Mini – Project

Report on Multiple Linear Regression Analysis

Report By

Chakradhar Varma

Table of Contents

1. Project Objective

The objective of the project is to perform Multivariate Data Analysis, perform Factor Analysis and derive underlying factors in the dataset, provide insights and test conclusions, build linear regression model and provide comments on the validity of the model. This report will address the following:

- Existence of Multicollinearity in the Dataset.
- Exploratory Factor Analysis.
- Interpretation and Naming of the Factors.
- Multiple Linear Regression.

2. Solutions

This Solutions section will answer the Questions asked in the following steps:

1. Preliminary Analysis.
2. Factor Analysis.
3. Multiple Linear regression Analysis.
4. Validation.

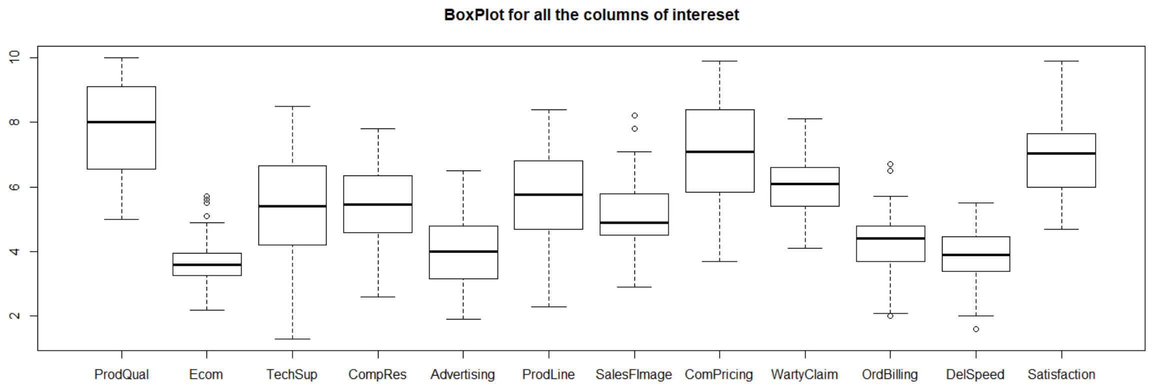
2.1 Preliminary Analysis

In this Preliminary Analysis the given dataset is analysed and validated whether the dataset is suitable for the Factor Analysis. As per the Factor Analysis Design, the sample data should have all metric variables and so is the given dataset. It has 100 observations and effectively 12 variables in the ratio of 8:1 which is more than desired. A descriptive five point summary of all the variables is presented below.

- **Five-point Summary of the data set(fh_data)**

	mean	sd	median	min	max
ProdQual	7.81	1.396279	8	5	10
Ecom	3.672	0.700516	3.6	2.2	5.7
TechSup	5.365	1.530457	5.4	1.3	8.5
CompRes	5.442	1.208403	5.45	2.6	7.8
Advertising	4.01	1.126943	4	1.9	6.5
ProdLine	5.805	1.315285	5.75	2.3	8.4
SalesFImage	5.123	1.07232	4.9	2.9	8.2
ComPricing	6.974	1.545055	7.1	3.7	9.9
WartyClaim	6.043	0.819738	6.1	4.1	8.1
OrdBilling	4.278	0.92884	4.4	2	6.7
DelSpeed	3.886	0.734437	3.9	1.6	5.5
Satisfaction	6.918	1.191839	7.05	4.7	9.9

The Boxplots for all the columns of interest are presented below and from the plot it is observed that few variables have outliers but at a very minimal number. All the variables are rated in the scale of 0 to 10.



2.2 Factor Analysis

Factor analysis is an interdependence technique whose primary purpose is to define the underlying structure among the variables in the analysis. It is used to identify a new, smaller set of uncorrelated variables to replace the original set of correlated variables for subsequent analysis such as Regression or Discriminant Analysis.

2.2.1 Assumptions

The Assumptions of the Factor Analysis are:

- A basic assumption is that some underlying structure does exist in the set of selected variables.
- The sample is homogeneous with respect to the underlying factor structure.
- Multicollinearity in the data is desirable because the objective is to identify interrelated set of variables.
- The data should be amenable for factor analysis. It should not be such that a variable is only correlated with itself and no correlation exists with any other variables.

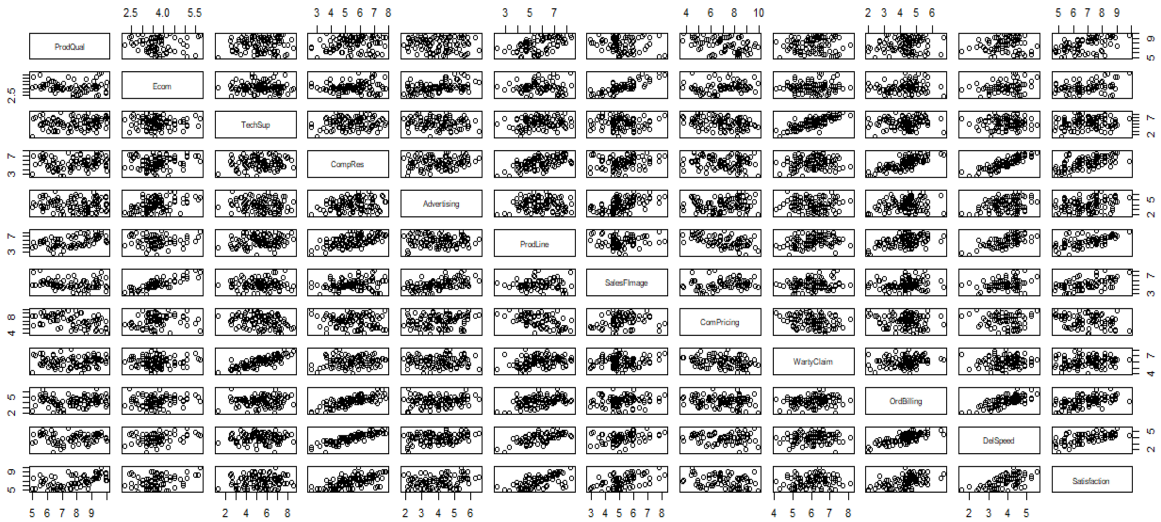
From the preliminary analysis it is evident that the given sample data is satisfying the design needs of factor analysis, but it should also be validated against the assumptions in the analysis to proceed further.

2.2.2 Validation of Assumptions

The given sample doesn't have any differing groups hence it is a homogeneous set. As per the Factor Analysis guidelines, mixing dependent and independent variables in a single factor analysis and then using the derived factors to support dependence relationships is inappropriate. So the correlation matrix is constructed excluding customer satisfaction (treating it as a dependent variable) and it is clear that customer satisfaction forms correlation with all the other variables in the sample.

Initial inspection of the correlation matrix shows that there are correlations between the variables and the scatter plots providing the evidence for the same is presented below.

Further it is required to prove the existence of multicollinearity statistically rather than conceptually, so **Bartlett test of sphericity test** is performed to check the statistical significance that the correlation matrix has significant correlations among at least some of the variables.



The p value from the test "1.79337e-96" shows that the sample has significant correlations. The next measure to quantify the degree of intercorrelations among the variables and the appropriateness of factor analysis is the **measure of sampling adequacy (MSA)** and it is measured using **Kaiser-Meyer-Olkin (KMO)** test. The result shows that all the variables have the required adequacy of more than 0.5 which implies no need to remove any variable at the moment and the overall MSA score is "0.65", which implies the **mediocre level** of adequacy to carry out the analysis.

The above tests form the basis to conclude that all the assumptions are satisfied and there is Multicollinerity present in the sample and the factor analysis can be performed.

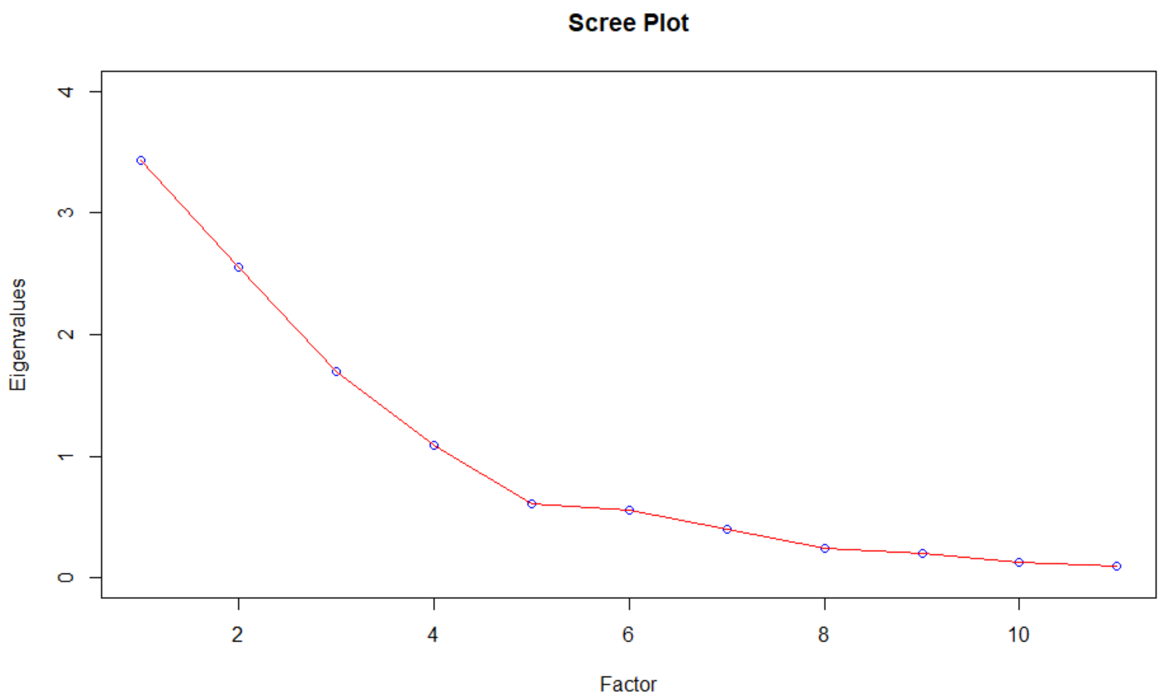
2.2.3 Deriving Factors

Now that the variables are specified and the correlation matrix is prepared, factor analysis is performed to identify the underlying structure of relationships.

Here in this analysis, principal component method is used to extract the factors and to decide upon number of factors scree plot is used in relation to Kaiser Rule. Eigen decomposition is calculated and a scree plot is constructed as shown below, with the Eigen values obtained for each variable and the number of factors.

From the Eigen value information, as per Kaiser Rule factors with eigenvalues greater than 1.0, four factors should be retained. The scree plot however, indicates that five factors may be appropriate when considering the changes in eigenvalues (i.e., identifying the "elbow" in the eigenvalues). As the eigenvalue for the fifth factor is low (.61) relative to the latent root criterion value of 1.0 it is not considered.

From the initial unrotated component data, it is observed that the four factors



retained represent 80 percent of the variance of the 11 variables, deemed sufficient in terms of total variance explained. Combining all these criteria together leads to the conclusion to retain four factors for further analysis.

2.2.4 Interpretation of Factors

Below table presents the unrotated Factor Matrix obtained from principal component method. The percentages of trace explained by each of the four factors (31%, 23%, 15%, and 10%, respectively) are shown in the proportion variance row of values.

	Unrotated Factor Matrix				
Variables	**F1**	**F2**	**F3**	**F4**	**communality**
ProdQual	0.25	**-0.50**	-0.08	**0.67**	0.77
Ecom	0.31	**0.71**	0.31	0.28	0.78
TechSup	0.29	-0.37	**0.79**	-0.20	0.89
CompRes	**0.87**	0.03	-0.27	-0.22	0.88
Advertising	0.34	**0.58**	0.11	0.33	0.58
ProdLine	**0.72**	**-0.45**	-0.15	0.21	0.79
SalesFImage	0.38	**0.75**	0.31	0.23	0.86
ComPricing	-0.28	**0.66**	-0.07	-0.35	0.64
WartyClaim	0.39	-0.31	**0.78**	-0.19	0.89
OrdBilling	**0.81**	0.04	-0.22	-0.25	0.77
DelSpeed	**0.88**	0.12	-0.30	-0.21	0.91
SS loadings	3.43	2.55	1.69	1.09	
Proportion Var	0.31	0.23	0.15	0.10	
Cumulative Var	0.31	0.54	0.70	0.80	

2.2.4.1 Unrotated Component Analysis

Based on this factor-loading pattern with a relatively large number of high loadings on factor 2 and only one high loading on factor 4, interpretation is difficult and theoretically less meaningful. Therefore, factor matrix needs to be rotated in order to redistribute the variance from the earlier factors to the later factors and rotation will result in a simpler and theoretically more meaningful factor pattern.

From the communality information in the above table it is observed that the communality figure of 0.58 for variable Advertising indicates that it has less in common with the other variables included in the analysis than does variable WarrantyClaim, which has a communality of 0.893. Both variables, however, still share more than one-half of their variance with the four factors. All of the communalities are sufficiently high to proceed with the rotation of the factor matrix.

2.2.4.2 Rotated Component Analysis

In this Rotated Component Analysis, **varimax rotation** is applied and its impact on the factor loadings is presented in the table below. It is observed that the total amount of variance extracted is the same in the rotated solution as it was in the unrotated one, 80 percent. Also, the communalities for each variable did not change when a rotation technique is applied.

The major changes that can be seen are first, the variance is redistributed so that the factor-loading pattern and the percentage of variance for each of the factors are slightly different. Specifically, in the rotated factor solution, the first factor accounts for 26 percent of the variance, compared to 31 percent in the unrotated solution. Likewise, the other factors also changed, the largest change being the fourth factor, increasing from 10 percent in the unrotated solution to 16 percent in the rotated solution.

	Rotated Factor Matrix				
Variables	**F1**	**F2**	**F3**	**F4**	**communality**
ProdQual				**0.88**	0.77
Ecom		**0.87**			0.78
TechSup			**0.94**		0.89
CompRes	**0.93**				0.88
Advertising		**0.74**			0.58
ProdLine	**0.59**			**0.64**	0.79
SalesFImage		**0.90**			0.86
ComPricing				**-0.72**	0.64
WartyClaim			**0.93**		0.89
OrdBilling	**0.86**				0.77
DelSpeed	**0.94**				0.91
SS loadings	2.89	2.23	1.86	1.77	
Proportion Var	0.26	0.20	0.17	0.16	
Cumulative Var	0.26	0.47	0.63	0.80	

Second, the interpretation of the factor matrix is simplified. In the rotated factor solution each of the variables has a significant loading on only one factor, except for Product Line, which cross-loads on two factors (factors 1 and 4). Moreover, all of the loadings are above 0.70, meaning that more than one-half of the variance is accounted for by the loading on a single factor.

The cross-loadings of Product Line variable are very close and it is not ignorable to avoid the ambiguity of selecting Product Line variable under a particular factor. And as the factor 4 is explaining relatively more percentage of variance (16), the number of factors cannot be reduced further. The observations from the correlation matrix tells that product line variable is highly correlated with multiple variables and have a correlation of around 0.55 with customer satisfaction. Though it is a significant value, as it has correlation with other variables it is assumed that the variance would be explained by those correlated variables. So the product line variable is removed and rotated factor matrix is calculated again and the data is presented in the table below.

	Rotated Factor Matrix(ProdLine deleted)				
Variables	**F1**	**F2**	**F3**	**F4**	**communality**
ProdQual				**0.89**	0.80
Ecom		**0.87**			0.78
TechSup			**0.94**		0.89
CompRes	**0.93**				0.89
Advertising		**0.74**			0.58
SalesFImage		**0.90**			0.86
ComPricing				**-0.73**	0.66
WartyClaim			**0.93**		0.89
OrdBilling	**0.89**				0.81
DelSpeed	**0.93**				0.89
SS loadings	2.59	2.22	1.85	1.41	
Proportion Var	0.26	0.22	0.18	0.14	
Cumulative Var	0.26	0.48	0.67	0.81	

The factor loadings for the 10 variables remain almost identical, exhibiting both the same pattern and almost the same values for the loadings. The amount of explained variance increases slightly to 81 percent. With the simplified pattern of loadings (all at significant levels), all communalities above 50 percent (and most much higher), and the overall level of explained variance high enough, the 10-variable/four-factor solution is accepted.

2.2.4.3 Naming of Factors

The process of naming the factors is intuitive and subjective and it is based on the significant loadings in each factor. A marked pattern of variables with high loadings for each factor is shown in the previous section. Factors 1 and 2 have three variables with significant loadings and factors 3 and 4 have two. Factor 4 has one loading with the negative sign.

The Names of the Factors derived are as below:

Factor 1 - **Customer Service**: complaint resolution, delivery speed and order & billing.

Factor 2 - **Marketing**: salesforce image, E-commerce and advertising.

Factor 3 - **Technical Support**: technical support and warranty & claims.

Factor 4 - **Product Value**: product quality and competitive pricing.

2.2.5 Conclusion

The exploratory factor analysis is carried out on the given sample data and found out that an optimal structure exists and four factors namely customer service, marketing, tech support and product value are derived from the interpretation of data through principal component method. Since there is no additional data available for testing the factor analysis solution, validation of the factor analysis is not carried out. These factors will be used as independent variables for further analysis to build the regression model.

2.3 Multiple Linear Regression

From the factor analysis implementation, four factors have been derived and these factors would form the basis for performing the regression analysis and to build a valid model. There are multiple methods available to proceed with data preparation for the analysis for example selecting the variable with the highest factor loading as a surrogate representative for a particular factor dimension and the other is replacing the original set of variables with an entirely new, smaller set of variables created either from summated scales or factor scores.

Here in this analysis factor scores procedure is employed and corresponding factor scores for all the four factors are calculated. And for calculating the factor scores "**Harman**" method is used as it is "univocal for Orthogonal Factors" which indicates that the factor score estimates will not be contaminated with variance from other orthogonal factors (in distinction to other factor score estimates) in the analysis which is desired. Below is the table showing the weights or the factor score coefficients obtained.

Factor Score Coefficients				
variables	**RC1**	**RC2**	**RC3**	**RC4**
ProdQual	-0.05141942	0.130336	-0.08253	0.696389
Ecom	-0.056648414	0.347772	0.003677	0.015731
TechSup	-0.032983868	-0.04021	0.509449	-0.0168
CompRes	0.405955888	-0.05768	-0.01794	0.013334
Advertising	-0.016550813	0.166555	-0.02416	0.070922
SalesFImage	-0.061328766	0.547227	0.017095	-0.01057
ComPricing	0.003344616	0.005666	-0.00985	-0.30228
WartyClaim	-0.003811491	0.00975	0.487834	-0.02513
OrdBilling	0.219331197	-0.03352	0.000643	0.001364
DelSpeed	0.421827735	-0.02295	-0.03915	-0.06397

Using the newly created factor scores as explanatory variables or regressors and customer satisfaction as dependent variable, linear regression is conducted with the assumptions of Normality of residuals, Homoscedasticity and no multicollinearity among the independent variables. The initial analysis with all the four factors has given the results as shown below.

Multiple Regression Analysis based on Factor Scores				
	Estimate	**Std. Error**	**t value**	**Pr(>|t|)**
(Intercept)	6.918	0.06569	105.321	< 2e-16
Factor 1	0.64544	0.06946	9.293	5.37E-15
Factor 2	0.60013	0.07209	8.325	6.20E-13
Factor 3	0.06797	0.0701	0.97	**0.335**
Factor 4	0.65243	0.07768	8.399	4.32E-13
Residual standard error: 0.6569 on 95 degrees of freedom				
Multiple R-squared: **0.7085**, Adjusted R-squared: 0.6963				
F-statistic: 57.73 on 4 and 95 DF, p-value: **< 2.2e-16**				

From the result it is observed that the regression is significant as p-value is less than 0.05 and all the estimate coefficients are greater than '0' and all the explanatory factors are significant except for factor 3(p-value > .05). The coefficient of determination R^2 value is around 0.71. Since factor 3 is insignificant and it is also evident from the preliminary analysis that both tech support and warranty & claim are having less correlation with satisfaction, it is removed and the regression is performed again with the remaining factors. The results of the new model is as shown below.

Multiple Regression Analysis (without Factor 3)				
	Estimate	**Std. Error**	**t value**	**Pr(>|t|)**
(Intercept)	6.918	0.06566	105.354	< 2e-16
Factor 1	0.64593	0.06943	9.303	4.68E-15
Factor 2	0.60056	0.07207	8.333	5.57E-13
Factor 4	0.65561	0.07759	8.45	3.15E-13
Residual standard error: 0.6566 on 96 degrees of freedom				
Multiple R-squared: **0.7057**, Adjusted R-squared: 0.6965				
F-statistic: 76.71 on 3 and 96 DF, p-value: **< 2.2e-16**				

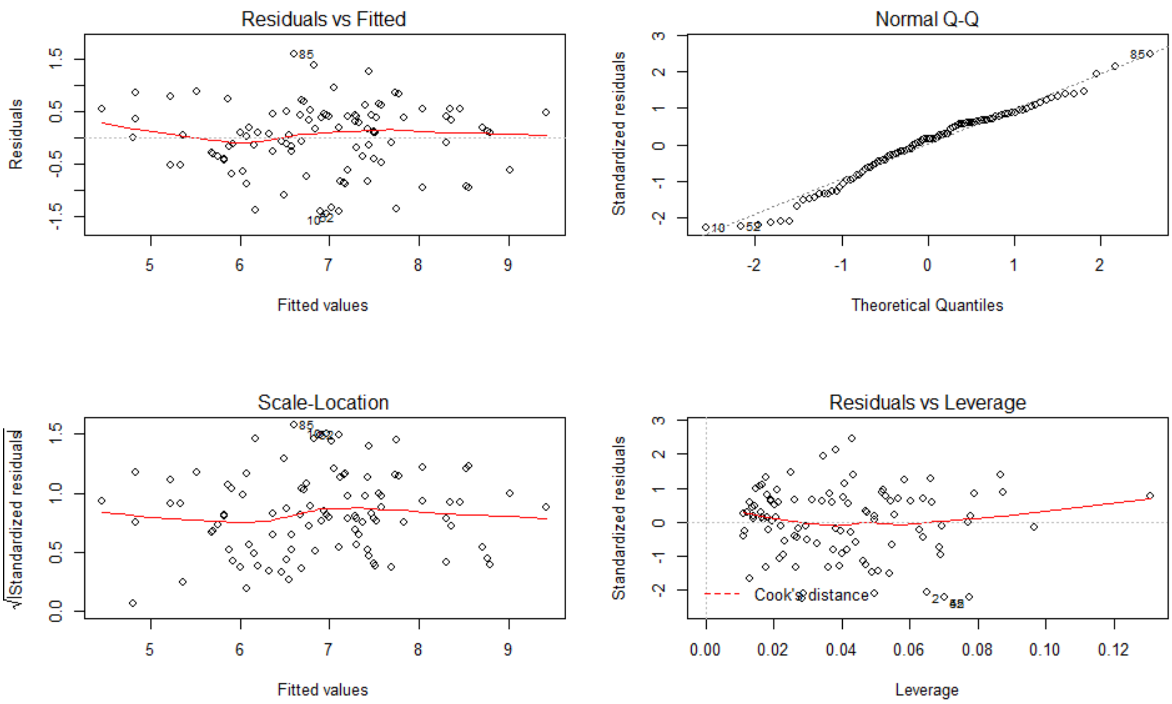
In the new regression analysis all the explanatory variables are significant and the overall model is also significant with R^2 value of 0.7057 which is slightly less at fraction level than the previous. Adjusted R^2 is also changed in fraction terms as insignificant variable is removed from the model.

2.4 Validation

The final model is tested for validation of assumptions. First, the mean of the residuals from the model is 0 ($-2.96e^{-17}$) and graph of Normal Q-Q plot below shows that all the points fall approximately along the reference line which implies normality holds. To further solidify this Saphiro-wilk test is carried out and the output, the p-value > 0.05 which implies that the distribution is not significantly different from normal.

Next, from the Residuals vs Fitted plot below it is observed that it's having a much flatted line indicating constant variance of the residuals with the fitted values. And also from the NCV test, p-value value > 0.05 which signifies homoscedasticity. The variance inflation factor for the model gives value of around 3.4 (< 5) indicating there is no multicollinearity.

So the interpretations and the tests of assumptions indicates that the model is valid and can be used for predicting the customer satisfaction using factor scores with the new set of data.



3. Conclusion

To conclude, all the questions that are asked as part of the project has been answered through this report. With the initial analysis proving to have multicollinearity, factor analysis has been performed to reduce the dimensions in the data set and to derive factors such that they form orthogonal(non collinear) components for further analysis. Those factors has been labelled based on the significant performers and the sign of the factor loadings. Next, a multiple linear regression analysis is performed to build a model for predicting the regressand customer satisfaction using the factor scores approach.

Later the interpretations and the assumptions of the model are validated using subjective graphical approach as well as statistical methods. Substituting the values of factor scores into the regression equation predicted an approximate value for the customer satisfaction. Assuming normal distribution of the sample dataset, it can be said that the model can be applied to population data as well. For further validating the fit and predictions of the model, the half approach of dividing the given dataset into two parts and testing the model on the other half can be done.

4. Appendix A – Source Code

```r
#set up of working directory
setwd("D:/BACP Program/R Directory")
getwd()

## [1] "D:/BACP Program/R Directory"

#Loading Libraries
library('nFactors')

library('psych')
library('car')

#create a data frame object and read the data file.
fh_data = read.csv("Factor-Hair-Revised.csv")
View(fh_data)

#Dimension and structure of the data set
dim(fh_data)

## [1] 100  13

str(fh_data)

## 'data.frame':    100 obs. of  13 variables:
##  $ ID          : int  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 ...

#Attach Data to the session
attach(fh_data)

#Find out summary of the data
summary(fh_data)

##        ID            ProdQual            Ecom            TechSup
##  Min.   :  1.00   Min.   : 5.000   Min.   :2.200   Min.   :1.300
##  1st Qu.: 25.75   1st Qu.: 6.575   1st Qu.:3.275   1st Qu.:4.250
##  Median : 50.50   Median : 8.000   Median :3.600   Median :5.400
##  Mean   : 50.50   Mean   : 7.810   Mean   :3.672   Mean   :5.365
##  3rd Qu.: 75.25   3rd Qu.: 9.100   3rd Qu.:3.925   3rd Qu.:6.625
##  Max.   :100.00   Max.   :10.000   Max.   :5.700   Max.   :8.500
##     CompRes        Advertising        ProdLine       SalesFImage
##  Min.   :2.600   Min.   :1.900   Min.   :2.300   Min.   :2.900
##  1st Qu.:4.600   1st Qu.:3.175   1st Qu.:4.700   1st Qu.:4.500
```

```
##  Median :5.450   Median :4.000   Median :5.750   Median :4.900  
##  Mean   :5.442   Mean   :4.010   Mean   :5.805   Mean   :5.123  
##  3rd Qu.:6.325   3rd Qu.:4.800   3rd Qu.:6.800   3rd Qu.:5.800  
##  Max.   :7.800   Max.   :6.500   Max.   :8.400   Max.   :8.200  
##    ComPricing      WartyClaim      OrdBilling       DelSpeed    
##  Min.   :3.700   Min.   :4.100   Min.   :2.000   Min.   :1.600  
##  1st Qu.:5.875   1st Qu.:5.400   1st Qu.:3.700   1st Qu.:3.400  
##  Median :7.100   Median :6.100   Median :4.400   Median :3.900  
##  Mean   :6.974   Mean   :6.043   Mean   :4.278   Mean   :3.886  
##  3rd Qu.:8.400   3rd Qu.:6.600   3rd Qu.:4.800   3rd Qu.:4.425  
##  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
```

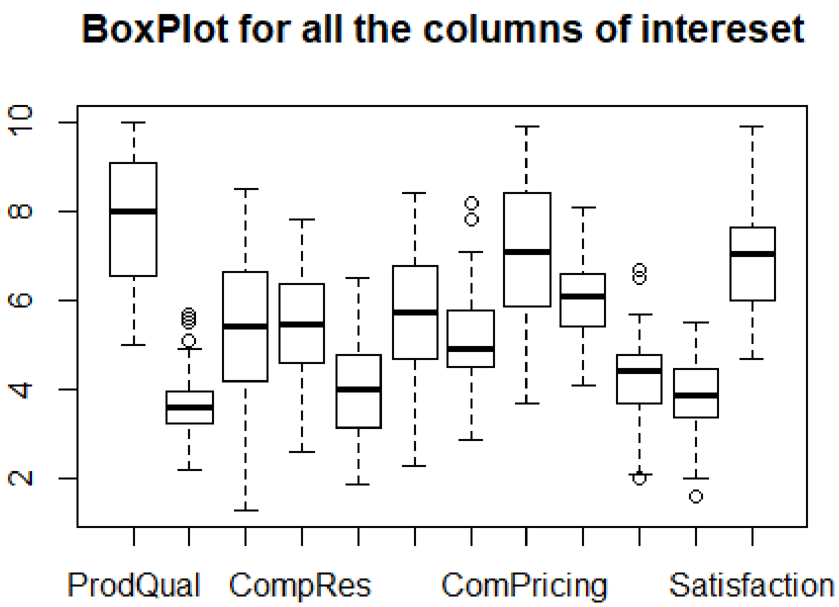
```
tmp <- do.call(data.frame,
               list(mean = apply(fh_data, 2, mean),
                    sd = apply(fh_data, 2, sd),
                    median = apply(fh_data, 2, median),
                    min = apply(fh_data, 2, min),
                    max = apply(fh_data, 2, max)))

print(tmp)
```

```
##                 mean         sd median min   max
## ID            50.500 29.0114920  50.50 1.0 100.0
## ProdQual       7.810  1.3962793   8.00 5.0  10.0
## Ecom           3.672  0.7005164   3.60 2.2   5.7
## TechSup        5.365  1.5304568   5.40 1.3   8.5
## CompRes        5.442  1.2084032   5.45 2.6   7.8
## Advertising    4.010  1.1269428   4.00 1.9   6.5
## ProdLine       5.805  1.3152850   5.75 2.3   8.4
## SalesFImage    5.123  1.0723198   4.90 2.9   8.2
## ComPricing     6.974  1.5450553   7.10 3.7   9.9
## WartyClaim     6.043  0.8197382   6.10 4.1   8.1
## OrdBilling     4.278  0.9288398   4.40 2.0   6.7
## DelSpeed       3.886  0.7344372   3.90 1.6   5.5
## Satisfaction   6.918  1.1918393   7.05 4.7   9.9
```

```
#check for any NA fields
any(is.na(fh_data))
```

```
## [1] FALSE
```

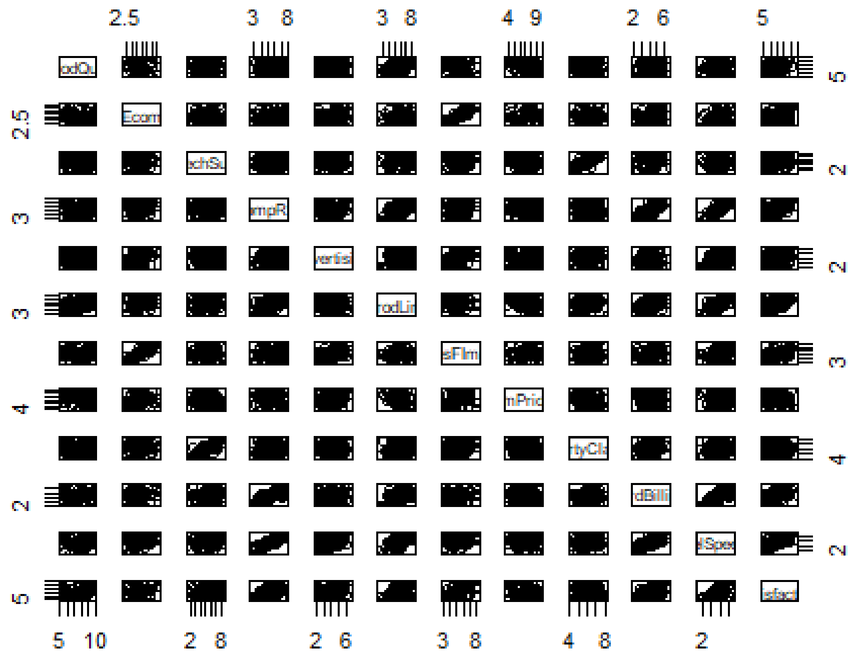
```
#Boxplot
boxplot(fh_data[,2:13], main = "BoxPlot for all the columns of intereset")
```




```
#saphiro test for checking normality
shapiro.test(ProdQual)

##
##  Shapiro-Wilk normality test
##
## data:  ProdQual
## W = 0.94972, p-value = 0.0007953

shapiro.test(Ecom)

##
##  Shapiro-Wilk normality test
##
## data:  Ecom
## W = 0.95852, p-value = 0.003157

shapiro.test(TechSup) #not siginificant

##
##  Shapiro-Wilk normality test
##
## data:  TechSup
## W = 0.98626, p-value = 0.39

shapiro.test(CompRes) #not siginificant

##
##  Shapiro-Wilk normality test
##
## data:  CompRes
## W = 0.98646, p-value = 0.4023
```

```r
shapiro.test(Advertising) #not siginificant

##
##  Shapiro-Wilk normality test
##
## data:  Advertising
## W = 0.97626, p-value = 0.06769

shapiro.test(ProdLine) #not siginificant

##
##  Shapiro-Wilk normality test
##
## data:  ProdLine
## W = 0.98692, p-value = 0.4324

shapiro.test(SalesFImage)

##
##  Shapiro-Wilk normality test
##
## data:  SalesFImage
## W = 0.97403, p-value = 0.04534

shapiro.test(ComPricing)

##
##  Shapiro-Wilk normality test
##
## data:  ComPricing
## W = 0.96758, p-value = 0.01448

shapiro.test(WartyClaim) #not siginificant

##
##  Shapiro-Wilk normality test
##
## data:  WartyClaim
## W = 0.99094, p-value = 0.7404

shapiro.test(OrdBilling)

##
##  Shapiro-Wilk normality test
##
## data:  OrdBilling
## W = 0.97405, p-value = 0.04549

shapiro.test(DelSpeed) #not siginificant

##
##  Shapiro-Wilk normality test
##
## data:  DelSpeed
## W = 0.98161, p-value = 0.177
```

```r
#Correlation Matrix Inspection
cor_fh_data = fh_data[,2:13]
pairs(cor_fh_data)
```



```r
cor(cor_fh_data)
```

```
##                 ProdQual          Ecom       TechSup    CompRes
## ProdQual      1.00000000 -0.1371632174  0.0956004542  0.1063700
## Ecom         -0.13716322  1.0000000000  0.0008667887  0.1401793
## TechSup       0.09560045  0.0008667887  1.0000000000  0.0966566
## CompRes       0.10637000  0.1401792611  0.0966565978  1.0000000
## Advertising  -0.05347313  0.4298907110 -0.0628700668  0.1969168
## ProdLine      0.47749341 -0.0526878383  0.1926254565  0.5614170
## SalesFImage  -0.15181287  0.7915437115  0.0169905395  0.2297518
## ComPricing   -0.40128188  0.2294624014 -0.2707866821 -0.1279543
## WartyClaim    0.08831231  0.0518981915  0.7971679258  0.1404083
## OrdBilling    0.10430307  0.1561473316  0.0801018246  0.7568686
## DelSpeed      0.02771800  0.1916360683  0.0254406935  0.8650917
## Satisfaction  0.48632500  0.2827450147  0.1125971788  0.6032626
##              Advertising    ProdLine SalesFImage  ComPricing  WartyClai
m
## ProdQual     -0.05347313  0.47749341 -0.15181287 -0.40128188  0.0883123
1
## Ecom          0.42989071 -0.05268784  0.79154371  0.22946240  0.0518981
9
## TechSup      -0.06287007  0.19262546  0.01699054 -0.27078668  0.7971679
3
## CompRes       0.19691685  0.56141695  0.22975176 -0.12795425  0.1404083
0
## Advertising   1.00000000 -0.01155082  0.54220366  0.13421689  0.0107920
7
```

```
## ProdLine     -0.01155082  1.00000000 -0.06131553 -0.49494840  0.2730775
3
## SalesFImage   0.54220366 -0.06131553  1.00000000  0.26459655  0.1074553
4
## ComPricing    0.13421689 -0.49494840  0.26459655  1.00000000 -0.2449860
5
## WartyClaim    0.01079207  0.27307753  0.10745534 -0.24498605  1.0000000
0
## OrdBilling    0.18423559  0.42440825  0.19512741 -0.11456703  0.1970651
2
## DelSpeed      0.27586308  0.60185021  0.27155126 -0.07287173  0.1093946
0
## Satisfaction  0.30466947  0.55054594  0.50020531 -0.20829569  0.1775448
2
##               OrdBilling    DelSpeed Satisfaction
## ProdQual      0.10430307  0.02771800    0.4863250
## Ecom          0.15614733  0.19163607    0.2827450
## TechSup       0.08010182  0.02544069    0.1125972
## CompRes       0.75686859  0.86509170    0.6032626
## Advertising   0.18423559  0.27586308    0.3046695
## ProdLine      0.42440825  0.60185021    0.5505459
## SalesFImage   0.19512741  0.27155126    0.5002053
## ComPricing   -0.11456703 -0.07287173   -0.2082957
## WartyClaim    0.19706512  0.10939460    0.1775448
## OrdBilling    1.00000000  0.75100307    0.5217319
## DelSpeed      0.75100307  1.00000000    0.5770423
## Satisfaction  0.52173191  0.57704227    1.0000000

#Testing for validation of assumptions
va_fh_data = fh_data[,2:12]
cortest.bartlett(cor(va_fh_data), n =100)

## $chisq
## [1] 619.2726
##
## $p.value
## [1] 1.79337e-96
##
## $df
## [1] 55

KMO(va_fh_data)

## Kaiser-Meyer-Olkin factor adequacy
## Call: KMO(r = va_fh_data)
## Overall MSA =  0.65
## MSA for each item =
##    ProdQual         Ecom      TechSup      CompRes Advertising     ProdLine
##        0.51         0.63         0.52         0.79        0.78         0.62
## SalesFImage  ComPricing  WartyClaim  OrdBilling     DelSpeed
##        0.62         0.75         0.51         0.76        0.67

#Factor Analysis
fa_fh_data = fh_data[,2:12]
```

```r
ev = eigen(cor(fa_fh_data))
ev
```

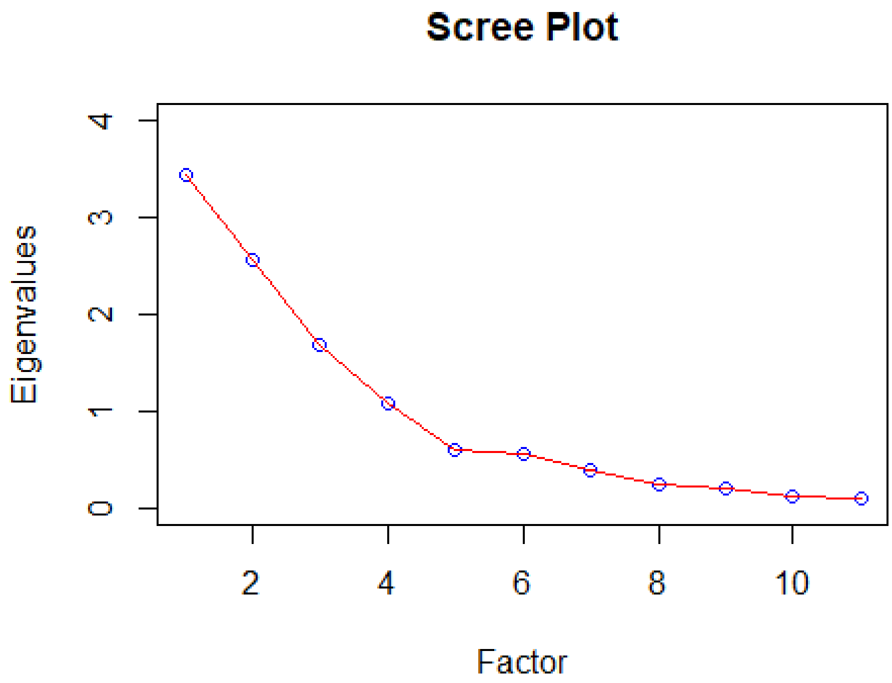
```
## eigen() decomposition
## $values
##  [1] 3.42697133 2.55089671 1.69097648 1.08655606 0.60942409 0.55188378
##  [7] 0.40151815 0.24695154 0.20355327 0.13284158 0.09842702
##
## $vectors
##             [,1]        [,2]        [,3]       [,4]        [,5]
##  [1,] -0.1337896  0.31349802  0.06227164  0.6431362  0.23166620
##  [2,] -0.1659528 -0.44650918 -0.23524791  0.2723803  0.42228844
##  [3,] -0.1576926  0.23096734 -0.61095105 -0.1933931 -0.02395667
##  [4,] -0.4706836 -0.01944394  0.21035078 -0.2063204  0.02865743
##  [5,] -0.1837350 -0.36366471 -0.08809705  0.3178945 -0.80387024
##  [6,] -0.3867652  0.28478056  0.11627864  0.2029023  0.11667416
##  [7,] -0.2036696 -0.47069599 -0.24134210  0.2221772  0.20437283
##  [8,]  0.1516886 -0.41345650  0.05304529 -0.3335435  0.24892601
##  [9,] -0.2129336  0.19167191 -0.59856398 -0.1853020 -0.03292706
## [10,] -0.4372177 -0.02639905  0.16892981 -0.2368536  0.02675377
## [11,] -0.4730891 -0.07305172  0.23262477 -0.1973299 -0.03543294
##              [,6]         [,7]        [,8]        [,9]       [,10]
##  [1,]  0.56456996 -0.191641317  0.13547311  0.03132810 -0.06659717
##  [2,] -0.26325703 -0.059626208 -0.12202642 -0.54251104 -0.28155772
##  [3,]  0.10876896  0.017199915  0.46470964 -0.35929961  0.38817090
##  [4,]  0.02815231  0.008499596  0.51339754  0.09324751 -0.53467243
##  [5,]  0.20056937  0.063069619 -0.05347713 -0.15468169 -0.03715799
##  [6,] -0.09819533  0.608147555 -0.33320710 -0.08415534  0.23479794
##  [7,] -0.10497225 -0.001437351  0.16910665  0.64489911  0.35341191
##  [8,]  0.70973595  0.308248871 -0.09883227 -0.09414389  0.04518224
##  [9,]  0.13983966  0.030640243 -0.44354040  0.31756604 -0.43534752
## [10,]  0.11947974 -0.659319893 -0.36601754 -0.09907265  0.30386545
## [11,] -0.02979992  0.234239274  0.06539059 -0.02188514  0.12010386
##              [,11]
##  [1,] -0.18279209
##  [2,] -0.06233863
##  [3,]  0.05192956
##  [4,]  0.36253352
##  [5,]  0.08118684
##  [6,]  0.38507778
##  [7,]  0.08469869
##  [8,]  0.10295751
##  [9,] -0.12893245
## [10,]  0.19415064
## [11,] -0.77563222
```

```r
Eigenvalues = ev$values
Eigenvalues
```

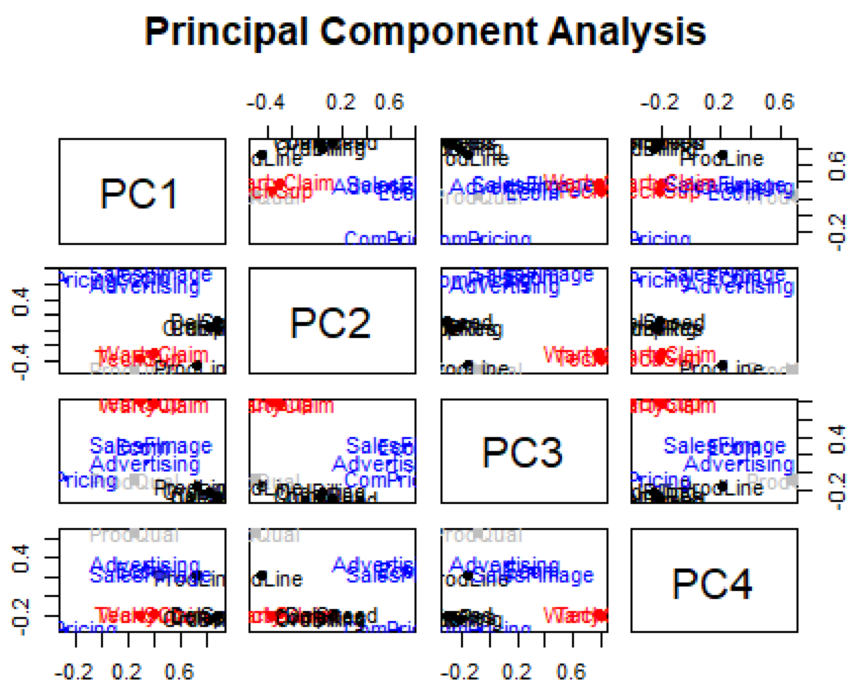
```
##  [1] 3.42697133 2.55089671 1.69097648 1.08655606 0.60942409 0.55188378
##  [7] 0.40151815 0.24695154 0.20355327 0.13284158 0.09842702
```

```r
Factor = 1:11

scree = data.frame(Factor,Eigenvalues)
```

```r
plot(scree,main="Scree Plot",col="Blue", ylim = c(0,4))
lines(scree,col="Red")
```



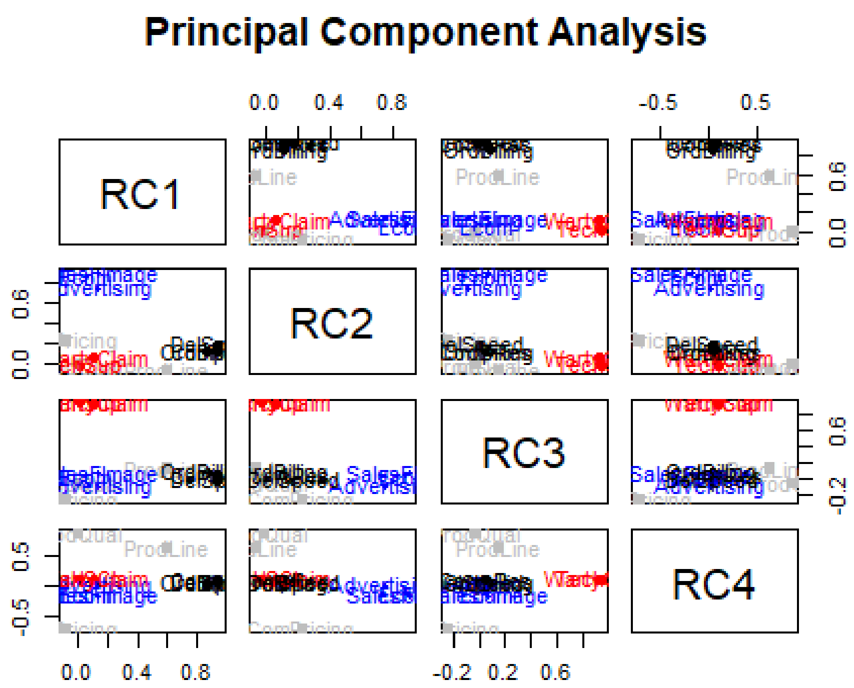

```r
#Unrotated PCA
unrotate = principal(fa_fh_data, nfactors = 4, rotate = "none")
unrotate$Vaccounted

##                             PC1       PC2       PC3        PC4
## SS loadings           3.4269713 2.5508967 1.6909765 1.08655606
## Proportion Var        0.3115428 0.2318997 0.1537251 0.09877782
## Cumulative Var        0.3115428 0.5434425 0.6971677 0.79594551
## Proportion Explained  0.3914123 0.2913512 0.1931352 0.12410124
## Cumulative Proportion 0.3914123 0.6827635 0.8758988 1.00000000

unrotatedprofile = plot(unrotate, row.names(unrotate$loadings))
```




```
#Rotated PCA
rotate = principal(fa_fh_data, nfactors = 4, rotate = "varimax")
rotate

## Principal Components Analysis
## Call: principal(r = fa_fh_data, nfactors = 4, rotate = "varimax")
## Standardized loadings (pattern matrix) based upon correlation matrix
##               RC1   RC2   RC3   RC4   h2    u2 com
## ProdQual     0.00 -0.01 -0.03  0.88 0.77 0.232 1.0
## Ecom         0.06  0.87  0.05 -0.12 0.78 0.223 1.1
## TechSup      0.02 -0.02  0.94  0.10 0.89 0.107 1.0
## CompRes      0.93  0.12  0.05  0.09 0.88 0.119 1.1
## Advertising  0.14  0.74 -0.08  0.01 0.58 0.424 1.1
## ProdLine     0.59 -0.06  0.15  0.64 0.79 0.213 2.1
## SalesFImage  0.13  0.90  0.08 -0.16 0.86 0.141 1.1
## ComPricing  -0.09  0.23 -0.25 -0.72 0.64 0.359 1.5
## WartyClaim   0.11  0.05  0.93  0.10 0.89 0.108 1.1
## OrdBilling   0.86  0.11  0.08  0.04 0.77 0.234 1.1
## DelSpeed     0.94  0.18  0.00  0.05 0.91 0.086 1.1
##
##                        RC1  RC2  RC3  RC4
## SS loadings           2.89 2.23 1.86 1.77
## Proportion Var        0.26 0.20 0.17 0.16
## Cumulative Var        0.26 0.47 0.63 0.80
## Proportion Explained  0.33 0.26 0.21 0.20
## Cumulative Proportion 0.33 0.59 0.80 1.00
##
## Mean item complexity =  1.2
## Test of the hypothesis that 4 components are sufficient.
##
## The root mean square of the residuals (RMSR) is  0.06
```

```
##  with the empirical chi square  39.02  with prob <  0.0018
##
## Fit based upon off diagonal values = 0.97
```

```
rotatedprofile = plot(rotate, row.names(rotate$loadings), cex=1.0)
```




```
#Data frame with prod line column removed
vd_fh_data = fa_fh_data[,-6]

#Rotated PCA with prodline removed
rotate = principal(vd_fh_data, nfactors = 4, rotate = "varimax")
rotate

## Principal Components Analysis
## Call: principal(r = vd_fh_data, nfactors = 4, rotate = "varimax")
## Standardized loadings (pattern matrix) based upon correlation matrix
##               RC1   RC2   RC3   RC4   h2   u2 com
## ProdQual     0.03 -0.01 -0.02  0.89 0.80 0.20 1.0
## Ecom         0.06  0.87  0.05 -0.14 0.78 0.22 1.1
## TechSup      0.02 -0.02  0.94  0.10 0.89 0.11 1.0
## CompRes      0.93  0.10  0.06  0.08 0.89 0.11 1.0
## Advertising  0.16  0.74 -0.08  0.04 0.58 0.42 1.1
## SalesFImage  0.14  0.90  0.08 -0.17 0.86 0.14 1.1
## ComPricing  -0.10  0.23 -0.26 -0.73 0.66 0.34 1.5
## WartyClaim   0.10  0.05  0.93  0.08 0.89 0.11 1.0
## OrdBilling   0.89  0.10  0.09  0.07 0.81 0.19 1.1
## DelSpeed     0.93  0.17  0.00  0.01 0.89 0.11 1.1
##
##                        RC1  RC2  RC3  RC4
## SS loadings           2.59 2.22 1.85 1.41
## Proportion Var        0.26 0.22 0.18 0.14
## Cumulative Var        0.26 0.48 0.67 0.81
```

```
## Proportion Explained  0.32 0.28 0.23 0.17
## Cumulative Proportion 0.32 0.60 0.83 1.00
##
## Mean item complexity =  1.1
## Test of the hypothesis that 4 components are sufficient.
##
## The root mean square of the residuals (RMSR) is  0.06
##  with the empirical chi square  35.71  with prob <  0.00019
##
## Fit based upon off diagonal values = 0.96
```

```
fs = factor.scores(vd_fh_data, f=rotate$loadings,  method = "Harman" )
fs
```

```
## $weights
##                     RC1          RC2           RC3          RC4
## ProdQual    -0.051419420  0.130336157 -0.0825314639  0.696389364
## Ecom        -0.056648414  0.347771887  0.0036772646  0.015730620
## TechSup     -0.032983868 -0.040214111  0.5094485681 -0.016797627
## CompRes      0.405955888 -0.057675645 -0.0179424006  0.013333569
## Advertising -0.016550813  0.166554632 -0.0241552561  0.070921665
## SalesFImage -0.061328766  0.547226619  0.0170950608 -0.010568414
## ComPricing   0.003344616  0.005665778 -0.0098516614 -0.302281814
## WartyClaim  -0.003811491  0.009749789  0.4878339957 -0.025129900
## OrdBilling   0.219331197 -0.033517305  0.0006425309  0.001364191
## DelSpeed     0.421827735 -0.022946340 -0.0391482608 -0.063966796
##
## $r.scores
##             RC1          RC2         RC3         RC4
## RC1 1.000000000  0.039254143 0.008751499  0.02859801
## RC2 0.039254143  1.000000000 0.003696889 -0.06233124
## RC3 0.008751499  0.003696889 1.000000000  0.04207727
## RC4 0.028598008 -0.062331242 0.042077267  1.00000000
##
## $missing
##   [1] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0
##  [36] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0
##  [71] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
##
## $R2
## [1] 0.9483534 0.9051060 0.9406820 0.8418479
```

```
fsd = data.frame(fs$scores)

#Multiple Linear regression analysis
rts = lm(Satisfaction~fsd$RC1+fsd$RC2+fsd$RC3+fsd$RC4, data=fsd)
aov(rts)
```

```
## Call:
##    aov(formula = rts)
##
## Terms:
##                   fsd$RC1  fsd$RC2  fsd$RC3  fsd$RC4 Residuals
```

```
## Sum of Squares  42.13155 26.31675  0.75709 30.43407  40.98814
## Deg. of Freedom        1        1        1        1        95
##
## Residual standard error: 0.6568517
## Estimated effects may be unbalanced
```

```
summary(rts)
```

```
##
## Call:
## lm(formula = Satisfaction ~ fsd$RC1 + fsd$RC2 + fsd$RC3 + fsd$RC4,
##     data = fsd)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.4343 -0.4421  0.0937  0.4581  1.5411
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.91800    0.06569 105.321  < 2e-16 ***
## fsd$RC1      0.64544    0.06946   9.293 5.37e-15 ***
## fsd$RC2      0.60013    0.07209   8.325 6.20e-13 ***
## fsd$RC3      0.06797    0.07010   0.970    0.335
## fsd$RC4      0.65243    0.07768   8.399 4.32e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6569 on 95 degrees of freedom
## Multiple R-squared:  0.7085, Adjusted R-squared:  0.6963
## F-statistic: 57.73 on 4 and 95 DF,  p-value: < 2.2e-16
```

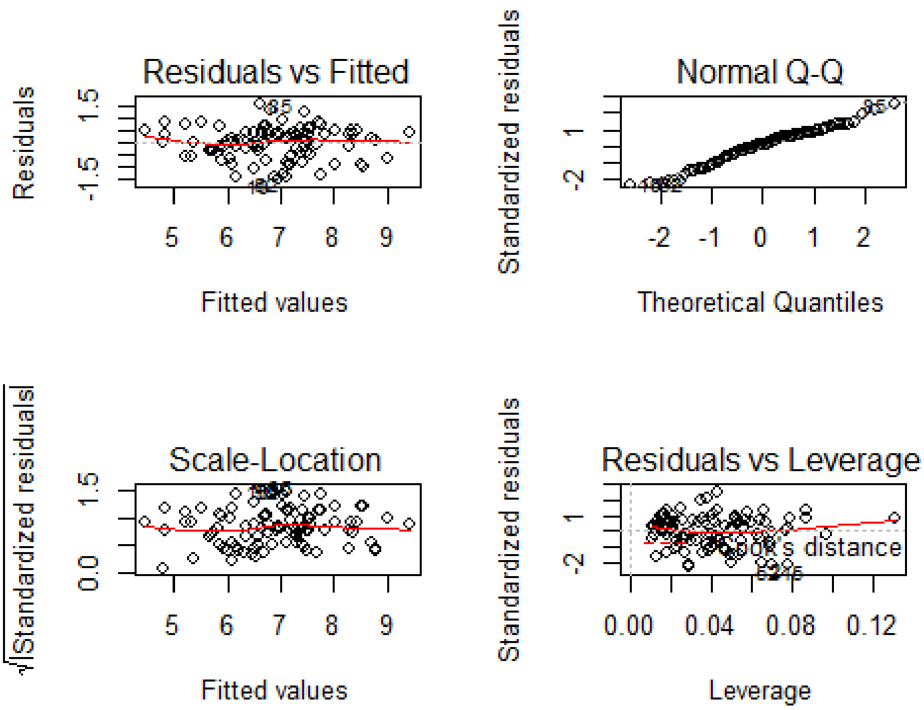
```
rts = lm(Satisfaction~fsd$RC1+fsd$RC2+fsd$RC4, data=fsd)
aov(rts)
```

```
## Call:
##    aov(formula = rts)
##
## Terms:
##                  fsd$RC1  fsd$RC2  fsd$RC4 Residuals
## Sum of Squares  42.13155 26.31675 30.78559  41.39370
## Deg. of Freedom        1        1        1        96
##
## Residual standard error: 0.6566463
## Estimated effects may be unbalanced
```

```
summary(rts)
```

```
##
## Call:
## lm(formula = Satisfaction ~ fsd$RC1 + fsd$RC2 + fsd$RC4, data = fsd)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.46140 -0.39990  0.09529  0.43526  1.59013
##
## Coefficients:
```

```
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.91800    0.06566 105.354  < 2e-16 ***
## fsd$RC1      0.64593    0.06943   9.303 4.68e-15 ***
## fsd$RC2      0.60056    0.07207   8.333 5.57e-13 ***
## fsd$RC4      0.65561    0.07759   8.450 3.15e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6566 on 96 degrees of freedom
## Multiple R-squared:  0.7057, Adjusted R-squared:  0.6965
## F-statistic: 76.71 on 3 and 96 DF,  p-value: < 2.2e-16

#Validation of Assumptions

mean(rts$residuals)

## [1] -2.962989e-17

#saphiro test for checking normality
shapiro.test(rts$residuals)

##
##  Shapiro-Wilk normality test
##
## data:  rts$residuals
## W = 0.97597, p-value = 0.06424

par(mfrow=c(2,2))
plot(rts)
```




```
ncvTest(rts)
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 0.09503834, Df = 1, p = 0.75787

spreadLevelPlot(rts)

##
## Suggested power transformation:  0.580008

vif = 1/(1-0.7057)
print(vif)

## [1] 3.397893

#vif<5, so multicollinearity not exists


#Random Test with all the variables
result = lm(Satisfaction~ProdQual+Ecom+TechSup+CompRes+Advertising+ProdLin
e+SalesFImage+ComPricing+WartyClaim+OrdBilling+DelSpeed, data = fh_data)
aov(result)

## Call:
##    aov(formula = result)
##
## Terms:
##                  ProdQual     Ecom  TechSup  CompRes Advertising ProdLin
e
## Sum of Squares  33.26012 17.50215  0.53038 35.45256     2.21484  0.9642
3
## Deg. of Freedom        1        1        1        1           1
1
##                  SalesFImage ComPricing WartyClaim OrdBilling DelSpeed
## Sum of Squares     21.30244    0.17718    0.08240    1.08747  0.22500
## Deg. of Freedom           1          1          1          1        1
##                  Residuals
## Sum of Squares   27.82884
## Deg. of Freedom        88
##
## Residual standard error: 0.5623494
## Estimated effects may be unbalanced

summary(result)

##
## Call:
## lm(formula = Satisfaction ~ ProdQual + Ecom + TechSup + CompRes +
##     Advertising + ProdLine + SalesFImage + ComPricing + WartyClaim +
##     OrdBilling + DelSpeed, data = fh_data)
##
## 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
```