Mini Project- Factor Hair Revised

Advanced Statistics

Table of Contents

1. Project Objective

The objective of the report is to explore the Factor Hair Data ("Factor-Hair-Revised.csv") in R and generate insights about the data set. This exploration report will consists of the following:

- Understanding the structure of dataset
- Graphical exploration
- Descriptive statistics
- General Insights from the dataset
- Checking for Outliers & Missing Value
- Test for Multicollinearity
- Simple Linear Regression between dependent and Independent Variable
- Perform Factor analysis
- Perform multiple linear regression on Factors generated
- Validate the Model

2. Assumptions

Following assumption we made for this analysis

- The Data Provided to us was not tempered.
- Linearity - Linearity assumes a straight line relationship between each of the two variables.
- Homoscedasticity - Homoscedasticity assumes that data is equally distributed about the regression line.

3. Exploratory Data Analysis Step by Step approach

A Typical Data exploration activity consists of the following steps:

1. Environment Set up and Data Import
2. Variable Identification
3. Univariate Analysis
4. Bi-Variate Analysis
5. Outlier Identification
6. Feature Creation & Exploration

We shall follow these steps in exploring the provided dataset.

3.1 Environment Set up and Data Import

3.1.1 Install necessary Packages and Invoke Libraries

Following are the Libraries are used in the analysis

- tidyverse
- dplyr
- ggplot2
- DataExplorer
- corrplot
- car

- Metrics
- GPArotation
- MASS
- psych

Code for loading library

```
#Libraries Required
library(tidyverse)
library(dplyr)
library(ggplot2)
library(DataExplorer)
library(corrplot)
library(car)
library(Metrics)
library(GPArotation)
library(MASS)
library(psych)
```

Please refer to Appendix A for Source Code.

3.1.2 Set up working Directory

Setting a working directory on starting of the R session makes importing and exporting data files and code files easier. Basically, working directory is the location/ folder on the PC where you have the data, codes etc. related to the project.

Code for setting working directory

```
#Setting the Working Directory
setwd("E:/000GL/000 0Projects/003 Factor Hair Revised")
getwd()
```

Please refer to Appendix A for Source Code.

3.1.3 Import and Read the Dataset

The given dataset is in .csv format. Hence, the command 'read.csv' is used for importing the file.

Code for Read the Dataset

```
# Importing Data
## Import the Cold_Storage_Temp_Data.csv
myData = read.csv("Factor-Hair-Revised.csv")
myData
```

Please refer to Appendix A for Source Code.

3.2 Variable Identification

Functions is used for variable identifications with there functionality:

- **class(myData):** To identify the class of Data
- **str(myData):** compactly display the (abbreviated) contents of lists.

- **names(myData):** Names of DataFrame variable
- **dim(myData):** Dimensions of Dataframe
- **head(myData):** Display top 6 elements of Variables
- **tail(myData):** Display last 6 elements of variables
- **summary(myData):** Provides an overview of Data
- **plot_missing(myData):** Plot if the variable having any data missing

Code for general Variable Identification

```
# General Analysis
#Variable Identification
##Check the Class of Data
class(myData)

## First Inspection of Dataset using str
str(myData)

## Find the name of variable
names(myData)

## find the dimension of Data
dim(myData)

## find first 6 elements of Data
head(myData)

## find last 5 elements of Data
tail(myData)

## find summary of myData to get Min,median,Mean and Max with First and 3rd quartile.
summary(myData)
```

Please refer to Appendix A for Source Code.

3.2.1 Variable Identification – Inferences

Our Data contain 100 obs. of 13 variables with 1 variables as factors(Which we will remove as it is just S.No. with no significance) and 12 numerical data.

Column name of our Data are:

- ID
- ProdQual
- Ecom
- TechSup
- CompRes
- Advertising

- ProdLine
- SalesFImage
- ComPricing
- WartyClaim
- OrdBilling
- DelSpeed
- Satisfaction

We also checked the top 6 and last 6 elements of each variable with command head and tail and summary of data as below.

<u>Satisfaction</u> is our Dependent Variable, <u>ID</u> needs to be removed (because just being a serial no.) and rest all are Independent Variable

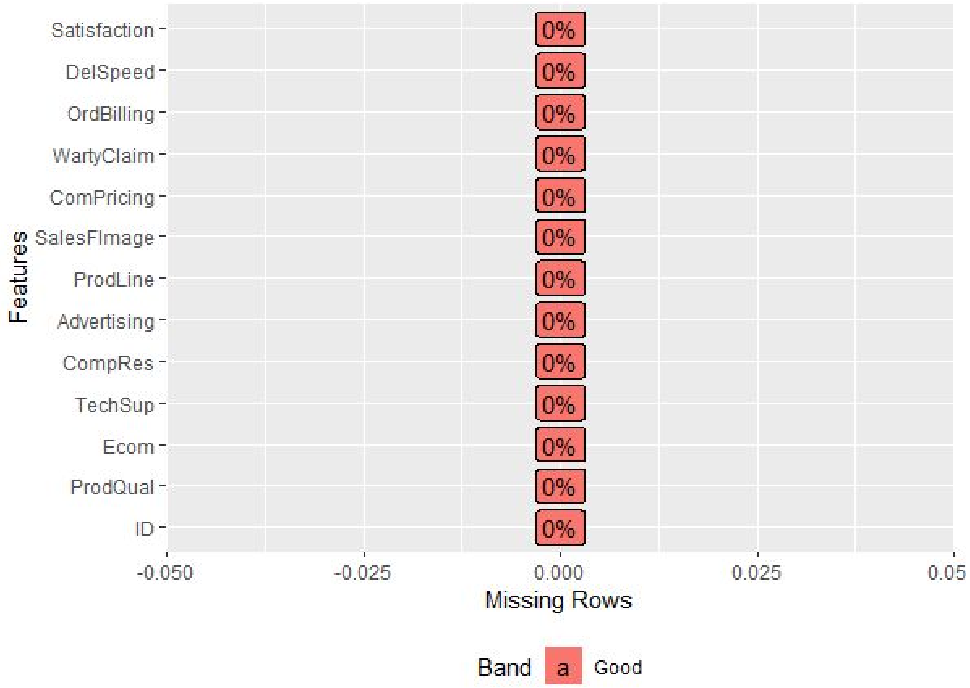
Command for variable identifications and Output

```
> # General Analysis
> #Variable Identification
> ##Check the Class of Data
> class(myData)
[1] "data.frame"
>
> ## First Inspection of Dataset using str
> str(myData)
'data.frame':	100 obs. of  13 variables:
 $ ID          : Factor w/ 100 levels "1","2","3","4",..: 1 2 3 4 5 6 7 8 9 10 ...
 $ ProdQual    : num  8.5 8.2 9.2 6.4 9 6.5 6.9 6.2 5.8 6.4 ...
 $ Ecom        : num  3.9 2.7 3.4 3.3 3.4 2.8 3.7 3.3 3.6 4.5 ...
 $ TechSup     : num  2.5 5.1 5.6 7 5.2 3.1 5 3.9 5.1 5.1 ...
 $ CompRes     : num  5.9 7.2 5.6 3.7 4.6 4.1 2.6 4.8 6.7 6.1 ...
 $ Advertising : num  4.8 3.4 5.4 4.7 2.2 4 2.1 4.6 3.7 4.7 ...
 $ ProdLine    : num  4.9 7.9 7.4 4.7 6 4.3 2.3 3.6 5.9 5.7 ...
 $ SalesFImage : num  6 3.1 5.8 4.5 4.5 3.7 5.4 5.1 5.8 5.7 ...
 $ ComPricing  : num  6.8 5.3 4.5 8.8 6.8 8.5 8.9 6.9 9.3 8.4 ...
 $ WartyClaim  : num  4.7 5.5 6.2 7 6.1 5.1 4.8 5.4 5.9 5.4 ...
 $ OrdBilling  : num  5 3.9 5.4 4.3 4.5 3.6 2.1 4.3 4.4 4.1 ...
 $ DelSpeed    : num  3.7 4.9 4.5 3 3.5 3.3 2 3.7 4.6 4.4 ...
 $ Satisfaction: num  8.2 5.7 8.9 4.8 7.1 4.7 5.7 6.3 7 5.5 ...
>
> ## Find the name of variable
> names(myData)
 [1] "ID"           "ProdQual"     "Ecom"         "TechSup"      "CompRes"      "Advertising"
 [7] "ProdLine"     "SalesFImage"  "ComPricing"   "WartyClaim"   "OrdBilling"   "DelSpeed"
[13] "Satisfaction"
>
> ## find the dimension of Data
> dim(myData)
[1] 100  13
>
> ## find first 6 elements of Data
> head(myData)
  ID ProdQual Ecom TechSup CompRes Advertising ProdLine SalesFImage ComPricing WartyClaim
```

```
1  1      8.5  3.9     2.5     5.9         4.8      4.9         6.0        6.8        4.7
2  2      8.2  2.7     5.1     7.2         3.4      7.9         3.1        5.3        5.5
3  3      9.2  3.4     5.6     5.6         5.4      7.4         5.8        4.5        6.2
4  4      6.4  3.3     7.0     3.7         4.7      4.7         4.5        8.8        7.0
5  5      9.0  3.4     5.2     4.6         2.2      6.0         4.5        6.8        6.1
6  6      6.5  2.8     3.1     4.1         4.0      4.3         3.7        8.5        5.1
  OrdBilling DelSpeed Satisfaction
1        5.0      3.7          8.2
2        3.9      4.9          5.7
3        5.4      4.5          8.9
4        4.3      3.0          4.8
5        4.5      3.5          7.1
6        3.6      3.3          4.7
>
> ## find last 5 elements of Data
> tail(myData)
     ID ProdQual Ecom TechSup CompRes Advertising ProdLine SalesFImage ComPricing WartyClaim
95   95      9.3  3.8     4.0     4.6         4.7      6.4         5.5        7.4        5.3
96   96      8.6  4.8     5.6     5.3         2.3      6.0         5.7        6.7        5.8
97   97      7.4  3.4     2.6     5.0         4.1      4.4         4.8        7.2        4.5
98   98      8.7  3.2     3.3     3.2         3.1      6.1         2.9        5.6        5.0
99   99      7.8  4.9     5.8     5.3         5.2      5.3         7.1        7.9        6.0
100 100      7.9  3.0     4.4     5.1         5.9      4.2         4.8        9.7        5.7
    OrdBilling DelSpeed Satisfaction
95         3.6      3.4          7.7
96         4.9      3.6          7.3
97         4.2      3.7          6.3
98         3.1      2.5          5.4
99         4.3      3.9          6.4
100        3.4      3.5          6.4
>
> ## find summary of myData to get Min,median,Mean and Max with First and 3rd quartile.
> summary(myData)
       ID         ProdQual            Ecom           TechSup          CompRes         Advertising
 1      : 1   Min.   : 5.000   Min.   :2.200   Min.   :1.300   Min.   :2.600   Min.   :1.900
 2      : 1   1st Qu.: 6.575   1st Qu.:3.275   1st Qu.:4.250   1st Qu.:4.600   1st Qu.:3.175
 3      : 1   Median : 8.000   Median :3.600   Median :5.400   Median :5.450   Median :4.000
 4      : 1   Mean   : 7.810   Mean   :3.672   Mean   :5.365   Mean   :5.442   Mean   :4.010
 5      : 1   3rd Qu.: 9.100   3rd Qu.:3.925   3rd Qu.:6.625   3rd Qu.:6.325   3rd Qu.:4.800
 6      : 1   Max.   :10.000   Max.   :5.700   Max.   :8.500   Max.   :7.800   Max.   :6.500
 (Other):94
    ProdLine       SalesFImage      ComPricing      WartyClaim      OrdBilling       DelSpeed
 Min.   :2.300   Min.   :2.900   Min.   :3.700   Min.   :4.100   Min.   :2.000   Min.   :1.600
 1st Qu.:4.700   1st Qu.:4.500   1st Qu.:5.875   1st Qu.:5.400   1st Qu.:3.700   1st Qu.:3.400
 Median :5.750   Median :4.900   Median :7.100   Median :6.100   Median :4.400   Median :3.900
 Mean   :5.805   Mean   :5.123   Mean   :6.974   Mean   :6.043   Mean   :4.278   Mean   :3.886
 3rd Qu.:6.800   3rd Qu.:5.800   3rd Qu.:8.400   3rd Qu.:6.600   3rd Qu.:4.800   3rd Qu.:4.425
 Max.   :8.400   Max.   :8.200   Max.   :9.900   Max.   :8.100   Max.   :6.700   Max.   :5.500

  Satisfaction
 Min.   :4.700
 1st Qu.:6.000
 Median :7.050
 Mean   :6.918
 3rd Qu.:7.625
 Max.   :9.900
```

```
>
> ## plot the missing value
> plot_missing(myData)
```

Please refer to Appendix A for Source Code.



(Missing Variable Plot)

3.3 Univariate Analysis

"summary" provides an overview of data for Univariate Analysis

"hist" is used to plot the histogram of numeric variables.

"boxplot" is used to plot the boxplot of numeric variables and also help us to find outliers.

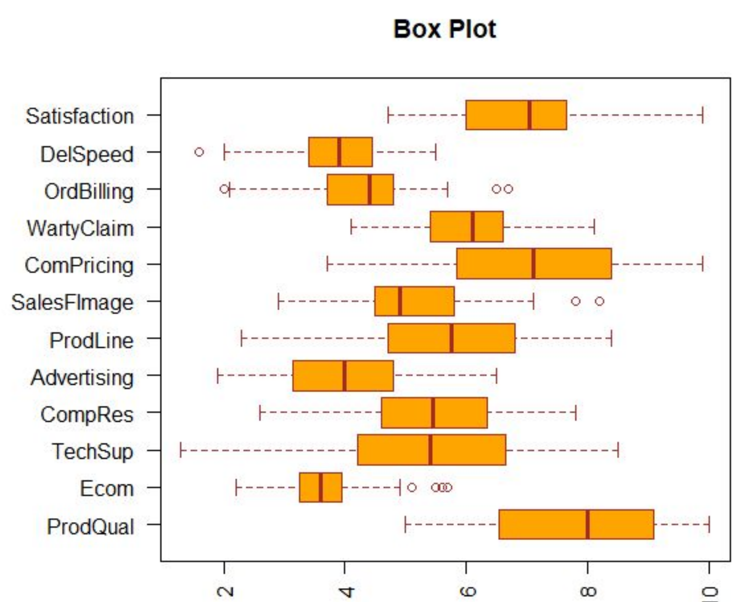
"sd" is used to find the standard deviation of numerical data

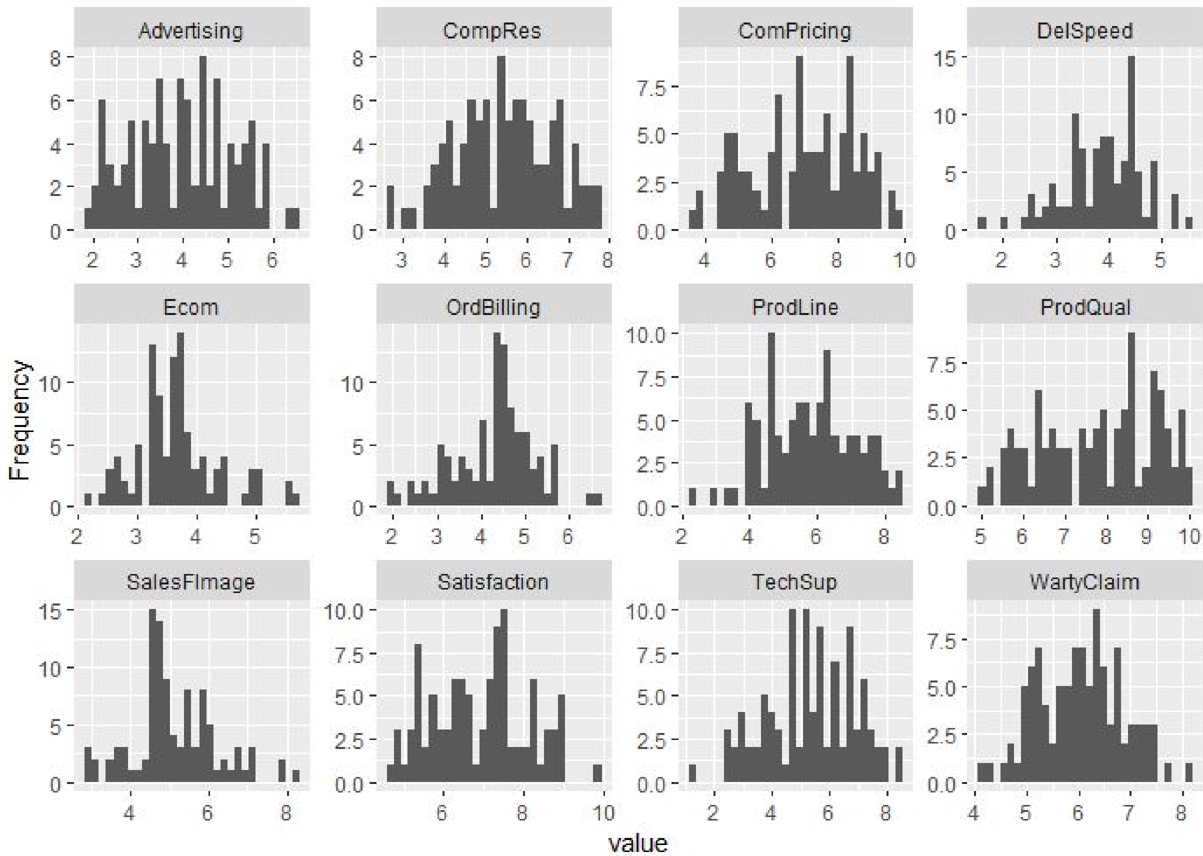
Inference:

Code for Univariate analysis with output

```
> ##UnivarientAnalysis
>
> ###Summary
> summary(myDataM)
    ProdQual          Ecom           TechSup         CompRes        Advertising       ProdLine
 Min.   : 5.000   Min.   :2.200   Min.   :1.300   Min.   :2.600   Min.   :1.900   Min.   :2.300
 1st Qu.: 6.575   1st Qu.:3.275   1st Qu.:4.250   1st Qu.:4.600   1st Qu.:3.175   1st Qu.:4.700
 Median : 8.000   Median :3.600   Median :5.400   Median :5.450   Median :4.000   Median :5.750
 Mean   : 7.810   Mean   :3.672   Mean   :5.365   Mean   :5.442   Mean   :4.010   Mean   :5.805
```

```
 3rd Qu.: 9.100   3rd Qu.:3.925   3rd Qu.:6.625   3rd Qu.:6.325   3rd Qu.:4.800   3rd Qu.:6.800
 Max.   :10.000   Max.   :5.700   Max.   :8.500   Max.   :7.800   Max.   :6.500   Max.   :8.400

  SalesFImage      ComPricing      WartyClaim      OrdBilling       DelSpeed      Satisfaction
 Min.   :2.900   Min.   :3.700   Min.   :4.100   Min.   :2.000   Min.   :1.600   Min.   :4.700
 1st Qu.:4.500   1st Qu.:5.875   1st Qu.:5.400   1st Qu.:3.700   1st Qu.:3.400   1st Qu.:6.000
 Median :4.900   Median :7.100   Median :6.100   Median :4.400   Median :3.900   Median :7.050
 Mean   :5.123   Mean   :6.974   Mean   :6.043   Mean   :4.278   Mean   :3.886   Mean   :6.918
 3rd Qu.:5.800   3rd Qu.:8.400   3rd Qu.:6.600   3rd Qu.:4.800   3rd Qu.:4.425   3rd Qu.:7.625
 Max.   :8.200   Max.   :9.900   Max.   :8.100   Max.   :6.700   Max.   :5.500   Max.   :9.900
> ##Histogram
> plot_histogram(myData,nrow = 4,ncol = 4)
> ##Box Plot
> par(mar=c(4,10,4,4))
> boxplot(myDataM,
+         horizontal = TRUE
+         ,las =2
+         ,main = "Box Plot"
+         ,col = "orange"
+         ,border = "brown")
> myDataM %>%
+   summarise_each(funs(sd(., na.rm=TRUE)))
  ProdQual      Ecom  TechSup  CompRes Advertising ProdLine SalesFImage ComPricing WartyClaim
1 1.396279 0.7005164 1.530457 1.208403    1.126943 1.315285     1.07232    1.545055  0.8197382
  OrdBilling  DelSpeed Satisfaction
1  0.9288398 0.7344372     1.191839
```
Please refer to Appendix A for Source Code.





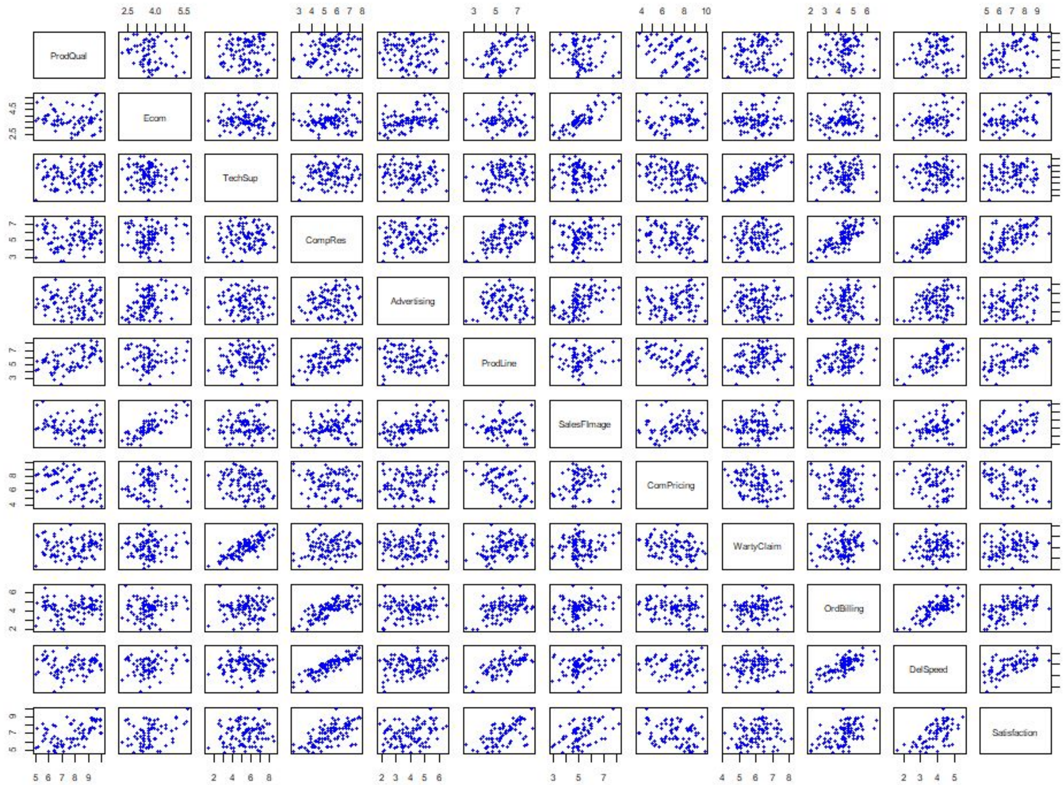
3.4 Bi-Variate Analysis

Since all the data is numeric in nature best way do Bi- vrient analysis is Scatter plot and Corplot

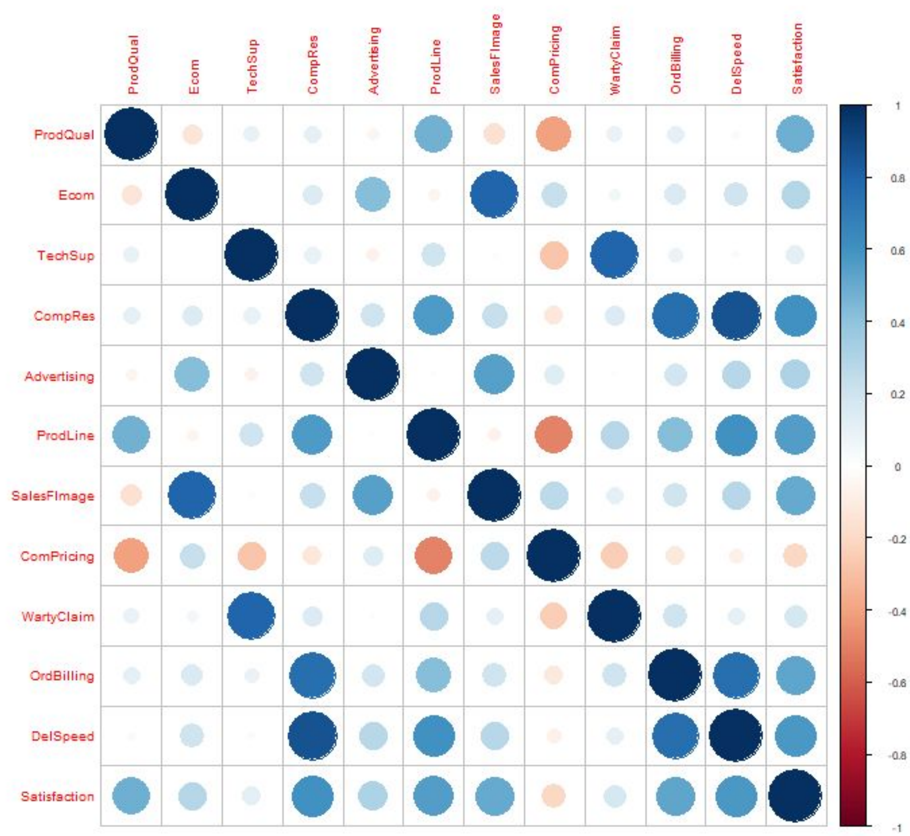
Plots and Output are as below

Code for bivariate Analysis

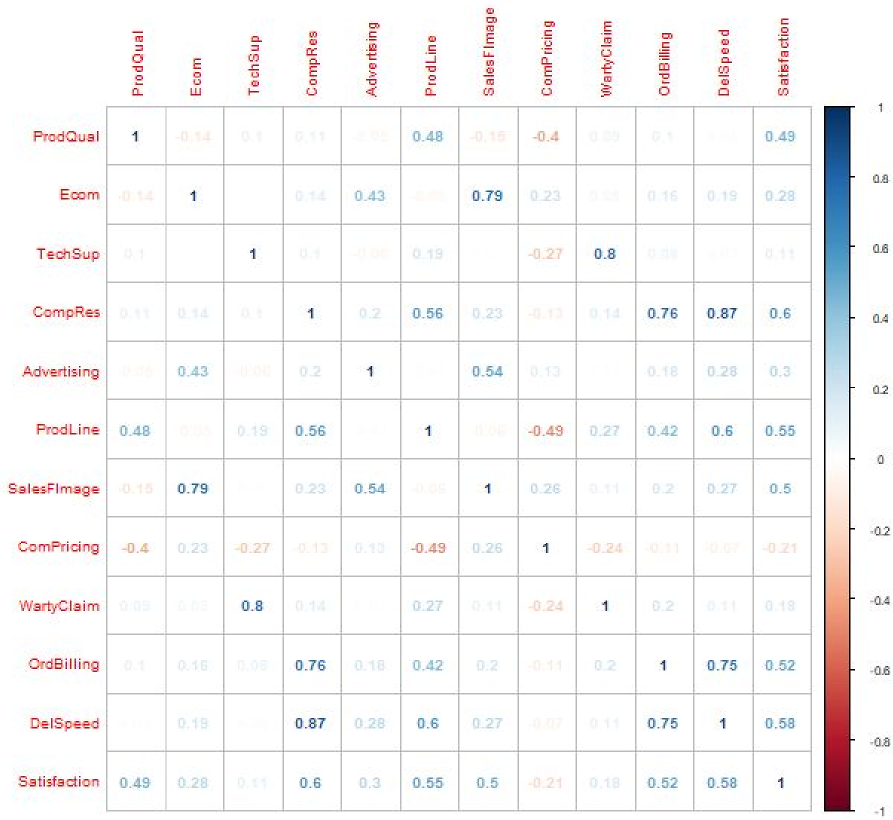
```
> ##Bivarient Analysis
>
> ##Scatterplot
> plot(myDataM, col="blue", cex.axis=0.75,cex.lab=5, pch=20)
>
> ##Corelation
> Data_cor <- cor(myDataM)
> cex.before <- par("cex")
> par(cex = 0.6)
> corrplot(Data_cor)
>
>
> ##Corelation
> corrplot(Data_cor, method = "number", number.digits = 2)
> par(cex = cex.before)
```



ProdQual
Ecom
TechSup
CompRes
Advertising
ProdLine
SalesFImage
ComPricing
WartyClaim
OrdBilling
DelSpeed
Satisfaction



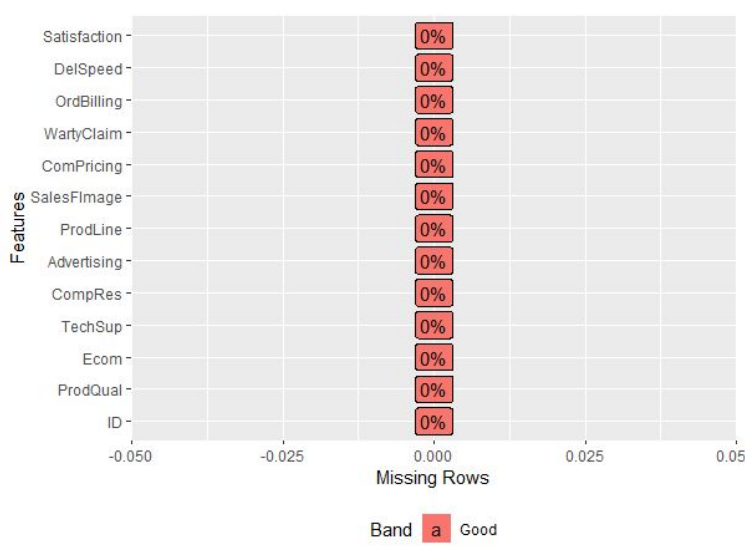
ProdQual
Ecom
TechSup
CompRes
Advertising
ProdLine
SalesFImage
ComPricing
WartyClaim
OrdBilling
DelSpeed
Satisfaction
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1



3.5 Missing Value Identification

plot_missing(myData) is used to check the missing variable and our data has no missing value

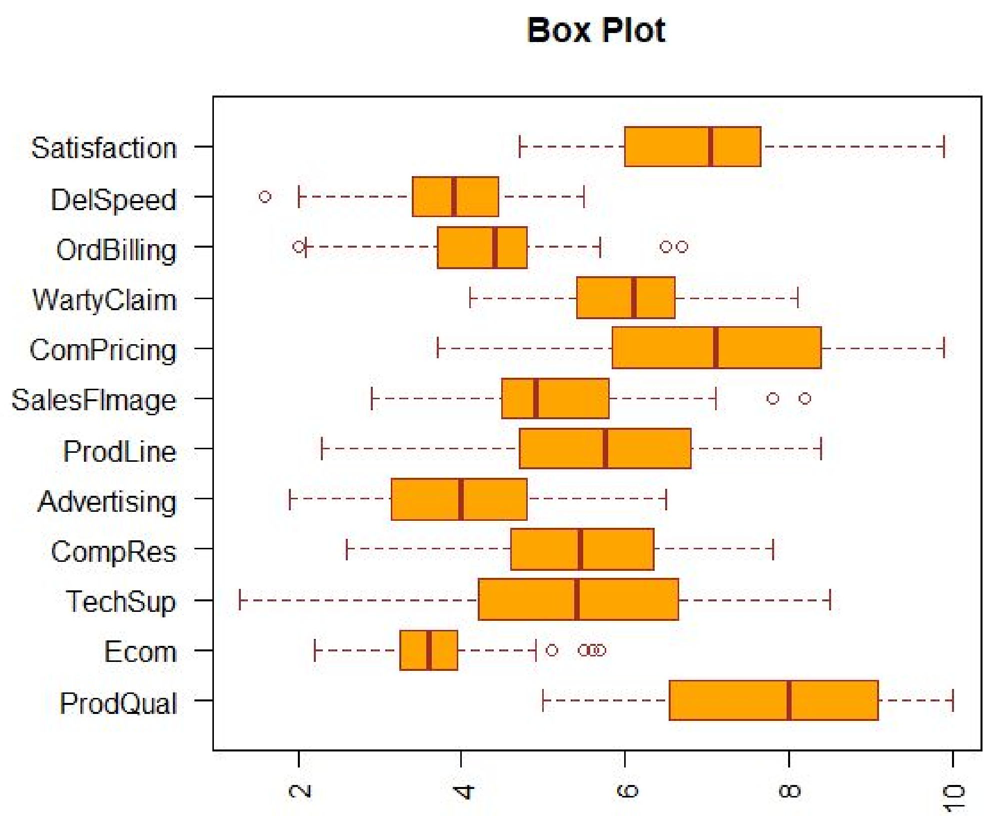
```
## plot the missing value
plot_missing(myData)
```



3.6 Outlier Identification

Inference:
Following Variables has Outliers:

- DelSpeed
- OrdBilling
- SalesFImage
- Ecom



4 Conclusion

5 Problem was assigned to us and here are the solutions

4.1 Exploratory data analysis on the dataset with charts, graphs. Outliers and missing values Analysis

All the Required points have been covered in the "3. Exploratory Data Analysis Step by Step approach"

4.2 Multicollinearity Analysis with Graphical Plot

To check multicollinearity we used the corplot and multicollinearity exist in our data.
We also found the variable having collinearity more than 0.6/-0.6

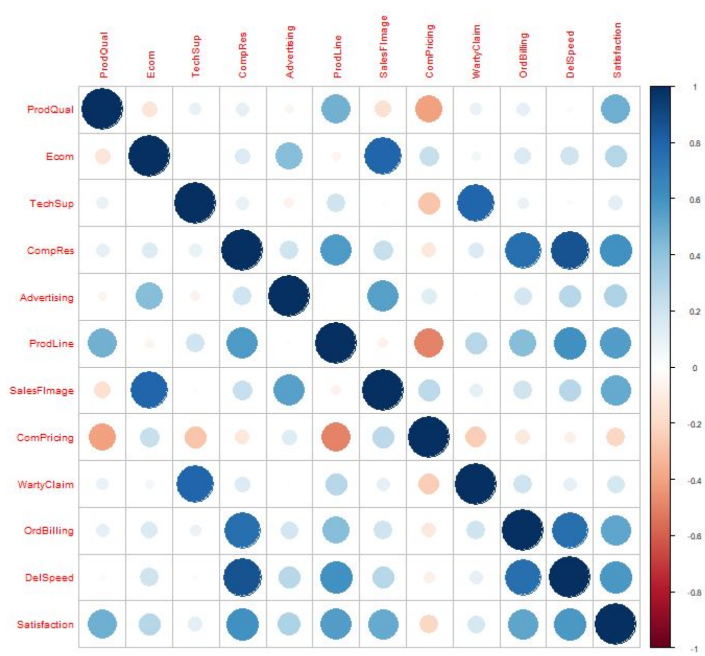
Inference:
Following Variable have multicollinearity more than 0.6/-0.6, which can be verified by Plots also

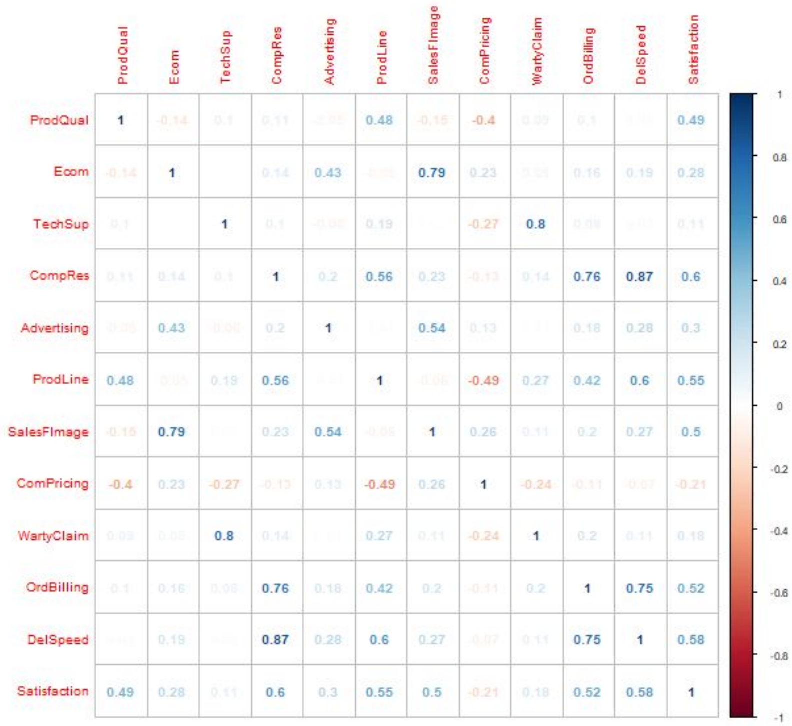
- Ecom : SalesFImage
- TechSup: WartyClaim
- CompRes: OrdBilling , DelSpeed, Satisfaction
- ProdLine: DelSpeed
- SalesFImage: Ecom
- WartyClaim: TechSup
- OrdBilling: CompRes, DelSpeed
- DelSpeed : CompRes, ProdLine, OrdBilling
- Satisfaction: CompRes

Plots and Output are as below
Code for bivariate Analysis

```
> ##Corelation
> Data_cor <- cor(myDataM)
> cex.before <- par("cex")
> par(cex = 0.6)
> corrplot(Data_cor)
>
>
> ##Corelation
> corrplot(Data_cor, method = "number", number.digits = 2)
> par(cex = cex.before)
>
> ##Variables having high corelation of more than 0.6
> for (i in 1:nrow(Data_cor)){
+   correlations <-  which((Data_cor[i,] > 0.6 | Data_cor[i,] < -0.6) & (Data_cor[i,] != 1))
+   if(length(correlations)> 0){
+     print(colnames(myDataM)[i])
```

```
+     print(correlations)
+   }
+ }
[1] "Ecom"
SalesFImage
          7
[1] "TechSup"
WartyClaim
         9
[1] "CompRes"
  OrdBilling     DelSpeed Satisfaction
          10           11           12
[1] "ProdLine"
DelSpeed
      11
[1] "SalesFImage"
Ecom
   2
[1] "WartyClaim"
TechSup
      3
[1] "OrdBilling"
 CompRes DelSpeed
       4       11
[1] "DelSpeed"
   CompRes    ProdLine OrdBilling
         4           6         10
[1] "Satisfaction"
CompRes
      4
```




ProdQual
Ecom
TechSup
CompRes
Advertising
ProdLine
SalesFImage
ComPricing
WartyClaim
OrdBilling
DelSpeed
Satisfaction
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1

4.3 Simple linear regression for the dependent variable with every independent variable

Dependent Variable :

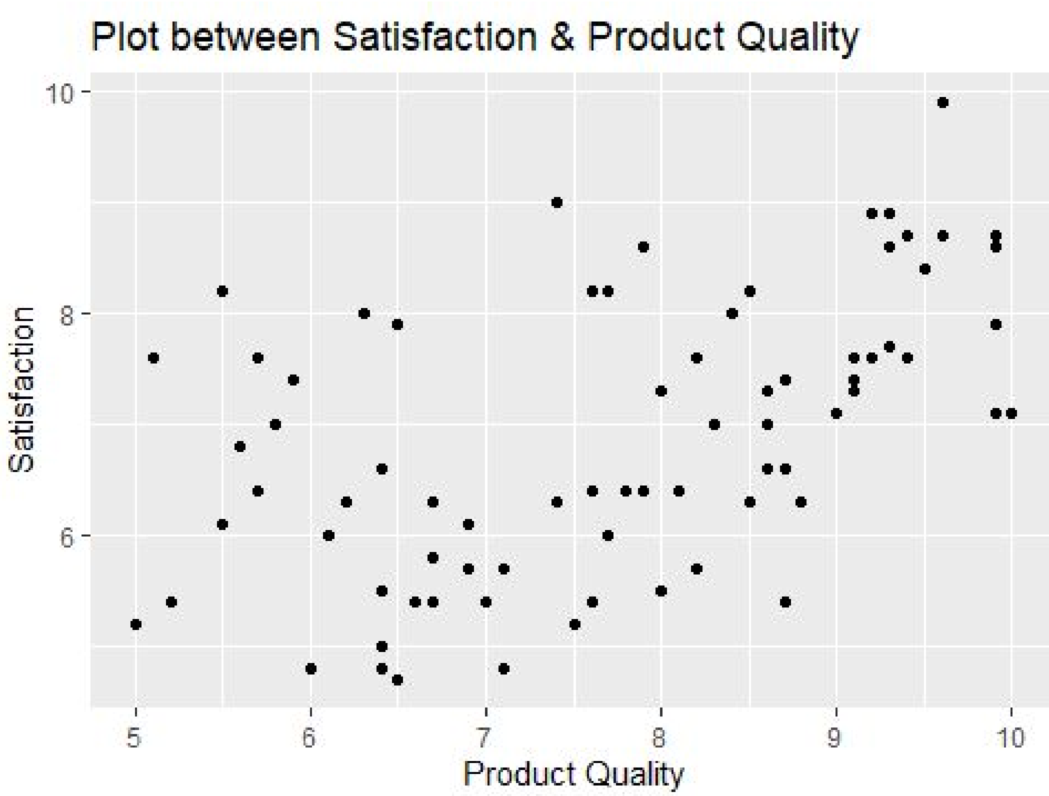
- Satisfaction

Independent Variable :

- ProdQual
- Ecom
- TechSup
- CompRes
- Advertising
- ProdLine
- SalesFImage
- ComPricing
- WartyClaim
- OrdBilling
- DelSpeed

4.3.1 Simple linear regression for the Satisfaction with ProdQual

Graphical Plot between Satisfaction with ProdQual



Correlation between Satisfaction & ProdQual is 0.486325

Residuals:

Residuals are essentially the difference between the actual observed response values (distance to stop dist in our case) and the response values that the model predicted

Our Residual are :

```
     Min       1Q   Median       3Q      Max 
-1.88746 -0.72711 -0.01577  0.85641  2.25220 
```

Coefficients:

The coefficients are two unknown constants that represent the intercept and slope terms in the linear model.

```
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  3.67593    0.59765   6.151 1.68e-08 ***
ProdQual     0.41512    0.07534   5.510 2.90e-07 ***
```

Model : Satisfaction = 3.67593 + 0.41512(ProdQual)

Coefficient - Estimate

The coefficient Estimate contains two rows; the first one is the intercept : 3.67593

The second row in the Coefficients is the slope: 0.41512

Coefficient - Standard Error

The coefficient Standard Error measures the average amount that the coefficient estimates vary from the actual average value of our response variable. We'd ideally want a lower number relative to its coefficients. The Standard Error can be used to compute an estimate of the expected difference in case we ran the model again and again.

So In our Model, We can say satisfaction can vary by 0.07534

Coefficient - t value

The coefficient t-value is a measure of how many standard deviations our coefficient estimate is far away from 0.

We want it to be far away from zero as this would indicate we could reject the null hypothesis - that is, we could declare a relationship between speed and distance exist

In our case, the t-statistic values are relatively far away from zero and are large relative to the standard error.

Coefficient - Pr(>t)

The Pr(>t) acronym found in the model output relates to the probability of observing any value equal or larger than t. A small p-value indicates that it is unlikely we will observe a relationship between Satisfaction & ProdQual variables due to chance. Typically, a p-value of 5% or less is a good cut-off point. In our model, the p-values are very close to zero.

Residual Standard Error

Residual Standard Error is measure of the quality of a linear regression fit. The Residual Standard Error is the average amount that the response (Satisfaction) will deviate from the true regression line.
Residual standard error: 1.047

Multiple R-squared

The R-squared (R2) statistic provides a measure of how well the model is fitting the actual data. It takes the form of a proportion of variance. R-Square is a measure of the linear relationship between our predictor variable (ProdQual) and our response / target variable (Satisfaction). It always lies between 0 and 1 (i.e.: a number near 0 represents a regression that does not explain the variance in the response variable and a number close to 1 does explain the observed variance in the response variable).
The R-square we get is 0.2365. Or roughly 23% of the variance found in the response variable (Satisfaction) can be explained by the predictor variable (ProdQual).

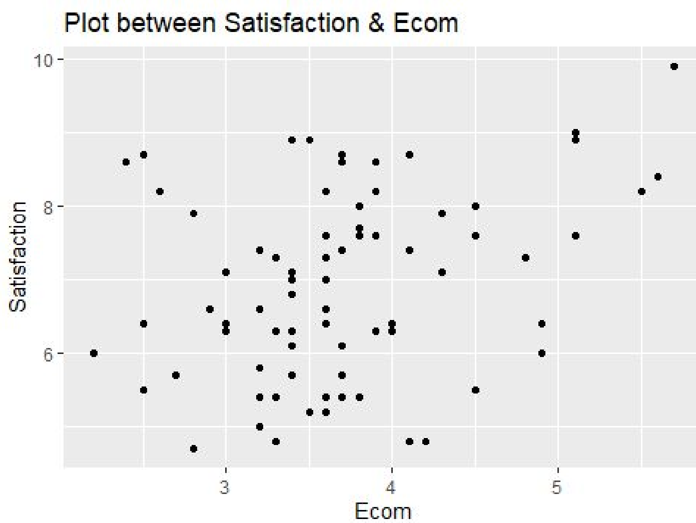
F-Statistic

F-statistic is a good indicator of whether there is a relationship between our predictor and the response variables. The further the F-statistic is from 1 the better it is. Lesser the P-Value the better is model
In our mode; the F-statistic is 30.36 which is relatively larger than 1 given the size of our data.
p-value: 2.901e-07

```
> #SLR Model for ProdQual & Satisfaction
> qplot(myDataSLR$ProdQual,myDataSLR$Satisfaction, main = "Plot between Satisfaction & Product Quality",
+       xlab = "Product Quality", ylab = "Satisfaction")
> cor(myDataSLR$ProdQual,myDataSLR$Satisfaction)
[1] 0.486325
> modSat_PQ =  lm(Satisfaction ~ ProdQual, data = myDataSLR)
> summary(modSat_PQ)

Call:
lm(formula = Satisfaction ~ ProdQual, data = myDataSLR)

Residuals:
     Min       1Q   Median       3Q      Max 
-1.88746 -0.72711 -0.01577  0.85641  2.25220 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  3.67593    0.59765   6.151 1.68e-08 ***
ProdQual     0.41512    0.07534   5.510 2.90e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.047 on 98 degrees of freedom
Multiple R-squared:  0.2365,	Adjusted R-squared:  0.2287 
F-statistic: 30.36 on 1 and 98 DF,  p-value: 2.901e-07
```

4.3.2 Simple linear regression for the Satisfaction with Ecom

Graphical Plot between Satisfaction with Ecom



Correlation between Satisfaction & Ecom is 0.282745

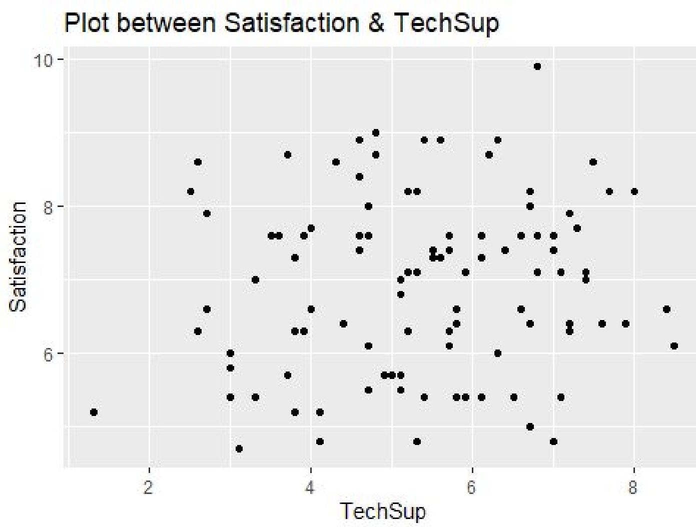
Model: Satisfaction = 5.1516 + 0.4811(Ecom)

Output:

```
> summary(modSat_Ecom)

Call:
lm(formula = Satisfaction ~ Ecom, data = myDataSLR)

Residuals:
     Min       1Q   Median       3Q      Max
-2.37200 -0.78971  0.04959  0.68085  2.34580

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   5.1516     0.6161   8.361 4.28e-13 ***
Ecom          0.4811     0.1649   2.918  0.00437 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.149 on 98 degrees of freedom
Multiple R-squared:  0.07994,	Adjusted R-squared:  0.07056
F-statistic: 8.515 on 1 and 98 DF,  p-value: 0.004368
```

4.3.3 Simple linear regression for the Satisfaction with TechSup

Graphical Plot between Satisfaction with TechSup



Correlation between Satisfaction & TechSup is 0.1125972

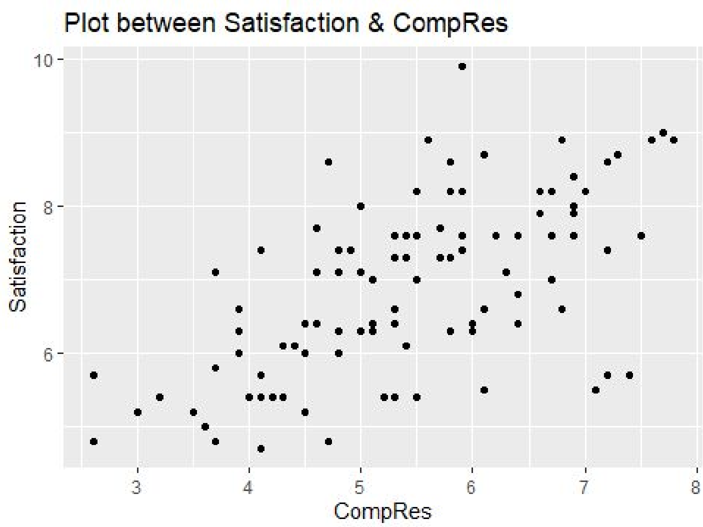
Model: Satisfaction = 6.44757 + 0.08768(TechSup)

Output:

```
> summary(modSat_TechSup)

Call:
lm(formula = Satisfaction ~ TechSup, data = myDataSLR)

Residuals:
     Min       1Q   Median       3Q      Max 
-2.26136 -0.93297  0.04302  0.82501  2.85617 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  6.44757    0.43592  14.791   <2e-16 ***
TechSup      0.08768    0.07817   1.122    0.265    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.19 on 98 degrees of freedom
Multiple R-squared:  0.01268,	Adjusted R-squared:  0.002603 
F-statistic: 1.258 on 1 and 98 DF,  p-value: 0.2647
```

4.3.4 Simple linear regression for the Satisfaction with CompRes

Graphical Plot between Satisfaction with CompRes



Correlation between Satisfaction & CompRes is 0.6032626

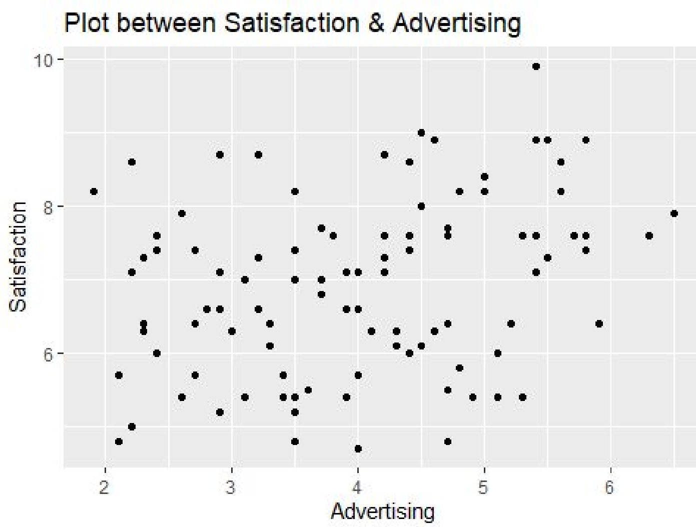
Model: Satisfaction = 3.68005 + 0.59499(CompRes)

Output:

```
> summary(modSat_CompRes)

Call:
lm(formula = Satisfaction ~ CompRes, data = myDataSLR)

Residuals:
     Min       1Q   Median       3Q      Max 
-2.40450 -0.66164  0.04499  0.63037  2.70949 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  3.68005    0.44285   8.310 5.51e-13 ***
CompRes      0.59499    0.07946   7.488 3.09e-11 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.9554 on 98 degrees of freedom
Multiple R-squared:  0.3639,	Adjusted R-squared:  0.3574 
F-statistic: 56.07 on 1 and 98 DF,  p-value: 3.085e-11
```

4.3.5 Simple linear regression for the Satisfaction with Advertising

Graphical Plot between Satisfaction with Advertising



Correlation between Satisfaction & Advertising is 0.3046695

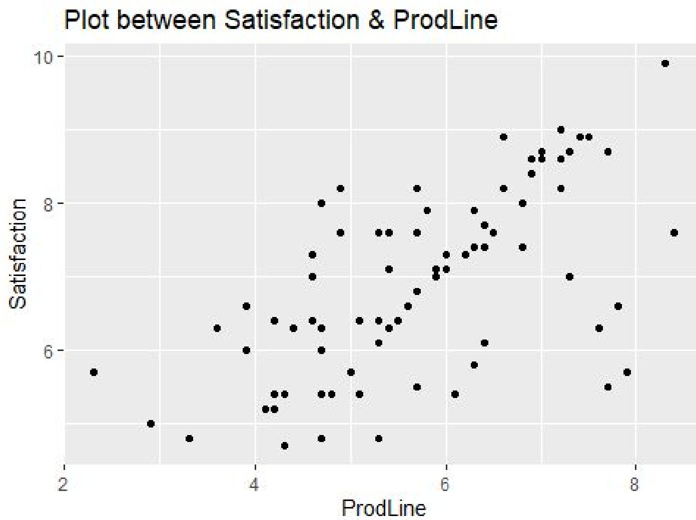
Model: Satisfaction = 5.6259 + 0.3222(Advertising)

Output:

```
> summary(modSat_Advertising)

Call:
lm(formula = Satisfaction ~ Advertising, data = myDataSLR)

Residuals:
     Min       1Q   Median       3Q      Max 
-2.34033 -0.92755  0.05577  0.79773  2.53412 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   5.6259     0.4237  13.279  < 2e-16 ***
Advertising   0.3222     0.1018   3.167  0.00206 ** 
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.141 on 98 degrees of freedom
Multiple R-squared:  0.09282,	Adjusted R-squared:  0.08357 
F-statistic: 10.03 on 1 and 98 DF,  p-value: 0.002056
```

4.3.6 Simple linear regression for the Satisfaction with ProdLine

Graphical Plot between Satisfaction with ProdLine



Correlation between Satisfaction & ProdLine is 0.5505459

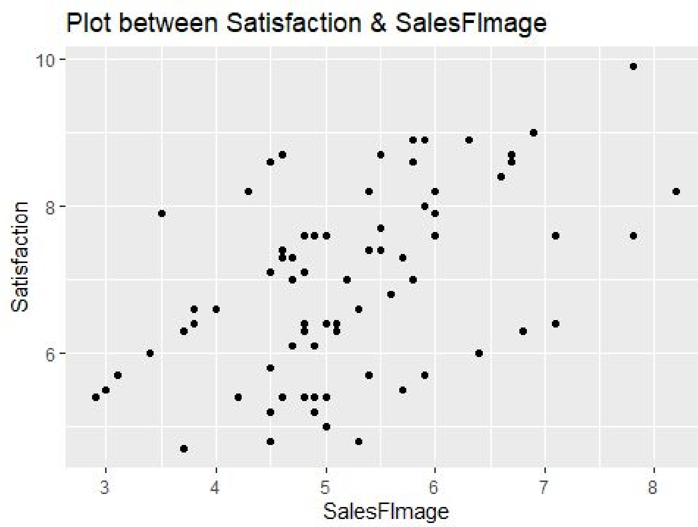
Model: Satisfaction = 4.02203 + 0.49887(ProdLine)

Output:

```
> summary(modSat_ProdLine)

Call:
lm(formula = Satisfaction ~ ProdLine, data = myDataSLR)

Residuals:
    Min      1Q  Median      3Q     Max 
-2.3634 -0.7795  0.1097  0.7604  1.7373 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  4.02203    0.45471   8.845 3.87e-14 ***
ProdLine     0.49887    0.07641   6.529 2.95e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1 on 98 degrees of freedom
Multiple R-squared:  0.3031,	Adjusted R-squared:  0.296 
F-statistic: 42.62 on 1 and 98 DF,  p-value: 2.953e-09
```

4.3.7 Simple linear regression for the Satisfaction with SalesFImage

Graphical Plot between Satisfaction with SalesFImage



Correlation between Satisfaction & SalesFImage is 0.5002053

Model: Satisfaction = 4.06983 + 0.55596(SalesFImage)

Output:

```
> summary(modSat_SalesFImage)

Call:
lm(formula = Satisfaction ~ SalesFImage, data = myDataSLR)

Residuals:
    Min      1Q  Median      3Q     Max
-2.2164 -0.5884  0.1838  0.6922  2.0728

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  4.06983    0.50874   8.000 2.54e-12 ***
SalesFImage  0.55596    0.09722   5.719 1.16e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.037 on 98 degrees of freedom
Multiple R-squared:  0.2502,    Adjusted R-squared:  0.2426
F-statistic:  32.7 on 1 and 98 DF,  p-value: 1.164e-07
```

4.3.8 Simple linear regression for the Satisfaction with ComPricing

Graphical Plot between Satisfaction with ComPricing

Correlation between Satisfaction & ComPricing is -0.2082957

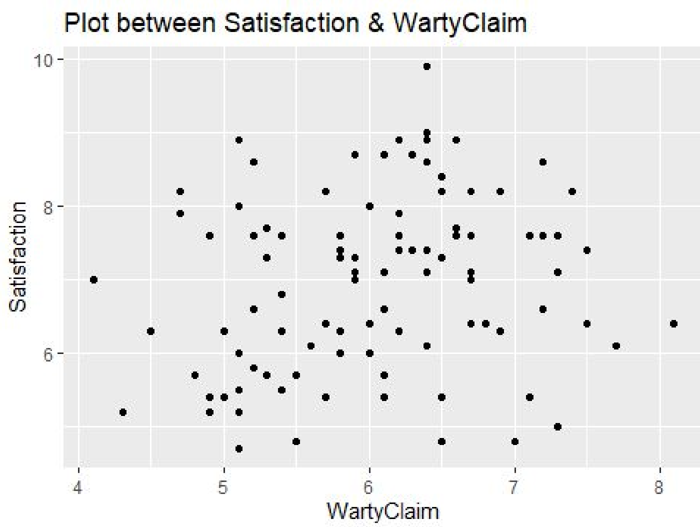
Model: Satisfaction = 8.03856 - 0.16068(ComPricing)

Output:

```
> summary(modSat_ComPricing)

Call:
lm(formula = Satisfaction ~ ComPricing, data = myDataSLR)

Residuals:
    Min      1Q  Median      3Q     Max
-1.9728 -0.9915 -0.1156  0.9111  2.5845

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  8.03856    0.54427  14.769   <2e-16 ***
ComPricing  -0.16068    0.07621  -2.108   0.0376 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.172 on 98 degrees of freedom
Multiple R-squared:  0.04339,    Adjusted R-squared:  0.03363
F-statistic: 4.445 on 1 and 98 DF,  p-value: 0.03756
```

4.3.9 Simple linear regression for the Satisfaction with WartyClaim

Graphical Plot between Satisfaction with WartyClaim



Correlation between Satisfaction & WartyClaim is 0.1775448

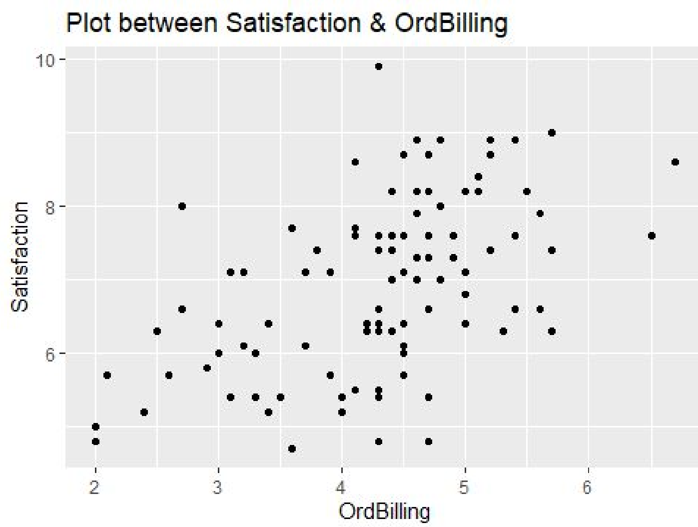
Model: Satisfaction = 5.3581 + 0.2581(WartyClaim)

Output:

```
> summary(modSat_WartyClaim)

Call:
lm(formula = Satisfaction ~ WartyClaim, data = myDataSLR)

Residuals:
     Min       1Q   Median       3Q      Max 
-2.36504 -0.90202  0.03019  0.90763  2.88985 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   5.3581     0.8813   6.079 2.32e-08 ***
WartyClaim    0.2581     0.1445   1.786   0.0772 .  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.179 on 98 degrees of freedom
Multiple R-squared:  0.03152,	Adjusted R-squared:  0.02164 
F-statistic:  3.19 on 1 and 98 DF,  p-value: 0.0772
```

4.3.10 Simple linear regression for the Satisfaction with OrdBilling

Graphical Plot between Satisfaction with OrdBilling



Correlation between Satisfaction & OrdBilling is 0.5217319

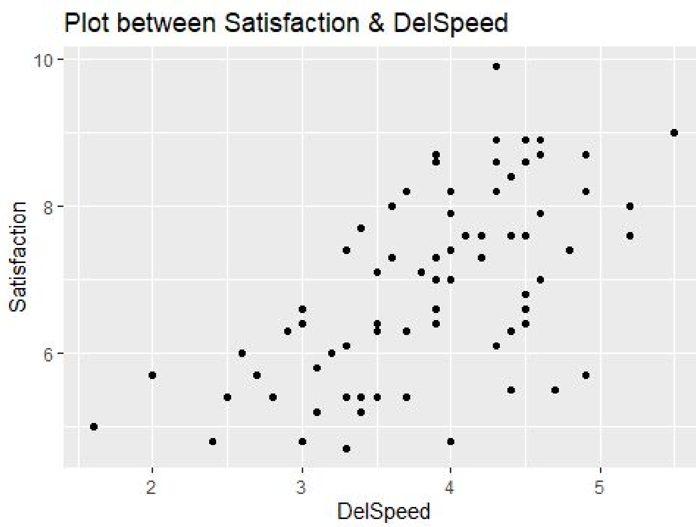
Model: Satisfaction = 4.0541 + 0.6695(OrdBilling)

Output:

```
> summary(modSat_OrdBilling)

Call:
lm(formula = Satisfaction ~ OrdBilling, data = myDataSLR)

Residuals:
    Min      1Q  Median      3Q     Max 
-2.4005 -0.7071 -0.0344  0.7340  2.9673 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   4.0541     0.4840   8.377 3.96e-13 ***
OrdBilling    0.6695     0.1106   6.054 2.60e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.022 on 98 degrees of freedom
Multiple R-squared:  0.2722,	Adjusted R-squared:  0.2648 
F-statistic: 36.65 on 1 and 98 DF,  p-value: 2.602e-08
```

4.3.11 Simple linear regression for the Satisfaction with DelSpeed

Graphical Plot between Satisfaction with DelSpeed



Correlation between Satisfaction & DelSpeed is 0.5770423

Model: Satisfaction = 3.2791 + 0.9364(DelSpeed)

Output:

```
> summary(modSat_DelSpeed)

Call:
lm(formula = Satisfaction ~ DelSpeed, data = myDataSLR)

Residuals:
     Min       1Q   Median       3Q      Max 
-2.22475 -0.54846  0.08796  0.54462  2.59432 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   3.2791     0.5294   6.194 1.38e-08 ***
DelSpeed      0.9364     0.1339   6.994 3.30e-10 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.9783 on 98 degrees of freedom
Multiple R-squared:  0.333,	Adjusted R-squared:  0.3262 
F-statistic: 48.92 on 1 and 98 DF,  p-value: 3.3e-10
```

4.3 Factor analysis & their Interpretation

4.4.1 Factor Analysis and Interpret the Eigen Values using Kaiser Normalization Rule

Although we are aware of the correlation among independent data, we built an initial multiple linear regression model on the data which is as follows

```
> summary(model0)

Call:
lm(formula = Satisfaction ~ ., data = myDataM)

Residuals:
     Min       1Q   Median       3Q      Max 
-1.43005 -0.31165  0.07621  0.37190  0.90120 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) -0.66961    0.81233  -0.824  0.41199    
ProdQual     0.37137    0.05177   7.173 2.18e-10 ***
Ecom        -0.44056    0.13396  -3.289  0.00145 ** 
TechSup      0.03299    0.06372   0.518  0.60591    
CompRes      0.16703    0.10173   1.642  0.10416    
Advertising -0.02602    0.06161  -0.422  0.67382    
ProdLine     0.14034    0.08025   1.749  0.08384 .  
SalesFImage  0.80611    0.09775   8.247 1.45e-12 ***
ComPricing  -0.03853    0.04677  -0.824  0.41235    
WartyClaim  -0.10298    0.12330  -0.835  0.40587    
OrdBilling   0.14635    0.10367   1.412  0.16160    
DelSpeed     0.16570    0.19644   0.844  0.40124    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.5623 on 88 degrees of freedom
Multiple R-squared:  0.8021,	Adjusted R-squared:  0.7774 
F-statistic: 32.43 on 11 and 88 DF,  p-value: < 2.2e-16
```

But when we checked vif(Variance Inflation Factor) it seems to be more than *3 for multiple variable (compRes, ProdLine, SalesFImage, WartyClaim, DelSpeed) Therefore model was rejected we have to move ahead with factor analysis.*
For Factor Analysis we have to test our data with KMO and bartlett Test

Kaiser-Meyer-Olkin factor adequacy
Kaiser-Meyer-Olkin (KMO) Test is a measure of how suited your data is for Factor Analysis. The test measures sampling adequacy for each variable in the model and for the complete model.

The Overall MSA is 0.65 and each variable is more than 0.5 which make our data suitable for Factor Analysis.

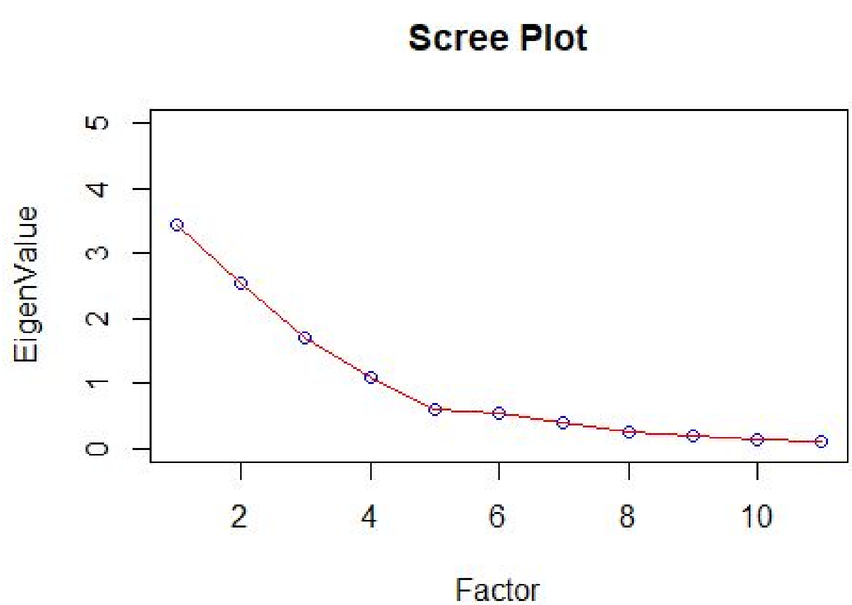
Bartlett test
To test the correlation matrix, Bartlett's test of sphericity is used . This method test is the correlation matrix similar with identity matrix (it means that each variable correlate only with itself).
The hypothesis tested are:
H0 : correlation matrix is an identity matrix (the factor analysis is inappropriate)
H1 : correlation matrix is not an identity matrix (the factor analysis is appropriate)

p.value 6.078303e-34 indicates the data correlation is not an identity matrix we can move ahead with factor analysis

EigenValue
In every factor analysis, there are the same number of factors as there are variables. Each factor captures a certain amount of the overall variance in the observed variables, and the factors are always listed in order of how much variation they explain.
The eigenvalue is a measure of how much of the variance of the observed variables a factor explains. Any factor with an eigenvalue ≥1 explains more variance than a single observed variable

We have 4 Factors having Eigenvalue more than therefore 4 factors will be good to explain most of the variance.

*In **Scree Plot** the point where the slope of the curve is clearly leveling off (the "elbow) indicates the number of factors that should be generated by the analysis is also 4.*



4.4.2 Interpretation and Naming of factors Generated

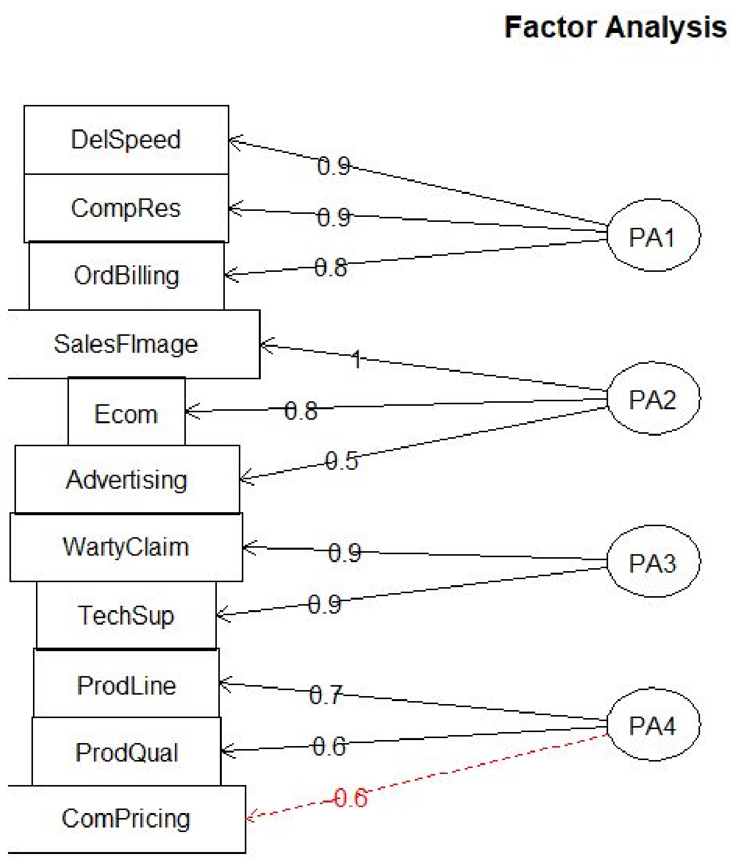
Factor Analysis
Using Varimax rotation we have done the factor Analysis which covers the 69% variance.

	PA1	PA2	PA3	PA4
ProdQual	0.02	-0.07	0.02	0.65
Ecom	0.07	0.79	0.03	-0.11
TechSup	0.02	-0.03	0.88	0.12
CompRes	0.90	0.13	0.05	0.13
Advertising	0.17	0.53	-0.04	-0.06
ProdLine	0.53	-0.04	0.13	0.71
SalesFImage	0.12	0.97	0.06	-0.13
ComPricing	-0.08	0.21	-0.21	-0.59
WartyClaim	0.10	0.06	0.89	0.13
OrdBilling	0.77	0.13	0.09	0.09
DelSpeed	0.95	0.19	0.00	0.09

Now we considered the loadings more than 0.5 and not loading on more than one factor. Negative values are acceptable here.

Here is how the factors can be interpreted:

PA1 (Product Delivery & Support)	PA2 (Availability & Marketing)	PA3 (After Sale Service)	PA4 (Quality & Pricing)
Complaint Resolution	E-Commerce	Technical Support	Product Quality
Order & Billing	Advertising	Warranty & Claims	Product Line
Delivery Speed	Salesforce Image		Competitive Pricing



Adequacy Test

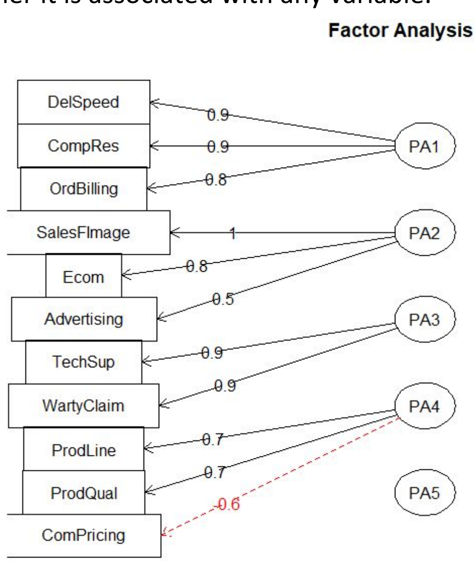
The root mean square of residuals (RMSR) is 0.02. This is acceptable as this value should be closer to 0. Next we should check RMSEA (root mean square error of approximation) index. Its value, 0.096 shows good model fit as the 90 % confidence intervals are 0.032 - 0.139. Finally, the Tucker-Lewis Index (TLI) is 0.921 - an acceptable value considering it's over 0.9.

Here is the Output

```
Factor Analysis using method =  pa
Call: fa(r = FData, nfactors = 4, rotate = "varimax", fm = "pa")
Standardized loadings (pattern matrix) based upon correlation matrix
              PA1   PA2   PA3   PA4   h2    u2 com
ProdQual     0.02 -0.07  0.02  0.65 0.42 0.576 1.0
Ecom         0.07  0.79  0.03 -0.11 0.64 0.362 1.1
TechSup      0.02 -0.03  0.88  0.12 0.79 0.205 1.0
CompRes      0.90  0.13  0.05  0.13 0.84 0.157 1.1
Advertising  0.17  0.53 -0.04 -0.06 0.31 0.686 1.2
ProdLine     0.53 -0.04  0.13  0.71 0.80 0.200 1.9
SalesFImage  0.12  0.97  0.06 -0.13 0.98 0.021 1.1
ComPricing  -0.08  0.21 -0.21 -0.59 0.44 0.557 1.6
WartyClaim   0.10  0.06  0.89  0.13 0.81 0.186 1.1
OrdBilling   0.77  0.13  0.09  0.09 0.62 0.378 1.1
```

```
DelSpeed     0.95  0.19  0.00  0.09 0.94 0.058 1.1

                       PA1  PA2  PA3  PA4
SS loadings           2.63 1.97 1.64 1.37
Proportion Var        0.24 0.18 0.15 0.12
Cumulative Var        0.24 0.42 0.57 0.69
Proportion Explained  0.35 0.26 0.22 0.18
Cumulative Proportion 0.35 0.60 0.82 1.00

Mean item complexity =  1.2
Test of the hypothesis that 4 factors are sufficient.

The degrees of freedom for the null model are  55  and the objective function was  6.55 with Chi Square
of  619.27
The degrees of freedom for the model are 17  and the objective function was  0.33

The root mean square of the residuals (RMSR) is  0.02
The df corrected root mean square of the residuals is  0.03

The harmonic number of observations is  100 with the empirical chi square  3.19  with prob <  1
The total number of observations was  100  with Likelihood Chi Square =  30.27  with prob <  0.024

Tucker Lewis Index of factoring reliability =  0.921
RMSEA index =  0.096  and the 90 % confidence intervals are  0.032 0.139
BIC =  -48.01
Fit based upon off diagonal values = 1
Measures of factor score adequacy
                                                 PA1  PA2  PA3  PA4
Correlation of (regression) scores with factors   0.98 0.99 0.94 0.88
Multiple R square of scores with factors          0.96 0.97 0.88 0.78
Minimum correlation of possible factor scores     0.93 0.94 0.77 0.55
```

So, to check the change we also created 5 factors whose is as follows
Cumulative Variance is 72% which don't provide any significant change from 4 factors. therefore 4 factors are good enough. Neither it is associated with any variable.



Code for factor analysis

```
> ##Initial Model
> model0 = lm(Satisfaction~., myDataM)
> summary(model0)

Call:
lm(formula = Satisfaction ~ ., data = myDataM)

Residuals:
     Min       1Q   Median       3Q      Max 
-1.43005 -0.31165  0.07621  0.37190  0.90120 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) -0.66961    0.81233  -0.824  0.41199    
ProdQual     0.37137    0.05177   7.173 2.18e-10 ***
Ecom        -0.44056    0.13396  -3.289  0.00145 ** 
TechSup      0.03299    0.06372   0.518  0.60591    
CompRes      0.16703    0.10173   1.642  0.10416    
Advertising -0.02602    0.06161  -0.422  0.67382    
ProdLine     0.14034    0.08025   1.749  0.08384 .  
SalesFImage  0.80611    0.09775   8.247 1.45e-12 ***
ComPricing  -0.03853    0.04677  -0.824  0.41235    
WartyClaim  -0.10298    0.12330  -0.835  0.40587    
OrdBilling   0.14635    0.10367   1.412  0.16160    
DelSpeed     0.16570    0.19644   0.844  0.40124    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.5623 on 88 degrees of freedom
Multiple R-squared:  0.8021,	Adjusted R-squared:  0.7774 
F-statistic: 32.43 on 11 and 88 DF,  p-value: < 2.2e-16

> vif(model0)
   ProdQual        Ecom     TechSup     CompRes Advertising    ProdLine SalesFImage  ComPricing 
   1.635797    2.756694    2.976796    4.730448    1.508933    3.488185    3.439420    1.635000 
 WartyClaim  OrdBilling    DelSpeed 
   3.198337    2.902999    6.516014 
> #Factor Analysis
> FData <- subset(myDataM, select = -c(12)) #Taking a subset of independent variables
> names(FData)
 [1] "ProdQual"    "Ecom"        "TechSup"     "CompRes"     "Advertising" "ProdLine"   
 [7] "SalesFImage" "ComPricing"  "WartyClaim"  "OrdBilling"  "DelSpeed"   
> datamatrix<-cor(FData)
> KMO(r=datamatrix) #MSA should be greater than 0.5
Kaiser-Meyer-Olkin factor adequacy
Call: KMO(r = datamatrix)
Overall MSA =  0.65
MSA for each item = 
   ProdQual        Ecom     TechSup     CompRes Advertising    ProdLine SalesFImage  ComPricing 
       0.51        0.63        0.52        0.79        0.78        0.62        0.62        0.75 
 WartyClaim  OrdBilling    DelSpeed 
       0.51        0.76        0.67 
> cortest.bartlett(datamatrix, n = 50) ### n is a sample size
$chisq
[1] 291.6151
```

```
$p.value
[1] 6.078303e-34

$df
[1] 55

> ev = eigen(datamatrix)
> EigenValue=ev$values
> EigenValue
 [1] 3.42697133 2.55089671 1.69097648 1.08655606 0.60942409 0.55188378 0.40151815 0.24695154
 [9] 0.20355327 0.13284158 0.09842702
> Factor=c(1:11)
> Scree=data.frame(Factor,EigenValue)
> plot(Scree,main="Scree Plot", col="Blue",ylim=c(0,5))
> lines(Scree,col="Red")
> #FA
> fa1<- fa(r=FData, nfactors = 4, rotate="varimax",fm="pa")
> print(fa1)
Factor Analysis using method =  pa
Call: fa(r = FData, nfactors = 4, rotate = "varimax", fm = "pa")
Standardized loadings (pattern matrix) based upon correlation matrix
              PA1   PA2   PA3   PA4   h2    u2 com
ProdQual     0.02 -0.07  0.02  0.65 0.42 0.576 1.0
Ecom         0.07  0.79  0.03 -0.11 0.64 0.362 1.1
TechSup      0.02 -0.03  0.88  0.12 0.79 0.205 1.0
CompRes      0.90  0.13  0.05  0.13 0.84 0.157 1.1
Advertising  0.17  0.53 -0.04 -0.06 0.31 0.686 1.2
ProdLine     0.53 -0.04  0.13  0.71 0.80 0.200 1.9
SalesFImage  0.12  0.97  0.06 -0.13 0.98 0.021 1.1
ComPricing  -0.08  0.21 -0.21 -0.59 0.44 0.557 1.6
WartyClaim   0.10  0.06  0.89  0.13 0.81 0.186 1.1
OrdBilling   0.77  0.13  0.09  0.09 0.62 0.378 1.1
DelSpeed     0.95  0.19  0.00  0.09 0.94 0.058 1.1

                       PA1  PA2  PA3  PA4
SS loadings           2.63 1.97 1.64 1.37
Proportion Var        0.24 0.18 0.15 0.12
Cumulative Var        0.24 0.42 0.57 0.69
Proportion Explained  0.35 0.26 0.22 0.18
Cumulative Proportion 0.35 0.60 0.82 1.00

Mean item complexity =  1.2
Test of the hypothesis that 4 factors are sufficient.

The degrees of freedom for the null model are  55  and the objective function was  6.55 with Chi Square
of  619.27
The degrees of freedom for the model are 17  and the objective function was  0.33

The root mean square of the residuals (RMSR) is  0.02
The df corrected root mean square of the residuals is  0.03

The harmonic number of observations is  100 with the empirical chi square  3.19  with prob <  1
The total number of observations was  100  with Likelihood Chi Square =  30.27  with prob <  0.024

Tucker Lewis Index of factoring reliability =  0.921
```

```
RMSEA index =  0.096  and the 90 % confidence intervals are  0.032 0.139
BIC =  -48.01
Fit based upon off diagonal values = 1
Measures of factor score adequacy
                                                   PA1  PA2  PA3  PA4
Correlation of (regression) scores with factors   0.98 0.99 0.94 0.88
Multiple R square of scores with factors          0.96 0.97 0.88 0.78
Minimum correlation of possible factor scores     0.93 0.94 0.77 0.55
> fa.diagram(fa1)
> plot(fa1$values, type="b")
```
Please refer to Appendix A for Source Code.

4.5 Multiple linear regression with customer satisfaction & factors Generated

4.5.1 Making of Data frame with 4 factors and "Satisfaction"

A new Dataframe is created using fa$scores & "Satisfaction" variable

```
> #Combining the factors in the data for regression analysis
> regdata <- cbind(myDataM[12], fa1$scores)
> head(regdata)
  Satisfaction        PA1        PA2          PA3         PA4
1          8.2 -0.1338871  0.9175166 -1.719604873  0.09135411
2          5.7  1.6297604 -2.0090053 -0.596361722  0.65808192
3          8.9  0.3637658  0.8361736  0.002979966  1.37548765
4          4.8 -1.2225230 -0.5491336  1.245473305 -0.64421384
5          7.1 -0.4854209 -0.4276223 -0.026980304  0.47360747
6          4.7 -0.5950924 -1.3035333 -1.183019401 -0.95913571
```

4.5.2 Model generation using Multiple Linear Regression

Creation of Train and Test Data

Train data (70% of Data) is used to build the model and the rest (30% of Data) is used to Validate the model.

Model Building

Model 1

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  6.91194    0.08216  84.131  < 2e-16 ***
PA1          0.66493    0.08396   7.919 4.06e-11 ***
PA2          0.40322    0.09817   4.107 0.000114 ***
PA3          0.05730    0.08822   0.649 0.518304    
PA4          0.60209    0.09709   6.201 4.35e-08 ***
```

It can be seen that, change in PA1, PA2 & PA4 are significantly associated with changes in Satisfaction while changes in PA3 is not significantly associated with Satisfaction.

We found that PA3 is not significant in the multiple regression model. This means that, for a fixed PA1, PA2 & PA4, changes in the PA3 will not significantly affect Satisfaction.

As the PA3 is not significant, it is possible to remove it from the model

In model2 we used 3 factors PA1,PA2 & PA4

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  6.92156    0.08046  86.028  < 2e-16 ***
PA1          0.66743    0.08351   7.993 2.73e-11 ***
PA2          0.40527    0.09769   4.149 9.77e-05 ***
PA4          0.60744    0.09631   6.307 2.71e-08 ***
```

Finally, our model equation can be written as follows:
Satisfaction = 6.92156 + 0.66743(PA1) + 0.40527(PA2) + 0.60744(PA4)

Coefficient-Std. Error
The standard deviation of an estimate is called the standard error. The standard error of the coefficient measures how precisely the model estimates the coefficient's unknown value. The standard error of the coefficient is always positive

Coefficient Pr(>|t|)
Individual p value for each parameter to accept or reject the null hypothesis, this is a statistical estimate of x and y. Lower the p value allow us to reject the null hypothesis. *All values are less than 0.05*

The confidence interval of the model coefficient can be extracted as follows:

```
> confint(model2)
                2.5 %    97.5 %
(Intercept) 6.7609225 7.0821982
PA1         0.5007069 0.8341568
PA2         0.2102286 0.6003181
PA4         0.4151466 0.7997324
```

Model accuracy assessment

R-squared:
In multiple linear regression, the R2 represents the correlation coefficient between the observed values of the outcome variable (y) and the fitted (i.e., predicted) values of y.
R2 represents the proportion of variance in the outcome variable y, that may be predicted by knowing the value of the x variables. An R2 value close to 1 indicates that the model explains a large portion of the variance in the outcome variable.
In our example, with PA1, PA2 & PA4 variables, *R2 is 0.63 that means the correlation coefficient between the satisfaction and 3 factors is 0.63*

Adjusted R Square:
The adjustment in the "Adjusted R Square" value in the summary output is a correction for the number of x variables included in the prediction model. *So 0.6116 i.e 61% of the variance in the measure of satisfaction can be predicted by these 3 variable.*

Residual Standard Error (RSE), or sigma:

The RSE estimate gives a measure of error of prediction. The lower the RSE, the more accurate the model (on the data in hand).

In our multiple regression case, the RSE is 0.6623 corresponding to 9.7% error rate.

```
error_rate = sigma(model2)/mean(train$Satisfaction)
```

F-statistic

F-statistic: 28.16 on 4 and 65 DF, p-value: 1.39e-13

This is showing the relationship between predictor and response, the higher the value will give more reasons to reject the null hypothesis, its significant overall model not any specific parameter

p-value

p-value: 1.39e-13

Overall p value on the basis of F-statistic, normally p value less than 0.05 indicate that overall model is significant.

Degree of Freedom:

Degree of freedom is like no of data point taken in consideration for estimation taking parameter in account, in this case, *we total have 70 data point and 3 variable so removed 4 data points (70–4) = 66 degrees of freedom*

Output:

```
> model1 = lm(Satisfaction~., train)
> summary(model1)

Call:
lm(formula = Satisfaction ~ ., data = train)

Residuals:
     Min       1Q   Median       3Q      Max 
-1.69261 -0.47602  0.09094  0.48715  1.12820 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  6.91194    0.08216  84.131  < 2e-16 ***
PA1          0.66493    0.08396   7.919 4.06e-11 ***
PA2          0.40322    0.09817   4.107 0.000114 ***
PA3          0.05730    0.08822   0.649 0.518304    
PA4          0.60209    0.09709   6.201 4.35e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.6652 on 65 degrees of freedom
Multiple R-squared:  0.6341,	Adjusted R-squared:  0.6116 
F-statistic: 28.16 on 4 and 65 DF,  p-value: 1.39e-13
```

```
> vif(model1)
     PA1      PA2      PA3      PA4 
1.003032 1.021759 1.009641 1.028411 
> model2 = lm(Satisfaction~PA1 + PA2 + PA4, train)
> summary(model2)

Call:
lm(formula = Satisfaction ~ PA1 + PA2 + PA4, data = train)

Residuals:
     Min       1Q   Median       3Q      Max 
-1.67785 -0.45521  0.09673  0.52682  1.09648 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  6.92156    0.08046  86.028  < 2e-16 ***
PA1          0.66743    0.08351   7.993 2.73e-11 ***
PA2          0.40527    0.09769   4.149 9.77e-05 ***
PA4          0.60744    0.09631   6.307 2.71e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.6623 on 66 degrees of freedom
Multiple R-squared:  0.6317,	Adjusted R-squared:  0.615 
F-statistic: 37.74 on 3 and 66 DF,  p-value: 2.532e-14

> vif(model2)
     PA1      PA2      PA4 
1.000927 1.020696 1.020996 
```

4.5.3 Model Testing

We Test the model on Test data (30% of Data) and the results are as follows

R Square Value : 0.6815526

Adjusted R square Values : 0.6448086

Which are in 10% range of our model so our model work perfectly on Test Data.

RMSE is 0.7974715

RMSE is an absolute measure of fit. As the square root of a variance, RMSE can be interpreted as the standard deviation of the unexplained variance, and has the useful property of being in the same units as the response variable.

With the Test data we can prove our model is working fine.

```
> pred=predict(model2, newdata = test, type = "response")
> pred
       1        2        3        6       12       18       20       21       22       36       40
7.259537 7.594863 8.338754 5.413472 5.949228 7.482209 7.809208 5.559574 9.215931 5.827328 6.342061
      43       44       45       49       52       57       62       63       66       67       70
8.147853 7.760535 7.492218 8.266969 7.714021 8.575765 6.663641 6.468533 7.512508 6.779930 6.324929
      73       75       79       92       94       96       97      100
7.679019 7.011674 8.593262 4.659920 8.603187 7.247490 6.351660 5.910752
> test$Satisfaction.Predict <- pred
> #Let's check the predictions.
> head(test[c(1,6)],10)
   Satisfaction Satisfaction.Predict
1           8.2             7.259537
2           5.7             7.594863
3           8.9             8.338754
6           4.7             5.413472
12          6.0             5.949228
18          7.4             7.482209
20          7.6             7.809208
21          5.4             5.559574
22          9.9             9.215931
36          5.4             5.827328
> SSE_val <- sum((test$Satisfaction - pred) ^ 2)
> SST_val <- sum((test$Satisfaction - mean(test$Satisfaction)) ^ 2)
> SSR_val=SST_val-SSE_val
> RSquare_val<-SSR_val/SST_val
> RSquare_val
[1] 0.6815526
> Term1<- (1-RSquare_val)
> Term2<- (count(as.data.frame(pred))-1)/(count(as.data.frame(pred))-3-1)
> AdjustedRSquare_val <- 1-(Term1*Term2)
> AdjustedRSquare_val
          n
1 0.6448086
> library(Metrics)
> rmse(test$Satisfaction,pred)
[1] 0.7974715
```

4.5.5 Model Interpretation

Final Model Equation

$$Satisfaction = 6.92156 + 0.66743(PA1) + 0.40527(PA2) + 0.60744(PA4)$$

Where:

- PA1 (Product Delivery & Support)
- PA2 (Availability & Marketing)
- PA4 (Quality & Pricing)

PA1 (Product Delivery & Support)	**PA2 (Availability & Marketing)**	**PA3 (After Sale Service)**	**PA4 (Quality & Pricing)**
Complaint Resolution	E-Commerce	Technical Support	Product Quality
Order & Billing	Advertising	Warranty & Claims	Product Line
Delivery Speed	Salesforce Image		Competitive Pricing

So, With every unit increase in “Product Delivery & Support” keeping all other factors constant Satisfaction will increase by 0.66743
with every unit increase in “Availability & Marketing” keeping all other factors constant Satisfaction will increase by 0.40527
with every unit increase in “Quality & Pricing” keeping all other factors constant Satisfaction will increase by 0.60744

The confidence interval of the model coefficient can be extracted as follows:

```
> confint(model2)
                2.5 %    97.5 %
(Intercept) 6.7609225 7.0821982
PA1         0.5007069 0.8341568
PA2         0.2102286 0.6003181
PA4         0.4151466 0.7997324
```

The range of variable can be in above mentioned range for 95% confidence

RSquare = 0.63, means the correlation coefficient between the satisfaction and 3 factors is 0.63
Adjusted R Square = 0.6116 i.e 61% of the variance in the measure of satisfaction can be predicted by these 3 variable.
We have 66 degrees of freedom.

5 Suggestion

Here are some suggestions

- Since the No. of datapoint was just 100 therefore more data points will help to build a better model.
- In depth explanation of variable can be more helpful for naming the factor.

6 Appendix A – Source Code

```
> #libraries Required
> library(tidyverse)
-- Attaching packages --------------------------------------- tidyverse 1.2.1 --
v ggplot2 3.2.1     v purrr   0.3.2
v tibble  2.1.3     v dplyr   0.8.3
v tidyr   0.8.3     v stringr 1.4.0
v readr   1.3.1     v forcats 0.4.0
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
Warning messages:
1: package 'tidyverse' was built under R version 3.6.1
2: package 'ggplot2' was built under R version 3.6.1
3: package 'tibble' was built under R version 3.6.1
4: package 'tidyr' was built under R version 3.6.1
5: package 'readr' was built under R version 3.6.1
6: package 'purrr' was built under R version 3.6.1
7: package 'dplyr' was built under R version 3.6.1
8: package 'stringr' was built under R version 3.6.1
9: package 'forcats' was built under R version 3.6.1
> library(dplyr)
> library(ggplot2)
> library(DataExplorer)
Warning message:
package 'DataExplorer' was built under R version 3.6.1
> library(corrplot)
corrplot 0.84 loaded
Warning message:
package 'corrplot' was built under R version 3.6.1
> library(car)
Loading required package: carData

Attaching package: 'car'

The following object is masked from 'package:dplyr':

    recode

The following object is masked from 'package:purrr':

    some

Warning message:
package 'car' was built under R version 3.6.1
> library(Metrics)
Warning message:
package 'Metrics' was built under R version 3.6.1
> library(GPArotation)
> library(MASS)

Attaching package: 'MASS'

The following object is masked from 'package:dplyr':
```

```
    select

Warning message:
package 'MASS' was built under R version 3.6.1
> library(psych)

Attaching package: 'psych'

The following object is masked from 'package:car':

    logit

The following objects are masked from 'package:ggplot2':

    %+%, alpha

> #Setting the Working Directory
> setwd("E:/000GL/000 0Projects/003 Factor Hair Revised")
> getwd()
[1] "E:/000GL/000 0Projects/003 Factor Hair Revised"
> # Importing Data
> myData = read.csv("Factor-Hair-Revised.csv")
> myData$ID = as.factor(myData$ID)
> # General Analysis
> #Variable Identification
> ##Check the Class of Data
> class(myData)
[1] "data.frame"
> ## First Inspection of Dataset using str
> str(myData)
'data.frame':	100 obs. of  13 variables:
 $ ID          : Factor w/ 100 levels "1","2","3","4",..: 1 2 3 4 5 6 7 8 9 10 ...
 $ ProdQual    : num  8.5 8.2 9.2 6.4 9 6.5 6.9 6.2 5.8 6.4 ...
 $ Ecom        : num  3.9 2.7 3.4 3.3 3.4 2.8 3.7 3.3 3.6 4.5 ...
 $ TechSup     : num  2.5 5.1 5.6 7 5.2 3.1 5 3.9 5.1 5.1 ...
 $ CompRes     : num  5.9 7.2 5.6 3.7 4.6 4.1 2.6 4.8 6.7 6.1 ...
 $ Advertising : num  4.8 3.4 5.4 4.7 2.2 4 2.1 4.6 3.7 4.7 ...
 $ ProdLine    : num  4.9 7.9 7.4 4.7 6 4.3 2.3 3.6 5.9 5.7 ...
 $ SalesFImage : num  6 3.1 5.8 4.5 4.5 3.7 5.4 5.1 5.8 5.7 ...
 $ ComPricing  : num  6.8 5.3 4.5 8.8 6.8 8.5 8.9 6.9 9.3 8.4 ...
 $ WartyClaim  : num  4.7 5.5 6.2 7 6.1 5.1 4.8 5.4 5.9 5.4 ...
 $ OrdBilling  : num  5 3.9 5.4 4.3 4.5 3.6 2.1 4.3 4.4 4.1 ...
 $ DelSpeed    : num  3.7 4.9 4.5 3 3.5 3.3 2 3.7 4.6 4.4 ...
 $ Satisfaction: num  8.2 5.7 8.9 4.8 7.1 4.7 5.7 6.3 7 5.5 ...
> ## Find the name of variable
> names(myData)
 [1] "ID"           "ProdQual"     "Ecom"         "TechSup"      "CompRes"      "Advertising"
 [7] "ProdLine"     "SalesFImage"  "ComPricing"   "WartyClaim"   "OrdBilling"   "DelSpeed"
[13] "Satisfaction"
> ## find the dimension of Data
> dim(myData)
[1] 100  13
> ## find first 6 elements of Data
> head(myData)
  ID ProdQual Ecom TechSup CompRes Advertising ProdLine SalesFImage ComPricing WartyClaim
```

```
1  1      8.5  3.9     2.5     5.9         4.8      4.9         6.0        6.8        4.7
2  2      8.2  2.7     5.1     7.2         3.4      7.9         3.1        5.3        5.5
3  3      9.2  3.4     5.6     5.6         5.4      7.4         5.8        4.5        6.2
4  4      6.4  3.3     7.0     3.7         4.7      4.7         4.5        8.8        7.0
5  5      9.0  3.4     5.2     4.6         2.2      6.0         4.5        6.8        6.1
6  6      6.5  2.8     3.1     4.1         4.0      4.3         3.7        8.5        5.1
  OrdBilling DelSpeed Satisfaction
1        5.0      3.7          8.2
2        3.9      4.9          5.7
3        5.4      4.5          8.9
4        4.3      3.0          4.8
5        4.5      3.5          7.1
6        3.6      3.3          4.7
> ## find last 5 elements of Data
> tail(myData)
     ID ProdQual Ecom TechSup CompRes Advertising ProdLine SalesFImage ComPricing WartyClaim
95   95      9.3  3.8     4.0     4.6         4.7      6.4         5.5        7.4        5.3
96   96      8.6  4.8     5.6     5.3         2.3      6.0         5.7        6.7        5.8
97   97      7.4  3.4     2.6     5.0         4.1      4.4         4.8        7.2        4.5
98   98      8.7  3.2     3.3     3.2         3.1      6.1         2.9        5.6        5.0
99   99      7.8  4.9     5.8     5.3         5.2      5.3         7.1        7.9        6.0
100 100      7.9  3.0     4.4     5.1         5.9      4.2         4.8        9.7        5.7
    OrdBilling DelSpeed Satisfaction
95         3.6      3.4          7.7
96         4.9      3.6          7.3
97         4.2      3.7          6.3
98         3.1      2.5          5.4
99         4.3      3.9          6.4
100        3.4      3.5          6.4
> ## find summary of myData to get Min,median,Mean and Max with First and 3rd quartile.
> summary(myData)
       ID         ProdQual            Ecom           TechSup          CompRes        Advertising
 1      : 1   Min.   : 5.000   Min.   :2.200   Min.   :1.300   Min.   :2.600   Min.   :1.900
 2      : 1   1st Qu.: 6.575   1st Qu.:3.275   1st Qu.:4.250   1st Qu.:4.600   1st Qu.:3.175
 3      : 1   Median : 8.000   Median :3.600   Median :5.400   Median :5.450   Median :4.000
 4      : 1   Mean   : 7.810   Mean   :3.672   Mean   :5.365   Mean   :5.442   Mean   :4.010
 5      : 1   3rd Qu.: 9.100   3rd Qu.:3.925   3rd Qu.:6.625   3rd Qu.:6.325   3rd Qu.:4.800
 6      : 1   Max.   :10.000   Max.   :5.700   Max.   :8.500   Max.   :7.800   Max.   :6.500
 (Other):94
    ProdLine       SalesFImage       ComPricing       WartyClaim       OrdBilling        DelSpeed
 Min.   :2.300   Min.   :2.900   Min.   :3.700   Min.   :4.100   Min.   :2.000   Min.   :1.600
 1st Qu.:4.700   1st Qu.:4.500   1st Qu.:5.875   1st Qu.:5.400   1st Qu.:3.700   1st Qu.:3.400
 Median :5.750   Median :4.900   Median :7.100   Median :6.100   Median :4.400   Median :3.900
 Mean   :5.805   Mean   :5.123   Mean   :6.974   Mean   :6.043   Mean   :4.278   Mean   :3.886
 3rd Qu.:6.800   3rd Qu.:5.800   3rd Qu.:8.400   3rd Qu.:6.600   3rd Qu.:4.800   3rd Qu.:4.425
 Max.   :8.400   Max.   :8.200   Max.   :9.900   Max.   :8.100   Max.   :6.700   Max.   :5.500

  Satisfaction
 Min.   :4.700
 1st Qu.:6.000
 Median :7.050
 Mean   :6.918
 3rd Qu.:7.625
 Max.   :9.900

> ## plot the missing value
```

```
> plot_missing(myData)
> ##removig id from data
> #Removing id variable
> myDataM <- subset(myData, select = -c(1))
> dim(myDataM)
[1] 100  12
> ###Summary
> summary(myDataM)
    ProdQual            Ecom           TechSup          CompRes       Advertising        ProdLine
 Min.   : 5.000   Min.   :2.200   Min.   :1.300   Min.   :2.600   Min.   :1.900   Min.   :2.300
 1st Qu.: 6.575   1st Qu.:3.275   1st Qu.:4.250   1st Qu.:4.600   1st Qu.:3.175   1st Qu.:4.700
 Median : 8.000   Median :3.600   Median :5.400   Median :5.450   Median :4.000   Median :5.750
 Mean   : 7.810   Mean   :3.672   Mean   :5.365   Mean   :5.442   Mean   :4.010   Mean   :5.805
 3rd Qu.: 9.100   3rd Qu.:3.925   3rd Qu.:6.625   3rd Qu.:6.325   3rd Qu.:4.800   3rd Qu.:6.800
 Max.   :10.000   Max.   :5.700   Max.   :8.500   Max.   :7.800   Max.   :6.500   Max.   :8.400
  SalesFImage      ComPricing      WartyClaim      OrdBilling       DelSpeed      Satisfaction
 Min.   :2.900   Min.   :3.700   Min.   :4.100   Min.   :2.000   Min.   :1.600   Min.   :4.700
 1st Qu.:4.500   1st Qu.:5.875   1st Qu.:5.400   1st Qu.:3.700   1st Qu.:3.400   1st Qu.:6.000
 Median :4.900   Median :7.100   Median :6.100   Median :4.400   Median :3.900   Median :7.050
 Mean   :5.123   Mean   :6.974   Mean   :6.043   Mean   :4.278   Mean   :3.886   Mean   :6.918
 3rd Qu.:5.800   3rd Qu.:8.400   3rd Qu.:6.600   3rd Qu.:4.800   3rd Qu.:4.425   3rd Qu.:7.625
 Max.   :8.200   Max.   :9.900   Max.   :8.100   Max.   :6.700   Max.   :5.500   Max.   :9.900
> ##Histogram
> plot_histogram(myData,nrow = 4,ncol = 4)
> ##Box Plot
> par(mar=c(4,10,4,4))
> boxplot(myDataM,
+         horizontal = TRUE
+         ,las =2
+         ,main = "Box Plot"
+         ,col = "orange"
+         ,border = "brown")
> ##Calculate SD of Each Data
> myDataM %>%
+   summarise_each(funs(sd(., na.rm=TRUE)))
  ProdQual      Ecom  TechSup  CompRes Advertising ProdLine SalesFImage ComPricing WartyClaim
1 1.396279 0.7005164 1.530457 1.208403    1.126943 1.315285     1.07232   1.545055  0.8197382
  OrdBilling  DelSpeed Satisfaction
1  0.9288398 0.7344372     1.191839
Warning message:
funs() is soft deprecated as of dplyr 0.8.0
Please use a list of either functions or lambdas:

  # Simple named list:
  list(mean = mean, median = median)

  # Auto named with `tibble::lst()`:
  tibble::lst(mean, median)

  # Using lambdas
  list(~ mean(., trim = .2), ~ median(., na.rm = TRUE))
This warning is displayed once per session.
> ##Scatterplot
> plot(myDataM, col="blue", cex.axis=0.75,cex.lab=5, pch=20)
> ##Corelation
> Data_cor <- cor(myDataM)
```

```
> cex.before <- par("cex")
> par(cex = 0.6)
> corrplot(Data_cor)
> ##Corelation
> corrplot(Data_cor, method = "number", number.digits = 2)
> par(cex = cex.before)
> ##Variables having high corelation of more than 0.6 or -0.6
> for (i in 1:nrow(Data_cor)){
+   correlations <-  which((Data_cor[i,] > 0.6 | Data_cor[i,] < -0.6) & (Data_cor[i,] != 1))
+   if(length(correlations)> 0){
+     print(colnames(myDataM)[i])
+     print(correlations)
+   }
+ }
[1] "Ecom"
SalesFImage
          7
[1] "TechSup"
WartyClaim
         9
[1] "CompRes"
  OrdBilling     DelSpeed Satisfaction
          10           11           12
[1] "ProdLine"
DelSpeed
      11
[1] "SalesFImage"
Ecom
   2
[1] "WartyClaim"
TechSup
      3
[1] "OrdBilling"
 CompRes DelSpeed
       4       11
[1] "DelSpeed"
   CompRes   ProdLine OrdBilling
         4          6         10
[1] "Satisfaction"
CompRes
      4
>
> #Simple Linear Regression of Dependent with each Idpendent Variable
> myDataSLR = myDataM
> dim(myDataSLR)
[1] 100  12
> view(myDataSLR)
> #SLR Model for ProdQual & Satisfaction
> qplot(myDataSLR$ProdQual,myDataSLR$Satisfaction, main = "Plot between Satisfaction & Product Quality",
+       xlab = "Product Quality", ylab = "Satisfaction")
> cor(myDataSLR$ProdQual,myDataSLR$Satisfaction)
[1] 0.486325
> modSat_PQ =  lm(Satisfaction ~ ProdQual, data = myDataSLR)
> summary(modSat_PQ)

Call:
```

```
lm(formula = Satisfaction ~ ProdQual, data = myDataSLR)

Residuals:
     Min       1Q   Median       3Q      Max 
-1.88746 -0.72711 -0.01577  0.85641  2.25220 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  3.67593    0.59765   6.151 1.68e-08 ***
ProdQual     0.41512    0.07534   5.510 2.90e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.047 on 98 degrees of freedom
Multiple R-squared:  0.2365,	Adjusted R-squared:  0.2287 
F-statistic: 30.36 on 1 and 98 DF,  p-value: 2.901e-07

> #SLR Model for Ecom & Satisfaction
> qplot(myDataSLR$Ecom,myDataSLR$Satisfaction, main = "Plot between Satisfaction & Ecom",
+       xlab = "Ecom", ylab = "Satisfaction")
> cor(myDataSLR$Ecom,myDataSLR$Satisfaction)
[1] 0.282745
> modSat_Ecom =  lm(Satisfaction ~ Ecom, data = myDataSLR)
> summary(modSat_Ecom)

Call:
lm(formula = Satisfaction ~ Ecom, data = myDataSLR)

Residuals:
     Min       1Q   Median       3Q      Max 
-2.37200 -0.78971  0.04959  0.68085  2.34580 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   5.1516     0.6161   8.361 4.28e-13 ***
Ecom          0.4811     0.1649   2.918  0.00437 ** 
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.149 on 98 degrees of freedom
Multiple R-squared:  0.07994,	Adjusted R-squared:  0.07056 
F-statistic: 8.515 on 1 and 98 DF,  p-value: 0.004368

> #SLR Model for TechSup & Satisfaction
> qplot(myDataSLR$TechSup,myDataSLR$Satisfaction, main = "Plot between Satisfaction & TechSup",
+       xlab = "TechSup", ylab = "Satisfaction")
> cor(myDataSLR$TechSup,myDataSLR$Satisfaction)
[1] 0.1125972
> modSat_TechSup =  lm(Satisfaction ~ TechSup, data = myDataSLR)
> summary(modSat_TechSup)

Call:
lm(formula = Satisfaction ~ TechSup, data = myDataSLR)

Residuals:
     Min       1Q   Median       3Q      Max 
```

```
-2.26136 -0.93297  0.04302  0.82501  2.85617

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  6.44757    0.43592  14.791   <2e-16 ***
TechSup      0.08768    0.07817   1.122    0.265
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.19 on 98 degrees of freedom
Multiple R-squared:  0.01268,	Adjusted R-squared:  0.002603
F-statistic: 1.258 on 1 and 98 DF,  p-value: 0.2647

> #SLR Model for CompRes & Satisfaction
> qplot(myDataSLR$CompRes,myDataSLR$Satisfaction, main = "Plot between Satisfaction & CompRes",
+       xlab = "CompRes", ylab = "Satisfaction")
> cor(myDataSLR$CompRes,myDataSLR$Satisfaction)
[1] 0.6032626
> modSat_CompRes =  lm(Satisfaction ~ CompRes, data = myDataSLR)
> summary(modSat_CompRes)

Call:
lm(formula = Satisfaction ~ CompRes, data = myDataSLR)

Residuals:
     Min       1Q   Median       3Q      Max 
-2.40450 -0.66164  0.04499  0.63037  2.70949

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  3.68005    0.44285   8.310 5.51e-13 ***
CompRes      0.59499    0.07946   7.488 3.09e-11 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.9554 on 98 degrees of freedom
Multiple R-squared:  0.3639,	Adjusted R-squared:  0.3574
F-statistic: 56.07 on 1 and 98 DF,  p-value: 3.085e-11

> #SLR Model for Advertising & Satisfaction
> qplot(myDataSLR$Advertising,myDataSLR$Satisfaction, main = "Plot between Satisfaction & Advertising",
+       xlab = "Advertising", ylab = "Satisfaction")
> cor(myDataSLR$Advertising,myDataSLR$Satisfaction)
[1] 0.3046695
> modSat_Advertising =  lm(Satisfaction ~ Advertising, data = myDataSLR)
> summary(modSat_Advertising)

Call:
lm(formula = Satisfaction ~ Advertising, data = myDataSLR)

Residuals:
     Min       1Q   Median       3Q      Max 
-2.34033 -0.92755  0.05577  0.79773  2.53412

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
```

```
(Intercept)   5.6259     0.4237  13.279  < 2e-16 ***
Advertising   0.3222     0.1018   3.167  0.00206 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.141 on 98 degrees of freedom
Multiple R-squared:  0.09282,	Adjusted R-squared:  0.08357
F-statistic: 10.03 on 1 and 98 DF,  p-value: 0.002056

> #SLR Model for ProdLine & Satisfaction
> qplot(myDataSLR$ProdLine,myDataSLR$Satisfaction, main = "Plot between Satisfaction & ProdLine",
+       xlab = "ProdLine", ylab = "Satisfaction")
> cor(myDataSLR$ProdLine,myDataSLR$Satisfaction)
[1] 0.5505459
> modSat_ProdLine =  lm(Satisfaction ~ ProdLine, data = myDataSLR)
> summary(modSat_ProdLine)

Call:
lm(formula = Satisfaction ~ ProdLine, data = myDataSLR)

Residuals:
    Min      1Q  Median      3Q     Max 
-2.3634 -0.7795  0.1097  0.7604  1.7373 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  4.02203    0.45471   8.845 3.87e-14 ***
ProdLine     0.49887    0.07641   6.529 2.95e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1 on 98 degrees of freedom
Multiple R-squared:  0.3031,	Adjusted R-squared:  0.296 
F-statistic: 42.62 on 1 and 98 DF,  p-value: 2.953e-09

> #SLR Model for SalesFImage & Satisfaction
> qplot(myDataSLR$SalesFImage,myDataSLR$Satisfaction, main = "Plot between Satisfaction & SalesFImage",
+       xlab = "SalesFImage", ylab = "Satisfaction")
> cor(myDataSLR$SalesFImage,myDataSLR$Satisfaction)
[1] 0.5002053
> modSat_SalesFImage =  lm(Satisfaction ~ SalesFImage, data = myDataSLR)
> summary(modSat_SalesFImage)

Call:
lm(formula = Satisfaction ~ SalesFImage, data = myDataSLR)

Residuals:
    Min      1Q  Median      3Q     Max 
-2.2164 -0.5884  0.1838  0.6922  2.0728 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  4.06983    0.50874   8.000 2.54e-12 ***
SalesFImage  0.55596    0.09722   5.719 1.16e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 1.037 on 98 degrees of freedom
Multiple R-squared:  0.2502,	Adjusted R-squared:  0.2426
F-statistic:  32.7 on 1 and 98 DF,  p-value: 1.164e-07

> #SLR Model for ComPricing & Satisfaction
> qplot(myDataSLR$ComPricing,myDataSLR$Satisfaction, main = "Plot between Satisfaction & ComPricing",
+       xlab = "ComPricing", ylab = "Satisfaction")
> cor(myDataSLR$ComPricing,myDataSLR$Satisfaction)
[1] -0.2082957
> modSat_ComPricing =  lm(Satisfaction ~ ComPricing, data = myDataSLR)
> summary(modSat_ComPricing)

Call:
lm(formula = Satisfaction ~ ComPricing, data = myDataSLR)

Residuals:
    Min      1Q  Median      3Q     Max 
-1.9728 -0.9915 -0.1156  0.9111  2.5845 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  8.03856    0.54427  14.769   <2e-16 ***
ComPricing  -0.16068    0.07621  -2.108   0.0376 *  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.172 on 98 degrees of freedom
Multiple R-squared:  0.04339,	Adjusted R-squared:  0.03363 
F-statistic: 4.445 on 1 and 98 DF,  p-value: 0.03756

> #SLR Model for WartyClaim & Satisfaction
> qplot(myDataSLR$WartyClaim,myDataSLR$Satisfaction, main = "Plot between Satisfaction & WartyClaim",
+       xlab = "WartyClaim", ylab = "Satisfaction")
> cor(myDataSLR$WartyClaim,myDataSLR$Satisfaction)
[1] 0.1775448
> modSat_WartyClaim =  lm(Satisfaction ~ WartyClaim, data = myDataSLR)
> summary(modSat_WartyClaim)

Call:
lm(formula = Satisfaction ~ WartyClaim, data = myDataSLR)

Residuals:
     Min       1Q   Median       3Q      Max 
-2.36504 -0.90202  0.03019  0.90763  2.88985 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   5.3581     0.8813   6.079 2.32e-08 ***
WartyClaim    0.2581     0.1445   1.786   0.0772 .  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.179 on 98 degrees of freedom
Multiple R-squared:  0.03152,	Adjusted R-squared:  0.02164 
F-statistic:  3.19 on 1 and 98 DF,  p-value: 0.0772
```

```
> #SLR Model for OrdBilling & Satisfaction
> qplot(myDataSLR$OrdBilling,myDataSLR$Satisfaction, main = "Plot between Satisfaction & OrdBilling",
+       xlab = "OrdBilling", ylab = "Satisfaction")
> cor(myDataSLR$OrdBilling,myDataSLR$Satisfaction)
[1] 0.5217319
> modSat_OrdBilling =  lm(Satisfaction ~ OrdBilling, data = myDataSLR)
> summary(modSat_OrdBilling)

Call:
lm(formula = Satisfaction ~ OrdBilling, data = myDataSLR)

Residuals:
    Min      1Q  Median      3Q     Max 
-2.4005 -0.7071 -0.0344  0.7340  2.9673 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   4.0541     0.4840   8.377 3.96e-13 ***
OrdBilling    0.6695     0.1106   6.054 2.60e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.022 on 98 degrees of freedom
Multiple R-squared:  0.2722,	Adjusted R-squared:  0.2648 
F-statistic: 36.65 on 1 and 98 DF,  p-value: 2.602e-08

> #SLR Model for DelSpeed & Satisfaction
> qplot(myDataSLR$DelSpeed,myDataSLR$Satisfaction, main = "Plot between Satisfaction & DelSpeed",
+       xlab = "DelSpeed", ylab = "Satisfaction")
> cor(myDataSLR$DelSpeed,myDataSLR$Satisfaction)
[1] 0.5770423
> modSat_DelSpeed =  lm(Satisfaction ~ DelSpeed, data = myDataSLR)
> summary(modSat_DelSpeed)

Call:
lm(formula = Satisfaction ~ DelSpeed, data = myDataSLR)

Residuals:
     Min       1Q   Median       3Q      Max 
-2.22475 -0.54846  0.08796  0.54462  2.59432 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   3.2791     0.5294   6.194 1.38e-08 ***
DelSpeed      0.9364     0.1339   6.994 3.30e-10 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.9783 on 98 degrees of freedom
Multiple R-squared:  0.333,	Adjusted R-squared:  0.3262 
F-statistic: 48.92 on 1 and 98 DF,  p-value: 3.3e-10

> ##Initial Model
> model0 = lm(Satisfaction~., myDataM)
> summary(model0)
```

```
Call:
lm(formula = Satisfaction ~ ., data = myDataM)

Residuals:
     Min       1Q   Median       3Q      Max 
-1.43005 -0.31165  0.07621  0.37190  0.90120 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) -0.66961    0.81233  -0.824  0.41199    
ProdQual     0.37137    0.05177   7.173 2.18e-10 ***
Ecom        -0.44056    0.13396  -3.289  0.00145 ** 
TechSup      0.03299    0.06372   0.518  0.60591    
CompRes      0.16703    0.10173   1.642  0.10416    
Advertising -0.02602    0.06161  -0.422  0.67382    
ProdLine     0.14034    0.08025   1.749  0.08384 .  
SalesFImage  0.80611    0.09775   8.247 1.45e-12 ***
ComPricing  -0.03853    0.04677  -0.824  0.41235    
WartyClaim  -0.10298    0.12330  -0.835  0.40587    
OrdBilling   0.14635    0.10367   1.412  0.16160    
DelSpeed     0.16570    0.19644   0.844  0.40124    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.5623 on 88 degrees of freedom
Multiple R-squared:  0.8021,	Adjusted R-squared:  0.7774 
F-statistic: 32.43 on 11 and 88 DF,  p-value: < 2.2e-16

> vif(model0)
   ProdQual        Ecom     TechSup     CompRes Advertising    ProdLine SalesFImage  ComPricing 
   1.635797    2.756694    2.976796    4.730448    1.508933    3.488185    3.439420    1.635000 
 WartyClaim  OrdBilling    DelSpeed 
   3.198337    2.902999    6.516014 
> #Factor Analysis
> FData <- subset(myDataM, select = -c(12)) #Taking a subset of independent variables
> names(FData)
 [1] "ProdQual"    "Ecom"        "TechSup"     "CompRes"     "Advertising" "ProdLine"   
 [7] "SalesFImage" "ComPricing"  "WartyClaim"  "OrdBilling"  "DelSpeed"   
> datamatrix<-cor(FData)
> KMO(r=datamatrix) #MSA should be greater than 0.5
Kaiser-Meyer-Olkin factor adequacy
Call: KMO(r = datamatrix)
Overall MSA =  0.65
MSA for each item = 
   ProdQual        Ecom     TechSup     CompRes Advertising    ProdLine SalesFImage  ComPricing 
       0.51        0.63        0.52        0.79        0.78        0.62        0.62        0.75 
 WartyClaim  OrdBilling    DelSpeed 
       0.51        0.76        0.67 
> cortest.bartlett(datamatrix, n = 50) ### n is a sample size
$chisq
[1] 291.6151

$p.value
[1] 6.078303e-34
```

```
$df
[1] 55

> ?cortest.bartlett
> ev = eigen(datamatrix)
> EigenValue=ev$values
> EigenValue
 [1] 3.42697133 2.55089671 1.69097648 1.08655606 0.60942409 0.55188378 0.40151815 0.24695154
 [9] 0.20355327 0.13284158 0.09842702
> Factor=c(1:11)
> Scree=data.frame(Factor,EigenValue)
> plot(Scree,main="Scree Plot", col="Blue",ylim=c(0,5))
> lines(Scree,col="Red")
> #FA
> fa1<- fa(r=FData, nfactors = 4, rotate="varimax",fm="pa")
> print(fa1)
Factor Analysis using method =  pa
Call: fa(r = FData, nfactors = 4, rotate = "varimax", fm = "pa")
Standardized loadings (pattern matrix) based upon correlation matrix
              PA1   PA2   PA3   PA4   h2    u2 com
ProdQual     0.02 -0.07  0.02  0.65 0.42 0.576 1.0
Ecom         0.07  0.79  0.03 -0.11 0.64 0.362 1.1
TechSup      0.02 -0.03  0.88  0.12 0.79 0.205 1.0
CompRes      0.90  0.13  0.05  0.13 0.84 0.157 1.1
Advertising  0.17  0.53 -0.04 -0.06 0.31 0.686 1.2
ProdLine     0.53 -0.04  0.13  0.71 0.80 0.200 1.9
SalesFImage  0.12  0.97  0.06 -0.13 0.98 0.021 1.1
ComPricing  -0.08  0.21 -0.21 -0.59 0.44 0.557 1.6
WartyClaim   0.10  0.06  0.89  0.13 0.81 0.186 1.1
OrdBilling   0.77  0.13  0.09  0.09 0.62 0.378 1.1
DelSpeed     0.95  0.19  0.00  0.09 0.94 0.058 1.1

                       PA1  PA2  PA3  PA4
SS loadings           2.63 1.97 1.64 1.37
Proportion Var        0.24 0.18 0.15 0.12
Cumulative Var        0.24 0.42 0.57 0.69
Proportion Explained  0.35 0.26 0.22 0.18
Cumulative Proportion 0.35 0.60 0.82 1.00

Mean item complexity =  1.2
Test of the hypothesis that 4 factors are sufficient.

The degrees of freedom for the null model are  55  and the objective function was  6.55 with Chi Square
of  619.27
The degrees of freedom for the model are 17  and the objective function was  0.33

The root mean square of the residuals (RMSR) is  0.02
The df corrected root mean square of the residuals is  0.03

The harmonic number of observations is  100 with the empirical chi square  3.19  with prob <  1
The total number of observations was  100  with Likelihood Chi Square =  30.27  with prob <  0.024

Tucker Lewis Index of factoring reliability =  0.921
RMSEA index =  0.096  and the 90 % confidence intervals are  0.032 0.139
BIC =  -48.01
Fit based upon off diagonal values = 1
```

```
Measures of factor score adequacy
                                                   PA1  PA2  PA3  PA4
Correlation of (regression) scores with factors   0.98 0.99 0.94 0.88
Multiple R square of scores with factors          0.96 0.97 0.88 0.78
Minimum correlation of possible factor scores     0.93 0.94 0.77 0.55
> fa.diagram(fa1)
> fa2<- fa(r=FData, nfactors = 5, rotate="varimax",fm="pa")
> print(fa2)
Factor Analysis using method =  pa
Call: fa(r = FData, nfactors = 5, rotate = "varimax", fm = "pa")
Standardized loadings (pattern matrix) based upon correlation matrix
              PA1   PA2   PA3   PA4   PA5   h2     u2 com
ProdQual     0.03 -0.07  0.01  0.68 -0.07 0.48 0.5214 1.0
Ecom         0.06  0.78  0.03 -0.11  0.01 0.63 0.3692 1.1
TechSup      0.02 -0.03  0.89  0.11  0.01 0.80 0.2001 1.0
CompRes      0.88  0.14  0.06  0.13  0.03 0.81 0.1867 1.1
Advertising  0.16  0.53 -0.04 -0.06 -0.02 0.31 0.6858 1.2
ProdLine     0.51 -0.04  0.14  0.71  0.40 0.95 0.0536 2.5
SalesFImage  0.11  0.98  0.06 -0.13  0.01 0.99 0.0073 1.1
ComPricing  -0.08  0.22 -0.22 -0.56 -0.03 0.42 0.5811 1.7
WartyClaim   0.10  0.06  0.88  0.12  0.01 0.81 0.1922 1.1
OrdBilling   0.82  0.13  0.09  0.11 -0.20 0.75 0.2472 1.2
DelSpeed     0.94  0.18  0.00  0.05  0.24 0.98 0.0199 1.2

                       PA1  PA2  PA3  PA4  PA5
SS loadings           2.65 1.98 1.65 1.38 0.27
Proportion Var        0.24 0.18 0.15 0.13 0.02
Cumulative Var        0.24 0.42 0.57 0.70 0.72
Proportion Explained  0.33 0.25 0.21 0.17 0.03
Cumulative Proportion 0.33 0.58 0.79 0.97 1.00

Mean item complexity =  1.3
Test of the hypothesis that 5 factors are sufficient.

The degrees of freedom for the null model are  55  and the objective function was  6.55 with Chi Square
of  619.27
The degrees of freedom for the model are 10  and the objective function was  0.13

The root mean square of the residuals (RMSR) is  0.01
The df corrected root mean square of the residuals is  0.02

The harmonic number of observations is  100 with the empirical chi square  1.19  with prob <  1
The total number of observations was  100  with Likelihood Chi Square =  12.1  with prob <  0.28

Tucker Lewis Index of factoring reliability =  0.979
RMSEA index =  0.056  and the 90 % confidence intervals are  0 0.124
BIC =  -33.96
Fit based upon off diagonal values = 1
Measures of factor score adequacy
                                                   PA1  PA2  PA3  PA4  PA5
Correlation of (regression) scores with factors   0.98 0.99 0.94 0.92 0.76
Multiple R square of scores with factors          0.96 0.99 0.88 0.85 0.57
Minimum correlation of possible factor scores     0.92 0.98 0.77 0.71 0.15
> fa.diagram(fa2)
> #Combining the factors in the data for regression analysis
> regdata <- cbind(myDataM[12], fa1$scores)
```

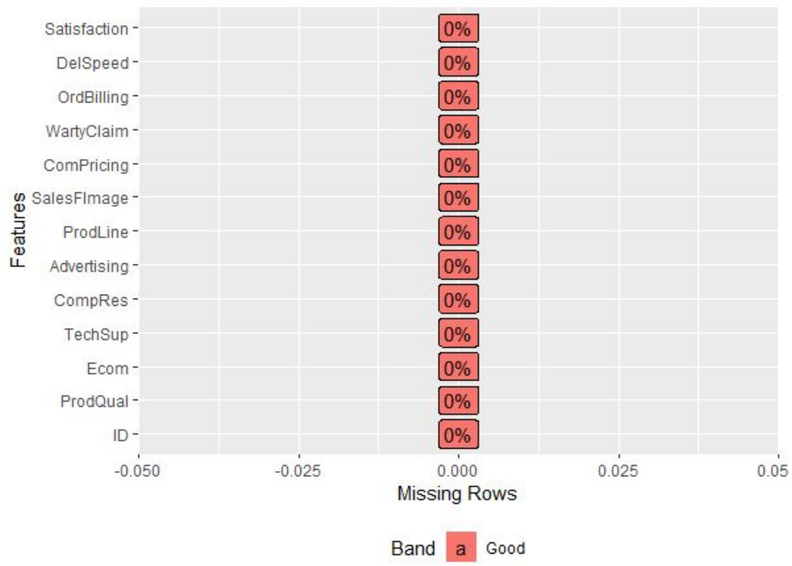
```
> head(regdata)
  Satisfaction        PA1        PA2          PA3         PA4
1          8.2 -0.1338871  0.9175166 -1.719604873  0.09135411
2          5.7  1.6297604 -2.0090053 -0.596361722  0.65808192
3          8.9  0.3637658  0.8361736  0.002979966  1.37548765
4          4.8 -1.2225230 -0.5491336  1.245473305 -0.64421384
5          7.1 -0.4854209 -0.4276223 -0.026980304  0.47360747
6          4.7 -0.5950924 -1.3035333 -1.183019401 -0.95913571

> #Splitting the data 70:30
> set.seed(10)
> indices= sample(1:nrow(regdata), 0.7*nrow(regdata))
> indices
 [1]  9 74 76 55 72 54 39 83 88 15 91 42 71 99 34 24 13  8  7 27 29 80 50 26 33 82 77 78 30 68 51
[32] 95 59 32 11 98 89 28 81 64 14 84 65 41 25 93 16 53 87 56 17 48 23 90 46 85 86  4 35 60 58 61
[63] 10 38 69 47 31 37  5 19
> train=regdata[indices,]
> test = regdata[-indices,]
> dim(train)
[1] 70  5
> dim(test)
[1] 30  5
> names(train)
[1] "Satisfaction" "PA1"          "PA2"          "PA3"          "PA4"
> #Regression Model using train data
> model1 = lm(Satisfaction~., train)
> summary(model1)

Call:
lm(formula = Satisfaction ~ ., data = train)

Residuals:
     Min       1Q   Median       3Q      Max 
-1.69261 -0.47602  0.09094  0.48715  1.12820 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  6.91194    0.08216  84.131  < 2e-16 ***
PA1          0.66493    0.08396   7.919 4.06e-11 ***
PA2          0.40322    0.09817   4.107 0.000114 ***
PA3          0.05730    0.08822   0.649 0.518304    
PA4          0.60209    0.09709   6.201 4.35e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.6652 on 65 degrees of freedom
Multiple R-squared:  0.6341,	Adjusted R-squared:  0.6116 
F-statistic: 28.16 on 4 and 65 DF,  p-value: 1.39e-13

> vif(model1)
     PA1      PA2      PA3      PA4 
1.003032 1.021759 1.009641 1.028411 
> model2 = lm(Satisfaction~PA1 + PA2 + PA4, train)
> summary(model2)

Call:
```

```
lm(formula = Satisfaction ~ PA1 + PA2 + PA4, data = train)

Residuals:
     Min       1Q   Median       3Q      Max 
-1.67785 -0.45521  0.09673  0.52682  1.09648 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  6.92156    0.08046  86.028  < 2e-16 ***
PA1          0.66743    0.08351   7.993 2.73e-11 ***
PA2          0.40527    0.09769   4.149 9.77e-05 ***
PA4          0.60744    0.09631   6.307 2.71e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.6623 on 66 degrees of freedom
Multiple R-squared:  0.6317,	Adjusted R-squared:  0.615 
F-statistic: 37.74 on 3 and 66 DF,  p-value: 2.532e-14

> vif(model2)
     PA1      PA2      PA4 
1.000927 1.020696 1.020996 
> #Now using the test data to predict
> pred=predict(model2, newdata = test, type = "response")
> pred
       1        2        3        6       12       18       20       21       22       36       40 
7.259537 7.594863 8.338754 5.413472 5.949228 7.482209 7.809208 5.559574 9.215931 5.827328 6.342061 
      43       44       45       49       52       57       62       63       66       67       70 
8.147853 7.760535 7.492218 8.266969 7.714021 8.575765 6.663641 6.468533 7.512508 6.779930 6.324929 
      73       75       79       92       94       96       97      100 
7.679019 7.011674 8.593262 4.659920 8.603187 7.247490 6.351660 5.910752 
> test$Satisfaction.Predict <- pred
> names(test)
[1] "Satisfaction"          "PA1"                   "PA2"                   "PA3"                  
[5] "PA4"                   "Satisfaction.Predict"
> #Let's check the predictions.
> head(test[c(1,6)],10)
   Satisfaction Satisfaction.Predict
1           8.2             7.259537
2           5.7             7.594863
3           8.9             8.338754
6           4.7             5.413472
12          6.0             5.949228
18          7.4             7.482209
20          7.6             7.809208
21          5.4             5.559574
22          9.9             9.215931
36          5.4             5.827328
> SSE_val <- sum((test$Satisfaction - pred) ^ 2)
> SST_val <- sum((test$Satisfaction - mean(test$Satisfaction)) ^ 2)
> SSR_val=SST_val-SSE_val
> RSquare_val<-SSR_val/SST_val
> RSquare_val
[1] 0.6815526
> Term1<- (1-RSquare_val)
> Term2<- (count(as.data.frame(pred))-1)/(count(as.data.frame(pred))-3-1)
```

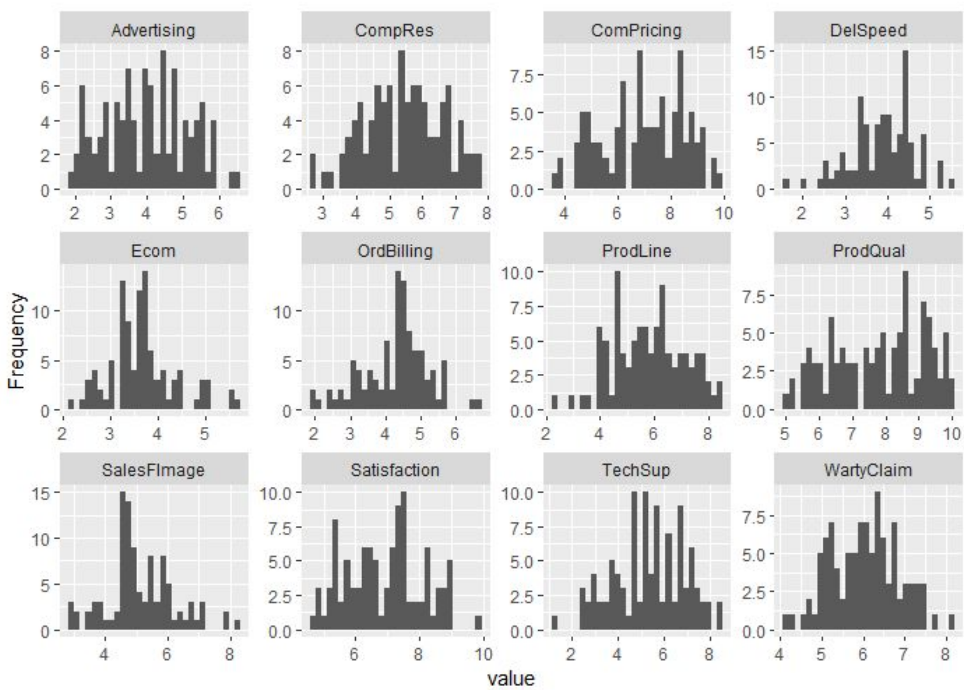
```
> AdjustedRSquare_val <- 1-(Term1*Term2)
> AdjustedRSquare_val
          n
1 0.6448086
> library(Metrics)
> rmse(test$Satisfaction,pred)
[1] 0.7974715
```

7 Appendix B – Graphs and Plot

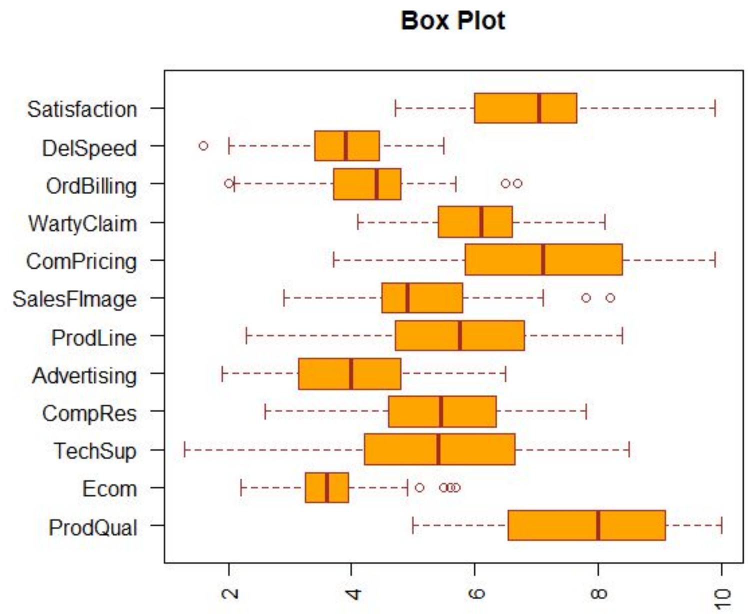
Missing Value Plot



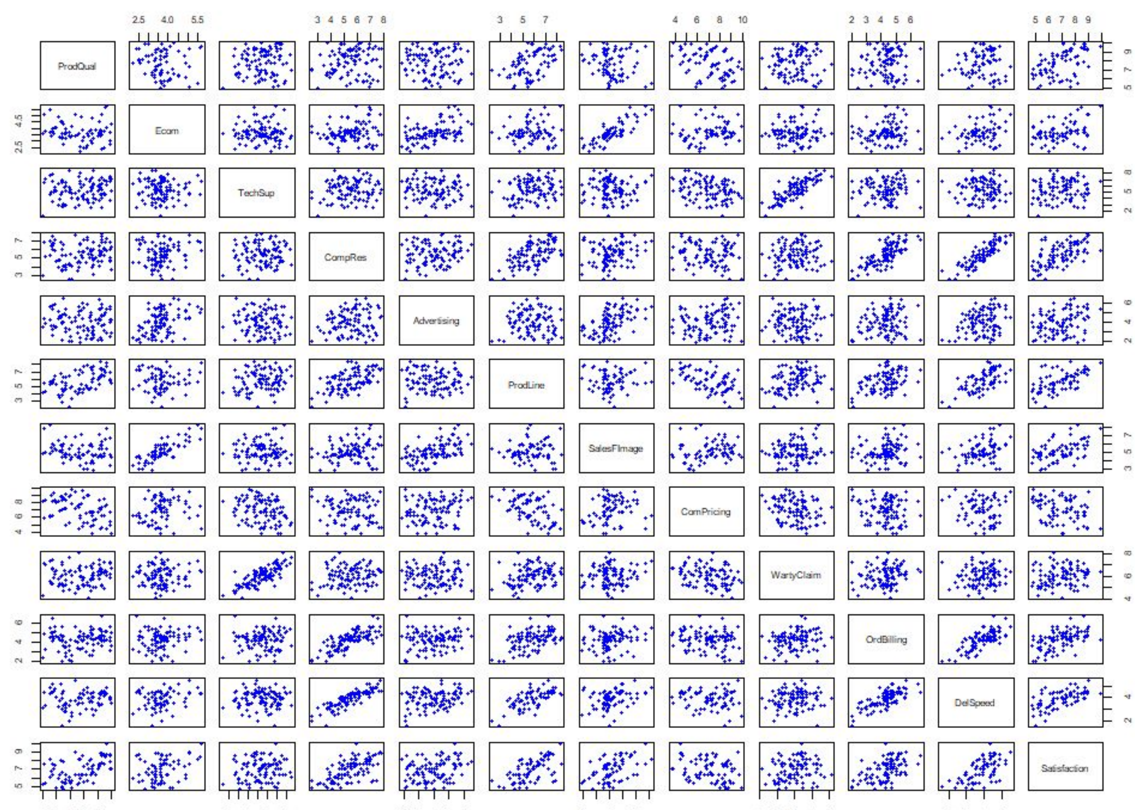
Histogram of Variable



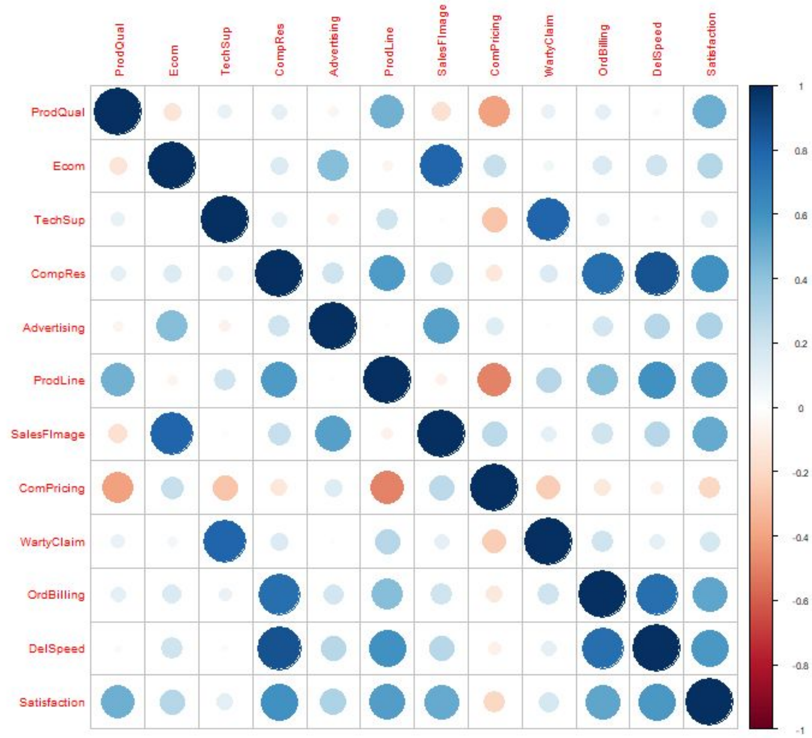
Box Plot to Check Outlier and data distribution

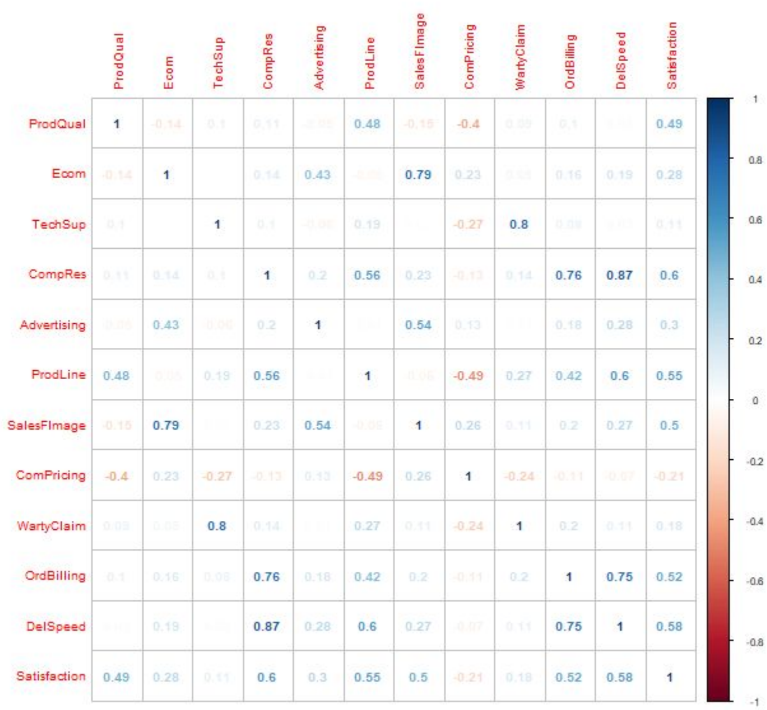


Scatter Plot between variable

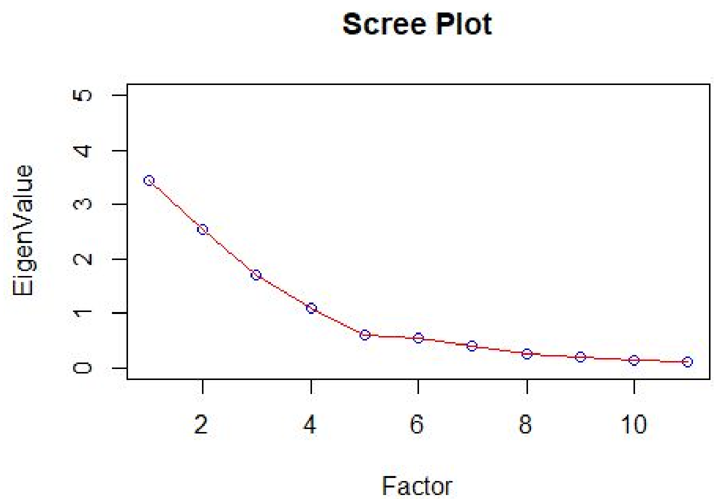


Correlation Plot

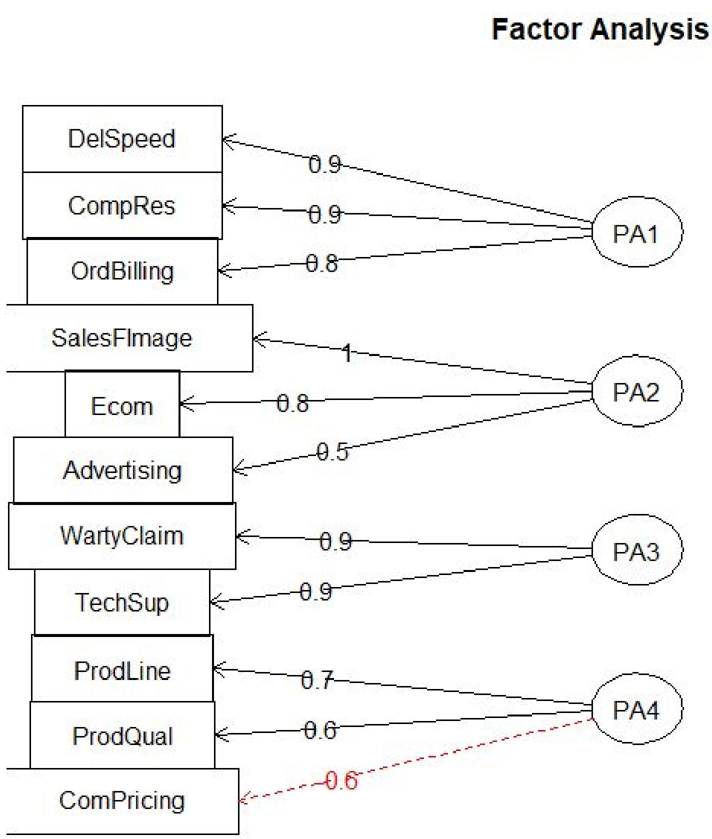




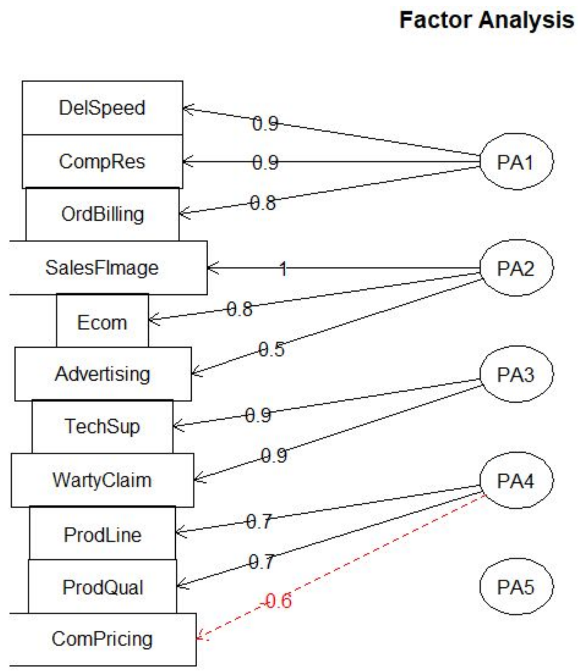
Scree Plot

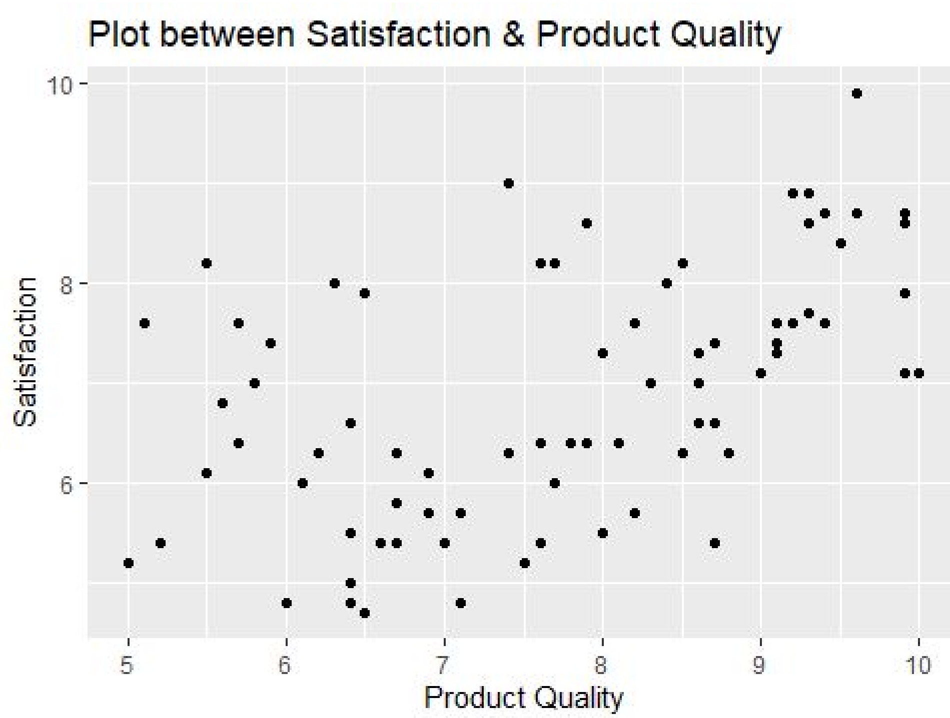


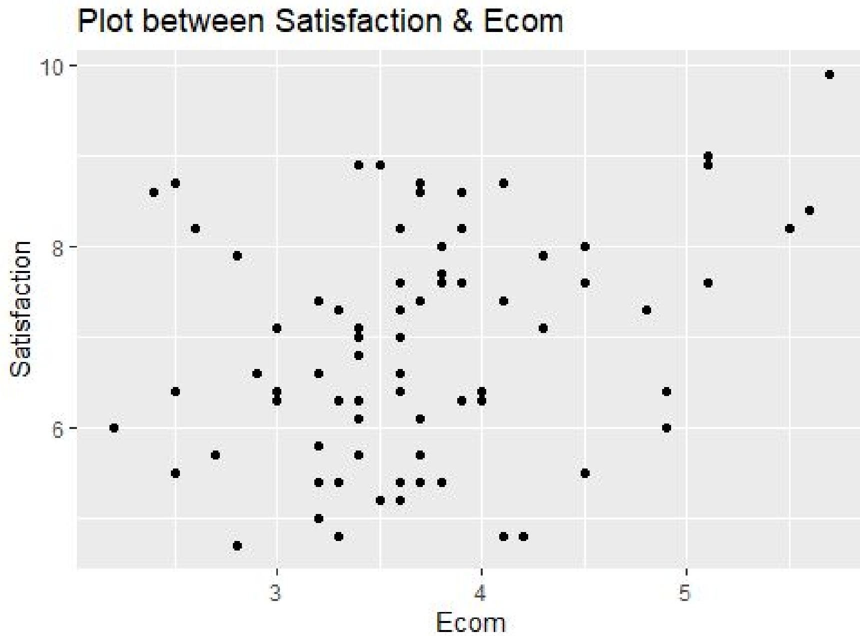
Factor Plot (4 Factor)

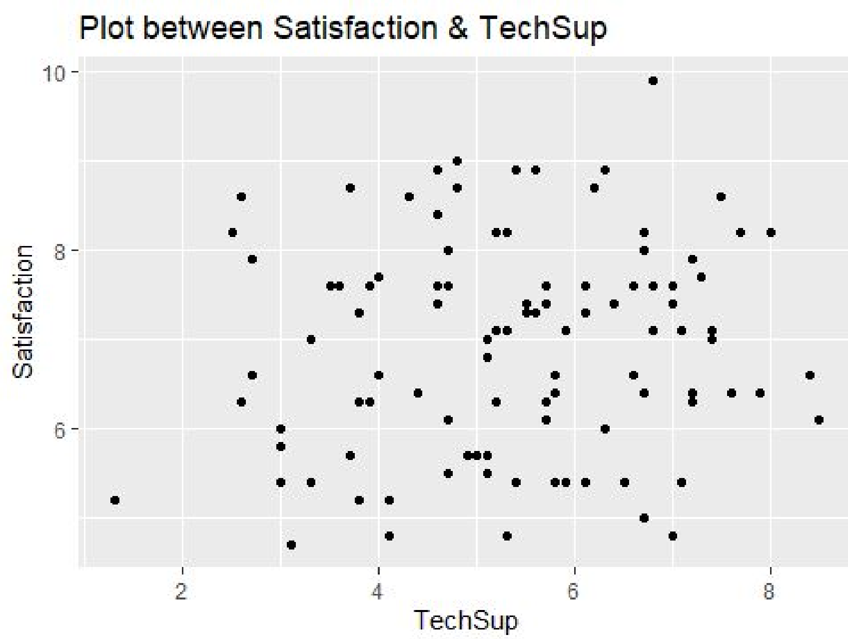


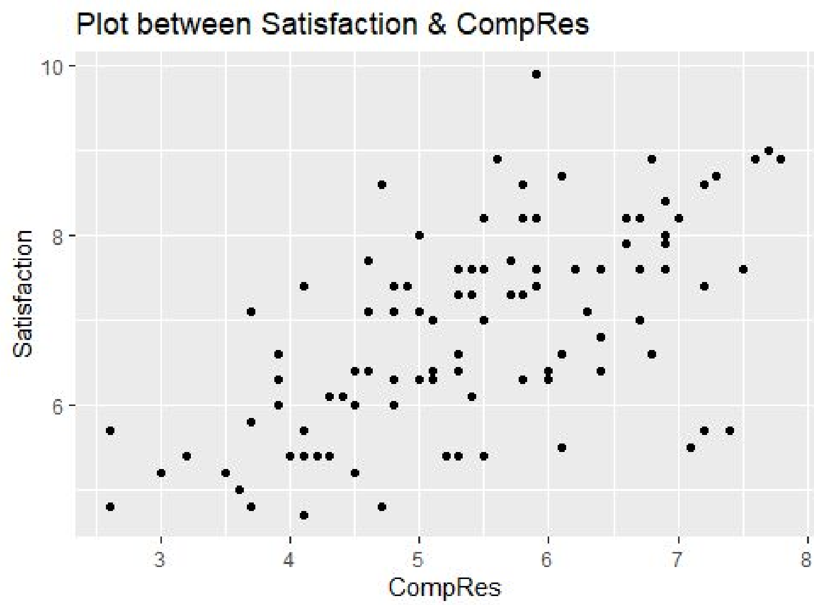
Factor Plot (5 Factor)

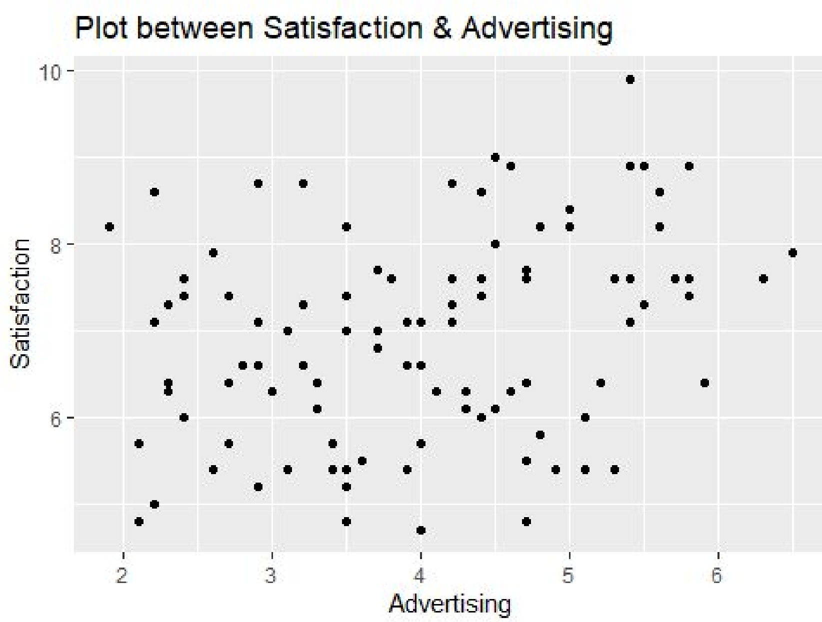




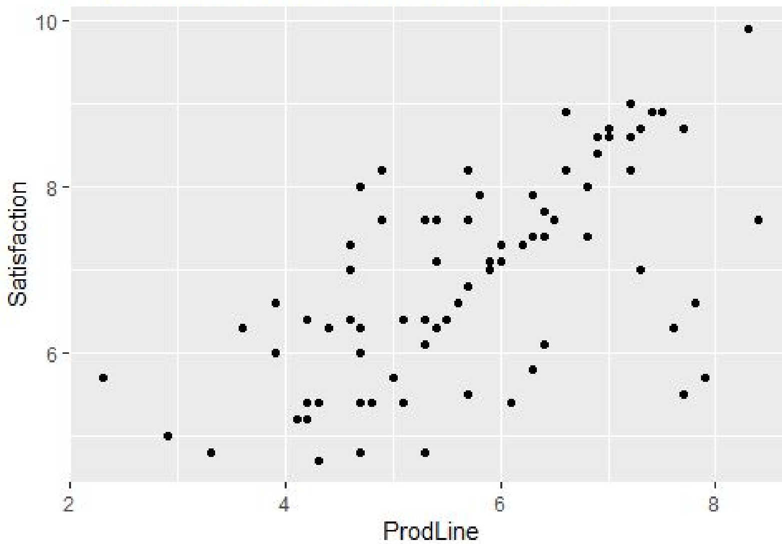

Plot between Satisfaction & Ecom
10
8
6
Satisfaction
3
4
5
Ecom


Plot between Satisfaction & TechSup
10
8
6
Satisfaction
2
4
6
8
TechSup

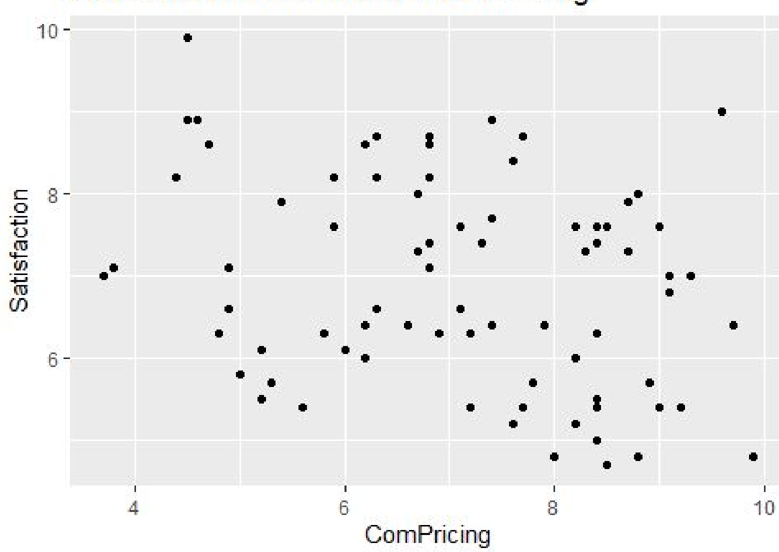

Plot between Satisfaction & CompRes
Satisfaction
CompRes
10
8
6
3
4
5
6
7
8

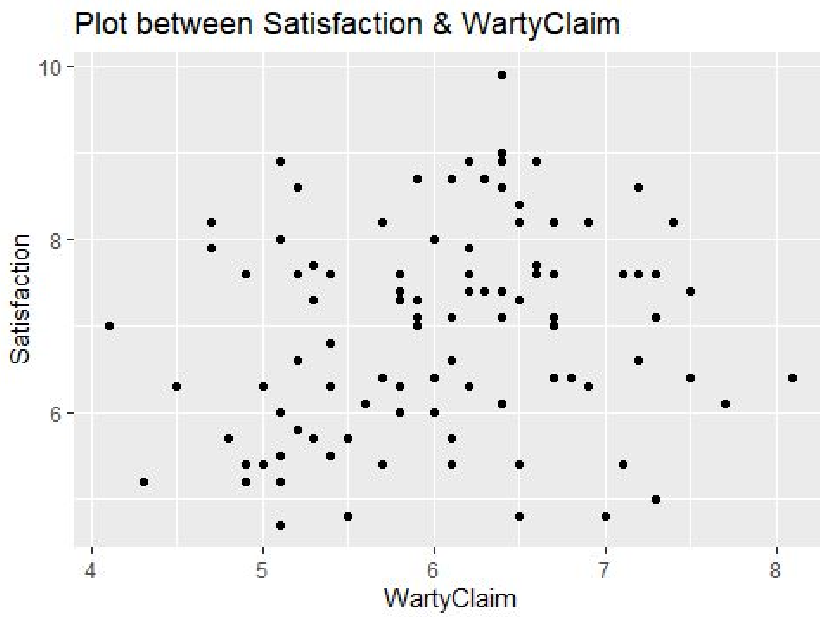

Plot between Satisfaction & Advertising
Satisfaction
Advertising
10
8
6
2
3
4
5
6

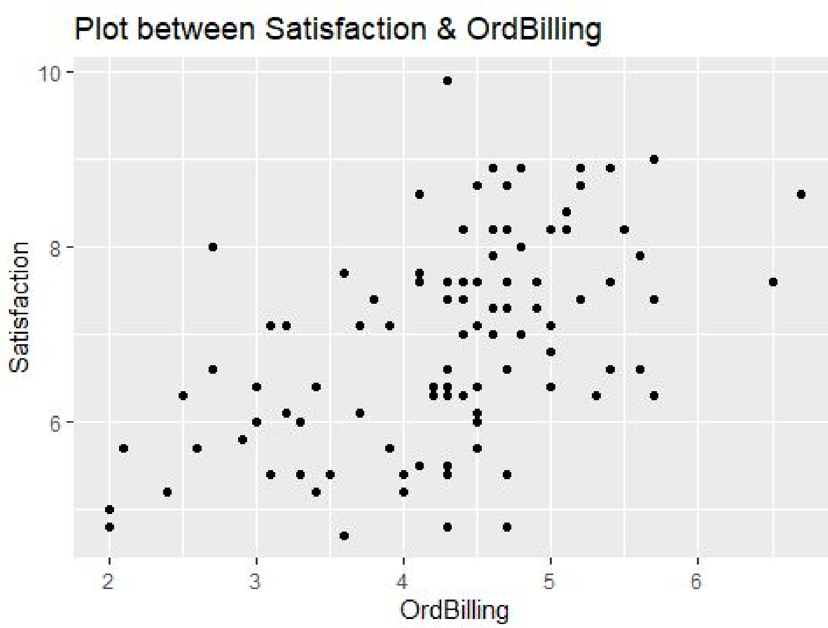
Plot between Satisfaction & ProdLine

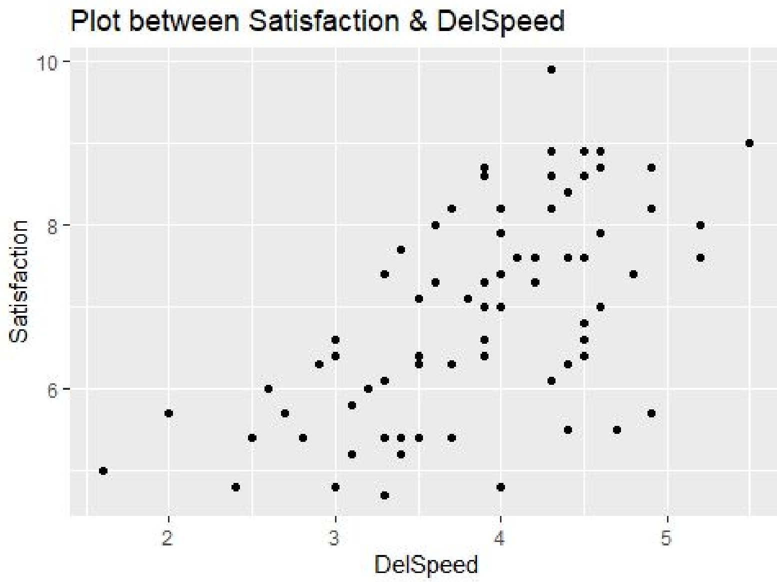


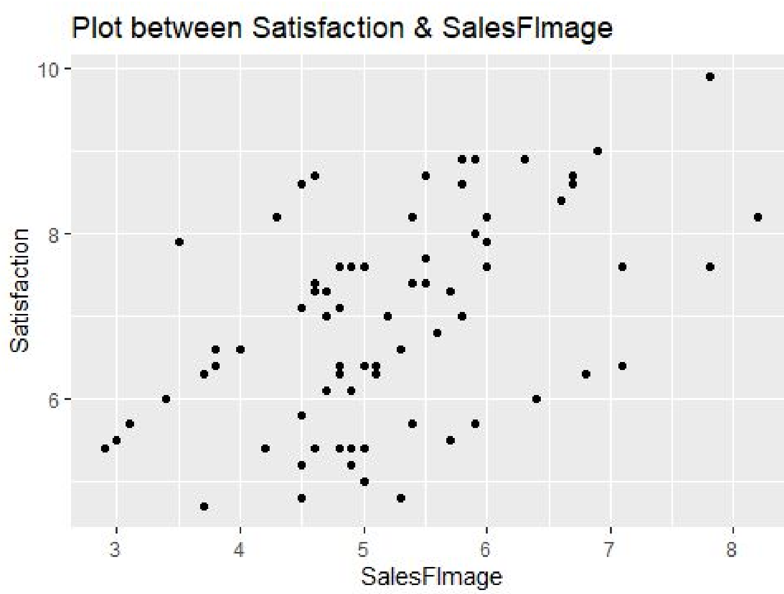
Plot between Satisfaction & ComPricing




Plot between Satisfaction & WartyClaim
Satisfaction
10
8
6
4
5
6
7
8
WartyClaim


Plot between Satisfaction & OrdBilling
Satisfaction
10
8
6
2
3
4
5
6
OrdBilling


Plot between Satisfaction & DelSpeed
Satisfaction
DelSpeed
10
8
6
2
3
4
5


Plot between Satisfaction & SalesFImage
Satisfaction
SalesFImage
10
8
6
3
4
5
6
7
8