~ test_SJ$RC_SJ, pch=6)
```



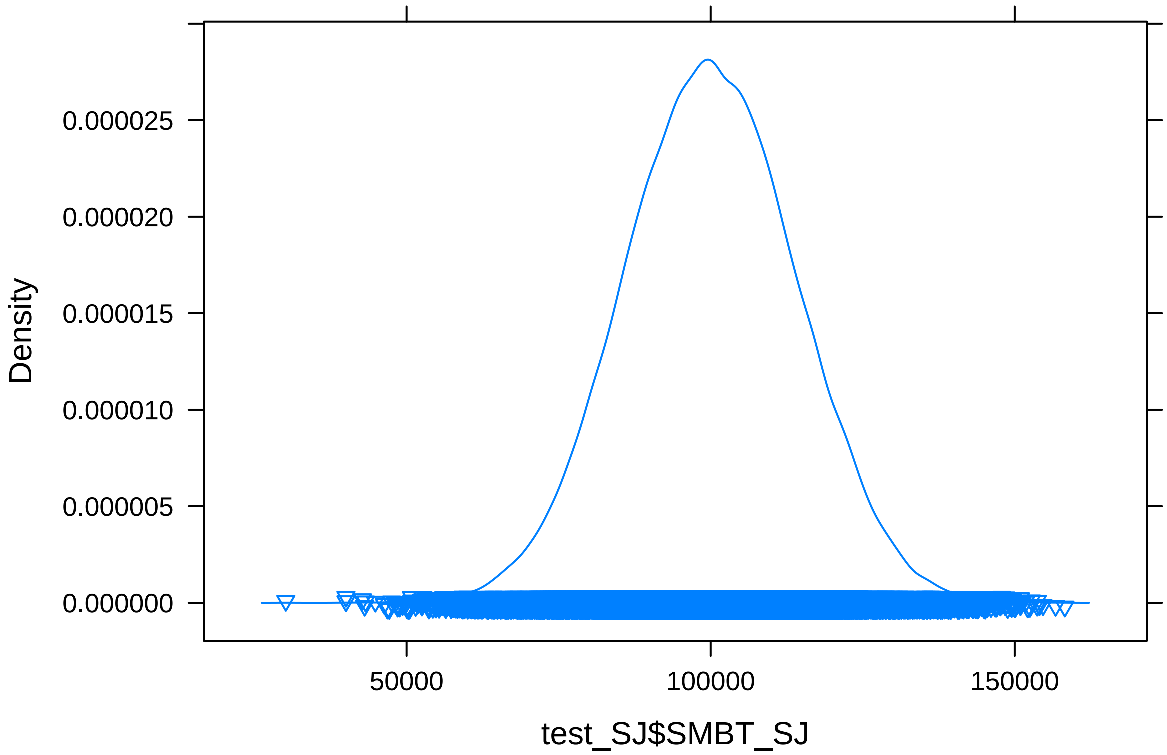

```
densityplot( ~ test_SJ$ST_SJ, pch=6)
```



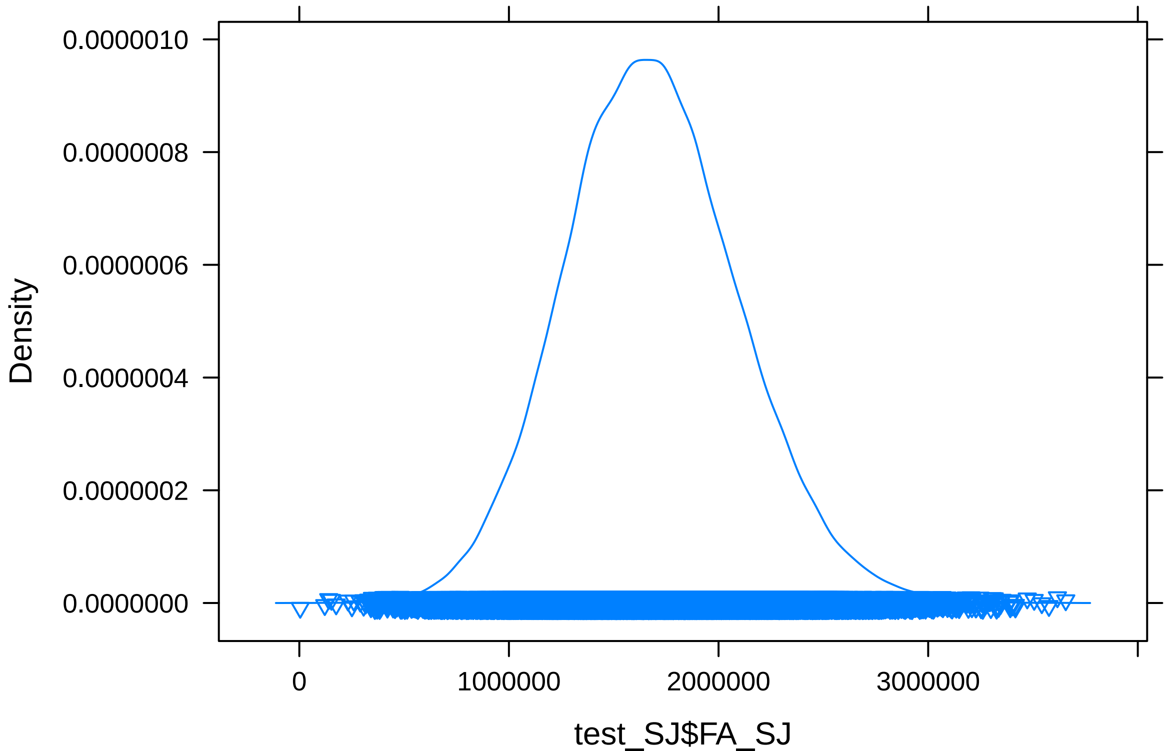

```
densityplot( ~ test_SJ$SMBR_SJ, pch=6)
```




```
densityplot( ~ test_SJ$SMBT_SJ, pch=6)
```

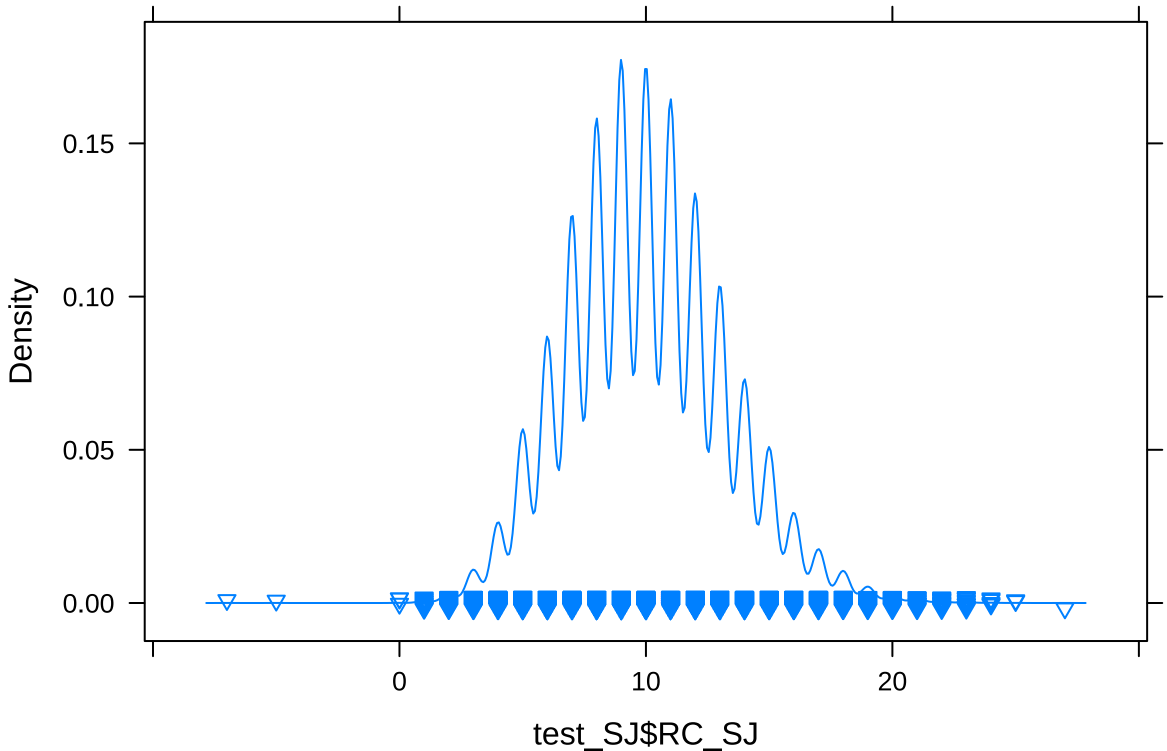



```
densityplot( ~ test_SJ$FA_SJ, pch=6)
```

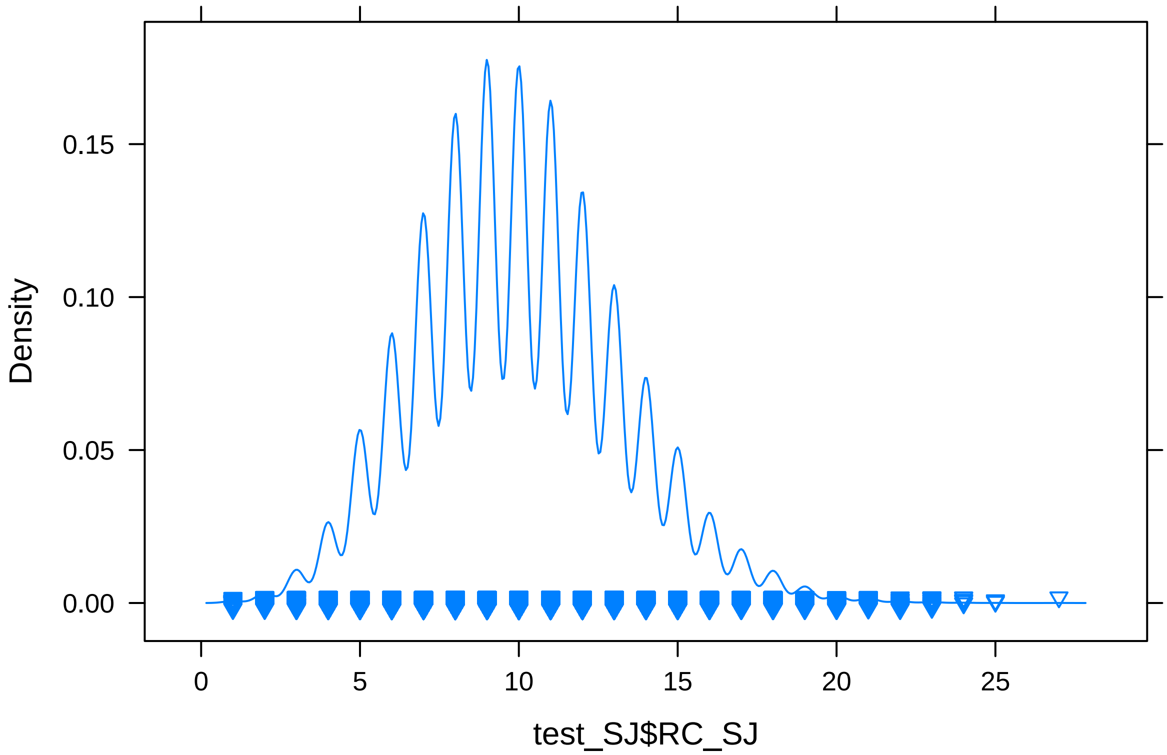



```
b. Comment on any outliers you see and deal with them
 appropriately. Make sure you explain why you dealt with them
 the way you decided to.

 -> 1.As we can see from the plots, Reconnections made has some values which seems like outliers
 has negative values which is not practically possible or it seems insensible or meaningless.
```

```
densityplot( ~ test_SJ$RC_SJ, pch=6)
```

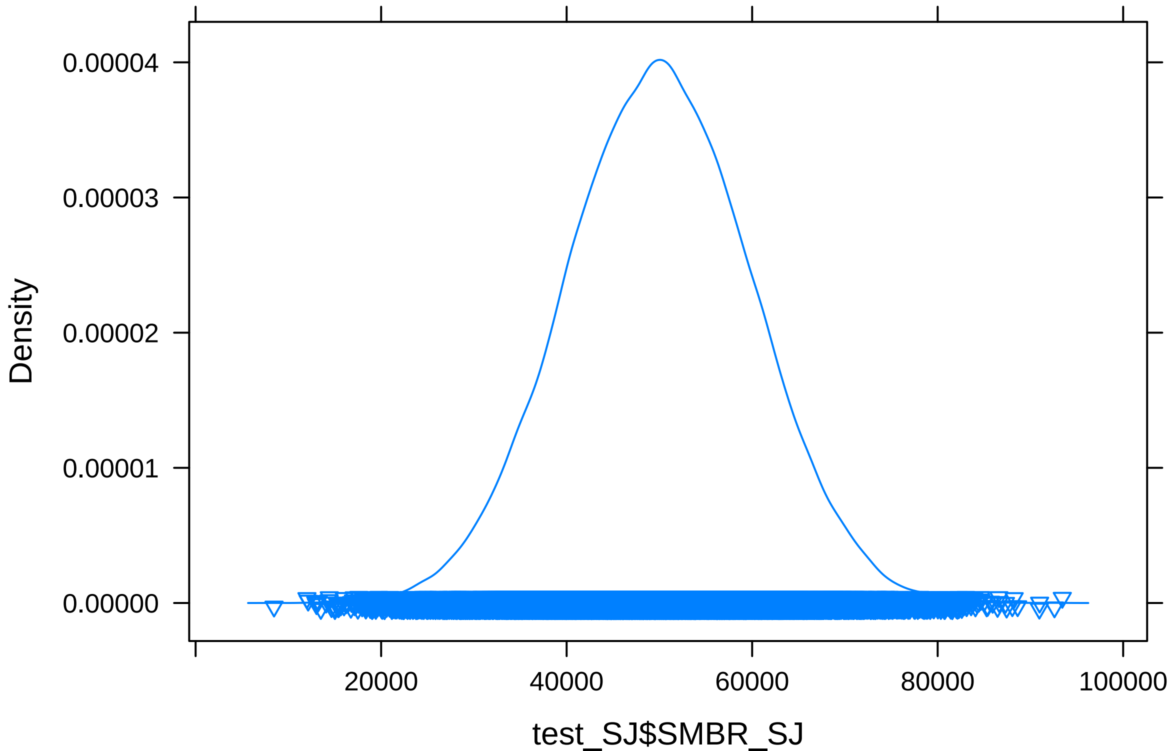



```
outlierInRC_SJ <- which(test_SJ$RC_SJ <= 0) ##Fond row number with RC less then or equal to 0
test_SJ <- test_SJ[-c(outlierInRC_SJ),]        #deleted the row
densityplot( ~ test_SJ$RC_SJ, pch=6)
```

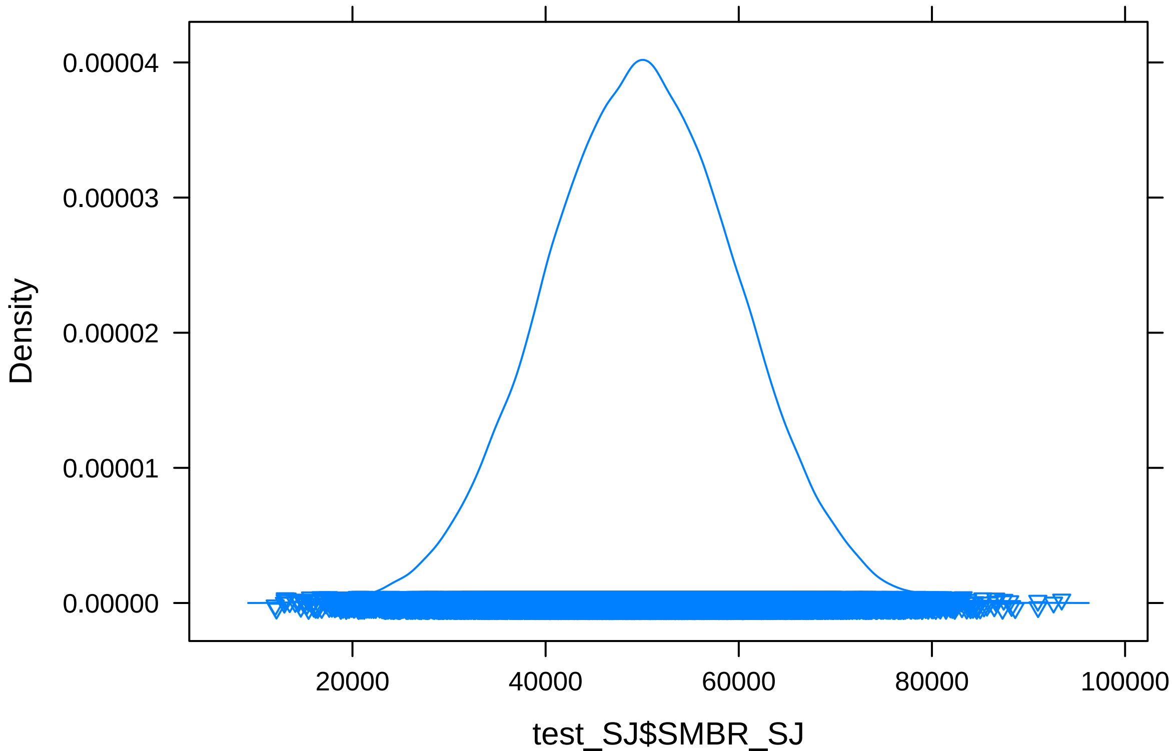


2.In the plots for the SMBR and SMBGT the minimum values which is near zero seems to be outliers, hence lets replace them with the mean values of subsequent columns that is SMBR_SJ and SMBR_ST.

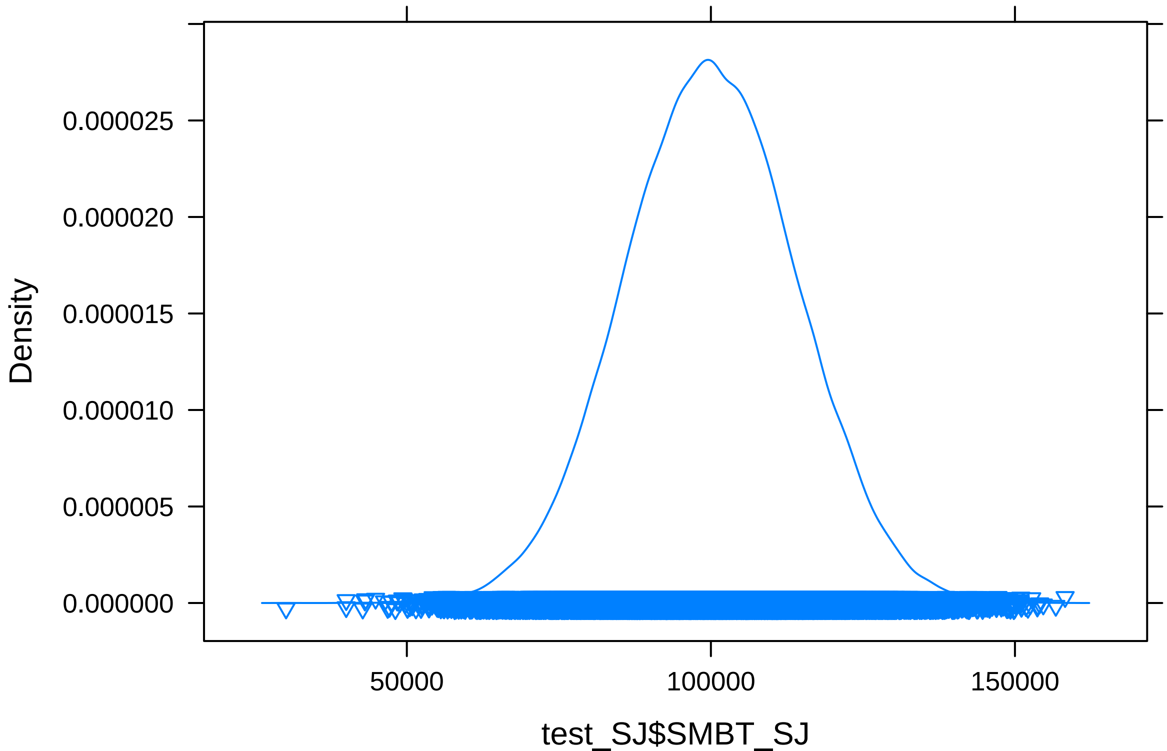
```
densityplot( ~ test_SJ$SMBR_SJ, pch=6)
```



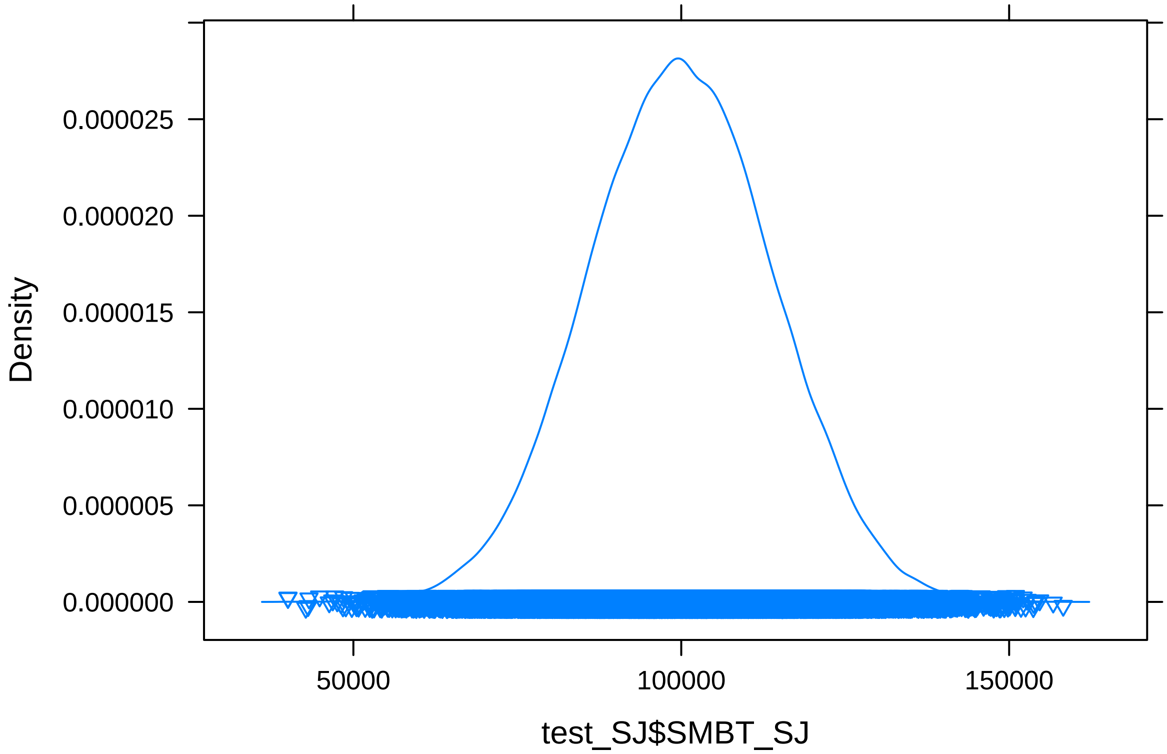

```
test_SJ$SMBR_SJ <- ifelse(test_SJ$SMBR_SJ == min(test_SJ$SMBR_SJ),mean(test_SJ$SMBR_SJ),
test_SJ$SMBR_SJ)                                                      #Replaces min value with the mean in
densityplot( ~ test_SJ$SMBR_SJ, pch=6)                                # Outlier Adjusted
```




```
densityplot( ~ test_SJ$SMBT_SJ, pch=6)
```

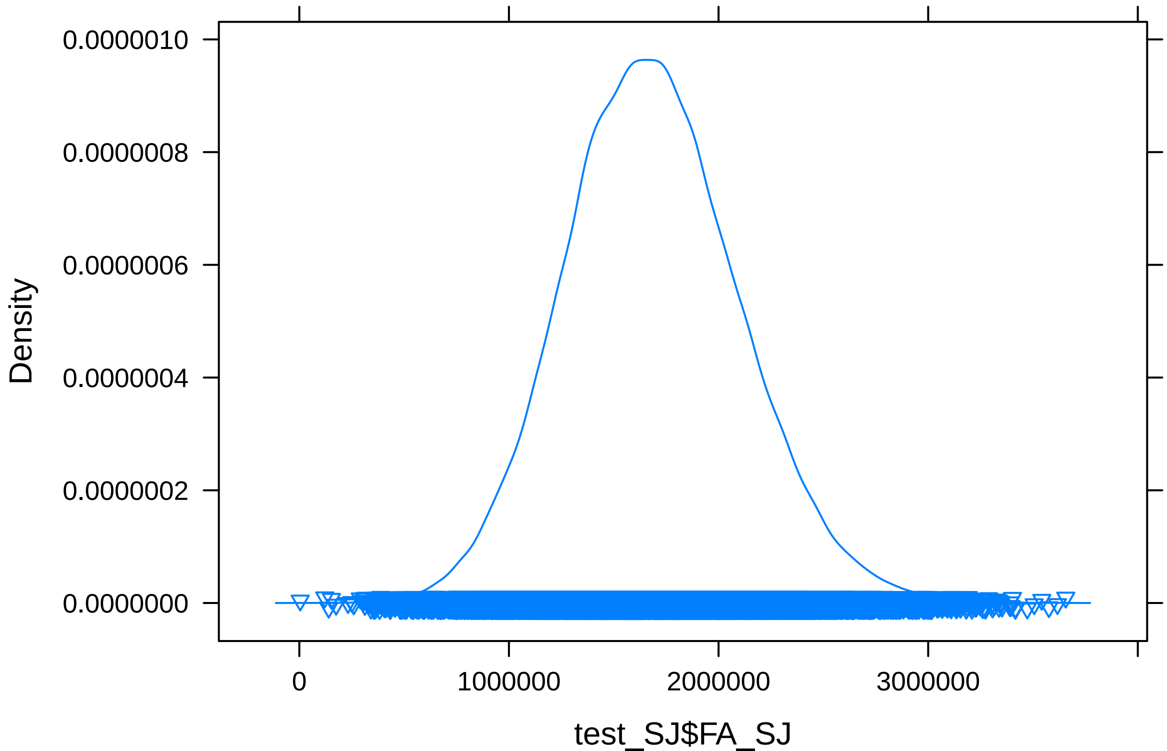



```
test_SJ$SMBT_SJ <- ifelse(test_SJ$SMBT_SJ == min(test_SJ$SMBT_SJ),mean(test_SJ$SMBT_SJ),test_SJ$SMBT_SJ)
                                                                                 #Replaces min value with the mean
densityplot( ~ test_SJ$SMBT_SJ, pch=6)                                            # Outlier Adjusted
```

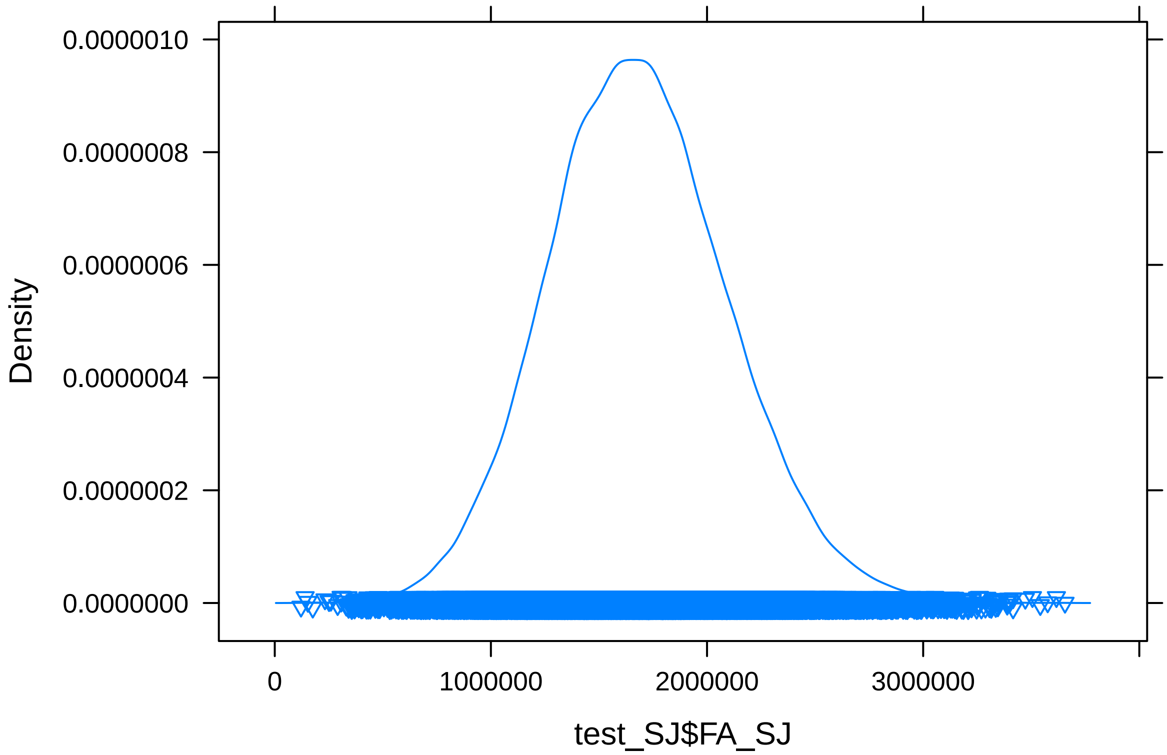


3. The same way for the Files Accessed, lets remove the minimum value with mean value of all Fil

```
densityplot( ~ test_SJ$FA_SJ, pch=6)
```

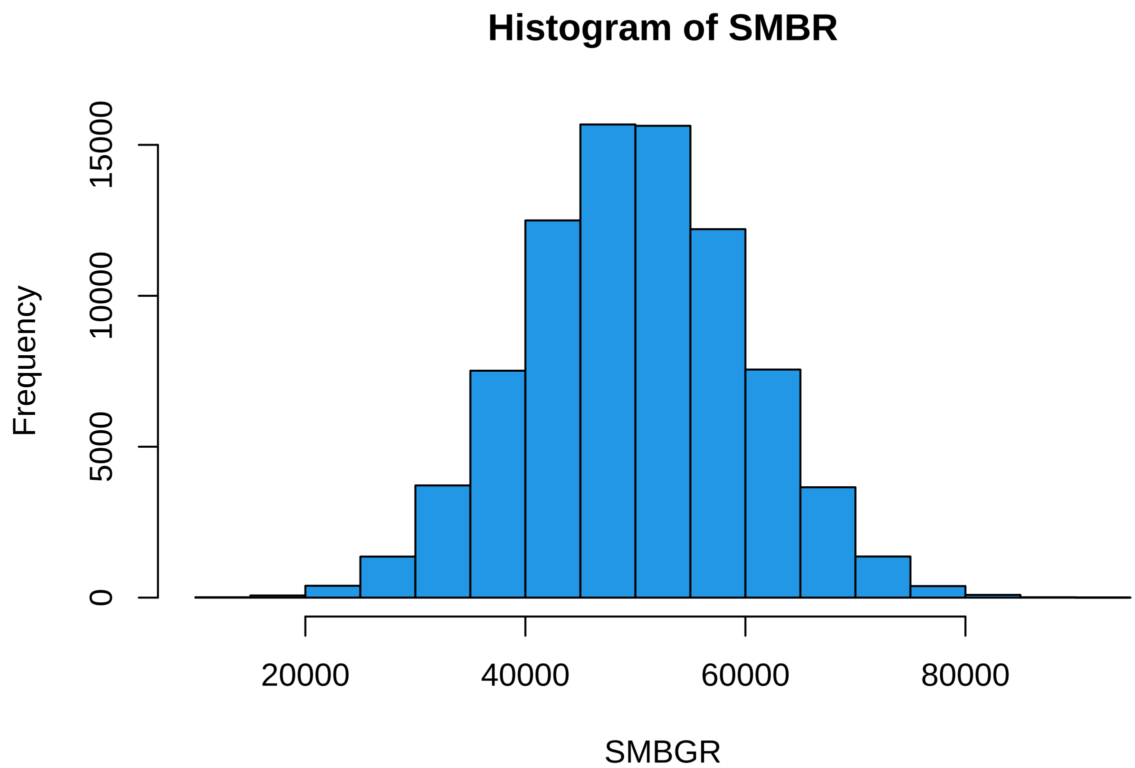



```r
test_SJ$FA_SJ <- ifelse(test_SJ$FA_SJ == min(test_SJ$FA_SJ),mean(test_SJ$FA_SJ),test_SJ$FA_SJ)
                                                                                 #Replaces min value with the mean
densityplot( ~ test_SJ$FA_SJ, pch=6)
```



2 Organizing Data 1. Scatter Plots a. Create a histogram for Server Message Blocks Received.

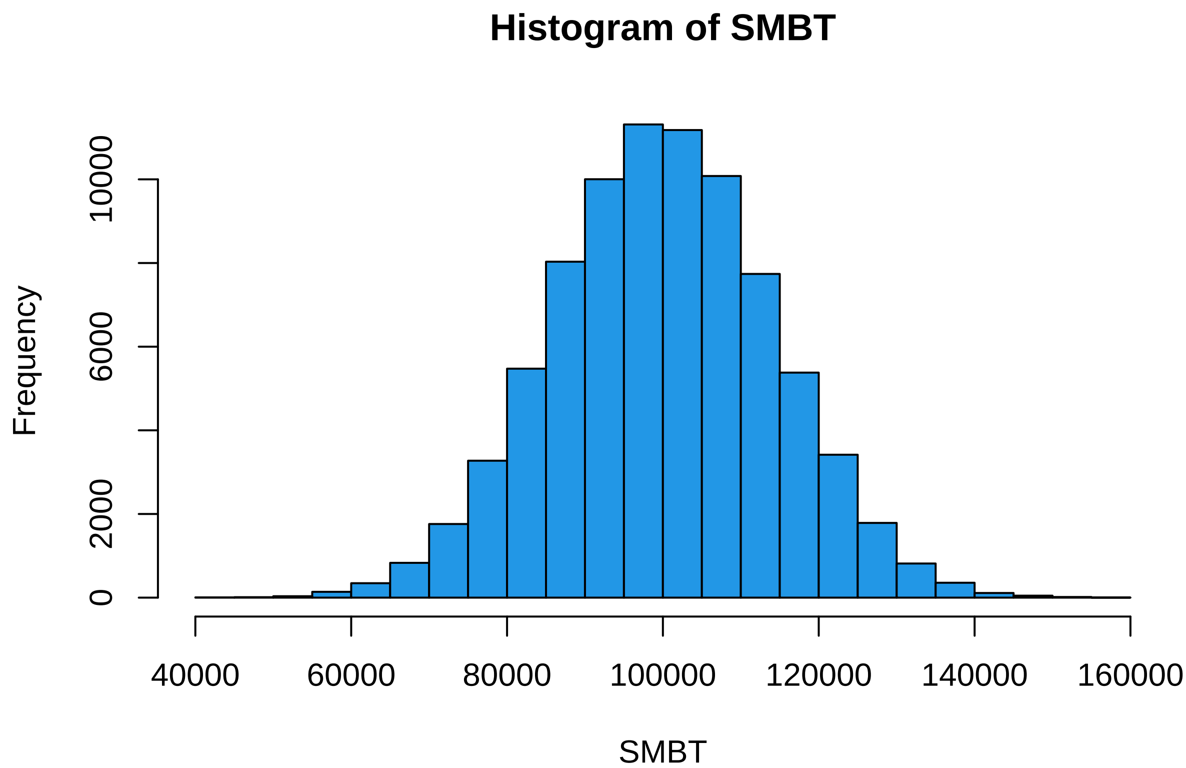
```
hist(test_SJ$SMBR_SJ,
      col = 100,
     xlab="SMBGR",
     main="Histogram of SMBR")
```



-> Above is the histogram plotted for SMBR, where we can identify the SMBRs having the most common frequency are in the range of 4,500 to 5,500.

b. Create a histogram for Server Message Blocks Transmitted.

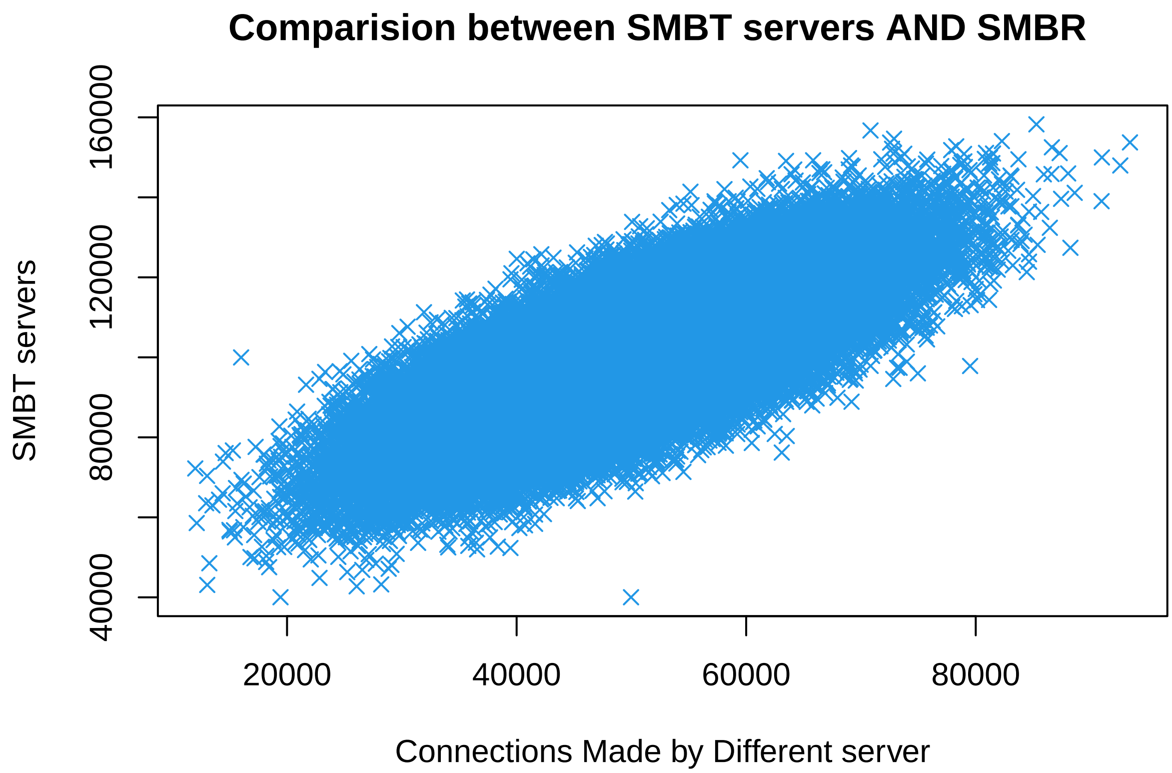
```
hist(test_SJ$SMBT_SJ,
     col = 100,
     xlab="SMBT",
     main="Histogram of SMBT")
```



-> In the histogram plotted for SMBT, where we can identify the SMBTs having the most common frequency are in the range of 9,500 to 10,500.

c. Create a scatter plot showing the relationship between SMBR and SMBT. (note: SMBR should be on the x-axis, SMBT should be the y-axis)

```
plot(SMBT_SJ ~ SMBR_SJ,
     data=test_SJ,
     col=100,
     pch=4,
     main="Comparision between SMBT servers AND SMBR ",
     xlab="Connections Made by Different server",
     ylab="SMBT servers"
     )
```



d. What conclusions, if any, can you draw from the chart?

-> From the scatter plot, we can say that there is a strong positive linear correlation between SMBT and SMBR.

e. Calculate a correlation coefficient between these two variables. Why did you choose the correlation coefficient you did? What conclusion you draw from it?

```
cor(test_SJ$SMBR_SJ,test_SJ$SMBT_SJ,method="pearson")
```

```
## [1] 0.7626193
```

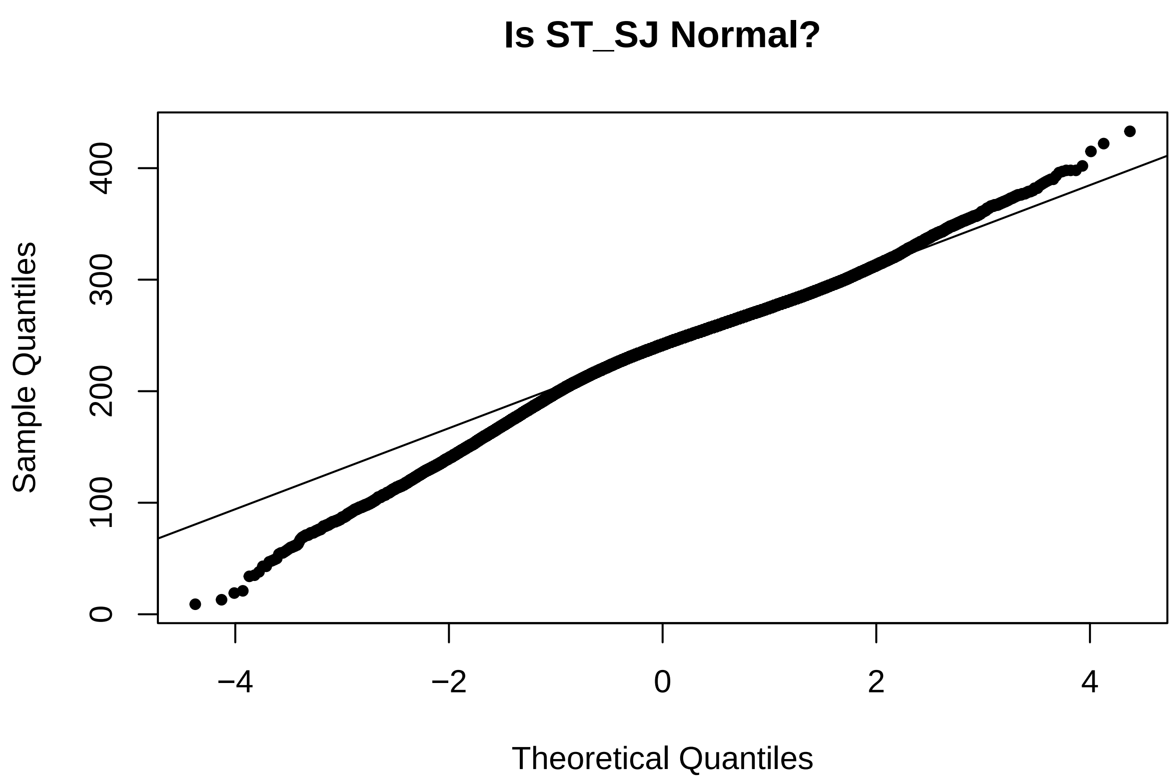
-> It can be clearly observed from scatter plot and histograms that SMBT and SMBR data is continuous and normally distributed. So for such data, we use Pearson method. By evaluating the value of correlation coefficient by Pearson method, we get 0.7626193 which signifies that it has strong linear positive relationship.

3 Inference

1. Normality

    a. Create a QQ Normal plot of for Sessions Timed Out.

```
qqnorm(test_SJ$ST_SJ, main="Is ST_SJ Normal?", pch=20)
qqline(test_SJ$ST_SJ)
```



-> By plotting the qqnorm, we can observe that the distribution is not normal.

```
b. Conduct a statistical test for normality on Sessions Timed
   Out.
```

```
ST_New_SJ<- sample(test_SJ$ST_SJ, 5000)
shapiro.test(ST_New_SJ)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  ST_New_SJ
## W = 0.98446, p-value < 2.2e-16
```

```
c. Is Sessions Times Out normally distributed? What led you to
    this conclusion?
```

```
-> As we have done the normality test by performing shapiro test, by observing the p-value,
   p_value is less then 0.05,we reject the null hypothesis that the distribution is normal
   and conclude that the distribution is not normal.
```

2. Statistically Significant Differences

    a. Compare Sessions Times Out between the two major Manufacturers in your data set using a suitable hypothesis test.

```
wilcox.test(ST_SJ ~ Manufacturer_SJ, data=test_SJ, exact=FALSE)
```

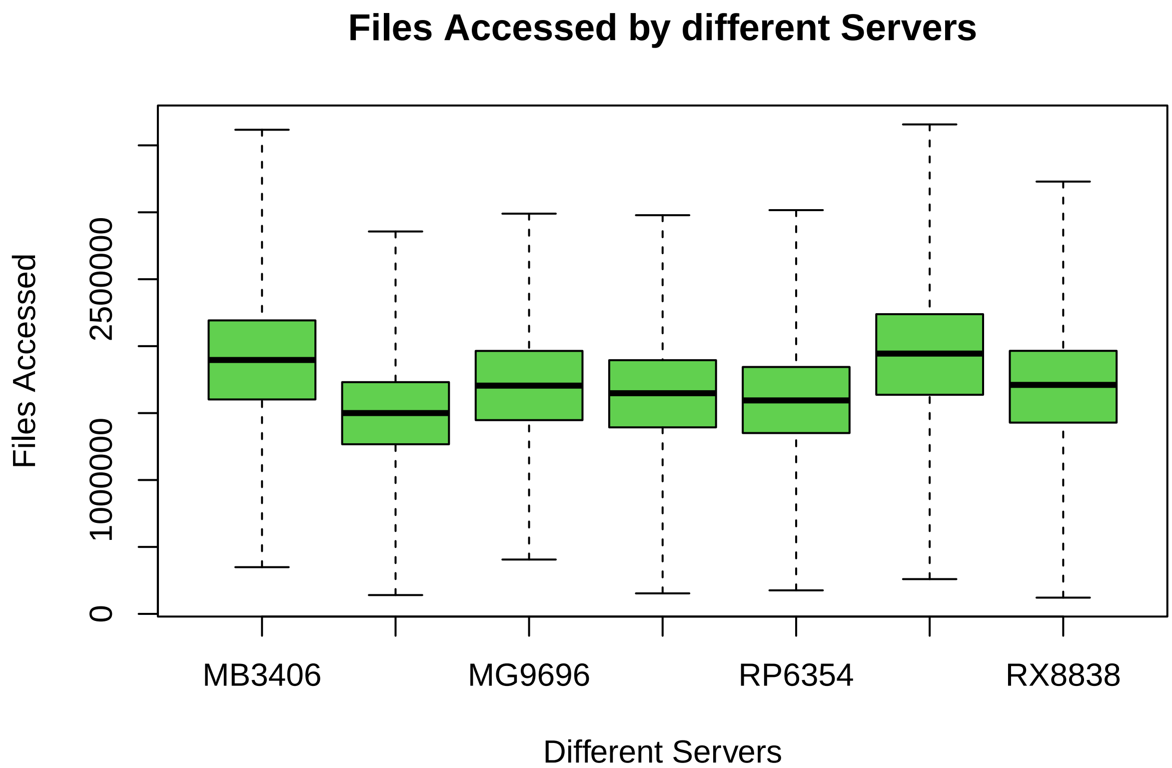
```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  ST_SJ by Manufacturer_SJ
## W = 1139321058, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0
```

```
b. Explain why you chose the test you did.


-> As Sessions times out is a continuous data,at the same time we see that there are
   only two groups being compared. We can also see it this way that it contains
   non-parametric data so we shoyuld perfrom Wilcoxon test.

c. Do you have strong evidence that Sessions Times Out are
     different between Manufacturers?

   -> Yes, we have strong evidence thatb ST are diff. from manufacturers as the p value
      very small.
```

3. Multiple Statistical Differences

    a. Determine if Files Accessed varies by Server using ANOVA (statistical) and a sequence of boxplots (graphical).

```
#Comparing Servers with Files Accessed

boxplot(test_SJ$FA_SJ ~ test_SJ$Server_SJ, data=test_SJ,
        main=" Files Accessed by different Servers ",
        xlab="Different Servers",
        ylab="Files Accessed",
        col=67,
        range=0)
```




```r
#One-Way ANOVA

summary(aov(test_SJ$FA_SJ ~ test_SJ$Server_SJ, data=test_SJ))
```

```
##                      Df            Sum Sq         Mean Sq F value Pr(>F)
## test_SJ$Server_SJ      6  1877409329942520 312901554990420    2080 <2e-16 ***
## Residuals          82144 12357397162657816     150435785482
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

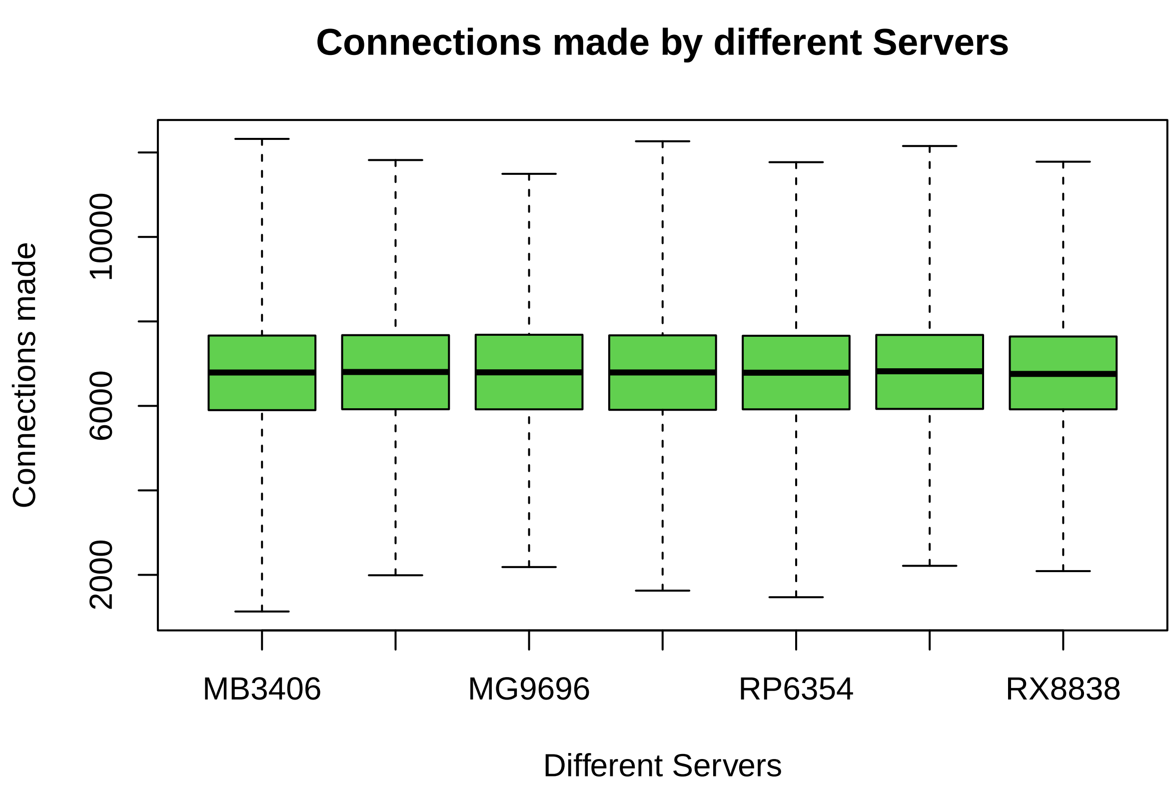
```r
ANOVA_For_Servers_SJ <- aov(test_SJ$FA_SJ~Server_SJ, data=test_SJ)
summary(ANOVA_For_Servers_SJ)
```

```
##                Df            Sum Sq         Mean Sq F value Pr(>F)
## Server_SJ       6  1877409329942520 312901554990420    2080 <2e-16 ***
## Residuals   82144 12357397162657816     150435785482
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

-> It is clearly observed that files Accessed varies by Server .

b. Determine if Connections Made varies by Server using
    ANOVA and a sequence of boxplots.

```r
#Comparing Servers with Connections Made

boxplot(test_SJ$Conn_SJ ~ test_SJ$Server_SJ, data=test_SJ,
        main=" Connections made by different Servers ",
        xlab="Different Servers",
        ylab="Connections made",
        col=67,
        range=0)
```




```r
#One-Way ANOVA

summary(aov(test_SJ$Conn_SJ ~ test_SJ$Server_SJ, data=test_SJ))
```

```
##                       Df       Sum Sq Mean Sq F value Pr(>F)
## test_SJ$Server_SJ      6      6745080 1124180   0.663  0.679
## Residuals          82144 139221596411 1694848
```

```r
ANOVA_For_Connections_SJ <- aov(test_SJ$Conn_SJ~Server_SJ, data=test_SJ)
summary(ANOVA_For_Connections_SJ)
```

```
##                 Df       Sum Sq Mean Sq F value Pr(>F)
```

```
## Server_SJ       6      6745080 1124180   0.663  0.679
## Residuals   82144 139221596411 1694848
```

-> It is observed that Connections made does not varry by Servers .