Significance and Inference

Divyang Vinubhai Hirpara

11/02/2023

It will clear all the plots, the console and the work-space. It also sets the overall format for numbers.

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

```
## Loading required package: HSAUR
```

```
## Loading required package: tools
```

```
library("HSAUR")
```

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

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

```
library("pastecs")
```

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

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

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

1. Data Transformation and Preparation

1. Initial Transformation

a. Rename all variables with your initials appended.

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

```
##   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
```

Interpretation

- The text(.txt) file shall be read with 'read.table' function in R.
- Text file is comma separated hence, sep=" ," is used to identify a rows and column.

- Header=TRUE is used due to the text file is generated with header in first line.
- By default, 6 records are displayed with 'head()' function as shown above.
- There are total 12 columns with manufacturer and serve of Character data type, and index, conn,RC, ST, SMBR, SMBT, BR, BT, UC, FA of integer datatype.

Rename Variables of column name

```
#Append data_DVH initials to column names
colnames(data_DVH) <- paste(colnames(data_DVH), "DVH", sep = "_")
head(data_DVH,10)
```

```
##    Index_DVH Manufacturer_DVH Server_DVH Conn_DVH RC_DVH ST_DVH SMBR_DVH
## 1          1             Lled     MB5755     5571     10    253    39806
## 2          2             Lled     MB3406     6684     12    282    56410
## 3          3           Ovonel     RQ8547     4790     10     83    55891
## 4          4             Lled     MB3406     6163     10    247    49546
## 5          5             Lled     MB5755     8939     12    252    61578
## 6          6           Ovonel     RP6354     7446      8    263    47692
## 7          7           Ovonel     RP6354     8618     13    118    50814
## 8          8           Ovonel     RP6354     7319      3    271    49125
## 9          9             Lled     MB5755     5853      7    283    62117
## 10        10           Ovonel     RL3777     7667     12    256    58279
##    SMBT_DVH BR_DVH BT_DVH UC_DVH  FA_DVH
## 1     91685     11     17   2000 1526223
## 2    115100     15     22   2000 1799882
## 3     98534     15     18   2000 1361793
## 4    116361     14     22   2000 2365969
## 5    104176     17     20   2000 2222282
## 6    102983     13     19   2000 2006826
## 7    102608     14     19   2000 1043945
## 8     99735     13     19   2000 1283390
## 9    127959     17     24   2000 1795163
## 10   109037     16     20   2000 1121878
```

Interpretation

Every column are replaced with initials.
Index –> Index_DVH
Manufacturer –> Manufacturer_DVH
Server –> Server_DVH
Conn –> Conn_DVH RC –> RC_DVH
ST –> ST_DVH
SMBR –> SMBR_DH
SMBT –> SMBT_DH
BR –> BR_DVH
BT –> BT_DVH
UC –> UC_DVH
FA –> FA_DVH

b. Transform character variables to factor variables.

```
data_DVH$Manufacturer_DVH <- as.factor(data_DVH$Manufacturer_DVH)
data_DVH$Server_DVH <- as.factor(data_DVH$Server_DVH)

head(data_DVH,8)
```

```
##   Index_DVH Manufacturer_DVH Server_DVH Conn_DVH RC_DVH ST_DVH SMBR_DVH
## 1         1             Lled     MB5755     5571     10    253    39806
## 2         2             Lled     MB3406     6684     12    282    56410
## 3         3           Ovonel     RQ8547     4790     10     83    55891
## 4         4             Lled     MB3406     6163     10    247    49546
## 5         5             Lled     MB5755     8939     12    252    61578
## 6         6           Ovonel     RP6354     7446      8    263    47692
## 7         7           Ovonel     RP6354     8618     13    118    50814
## 8         8           Ovonel     RP6354     7319      3    271    49125
##   SMBT_DVH BR_DVH BT_DVH UC_DVH  FA_DVH
## 1    91685     11     17   2000 1526223
## 2   115100     15     22   2000 1799882
## 3    98534     15     18   2000 1361793
## 4   116361     14     22   2000 2365969
## 5   104176     17     20   2000 2222282
## 6   102983     13     19   2000 2006826
## 7   102608     14     19   2000 1043945
## 8    99735     13     19   2000 1283390
```

Interpretation

Manufacture_DVH and Server_DVH are character data type. hence it is changed to factor data type by using as.factor() function.

2. Reduce Dimensionality

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

```
summary(data_DVH)
```

```
##    Index_DVH     Manufacturer_DVH  Server_DVH       Conn_DVH
RC_DVH
##  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_DVH         SMBR_DVH        SMBT_DVH          BR_DVH
```

```
##  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_DVH          UC_DVH         FA_DVH
##  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
##
```

Interpretation

The Summary () function displays statistical information such as min, 1st Quarter, 3rd Quarter, Median, Mean, Max, and missing values.

Looking at the above summary table of all columns, it seems there is no missing value available in any column.

If any missing value is available in any column, it is supposed to look like this - NA's 2. where 2 represents the number of missing values.

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

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

```
##                      Index_DVH Manufacturer_DVH Server_DVH
Conn_DVH
## 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_DVH           ST_DVH           SMBR_DVH
## 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_DVH            BR_DVH            BT_DVH
## 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_DVH               FA_DVH
## 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
```

Interpretation
From the above stat values, it seems UC_DVH (0.00002252) is likely very low in terms of Coef.var.

```
table(data_DVH$UC_DVH)

##
##  2000  2001
## 81989   167
```

Interpretation
From the above output, a high number of repeating values occurred for '2000', which is 81989. And, only 167 numbers appeared for '2001'. To conclude, it needs more balanced data for column 'UC_DVH.' for analysis.

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

```
numeric_data_DVH <- data_DVH[-c(2:3)]
head(numeric_data_DVH,3)

##   Index_DVH Conn_DVH RC_DVH ST_DVH SMBR_DVH SMBT_DVH BR_DVH BT_DVH UC_DVH
## 1         1     5571     10    253    39806    91685     11     17   2000
## 2         2     6684     12    282    56410   115100     15     22   2000
## 3         3     4790     10     83    55891    98534     15     18   2000
##    FA_DVH
## 1 1526223
## 2 1799882
## 3 1361793
```

Interpretation
Removed non numeric column to find high correlation of data. And, Stored numeric columns to variable 'numeric_data_DVH' for further analysis.

```
cor(numeric_data_DVH,method="spearman")

##               Index_DVH      Conn_DVH       RC_DVH        ST_DVH
SMBR_DVH
## Index_DVH  1.0000000000  0.0042298416  0.002637313  0.0039510097 -
0.0072618256
## Conn_DVH   0.0042298416  1.0000000000  0.002553713 -0.0051813266 -
0.0007441048
## RC_DVH     0.0026373131  0.0025537135  1.000000000  0.0040373249
0.0048533195
## ST_DVH     0.0039510097 -0.0051813266  0.004037325  1.0000000000 -
0.0000386051
## SMBR_DVH  -0.0072618256 -0.0007441048  0.004853319 -0.0000386051
1.0000000000
## SMBT_DVH  -0.0069800768 -0.0009864149  0.003980611 -0.0005576250
0.7476258171
```

```
## BR_DVH    -0.0079967459 -0.0005187586  0.004454767 -0.0001081305
0.9938996733
## BT_DVH    -0.0063522475 -0.0016882428  0.004523766 -0.0008330024
0.7430656692
## UC_DVH     0.0037172910  0.0017831677  0.004990823 -0.0044092114 -
0.0017005879
## FA_DVH     0.0005723258  0.0074740748 -0.002028123  0.0025137913
0.0016462715
##                SMBT_DVH        BR_DVH        BT_DVH        UC_DVH
FA_DVH
## Index_DVH -0.0069800768 -0.0079967459 -0.0063522475  0.0037172910
0.0005723258
## Conn_DVH  -0.0009864149 -0.0005187586 -0.0016882428  0.0017831677
0.0074740748
## RC_DVH     0.0039806108  0.0044547674  0.0045237655  0.0049908235 -
0.0020281233
## ST_DVH    -0.0005576250 -0.0001081305 -0.0008330024 -0.0044092114
0.0025137913
## SMBR_DVH   0.7476258171  0.9938996733  0.7430656692 -0.0017005879
0.0016462715
## SMBT_DVH   1.0000000000  0.7438861976  0.9935613424 -0.0017384137
0.0014519063
## BR_DVH     0.7438861976  1.0000000000  0.7393055744 -0.0021123454
0.0015287512
## BT_DVH     0.9935613424  0.7393055744  1.0000000000 -0.0012851327
0.0019115184
## UC_DVH    -0.0017384137 -0.0021123454 -0.0012851327  1.0000000000 -
0.0002994613
## FA_DVH     0.0014519063  0.0015287512  0.0019115184 -0.0002994613
1.0000000000
```

Interpretation

With Correlation function cor(), method="spearman" basically it refers to calculation of the Spearman's rank correlation coefficient. It helps find the high correlation between two variable.

From above values, there are variable with highly correlated values displayed below.

SMBR_DVH –> BR_DVH (0.99389967)
SMBT_DVH –> BT_DVH(0.9935613)
BR_DVH –> SMBR_DVH (0.99389967)
BT_DVH –> SMBT_DVH(0.99356134)

Hence, there is no need of considering all variables for analysis, it be any two need either **SMBR_DVH and SMBT_DVH** or **BR_DVH and BT_DVH**.

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

```
data_DVH <- data_DVH[, !colnames(data_DVH) %in% c("UC_DVH")]
data_DVH <- data_DVH[, !colnames(data_DVH) %in% c("BR_DVH")]
data_DVH <- data_DVH[, !colnames(data_DVH) %in% c("BT_DVH")]
```

```
data_DVH <- data_DVH[, !colnames(data_DVH) %in% c("Index_DVH")]
head(data_DVH,3)

##   Manufacturer_DVH Server_DVH Conn_DVH RC_DVH ST_DVH SMBR_DVH SMBT_DVH
FA_DVH
## 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
```

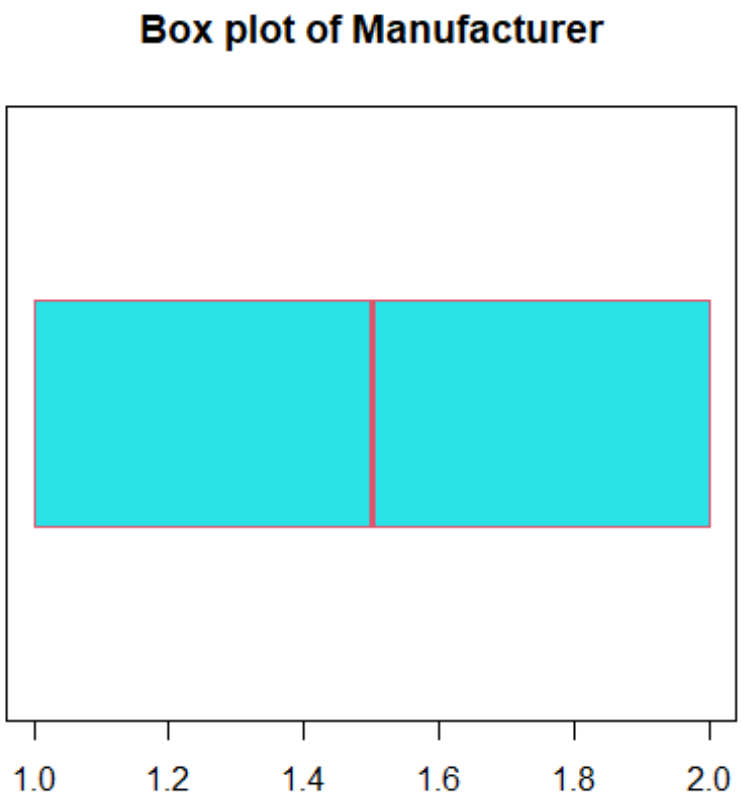
Interpretation

From (b), there is low variance of UC_DVH column, which is 0.00002252. From (c), high correlation found between SMBR_DVH - BR_DVH and SMBT_DVH - BT_DVH.
Hence, There are total 4 variables UC_DVH, BR_DVH, Index_DVH and BT_DVH are dropped as they do not contribute any useful analytical information at all.

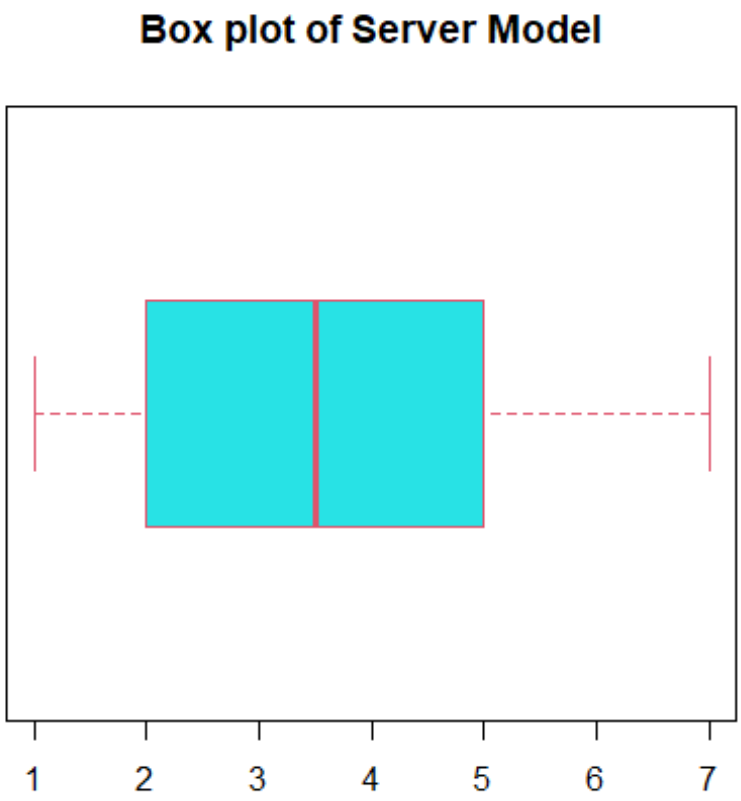
3. Outliers

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

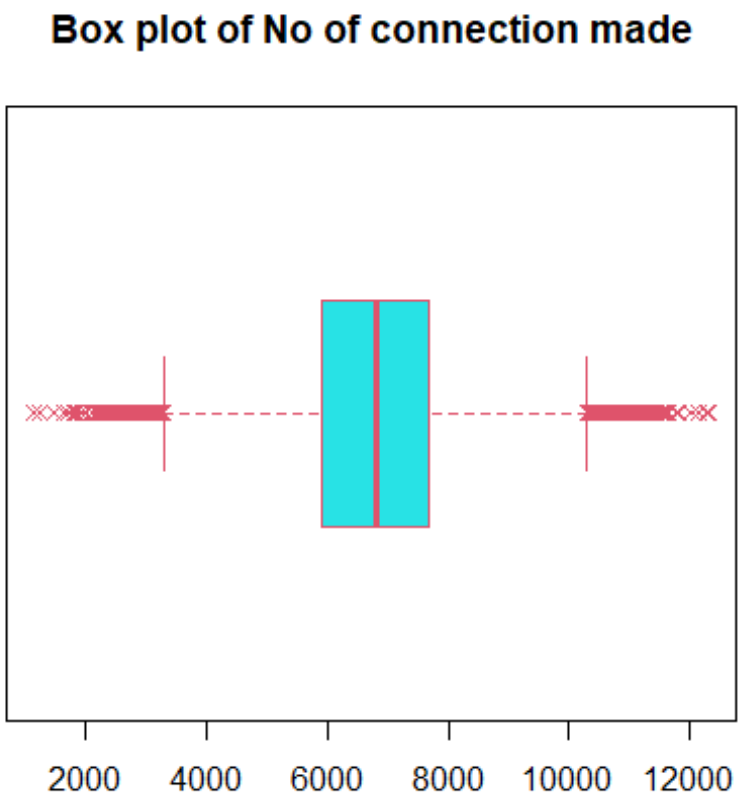
```
boxplot(data_DVH$Manufacturer_DVH, horizontal=TRUE, pch=4, col=5, border = 2,
main="Box plot of Manufacturer")
```



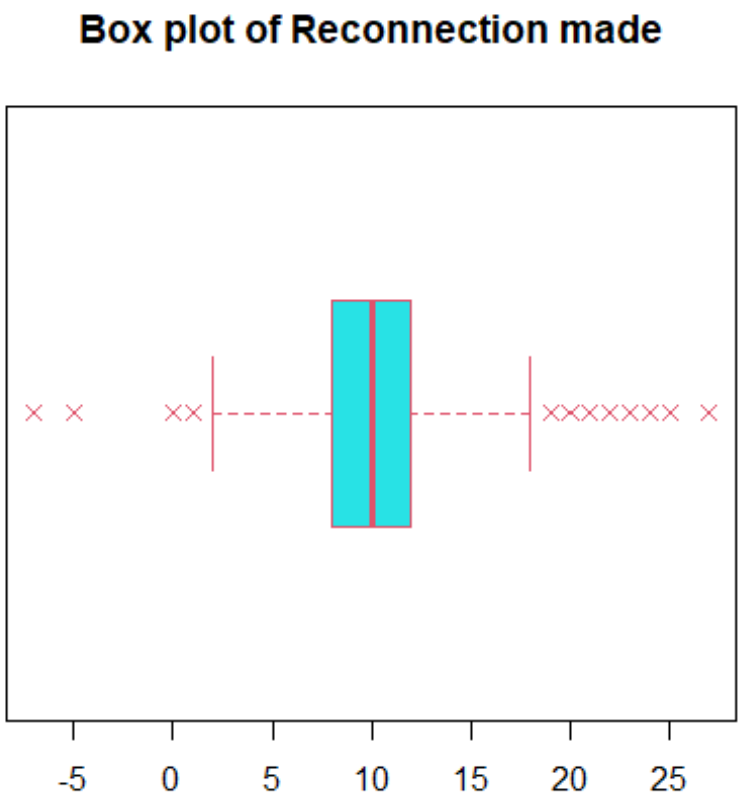

```
boxplot(data_DVH$Server_DVH, horizontal=TRUE, pch=4,col=5, border = 2,
main="Box plot of Server Model")
```



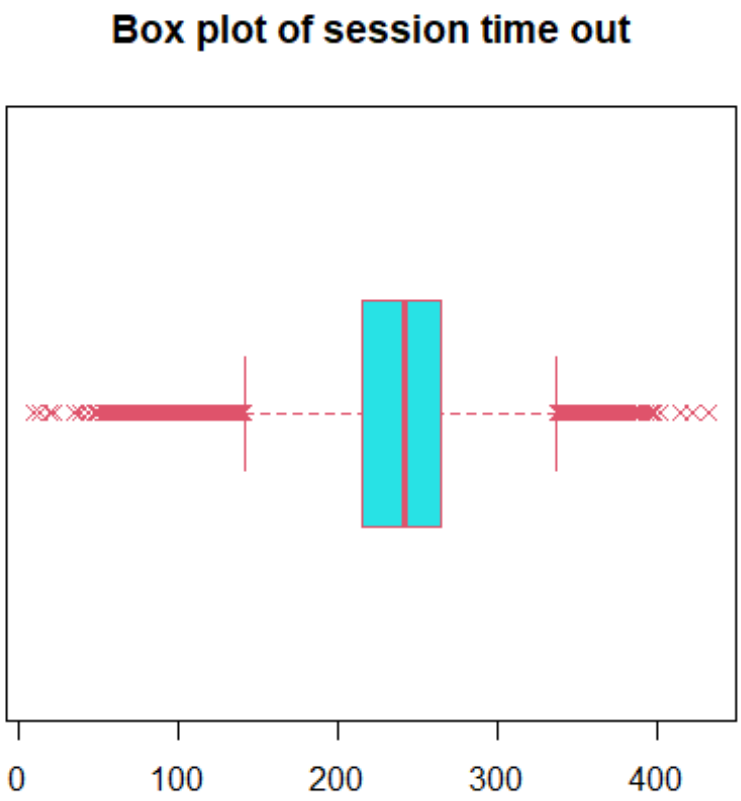

```
boxplot(data_DVH$Conn_DVH, horizontal=TRUE, pch=4,col=5, border = 2,
main="Box plot of No of connection made")
```



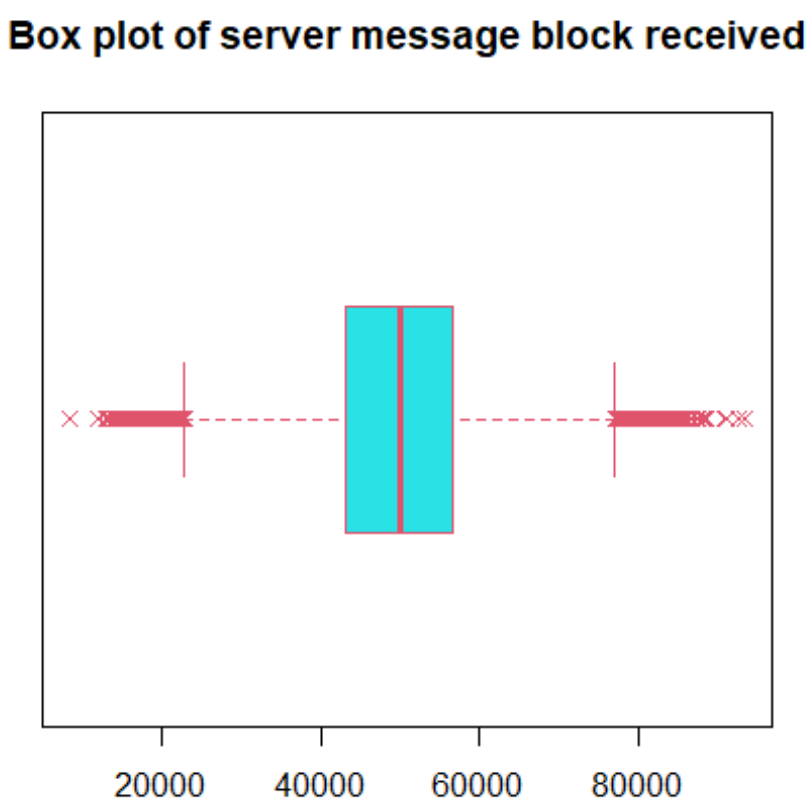

```
boxplot(data_DVH$RC_DVH, horizontal=TRUE, pch=4,col=5, border = 2, main="Box
plot of Reconnection made")
```



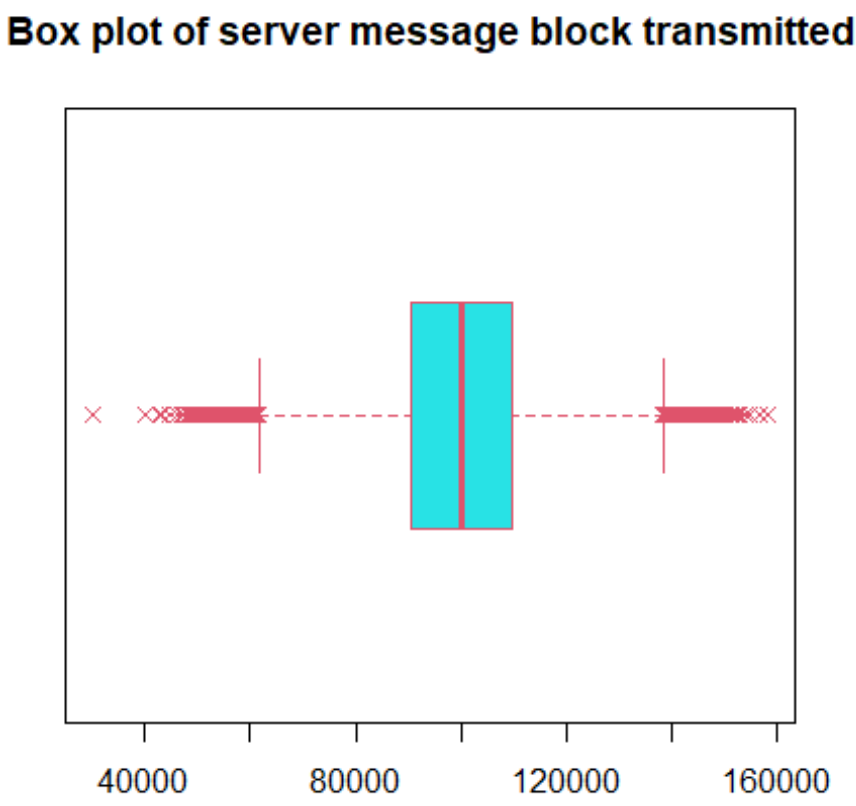

```
boxplot(data_DVH$ST_DVH, horizontal=TRUE, pch=4,col=5, border = 2, main="Box
plot of session time out")
```



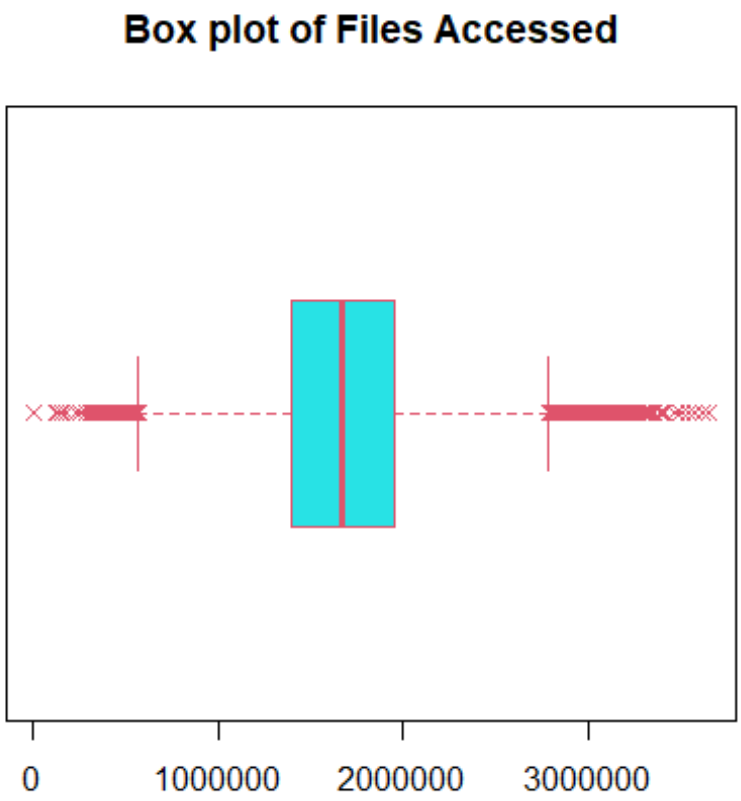

```
boxplot(data_DVH$SMBR_DVH, horizontal=TRUE, pch=4,col=5, border = 2,
main="Box plot of server message block received")
```




```
boxplot(data_DVH$SMBT_DVH, horizontal=TRUE, pch=4,col=5, border = 2,
main="Box plot of server message block transmitted")
```




```
boxplot(data_DVH$FA_DVH, horizontal=TRUE, pch=4,col=5, border = 2, main="Box
plot of Files Accessed")
```



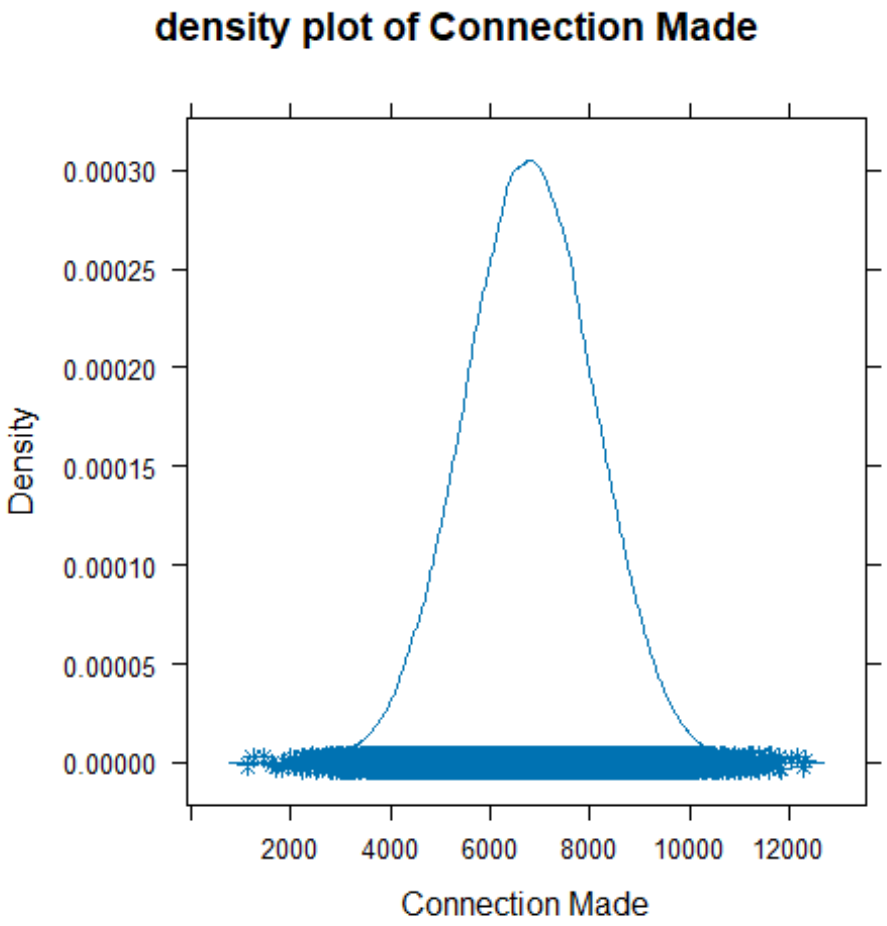
Interpretation

To find a outliers from data, there are different techniques used. Here box plot is displayed for all variables.

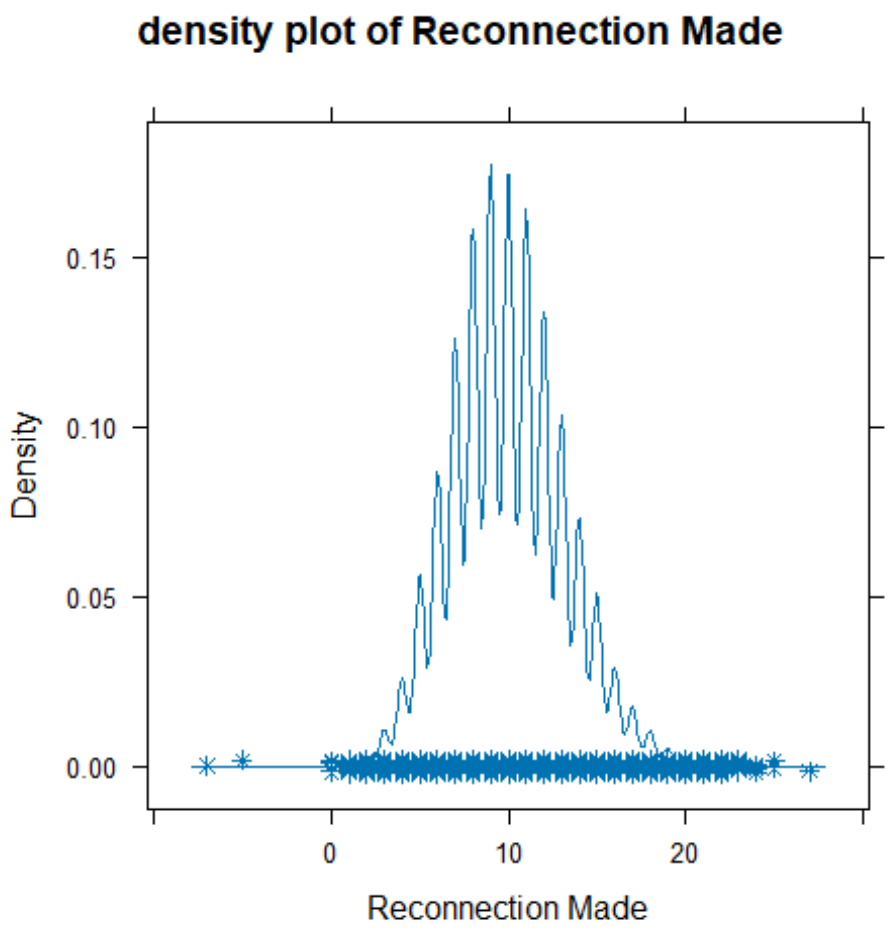
By focusing more on numeric variables from box plot, each variable has outliers present in data.

Re-connection made and server message block transmitted have some outliers which are more separated from other outliers as seen in box plot. Let's dig deeper on outliers with density plot.

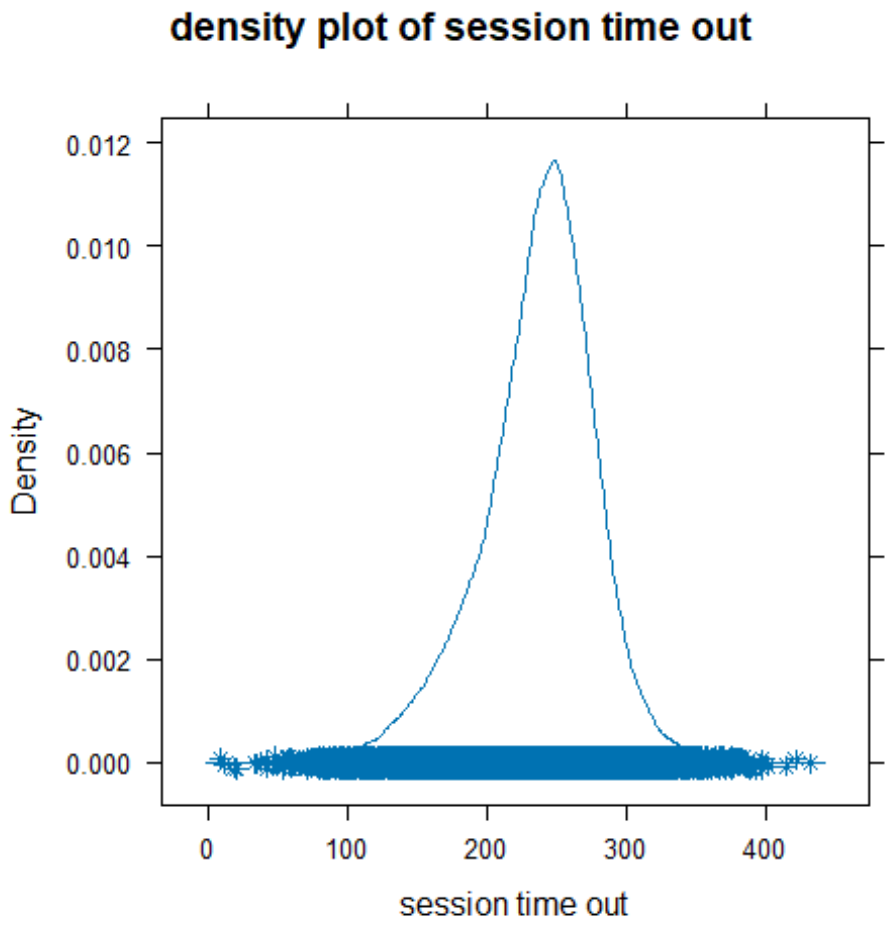
```
densityplot( ~ data_DVH$Conn_DVH, pch=8,main="density plot of Connection
Made",xlab="Connection Made")
```



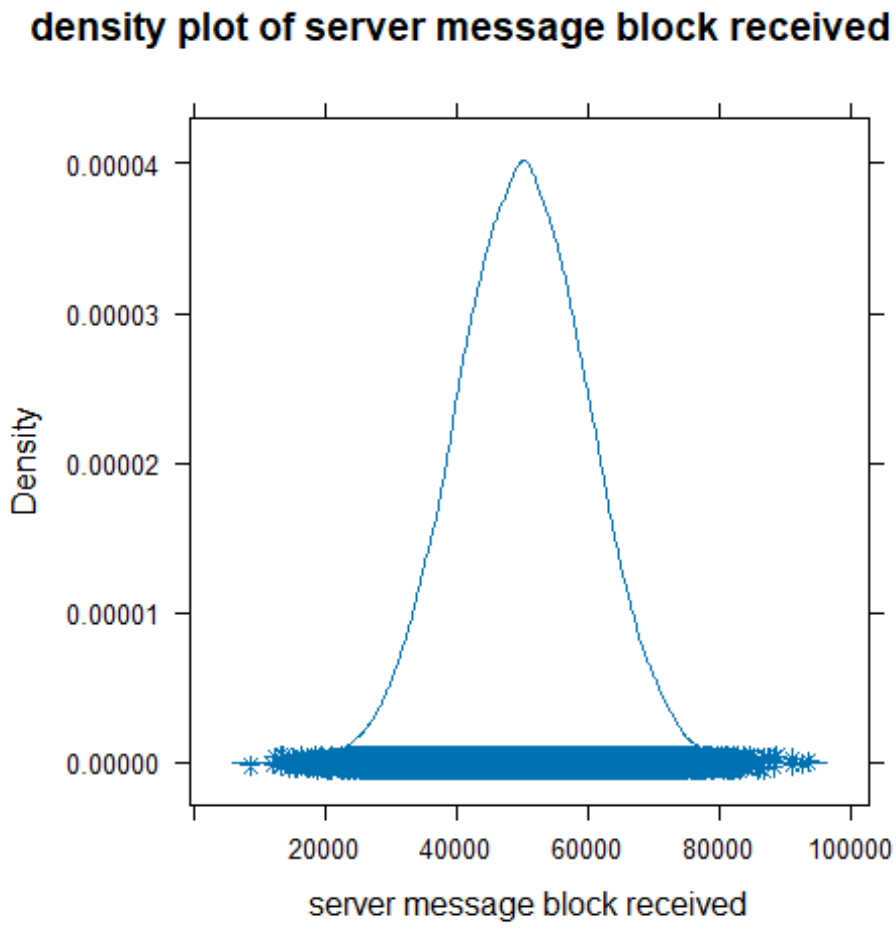

```
densityplot( ~ data_DVH$RC_DVH, pch=8,main="density plot of Reconnection
Made",xlab="Reconnection Made")
```



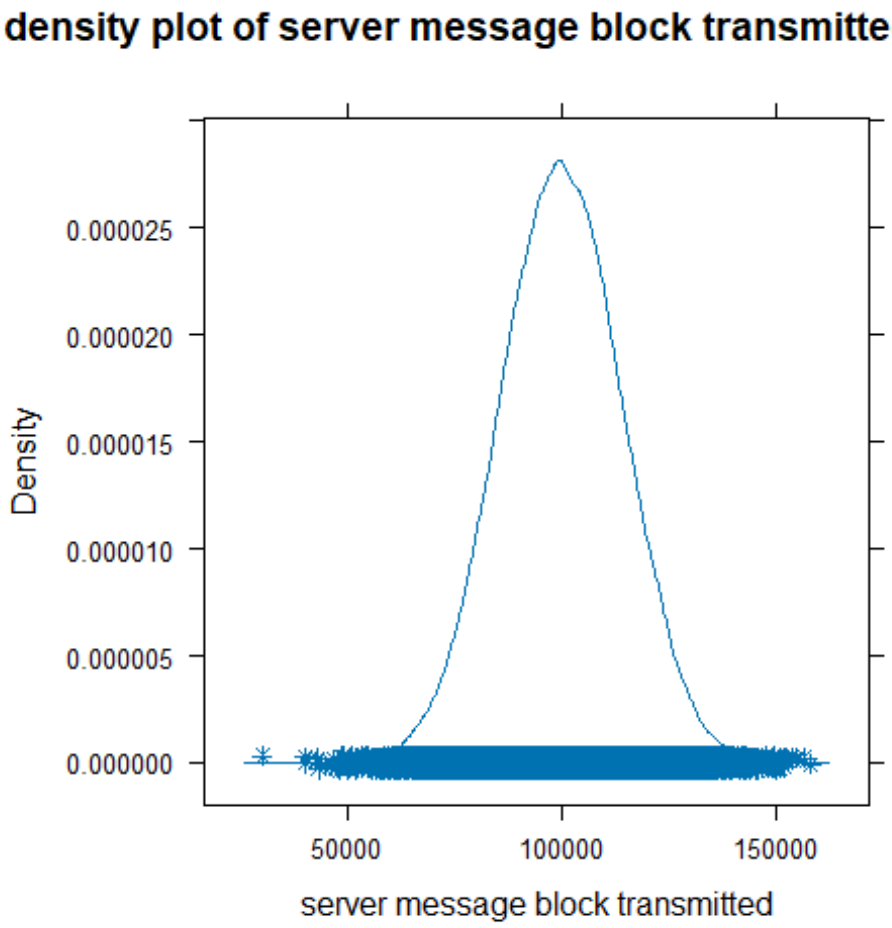

```
densityplot( ~ data_DVH$ST_DVH, pch=8,main="density plot of session time
out",xlab="session time out")
```



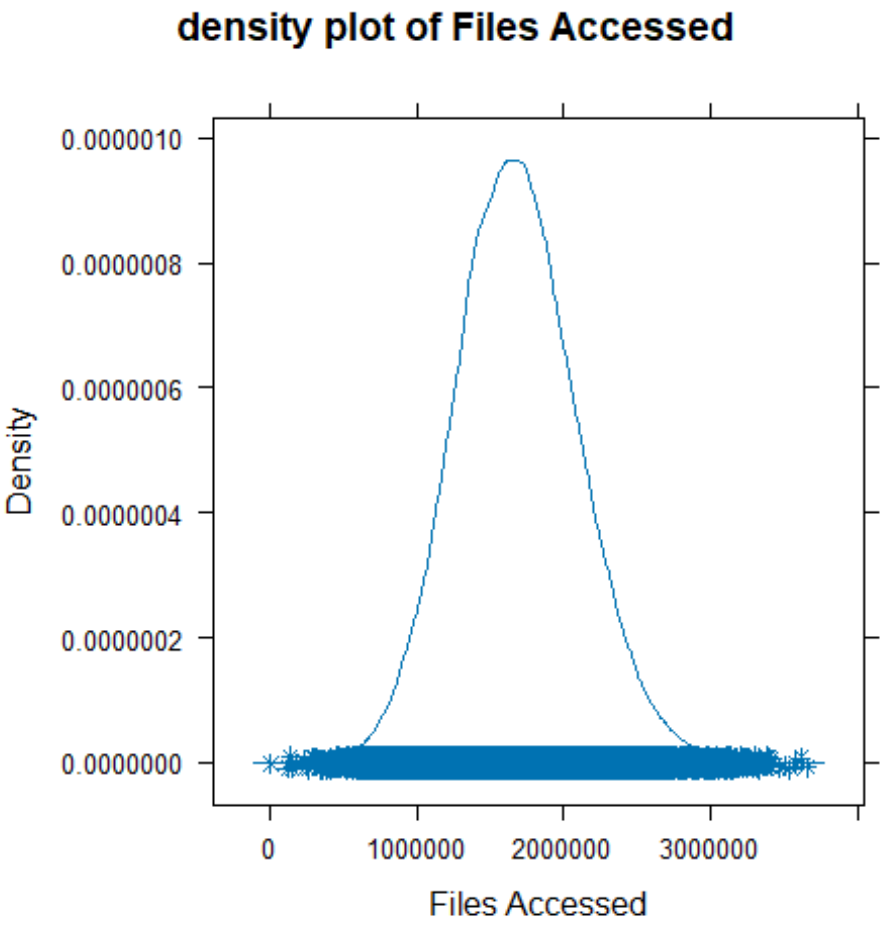

```
densityplot( ~ data_DVH$SMBR_DVH, pch=8,main="density plot of server message
block received",xlab="server message block received")
```




```
densityplot( ~ data_DVH$SMBT_DVH, pch=8,main="density plot of server message
block transmitted",xlab="server message block transmitted")
```




```
densityplot( ~ data_DVH$FA_DVH, pch=8,main="density plot of Files
Accessed",xlab="Files Accessed")
```



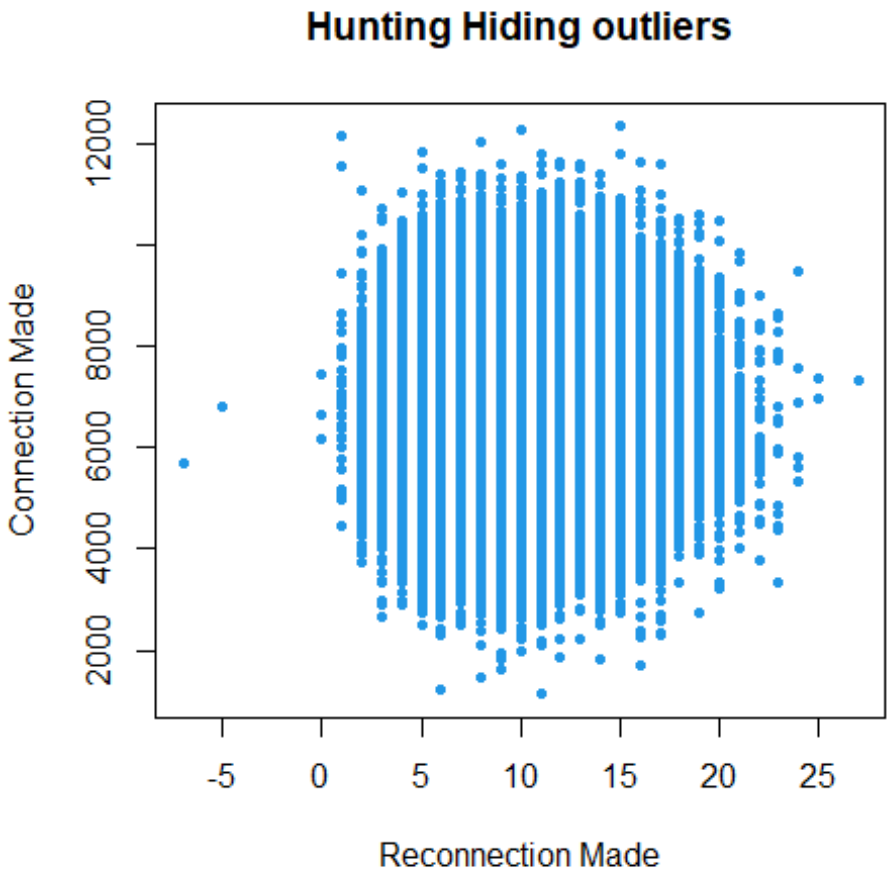
Interpretation

From the above density plot, Reconnection Made has two outliers which is way far from cluster.

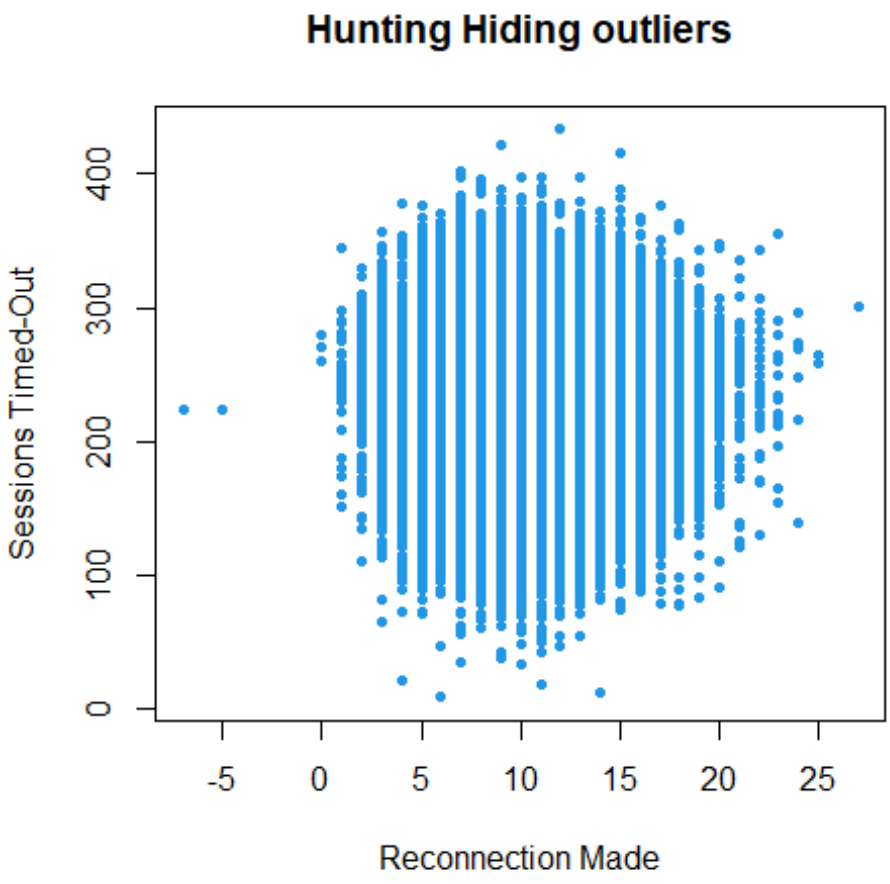
To visualize of Reconnection Made with other variables on outliers, there are scatter plot displayed below.

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.

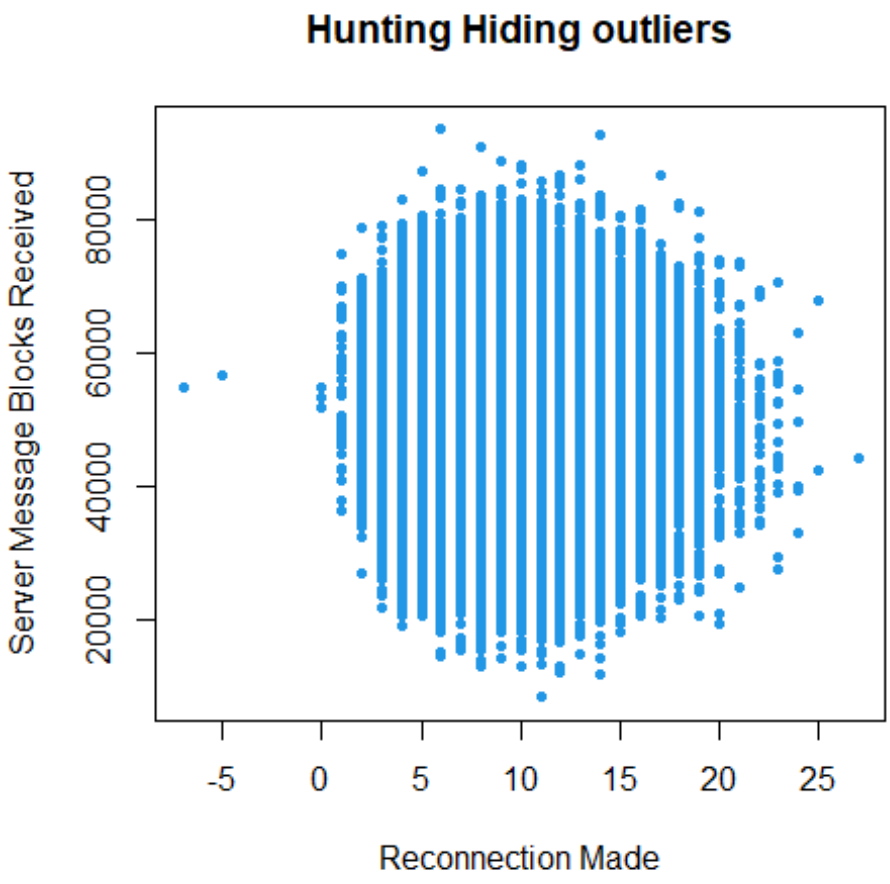
```
plot(data_DVH$RC_DVH,data_DVH$Conn_DVH, main='Hunting Hiding
outliers',pch=20,xlab = "Reconnection Made",ylab = "Connection Made",col=4)
```



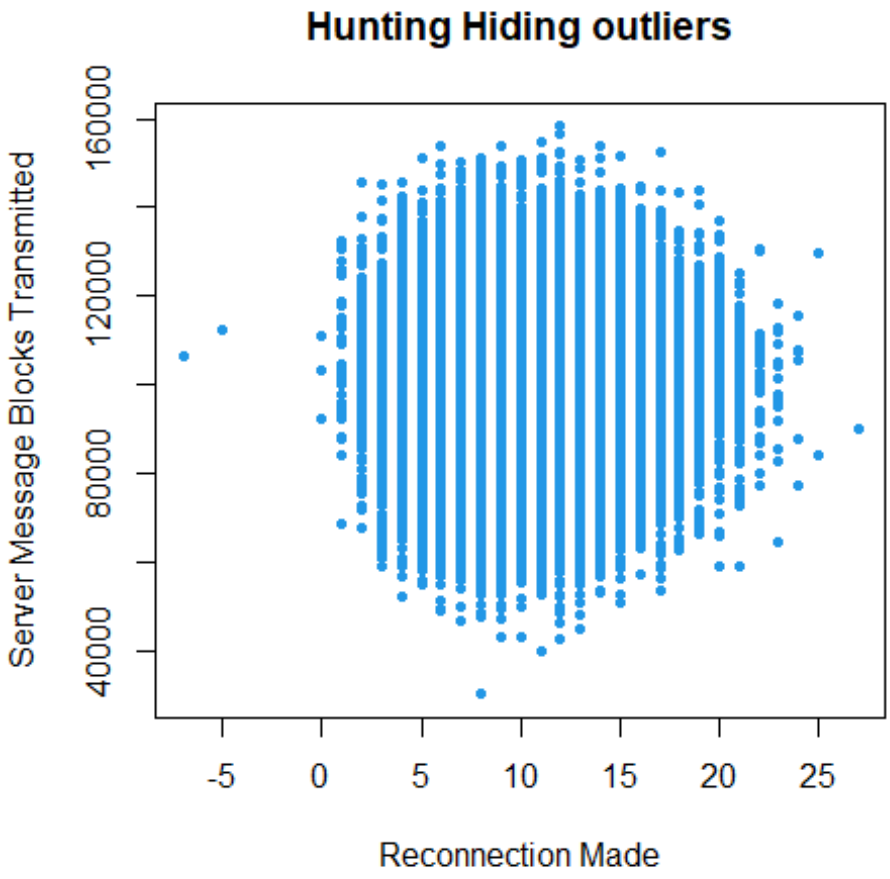

```
plot(data_DVH$RC_DVH,data_DVH$ST_DVH, main='Hunting Hiding
outliers',pch=20,xlab = "Reconnection Made",ylab = "Sessions Timed-
Out",col=4)
```



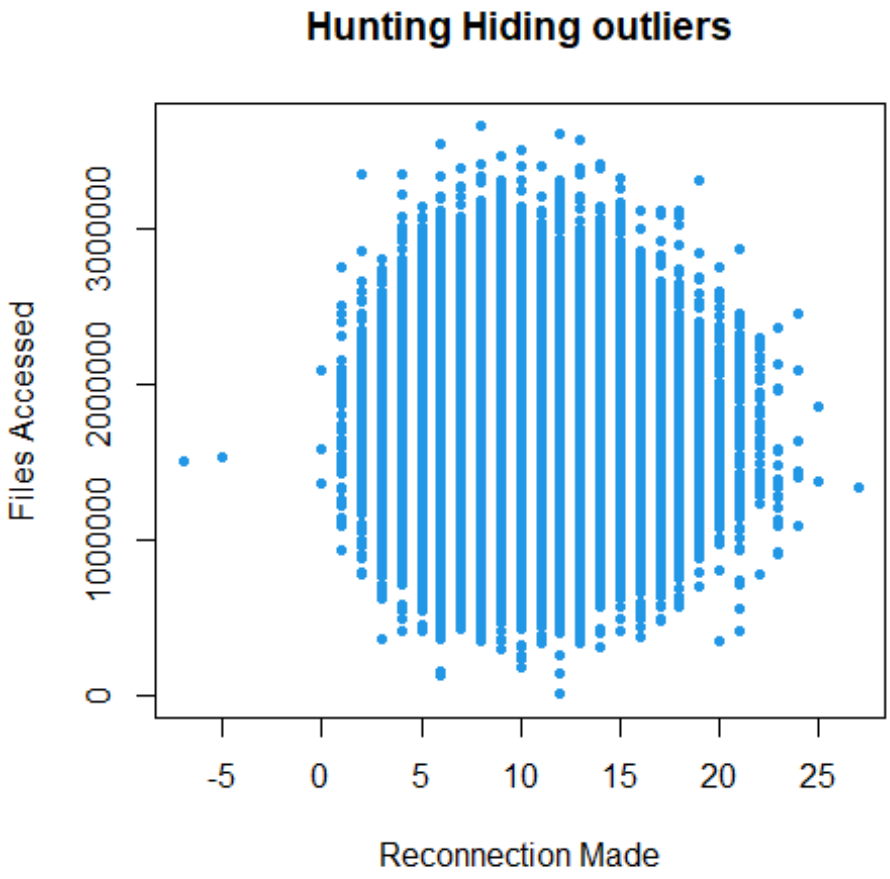

```
plot(data_DVH$RC_DVH,data_DVH$SMBR_DVH, main='Hunting Hiding
outliers',pch=20,xlab = "Reconnection Made",ylab = "Server Message Blocks
Received",col=4)
```




```
plot(data_DVH$RC_DVH,data_DVH$SMBT_DVH, main='Hunting Hiding
outliers',pch=20,xlab = "Reconnection Made",ylab = "Server Message Blocks
Transmitted",col=4)
```




```
plot(data_DVH$RC_DVH,data_DVH$FA_DVH, main='Hunting Hiding
outliers',pch=20,xlab = "Reconnection Made",ylab = "Files Accessed",col=4)
```

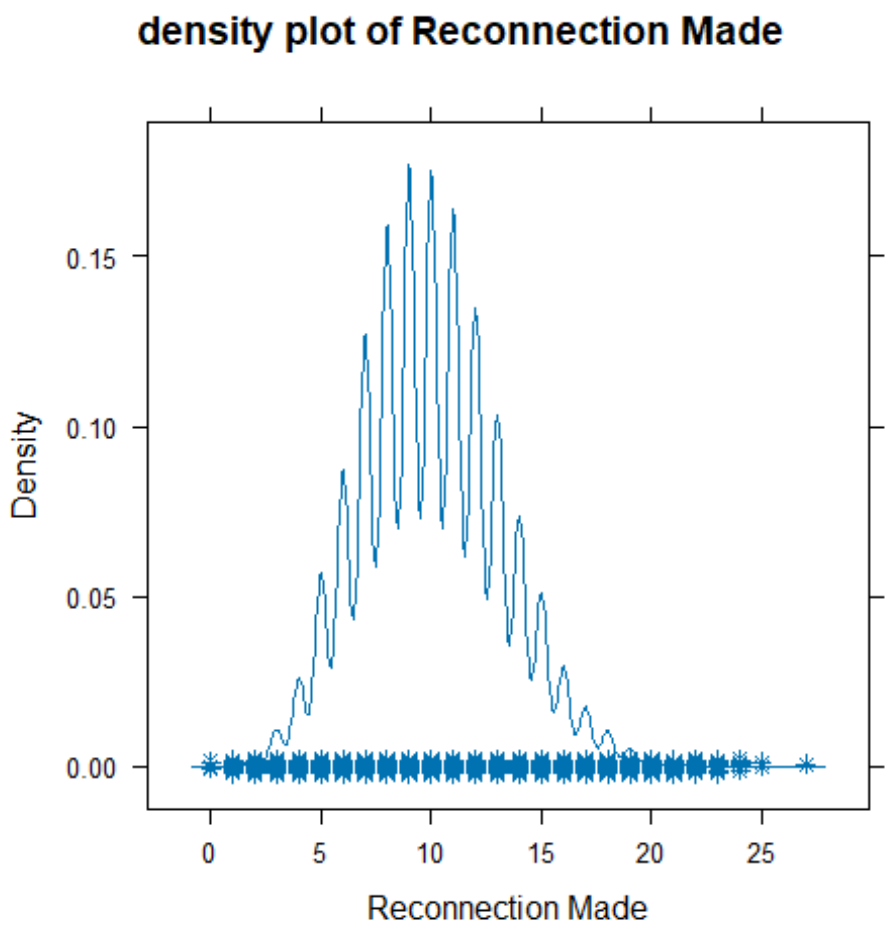


Interpretation

From the above scatter plot, it is clearly seen that Reconnection made has two outliers which is lower than ~ 0 and get separated from cluster.

As it is only two data and add no value for analysis, it is good decision to remove them from dataset.

```
nr <- which(data_DVH$RC_DVH < 0)  #Find row number with RC_DVH < 0
data_DVH <- data_DVH[-c(nr),]
densityplot( ~ data_DVH$RC_DVH, pch=8,main="density plot of Reconnection
Made",xlab="Reconnection Made")
```



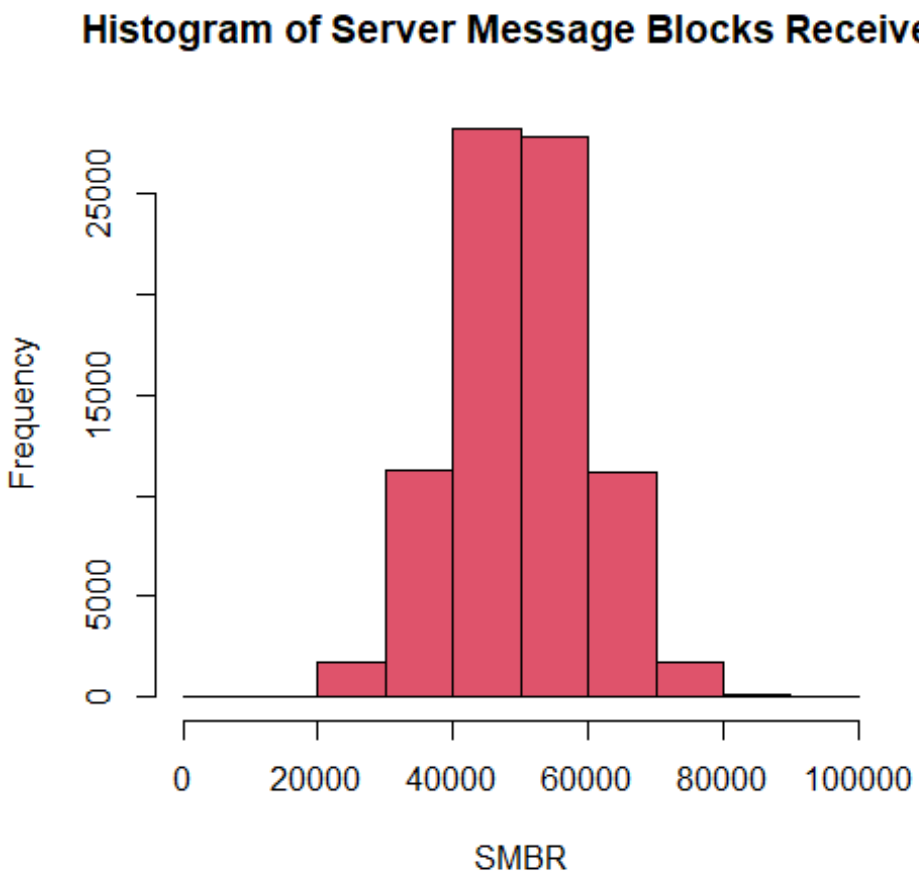
Interpretation
Above code filter data which are lower than ~ -5 as and remove filtered value from original dataset.
Above density plot is evidence that two outliers are successfully removed from dataset.

2. Organizing Data

1. Scatter Plots

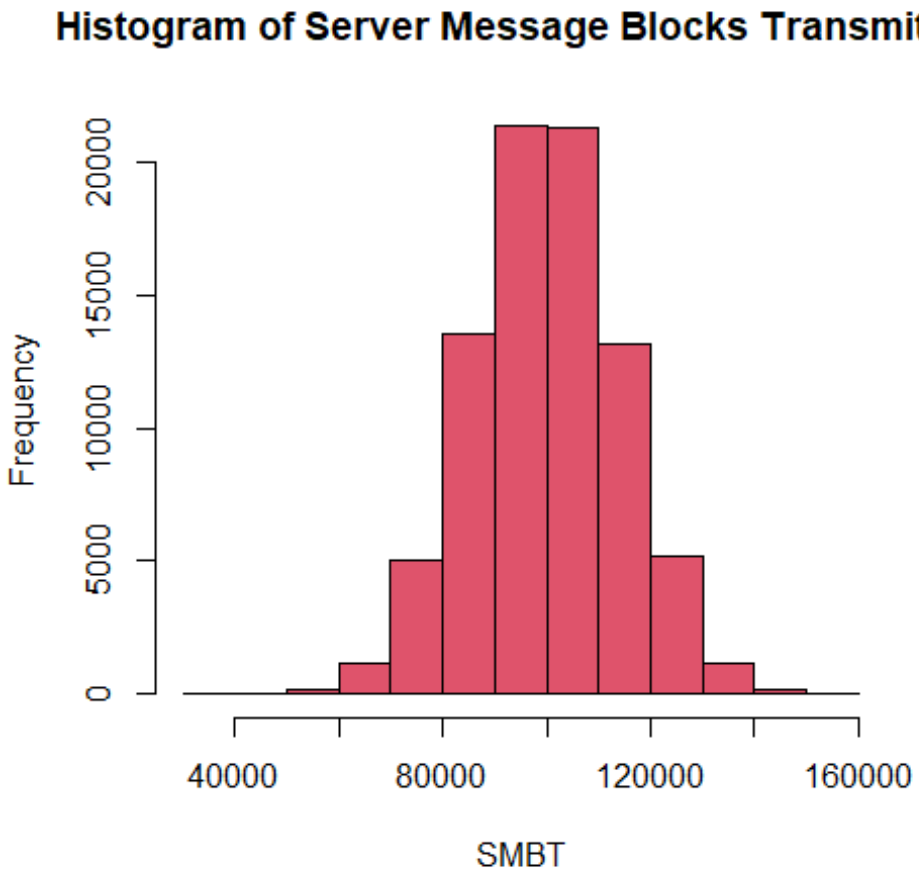
a. Create a histogram for Server Message Blocks Received.

```
hist(data_DVH$SMBR_DVH,
     col=2,
     border = 1,
     main="Histogram of Server Message Blocks Received",
     xlab = "SMBR",
     breaks = 10
          )
```



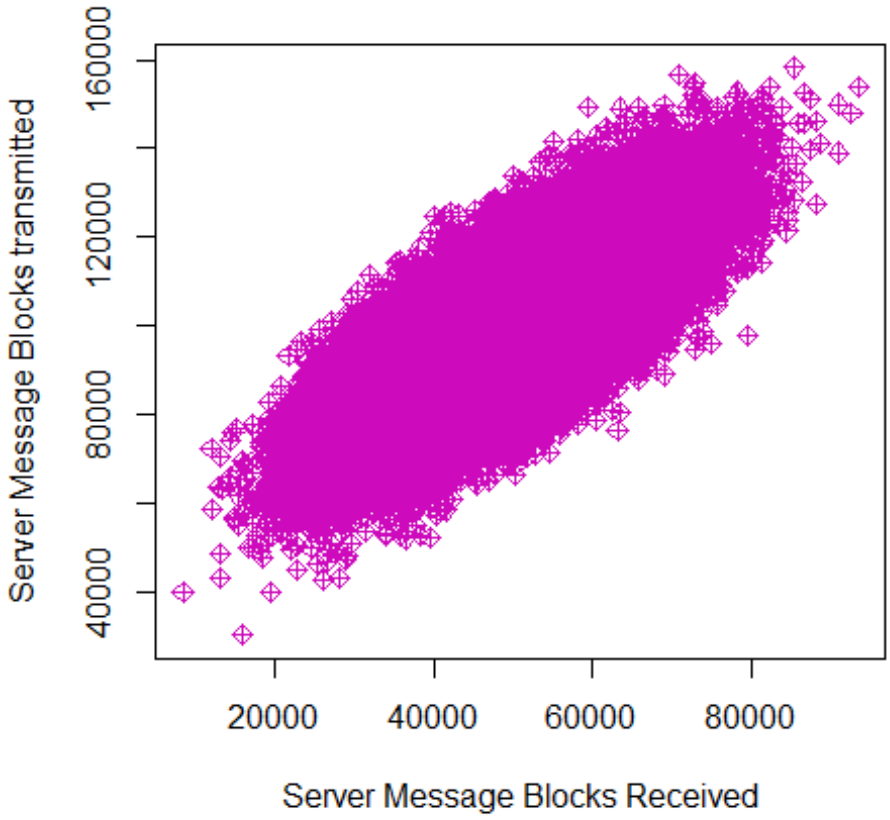
b. Create a histogram for Server Message Blocks Transmitted.

```
hist(data_DVH$SMBT_DVH,
     col=2,
     border = 1,
     main="Histogram of Server Message Blocks Transmitted",
     xlab = "SMBT",
     breaks = 10
          )
```



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(data_DVH$SMBR_DVH,data_DVH$SMBT_DVH,
    pch=9,
    xlab = "Server Message Blocks Received",
    ylab = "Server Message Blocks transmitted",
    col=6
    )
```



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

Ans.
The histograms will show the frequency distribution of the Server Message Blocks Received and Server Message Blocks Transmitted variables, respectively. The scatter plot will show the relationship between the two variables, and any outliers will be visible. From scatter plot, it is clear that data trend is increasing in positive direction.

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

```
round(cor(data_DVH$SMBR_DVH,data_DVH$SMBT_DVH),3)

## [1] 0.763
```

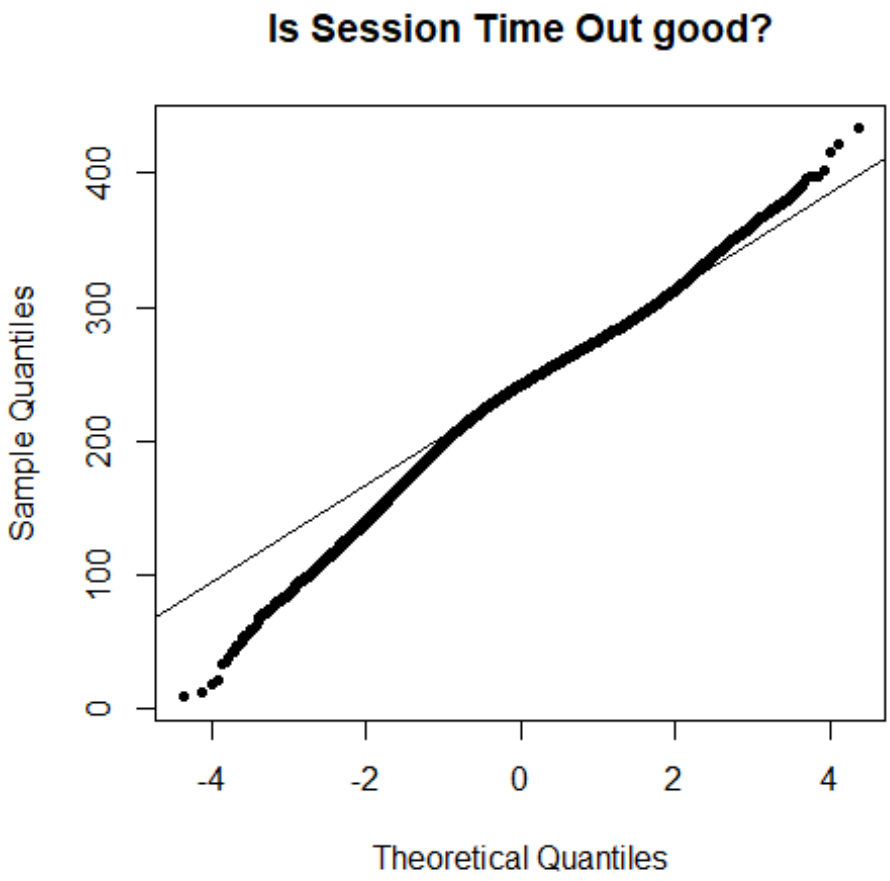
Interpretation
Reason for choosing coefficient is to measures the linear relationship between two variables.
To Conclude, correlation coefficient is 0.763, it means that there is a strong positive linear relationship between the server message block received and server message block transmitted.

3. Inference

1. Normality

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

```
qqnorm(data_DVH$ST_DVH, main="Is Session Time Out good?", pch=20)
qqline(data_DVH$ST_DVH)
```



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

```
sample_ST_DVH <- sample(data_DVH$ST_DVH,5000)
shapiro.test(sample_ST_DVH)

##
##  Shapiro-Wilk normality test
##
## data:  sample_ST_DVH
## W = 0.97901, p-value < 2.2e-16
```

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

Ans.
Session Times out is not normally distributed. Because, the QQ Normal plot for Sessions Timed Out shows that the data is not quite normally distributed. There is some deviation from the diagonal line in both tails of the plot, indicating that the data may be skewed.
A Shapiro-Wilk test for normality was also conducted. The null hypothesis is that the data is normally distributed, and the alternative hypothesis is that the data is not normally distributed. The p-value for the test was less than 0.05, which means that we can reject the null hypothesis and conclude that the data is not normally distributed.

2. Statistically Significant Differences

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

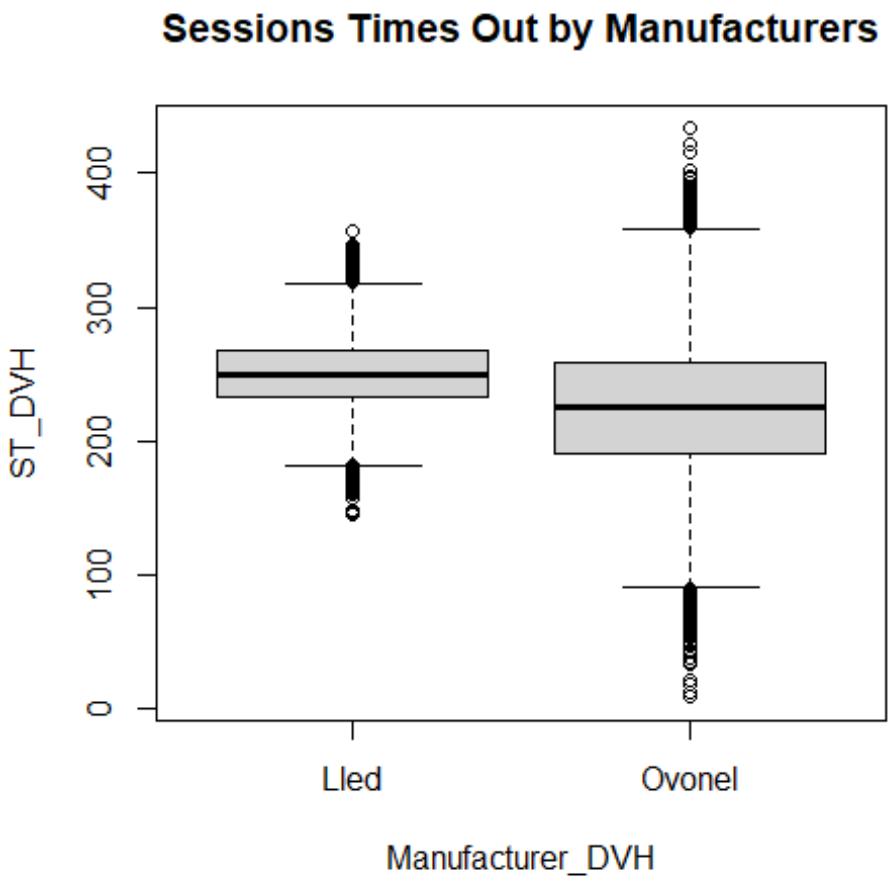
```
manufacturer_Lled_DVH <- data_DVH$ST_DVH[data_DVH$Manufacturer_DVH == "Lled"]
manufacturer_Ovonel_DVH <- data_DVH$ST_DVH[data_DVH$Manufacturer_DVH ==
"Ovonel"]


var.test(manufacturer_Lled_DVH, manufacturer_Ovonel_DVH)

##
##  F test to compare two variances
##
## data:  manufacturer_Lled_DVH and manufacturer_Ovonel_DVH
## F = 0.24907, num df = 41076, denom df = 41076, p-value < 2.2e-16
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.2443023 0.2539378
## sample estimates:
## ratio of variances
##          0.2490734

wilcox.test(ST_DVH ~ Manufacturer_DVH,data=data_DVH ,var.equal = FALSE)

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  ST_DVH by Manufacturer_DVH
## W = 1139390793, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0
```

b. Explain why you chose the test you did.

Ans.
T-test can not be applied as variance is not close to 1.

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

```
boxplot( ST_DVH ~ Manufacturer_DVH,
data=data_DVH,
main="Sessions Times Out by Manufacturers")
```



Ans.

Since Wilcoxon Test resulted to p-value < 2.2e-16. Null Hypothesis can be rejected and we have strong evidence to go with Alternate Hypothesis. Therefore, Sessions Times Out are different between Manufacturers.

3. Multiple Statistical Differences

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

```
# One Way ANOVA
summary(aov(FA_DVH~Server_DVH, data=data_DVH))
```

```
##                Df            Sum Sq         Mean Sq F value Pr(>F)
## Server_DVH      6  1877294907018140 312882484503023    2079 <2e-16 ***
## Residuals   82147 12360616982600226    150469487414
```

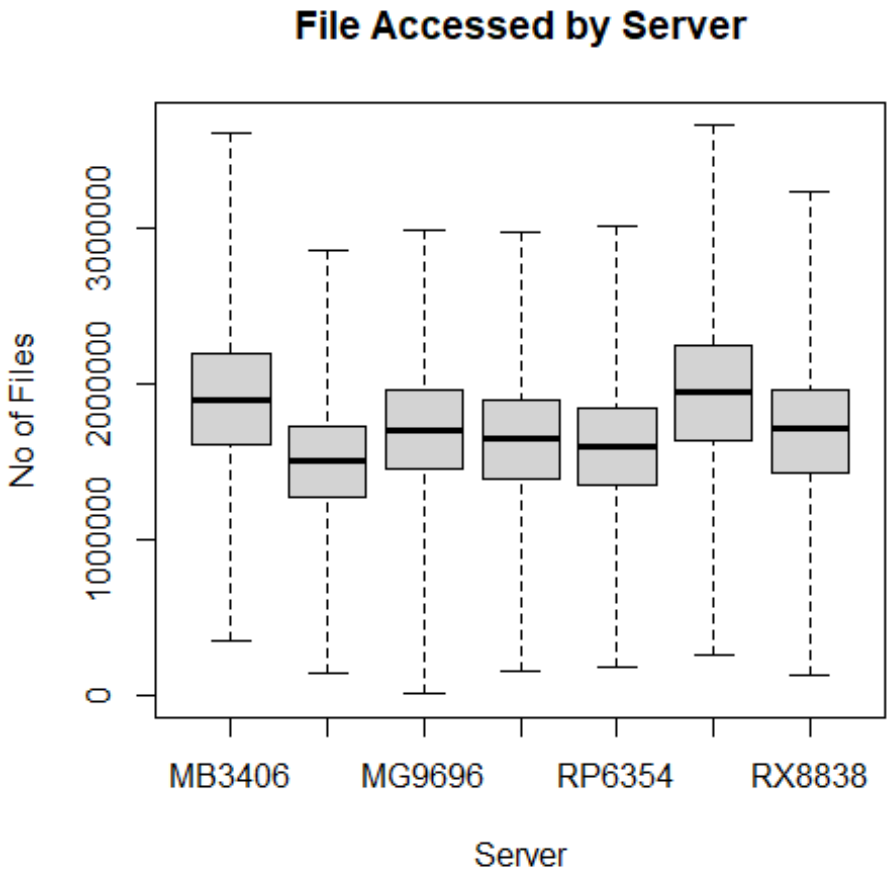
```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Interpretation

Based on the ANOVA output, there is strong evidence to suggest that the mean number of Files Accessed varies by Server, with a very low p-value of less than 0.001. The F-value of 2079 also indicates a large difference in means between the groups, which reinforces the statistical significance of the result.

These results suggest that there is a statistically significant difference in the number of Files Accessed across different servers, and further investigation may be warranted to explore the nature of this difference.

```
#Comparing Files by server

boxplot(FA_DVH~Server_DVH, data=data_DVH,
        main="File Accessed by Server",
        xlab="Server",
        ylab ="No of Files",
        range=0)
```



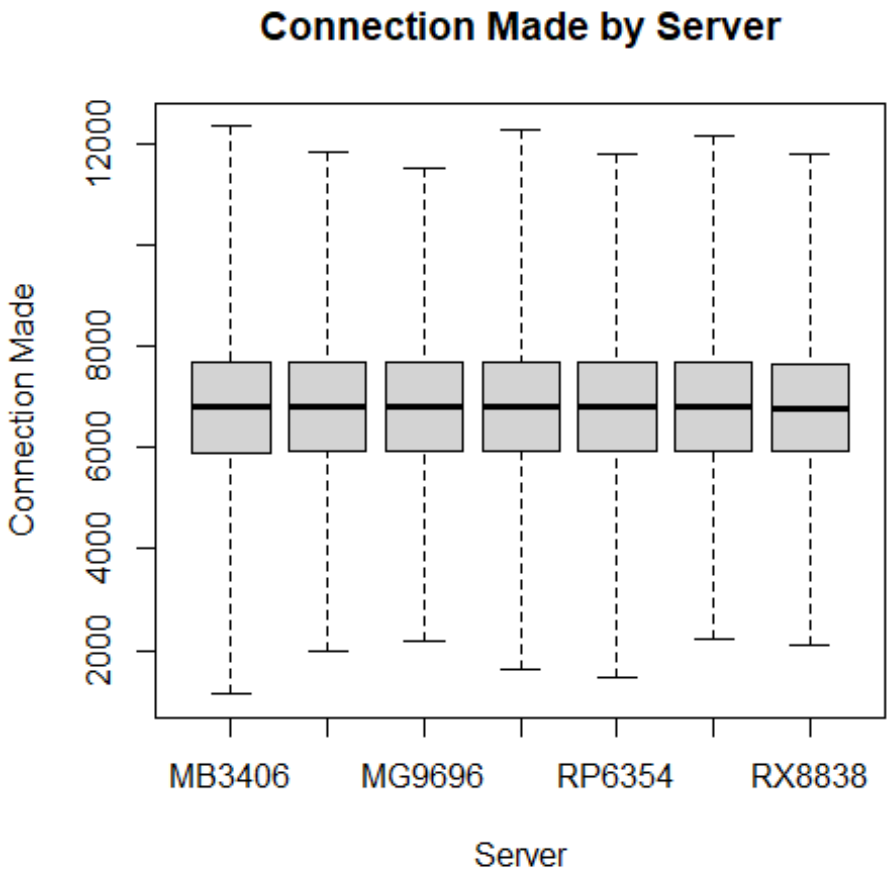
Interpretation
The boxplot show the distribution of Files Accessed for each Server. There is a significant difference between the means of the groups, as we are able to see it in the above boxplot.

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

```
# One Way ANOVA
summary(aov(Conn_DVH~Server_DVH, data=data_DVH))

##                Df       Sum Sq Mean Sq F value Pr(>F)
## Server_DVH      6      6715799 1119300    0.66  0.682
## Residuals   82147 139222450571 1694797
```

Interpretation
The output shows the results of an ANOVA test for the variable “Connections Made” across different servers. The null hypothesis in this test is that there is no significant difference in the mean number of connections made across the different servers.
Based on the output provided, the p-value for the F-test is 0.682, which is greater than the significance level of 0.05. Therefore, we fail to reject the null hypothesis, and there is not enough evidence to suggest that the mean number of connections made varies significantly across the different servers.

```
#Comparing Files by server

boxplot(Conn_DVH~Server_DVH, data=data_DVH,
        main="Connection Made by Server",
        xlab="Server",
        ylab ="Connection Made",
        range=0)
```



Interpretation

The boxplot show the distribution of Connection made for each Server. There is no such significant difference between the means of the groups, as we are able to see it in the above boxplot.