

Case study 2: Factor Analysis

Oscar Wieland, Pablo Huber

2024-04-07

Preparation

```
rm(list = ls()) #Clean the entire environment  
cat("\014") # clean console
```

```
#setwd("~/Desktop/Master-Business analytics/Spring semester/Advanced Data Driven/Case study/
```

Install required packages

```
#install.packages('rcompanion')
#install.packages('nortest')
#install.packages('corrplot')
#install.packages('olsrr')
#install.packages('dplyr')
#install.packages('pastecs')
#install.packages('REdaS')
#install.packages('psych')
#install.packages('lm.beta')
#install.packages("mice")
#install.packages("naniar")
```

Import required packages

```
library(rcompanion) # Histogram and Normal Curve
library(nortest) # Kolmogorov-Smirnov-Test
library(corrplot) # Correlation matrix plot
library(olsrr) # VIF and Tolerance Values
library(dplyr)
library(pastecs)
library(REdaS) # Bartlett's Test
library(psych)
library(lm.beta)
library(mice) # MCAR plot
library(naniar) # MCAR test
library(VIM) # Visualisation of missing values
```

Context

Quality significantly impact effective corporate performance and is a key driver of increased profitability for companies. It remains a vital force behind corporate success, influencing both customer happiness and operational effectiveness. Thus, companies want to quantify quality in order to take advantage of it. One prominent smartphone manufacturer's product management team, started a project to develop a framework for evaluating smartphone quality from the standpoint of user experience. They want to know in greater depth which aspects of quality account for a major portion of the outcome factors such as willingness to pay premium (WTPP) and repurchase intentions (RI) to have a criterion variable to determine whether their efforts to increase quality are successful or not.

Indeed, there is currently no clear definition or correct measurement of quality. The product management team seeks to define product quality as the consumer's evaluation of a product's overall quality, value, or superiority over alternatives. Unfortunately, in the literature, there is mainly the following quality dimensions, which may not be entirely distinct: aesthetics, durability, ease of use, features, conformance, performance, reliability, serviceability and distinctiveness.

Our goal

We have been contacted by the product management team to identify relevant product quality dimensions from a customer point of view. Qualitative interviews, consisting of 33 questions to measure perceived product quality of 1014 smartphone customers in the US were put at our disposal.

Here is the dataset:

```
survey = read.csv("Data File_Case_Study_Factor Analysis_MD.csv")
head(survey)
```

```
## ownership usage_freq usingSince typicalUse brand cost starRating intro_q qd1
## 1      1      2      4      4      1      1      3      1      3
## 2      1      2      3      3      1      1      4      1      4
## 3      1      2      5      4      1      1      4      1      3
## 4      1      2      4      4      1      2      4      1      5
## 5      1      2      5      4      1      1      4      1      4
## 6      1      2      5      4      1      2      3      1      3
##   qd2 qd3 qd4 qd5 qd6 qd7 qd8 qd9 qd10 qd11 qd12 qd13 qd14 qd15 qd16 qd17 qd18
## 1  4   3   3   4   3   4   3   3   3   3   4   3   3   4   3   3   3
## 2  4   3   6   4   4   4   4   4   3   4   4   4   4   4   4   4   4
## 3  4   3   2   3   3   3   4   3   3   4   5   3   3   3   4   3   3
## 4  5   4   3   5   5   5   5   4   5   5   5   5   4   4   4   5   4
## 5  4   5   2   3   3   4   4   3   4   5   4   4   3   4   3   4   4
## 6  4   3   5   3   3   4   3   3   3   3   4   3   3   4   4   3   3
##   qd19 qd20 qd21 qd22 qd23 qd24 qd25 qd26 qd27 qd28 qd29 qd30 qd31 qd32 qd33
## 1   3   3   3   3   3   4   3   3   4   3   3   3   4   4   3   3
```

```

## 2   4   4   4   3   5   4   4   3   4   3   3   3   4   3   4   3
## 3   3   3   3   4   2   3   3   4   3   4   4   4   4   4   4   3   3
## 4   4   5   5   5   1   4   5   4   5   4   5   5   5   5   5   5   5
## 5   3   4   3   4   2   3   2   4   4   4   4   4   4   4   4   4   4
## 6   3   3   3   4   4   3   3   4   3   3   3   4   3   3   3   3   4
##   intro_b wtp1 wtp2 wtp3 Q01 Q02 bc1 bc2 Q03 ri1 ri2 bc3 bc4 sat3 sat2 Q04 sat1
## 1   1   5   4   5   6   5   5   6   6   5   5   6   6   6   6   6   6   6
## 2   1   4   5   4   5   6   5   5   5   5   6   4   5   6   6   6   5   6
## 3   1   4   4   4   4   4   4   4   4   5   6   4   4   6   5   5   5   6
## 4   1   1   1   1   6   6   2   3   5   2   6   5   5   6   7   7   6   6
## 5   1   4   2   2   6   5   2   3   6   6   5   5   6   6   6   6   6   6
## 6   1   3   3   2   4   4   3   4   5   5   5   4   4   5   5   4   4   5
##   defect1 defectsFaced noOfDefects defectsWarranty fRecall def_sev1 def_sev2
## 1   1           2          NA          NA          NA          NA          NA
## 2   1           2          NA          NA          NA          NA          NA
## 3   1           2          NA          NA          NA          NA          NA
## 4   1           2          NA          NA          NA          NA          NA
## 5   1           1          1           1           2           1           1
## 6   1           2          NA          NA          NA          NA          NA
##   didComplain reactionComplain nextPurchase intro_d gender age education income
## 1   NA          NA          NA          1           2           5           3           17
## 2   NA          NA          NA          1           2           5           3           4
## 3   NA          NA          NA          1           1           6           7           17
## 4   NA          NA          NA          1           1           6           5           2
## 5   2           NA          NA          2           1           2           6           5           2
## 6   NA          NA          NA          1           1           5           4           3
##   brandrec
## 1   1
## 2   1
## 3   1
## 4   1
## 5   1
## 6   1

```

Question 1

To do so, we decided to perform a factor analysis. Factor analysis is a great tool to condense large datasets into a small number of meaningful dimensions that can uncover key aspects of product quality. We will first run a principal axis factoring with varimax rotation. Once we have an appropriate solution, we'll conduct a principal component analysis and compare the results.

Our factor analysis will be performed on the 33 quality questions of the survey. Consequently, we select only the appropriate columns:

```
QI = survey[, paste0("qd", seq(1, 33))]
```

Missing values

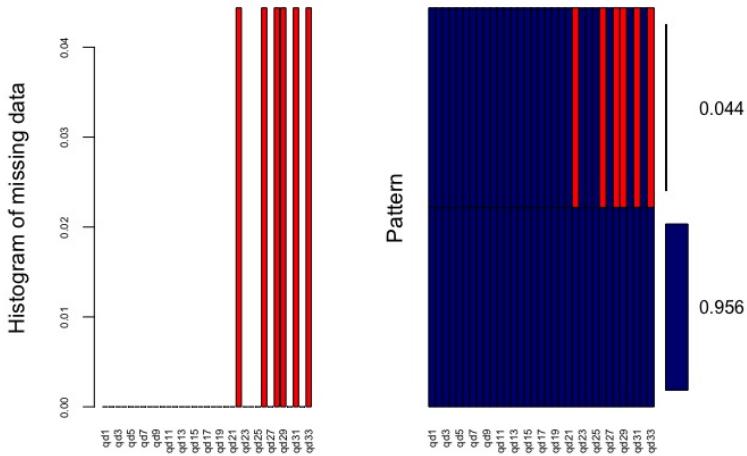
The first thing to do is check if there are missing values as they could produce biased results.

```
list_na = colnames(QI)[apply(QI, 2, anyNA)]
list_na
```

```
## [1] "qd22" "qd26" "qd28" "qd29" "qd31" "qd33"
```

Some columns seem to contain missing values. If the missing values are not missing completely at random (MCAR) or represent a huge proportion of the dataset, it could be a problem. First we will visualize the patterns and then run a test to see if the missing values are MCAR.

```
MissingValues_plot = agrgr(QI, col=c('navyblue', 'red'), numbers=TRUE, cex.axis=.6, gap=5, yl
```



We can see in the right plot that ~96% of the samples are not missing any information and ~4% of the samples are missing values for all the 6 questions. Strangely enough, those questions are not optional as the scale goes from 1 to 5. The reason behind this pattern could be investigated, but as it only affect a small proportion of the dataset, we will keep using the rest of the data for this study.

The MCAR test strangely don't confirm the result of the plot:

- The MCAR test Null hypothesis is: Data is MCAR
- The high p-value suggest that we fail to reject the hypothesis that the data is MCAR. (However as there is only one pattern for missing data this result doesn't make sense.)

```
# MCAR Test
mcar_test(QI)
```

```
## # A tibble: 1 × 4
##   statistic    df  p.value missing.patterns
##       <dbl>  <dbl>     <dbl>          <int>
## 1     31.1     27    0.267             2
```

Removing the rows with missing values shrink the dataset from 1014 rows to 969:

```
QI = na.omit(QI)
dim(QI)[1]
```

```
## [1] 969
```

Now our dataset no longer contains any missing values:

```
list_na = colnames(QI)[apply(QI, 2, anyNA)]
list_na
```

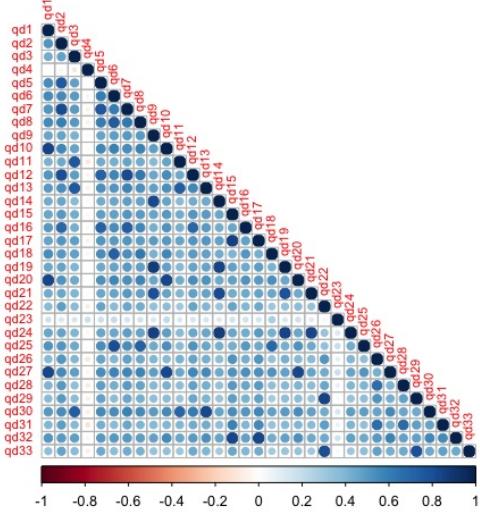
```
## character(0)
```

Correlation among variables

The first step of a factor analysis is to analyze the correlation matrix. It will help us examine the relationships among variables.

Correlation matrix

```
raqMatrix = cor(QI)
corrplot(as.matrix(raqMatrix), type = "lower", tl.cex=0.7)
```



```
# If you want the numbers: corrplot(as.matrix(rqMatrix), type = "lower", tl.cex=0.7, method
```

As we can see, most of the correlation are higher than 0.3 which is a good point. Q4 and Q23 don't have much correlation with other variables. We should keep an eye out for these variables. It is also interesting to note that all the correlations are positive.

Before performing factor analysis, we have to check the suitability of the data and the adequacy of the correlation matrix. This can be done with the Bartlett test, KMO test and the Anti-image correlation.

Bartlett's Test of sphericity

This test assesses whether there are significant correlations among the variables in the dataset. The null hypothesis of the test is that all variables are uncorrelated. Therefore, a small p-value from the test indicates evidence to reject the null hypothesis, suggesting that the variables are indeed correlated.

```
bart_spher(QI)
```

```
##  Bartlett's Test of Sphericity
##
## Call: bart_spher(x = QI)
##
##      X2 = 30857.911
##      df = 528
##  p-value < 2.22e-16
```

We have a small p-value so we can reject the null hypothesis which means some variables are correlated.

KMO

KMO provides information about the proportion of variance in the variables that might be caused by underlying factors.

It is the ratio of the sum of squared correlations to the sum of squared correlations plus the sum of squared partial correlations:

$$KMO = \frac{\text{Sum of Squared Correlations}}{\text{Sum of Squared Correlations} + \text{Sum of Squared Partial Correlations}}$$

This means that a high KMO is associated with a small sum of squared partial correlations. This suggests that a greater proportion of the variance among the variables is shared, rather than being unique to individual variables.

Higher values indicate more suitable data for factor analysis. The criterion is that the KMO should be above 0.6.

```
KMOTEST=KMOS(QI)
KMOTEST$KMO
```

```
## [1] 0.961722
```

We have a value higher than 0.6 which means it's good for factor analysis.

Anti-Image

If a value in the diagonal is below 0.5, the corresponding variable might be removed from factor analysis. The value on the diagonal represent the Measure of Sampling Adequacy (MSA) for each variable. It

indicates how well each variable correlate with all the other variables and thus indicate if they share common factors, the closer to 1 the better (however a extremely high value might indicate that there is too much redundancy in some questions).

```
#Here the values are the diagonal elements for each variable
sort(KMOTEST$MSA)
```

```
##      qd4      qd23      qd22      qd29      qd15      qd33      qd17      qd11
## 0.6422860 0.9025359 0.9273572 0.9348269 0.9463654 0.9502085 0.9526272 0.9553342
##      qd25      qd6      qd3      qd24      qd31      qd19      qd13      qd14
## 0.9554307 0.9562057 0.9570940 0.9572029 0.9595275 0.9596475 0.9604076 0.9606687
##      qd1      qd9      qd28      qd32      qd30      qd26      qd20      qd27
## 0.9611377 0.9614243 0.9620148 0.9632269 0.9651743 0.9658695 0.9660218 0.9662994
##      qd2      qd12     qd18      qd10      qd7      qd21      qd5      qd8
## 0.9665563 0.9692788 0.9694107 0.9700651 0.9700763 0.9745578 0.9794715 0.9809143
##      qd16
## 0.9818514
```

All variables are higher than 0.5. No need to remove some variables. We can still notice that q4 has a small value compare to the others so we should keep it in mind.

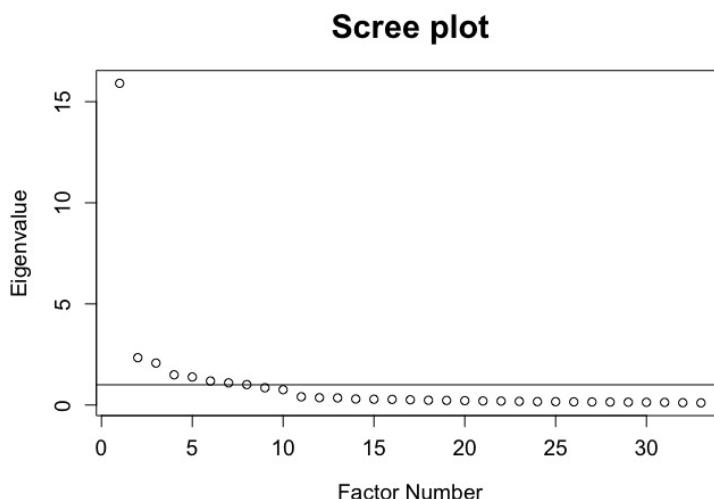
Factor Analysis

Before performing the factor analysis, we should carefully choose the number of factors. To do so, we use the kaiser criterion and the scree-test.

Scree-test and kaiser criterion

```
FA0 <- psych::fa(QI, rotate="varimax", scores=TRUE)

plot(FA0$e.values, xlab="Factor Number", ylab="Eigenvalue", main="Scree plot", cex.lab=1.2, cex.a
```



```
## integer(0)
```

```
EigenValue=FA0$e.values

Variance=EigenValue/ncol(QI)*100

SumVariance=cumsum(EigenValue/ncol(QI))

Total_Variance_Explained=cbind(EigenValue=EigenValue[EigenValue>0], Variance=Variance[Varianc

Total_Variance_Explained
```

```
##      EigenValue  Variance Total_Variance
## [1,] 15.9082244 48.2067407 0.4820674
## [2,] 2.3398002  7.0903037 0.5529704
## [3,] 2.0741379  6.2852665 0.6158231
## [4,] 1.4922324  4.5219164 0.6610423
## [5,] 1.3853335  4.1979804 0.7030221
## [6,] 1.1868699  3.5965755 0.7389878
## [7,] 1.0936900  3.3142121 0.7721300
## [8,] 1.0096776  3.0596291 0.8027262
## [9,] 0.8530507  2.5850023 0.8285763
```

```

## [10,] 0.7579386 2.2967837 0.8515441
## [11,] 0.4066927 1.2324022 0.8638681
## [12,] 0.3605823 1.0926736 0.8747949
## [13,] 0.3509690 1.0635424 0.8854303
## [14,] 0.2970630 0.9001911 0.8944322
## [15,] 0.2858756 0.8662897 0.9030951
## [16,] 0.2773086 0.8403290 0.9114984
## [17,] 0.2574310 0.7800940 0.9192993
## [18,] 0.2361584 0.7156314 0.9264556
## [19,] 0.2273536 0.6889502 0.9333451
## [20,] 0.2155112 0.6530643 0.9398758
## [21,] 0.2007128 0.6082205 0.9459580
## [22,] 0.1929568 0.5847176 0.9518052
## [23,] 0.1801325 0.5458562 0.9572637
## [24,] 0.1682817 0.5099446 0.9623632
## [25,] 0.1640197 0.4970294 0.9673335
## [26,] 0.1555231 0.4712820 0.9720463
## [27,] 0.1520528 0.4607661 0.9766539
## [28,] 0.1461674 0.4429317 0.9810833
## [29,] 0.1384577 0.4195689 0.9852790
## [30,] 0.1338676 0.4056595 0.9893355
## [31,] 0.1288620 0.3904910 0.9932405
## [32,] 0.1156120 0.3503395 0.9967439
## [33,] 0.1074529 0.3256148 1.0000000

```

The scree-plot is a graphical representation of eigen values by ordering them according to their size. To choose the number of factors, the rule of thumb is to choose the number before the elbow of the curve. Unfortunately, this criterion is not very clear and different people may have different choice of the elbow. Thus, it is important to include the kaiser criterion. This criterion says that only factors with eigen value higher than 1 should be extracted. Using these two criteria and knowing the lack of precision, we decide to test different situations around the kaiser criterion. We try the factor analysis with 7 and 8 factors. (little elbow after 8)

The first component has an eigenvalue of 15.9 indicating that it can explain around 48% of the overall variance. The 7 derived factors explain a total variance of 77% and the 8 derived factors 80%.

Now that we have decided the number of factors, we are ready to perform the factor analysis.

Principal axis factoring with 7 factors and Varimax rotation

```

FA7 = fa(QI, nfactors=7, rotate="varimax")

print(FA7$loadings, cutoff=0.3, sort=TRUE)

```

```

##
## Loadings:
##      MR2     MR1     MR5     MR3     MR7     MR6     MR4
## qd9   0.828
## qd14  0.850
## qd19  0.832
## qd21  0.786
## qd24  0.860
## qd2    0.730
## qd5   0.639      0.308
## qd7   0.734
## qd12  0.698
## qd16  0.613
## qd1    0.786
## qd10  0.744
## qd20  0.766
## qd27  0.746
## qd22      0.841
## qd29      0.816
## qd33      0.788
## qd6    0.751
## qd8   0.341      0.567
## qd18      0.643
## qd25      0.767
## qd15      0.771
## qd17      0.740
## qd32      0.716
## qd3      0.750
## qd11      0.740
## qd13      0.725
## qd30      0.665
## qd4
## qd23
## qd26      0.408      0.414
## qd28      0.318      0.357      0.395
## qd31      0.381      0.497
## 

```

```

##          MR2    MR1    MR5    MR3    MR7    MR6    MR4
## SS loadings   4.588  3.609  3.352  3.181  3.060  3.038  2.943
## Proportion Var 0.139  0.109  0.102  0.096  0.093  0.092  0.089
## Cumulative Var 0.139  0.248  0.350  0.446  0.539  0.631  0.720

```

Principal axis factoring with 8 factors using Varimax rotation

```

FA8 = fa(QI, nfactors=8, rotate="varimax")

print(FA8$loadings, cutoff=0.3, sort=TRUE)

```

```

## 
## Loadings:
##      MR2    MR1    MR5    MR7    MR4    MR3    MR6    MR8
## qd9   0.829
## qd14  0.852
## qd19  0.834
## qd21  0.787
## qd24  0.862
## qd2   0.725
## qd5   0.646      0.301
## qd7   0.737
## qd12  0.710
## qd16  0.616
## qd1   0.794
## qd10  0.745
## qd20  0.769
## qd27  0.742
## qd6   0.748
## qd8   0.334      0.567
## qd18  0.636
## qd25  0.768
## qd3   0.752
## qd11  0.734
## qd13  0.724
## qd30  0.663
## qd22  0.828
## qd29  0.805
## qd33  0.773
## qd15  0.761
## qd17  0.743
## qd32  0.720
## qd26  0.616
## qd28  0.659
## qd31  0.309  0.662
## qd4
## qd23
## 
##      MR2    MR1    MR5    MR7    MR4    MR3    MR6    MR8
## SS loadings   4.618  3.584  3.368  2.984  2.904  2.700  2.417  1.922
## Proportion Var 0.140  0.109  0.102  0.090  0.088  0.082  0.073  0.058
## Cumulative Var 0.140  0.249  0.351  0.441  0.529  0.611  0.684  0.742

```

As we can see, q4 and q23 loadings don't appear (smaller than 0.3). It makes sense as we saw in our previous step that qd4 and qd23 have really low correlation. Thus, we decide to delete them from our dataset.

```

QI$qd4 = NULL
QI$qd23 = NULL
head(QI, n=1)

```

```

##   qd1  qd2  qd3  qd5  qd6  qd7  qd8  qd9  qd10  qd11  qd12  qd13  qd14  qd15  qd16  qd17  qd18
## 1   3    4    3    4    3    4    3    3    3    3    3    3    4    3    3    4    3    3
##   qd19  qd20  qd21  qd22  qd24  qd25  qd26  qd27  qd28  qd29  qd30  qd31  qd32  qd33
## 1   3    3    3    3    3    3    4    3    3    3    4    4    3    3

```

We have to reperform the factor analysis without q4 and q23.

Principal axis factoring with 7 factors using Varimax rotation

```

FA7 = fa(QI, nfactors=7, rotate="varimax")

print(FA7$loadings, cutoff=0.3, sort=TRUE)

```

```

## 
## Loadings:
##      MR2    MR1    MR5    MR6    MR3    MR4    MR7
## qd9   0.826

```

```

## qd14 0.849
## qd19 0.830
## qd21 0.784
## qd24 0.859
## qd2      0.731
## qd5      0.643
## qd7      0.737
## qd12     0.701
## qd16     0.617
## qd1      0.792
## qd10     0.750
## qd20     0.770
## qd27     0.749
## qd15     0.776
## qd17     0.748
## qd31     0.511 0.370
## qd32     0.722
## qd22     0.843
## qd29     0.818
## qd33     0.788
## qd3      0.758
## qd11     0.745
## qd13     0.738
## qd30     0.675
## qd6      0.742
## qd8      0.346      0.556
## qd18     0.601
## qd25     0.761
## qd26     0.427 0.400
## qd28     0.318      0.409 0.346
##
##          MR2   MR1   MR5   MR6   MR3   MR4   MR7
## SS loadings  4.537 3.653 3.393 3.194 3.127 3.091 2.678
## Proportion Var 0.146 0.118 0.109 0.103 0.101 0.100 0.086
## Cumulative Var 0.146 0.264 0.374 0.477 0.578 0.677 0.764

```

We can see that qd26, qd28 and qd31 are loading low and on multiple factors. This means they don't have a strong association with either of those factors. It should be interesting to see if something can change by adding a new factor.

Principal axis factoring with 8 factors using Varimax rotation

```

FA8 = fa(QI, nfactors=8, rotate="varimax")

print(FA8$loadings, cutoff=0.3, sort=TRUE)

```

```

##
## Loadings:
##          MR2   MR1   MR5   MR6   MR3   MR4   MR8
## qd9      0.827
## qd14     0.851
## qd19     0.832
## qd21     0.785
## qd24     0.860
## qd2      0.723
## qd5      0.645
## qd7      0.738
## qd12     0.711
## qd16     0.616
## qd1      0.797
## qd10     0.749
## qd20     0.772
## qd27     0.747
## qd3      0.759
## qd11     0.749
## qd13     0.739
## qd30     0.677
## qd22     0.838
## qd29     0.813
## qd33     0.779
## qd6      0.741
## qd8      0.341      0.558
## qd18     0.601
## qd25     0.763
## qd15     0.750
## qd17     0.744
## qd32     0.715
## qd26     0.619
## qd28     0.673
## qd31     0.660
##
```

```

##                  MR2      MR1      MR5      MR4      MR3      MR7      MR6      MR8
## SS loadings     4.569   3.577   3.404   3.121   2.781   2.675   2.324   1.935
## Proportion Var  0.147   0.115   0.110   0.101   0.090   0.086   0.075   0.062
## Cumulative Var 0.147   0.263   0.373   0.473   0.563   0.649   0.724   0.787

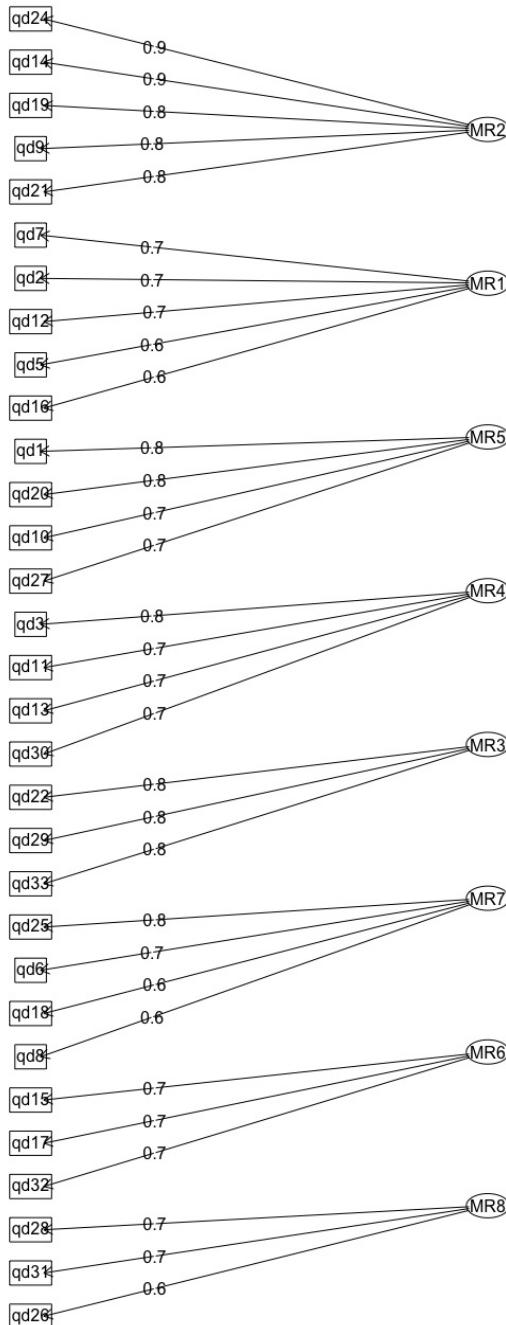
```

With 8 factors it's better. For the Total Variance Explained, the 8-factor model explains more total variance (78.7%) than the 7-factor (76.4%) model. Regarding Interpretability, there are less low loadings and qd26, qd28, qd31 load only on 1 factor now.

Thus, using 8 factors is preferred.

```
fa.diagram(FA8, main="Principal axis factoring with 8 factors")
```

Principal axis factoring with 8 factors



Interpretation

Now that we have identified the factors, it's crucial to understand what these factors represent. Therefore, we need to examine the items that constitute them and interpret the factors accordingly. We will help ourselves from the literature review on quality dimensions.

MR1 (qd2,5,7,8,12,16): Performance

MR2 (qd9,14,19,21,24): Serviceability (Customer service)

MR3 (qd22,29,33): Reliability/Flawlessness (defects)

- MR4 (qd3,11,13,30): Ease of Use (user experience)
 MR5 (qd1,10,20,27): Aesthetics/Appearance (attractiveness/look)
 MR6 (qd15,17,32): Conformance (materials)
 MR7 (qd6,8,18,25): Features/Versatility (extra features)
 MR8 (qd26,28,31): Durability

PCA

We will perform a principal component analysis because it is always interesting to employ different extraction methods and assess how similar or different the factor structures are. Then, we can choose which method seems more interpretative.

```
PCA = psych::principal(QI,
                       rotate="varimax",
                       nfactors=8,
                       scores=TRUE)

PCA_communalities=data.frame(sort(PCA$communality))
PCA_communalities
```

```
##      sort.PCA.communality.
## qd16          0.7281547
## qd18          0.7417442
## qd8           0.7445806
## qd26          0.7696596
## qd5            0.7781026
## qd31          0.7916366
## qd28          0.8042188
## qd30          0.8142034
## qd11          0.8172090
## qd12          0.8230745
## qd21          0.8252227
## qd3            0.8320578
## qd2            0.8362610
## qd13          0.8401490
## qd25          0.8514445
## qd7            0.8541838
## qd33          0.8570641
## qd6            0.8581700
## qd9            0.8717486
## qd10           0.8732182
## qd19           0.8739140
## qd27           0.8771753
## qd20           0.8792219
## qd29           0.8799137
## qd14           0.8809326
## qd32           0.8882966
## qd1            0.8907906
## qd22           0.8911649
## qd24           0.8980463
## qd17           0.9009975
## qd15           0.9012585
```

We have high communalities which is good. (even better than with PAF)

```
EigenValue=PCA$values

Variance=EigenValue/ncol(QI)*100

SumVariance=cumsum(EigenValue/ncol(QI))

Total_Variance_Explained=cbind(EigenValue=EigenValue[EigenValue>0], Variance=Variance[Varianc

Total_Variance_Explained
```

	EigenValue	Variance	Total_Variance
[1,]	15.8598680	51.1608646	0.5116086
[2,]	2.3217762	7.4896008	0.5865047
[3,]	2.0258331	6.5349453	0.6518541
[4,]	1.4781631	4.7682680	0.6995368
[5,]	1.3712708	4.4234540	0.7437713
[6,]	1.1343846	3.6593051	0.7803644
[7,]	1.0135713	3.2695850	0.8130602
[8,]	0.8689483	2.8030591	0.8410908
[9,]	0.4176516	1.3472632	0.8545635
[10,]	0.3618149	1.1671450	0.8662349
[11,]	0.3511069	1.1326029	0.8775609
[12,]	0.2984164	0.9626335	0.8871873
[13,]	0.2897893	0.9348042	0.8965353

```

## [14,] 0.2773548 0.8946931 0.9054822
## [15,] 0.2574651 0.8305324 0.9137876
## [16,] 0.2371933 0.7651396 0.9214390
## [17,] 0.2283675 0.7366692 0.9288057
## [18,] 0.2159573 0.6966366 0.9357720
## [19,] 0.2020993 0.6519334 0.9422914
## [20,] 0.1931866 0.6231825 0.9485232
## [21,] 0.1811551 0.5843713 0.9543669
## [22,] 0.1692323 0.5459106 0.9598260
## [23,] 0.1641484 0.5295110 0.9651211
## [24,] 0.1556840 0.5022064 0.9701432
## [25,] 0.1522619 0.4911674 0.9750548
## [26,] 0.1469643 0.4740784 0.9797956
## [27,] 0.1389878 0.4483478 0.9842791
## [28,] 0.1348376 0.4349599 0.9886287
## [29,] 0.1291215 0.4165208 0.9927939
## [30,] 0.1158191 0.3736102 0.9965300
## [31,] 0.1075695 0.3469985 1.0000000

```

With 8 factors, we can explain 84% of the overall variance. Even with 7 factors you are above 80%. With principal axis factoring, it was only 78%. Thus, PCA seems to work well. Remember, with PCA, all variance in observed variables can be completely accounted for by the extracted factors.

```
print(PCA$loadings, cutoff=0.3, sort=TRUE)
```

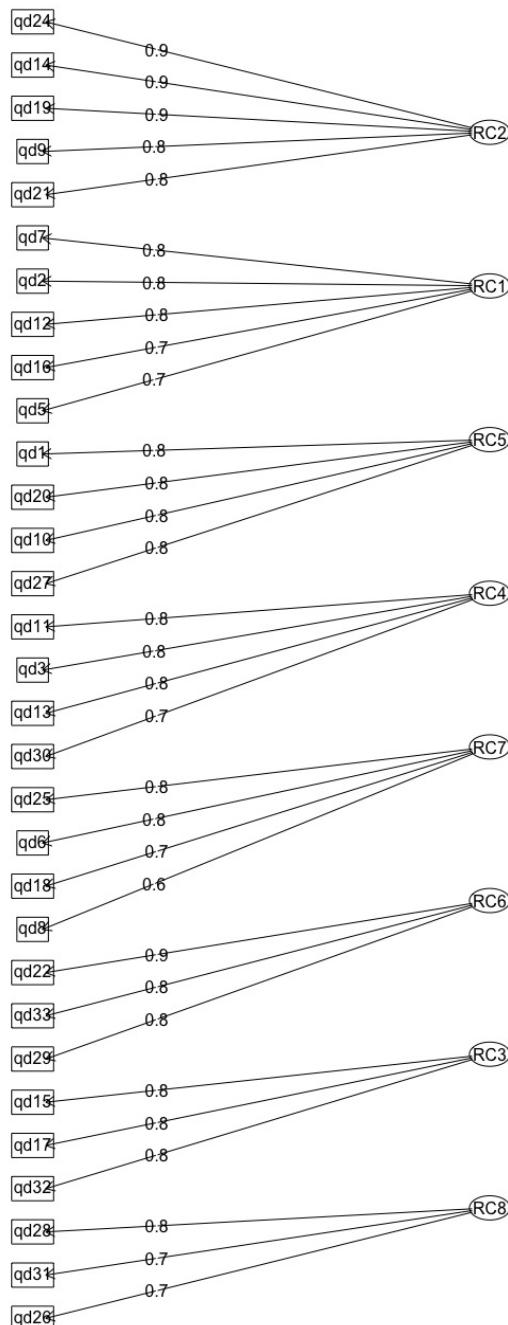
```

##
## Loadings:
##      RC2    RC1    RC5    RC4    RC7    RC6    RC3    RC8
## qd9  0.847
## qd14 0.868
## qd19 0.851
## qd21 0.823
## qd24 0.869
## qd2      0.754
## qd5      0.704
## qd7      0.758
## qd12     0.750
## qd16     0.708
## qd1      0.822
## qd10     0.778
## qd20     0.803
## qd27     0.774
## qd3      0.807
## qd11     0.819
## qd13     0.763
## qd30     0.706
## qd6      0.762
## qd8      0.339   0.626
## qd18     0.715
## qd25     0.806
## qd22     0.854
## qd29     0.841
## qd33     0.842
## qd15     0.787
## qd17     0.780
## qd32     0.765
## qd26     0.727
## qd28     0.773
## qd31     0.738
##
##      RC2    RC1    RC5    RC4    RC7    RC6    RC3    RC8
## SS loadings  4.673 3.908 3.492 3.297 3.020 2.856 2.492 2.337
## Proportion Var 0.151 0.126 0.113 0.106 0.097 0.092 0.080 0.075
## Cumulative Var 0.151 0.277 0.389 0.496 0.593 0.685 0.766 0.841

```

```
fa.diagram(PCA$loadings, main="Principal component analysis with 8 factors")
```

Principal component analysis with 8 factors



Comparing PCA and Principal axis factoring with 8 factors

PCA:

	RC2	RC1	RC5	RC4	RC7	RC6	RC3	RC8
SS loadings	4.673	3.908	3.492	3.297	3.020	2.856	2.492	2.337
Proportion Var	0.151	0.126	0.113	0.106	0.097	0.092	0.080	0.075
Cumulative Var	0.151	0.277	0.389	0.496	0.593	0.685	0.766	0.841

Principal axis factoring

	MR2	MR1	MR5	MR4	MR3	MR7	MR6	MR8
SS loadings	4.569	3.577	3.404	3.121	2.781	2.675	2.324	1.935
Proportion Var	0.147	0.115	0.110	0.101	0.090	0.086	0.075	0.062
Cumulative Var	0.147	0.263	0.373	0.473	0.563	0.649	0.724	0.787

Principal axis factoring and PCA groups the items identically. The loadings seem a bit better with PCA.

Almost all loadings are higher than 0.7. Also the Sum of Squares Loadings, which indicate the amount of variance in the original variables explained by the factor, are better. Higher SS loadings for a factor indicate that the factor explains more variance in the data, so it's better.

We can also see that with PCA the fifth, sixth and seventh factors (regarding the SS loadings) are different than those that we obtain with the Principal axis factoring.

Regarding eigenvalues:

Eigenvalues themselves do not directly indicate relevance for purchase decisions or any behavior. They simply represent the amount of variance explained by each factor in a statistical analysis. Therefore, while a higher eigenvalue might suggest a stronger influence of a particular factor, it doesn't necessarily mean that the corresponding dimension is more relevant for explaining behavior, such as repurchase behavior.

Question 2

For now, we only used methods with "varimax". This means we used orthogonal rotation method, meaning it forces the factors to be uncorrelated with each other. This is useful when the researcher assumes that factors are independent of each other. However, in our case, the factors may be dependent because they represent the same concept, Quality. Therefore, we will try to use "promax" which is an oblique rotation method, allowing factors to be correlated with each other. We will use the same number of factors as before (8).

```
PA8_ob = principal(
  QI,
  rotate="promax",
  nfactors=8,
  scores=TRUE)
PA8_ob
```

```
## Principal Components Analysis
## Call: principal(r = QI, nfactors = 8, rotate = "promax", scores = TRUE)
## Standardized loadings (pattern matrix) based upon correlation matrix
##          RC2   RC1   RC5   RC4   RC7   RC6   RC3   RC8   h2   u2 com
## qd1    0.00 -0.03  0.98 -0.04  0.02  0.01 -0.06  0.04  0.89  0.109 1.0
## qd2    0.00  0.92  0.00 -0.02 -0.06  0.03 -0.01  0.07  0.84  0.164 1.0
## qd3   -0.02 -0.06 -0.05  0.95  0.10  0.04 -0.02 -0.05  0.83  0.168 1.0
## qd5   -0.02  0.83  0.02 -0.02  0.08  0.03  0.01 -0.01  0.78  0.222 1.0
## qd6   -0.01  0.02  0.00  0.01  0.89 -0.01  0.05 -0.01  0.86  0.142 1.0
## qd7    0.04  0.92 -0.04  0.02  0.01 -0.03  0.00  0.00  0.85  0.146 1.0
## qd8    0.02  0.14  0.02  0.03  0.67  0.06 -0.02  0.04  0.74  0.255 1.1
## qd9    0.92 -0.02  0.01  0.01  0.01  0.02  0.00 -0.01  0.87  0.128 1.0
## qd10   0.02  0.03  0.89  0.02  0.03 -0.04  0.02 -0.01  0.87  0.127 1.0
## qd11   -0.02 -0.04 -0.01  0.98 -0.05 -0.08 -0.03  0.08  0.82  0.183 1.0
## qd12   -0.01  0.92  0.00  0.04 -0.04  0.02  0.02 -0.04  0.82  0.177 1.0
## qd13   0.02  0.09  0.04  0.86 -0.04  0.01  0.00 -0.02  0.84  0.160 1.0
## qd14   0.96  0.01 -0.02 -0.02  0.01  0.00  0.02 -0.04  0.88  0.119 1.0
## qd15   0.00  0.03 -0.04 -0.04  0.01  0.01  0.94  0.04  0.90  0.099 1.0
## qd16   0.00  0.86  0.04 -0.04  0.02 -0.04  0.01 -0.01  0.73  0.272 1.0
## qd17   0.03  0.03 -0.05  0.01  0.02 -0.02  0.93  0.00  0.90  0.099 1.0
## qd18   0.00 -0.02  0.06  0.05  0.84 -0.06 -0.02 -0.01  0.74  0.258 1.0
## qd19   0.93  0.00  0.01  0.00 -0.02  0.01  0.00  0.02  0.87  0.126 1.0
## qd20   0.00  0.01  0.94  0.01  0.01 -0.03 -0.01  0.01  0.88  0.121 1.0
## qd21   0.89  0.03 -0.01  0.02  0.03 -0.01 -0.04  0.00  0.83  0.175 1.0
## qd22   0.01  0.01  0.02 -0.02  0.00  0.95 -0.03  0.01  0.89  0.109 1.0
## qd24   0.95 -0.01  0.02 -0.01 -0.02 -0.02  0.01  0.03  0.90  0.102 1.0
## qd25   0.00 -0.05 -0.03 -0.05  0.99  0.03  0.00  0.00  0.85  0.149 1.0
## qd26   0.03  0.02 -0.03  0.04 -0.02  0.07 -0.02  0.83  0.77  0.230 1.0
## qd27   0.00  0.02  0.89  0.02 -0.04  0.08  0.06 -0.04  0.88  0.123 1.0
## qd28   -0.02  0.07  0.01 -0.04  0.07 -0.05 -0.08  0.91  0.80  0.196 1.0
## qd29   0.01  0.02  0.00 -0.02 -0.02  0.92  0.03  0.01  0.88  0.120 1.0
## qd30   0.04  0.08  0.03  0.76 -0.01  0.05  0.06 -0.01  0.81  0.186 1.1
## qd31   0.00 -0.08  0.03  0.02 -0.05  0.01  0.15  0.83  0.79  0.208 1.1
## qd32   -0.01 -0.03  0.09  0.02  0.01  0.00  0.91 -0.02  0.89  0.112 1.0
## qd33   -0.02 -0.02 -0.02  0.03  0.03  0.94 -0.01 -0.01  0.86  0.143 1.0
##
##          RC2   RC1   RC5   RC4   RC7   RC6   RC3   RC8
## SS loadings 4.38 4.15 3.54 3.23 3.07 2.69 2.71 2.32
## Proportion Var 0.14 0.13 0.11 0.10 0.10 0.09 0.09 0.07
## Cumulative Var 0.14 0.27 0.39 0.49 0.59 0.68 0.77 0.84
## Proportion Explained 0.17 0.16 0.14 0.12 0.12 0.10 0.10 0.09
## Cumulative Proportion 0.17 0.33 0.46 0.59 0.70 0.81 0.91 1.00
##
## With component correlations of
##          RC2   RC1   RC5   RC4   RC7   RC6   RC3   RC8
## RC2  1.00 0.55 0.52 0.51 0.53 0.38 0.52 0.45
## RC1  0.55 1.00 0.63 0.61 0.68 0.53 0.57 0.57
## RC5  0.52 0.63 1.00 0.57 0.64 0.43 0.57 0.47
## RC4  0.51 0.61 0.57 1.00 0.57 0.45 0.57 0.50
## RC7  0.53 0.68 0.64 0.57 1.00 0.42 0.51 0.46
```

```

## RC6 0.38 0.53 0.43 0.45 0.42 1.00 0.51 0.57
## RC3 0.52 0.57 0.57 0.57 0.51 0.51 1.00 0.60
## RC8 0.45 0.57 0.47 0.50 0.46 0.57 0.60 1.00
##
## Mean item complexity = 1
## Test of the hypothesis that 8 components are sufficient.
##
## The root mean square of the residuals (RMSR) is 0.02
## with the empirical chi square 346.58 with prob < 2e-05
##
## Fit based upon off diagonal values = 1

```

Here, 84% of the total variance is explained (Cumulative var). It is the same as PCA with varimax rotation.

Let's take a look at the factors and their items:

```
print(PA8_ob$loadings,cutoff=0.3,sort=TRUE)
```

```

##
## Loadings:
##      RC2     RC1     RC5     RC4     RC7     RC6     RC3     RC8
## qd9    0.921
## qd14   0.956
## qd19   0.927
## qd21   0.892
## qd24   0.951
## qd2     0.920
## qd5    0.826
## qd7    0.918
## qd12   0.916
## qd16   0.863
## qd1     0.979
## qd10   0.891
## qd20   0.939
## qd27   0.889
## qd3    0.949
## qd11   0.981
## qd13   0.857
## qd30   0.761
## qd6    0.891
## qd8    0.673
## qd18   0.838
## qd25   0.988
## qd22   0.946
## qd29   0.922
## qd33   0.937
## qd15   0.943
## qd17   0.934
## qd32   0.907
## qd26   0.826
## qd28   0.912
## qd31   0.833
##
##      RC2     RC1     RC5     RC4     RC7     RC6     RC3     RC8
## SS loadings  4.332 4.014 3.453 3.197 2.967 2.665 2.635 2.237
## Proportion Var 0.140 0.129 0.111 0.103 0.096 0.086 0.085 0.072
## Cumulative Var 0.140 0.269 0.381 0.484 0.579 0.665 0.750 0.823

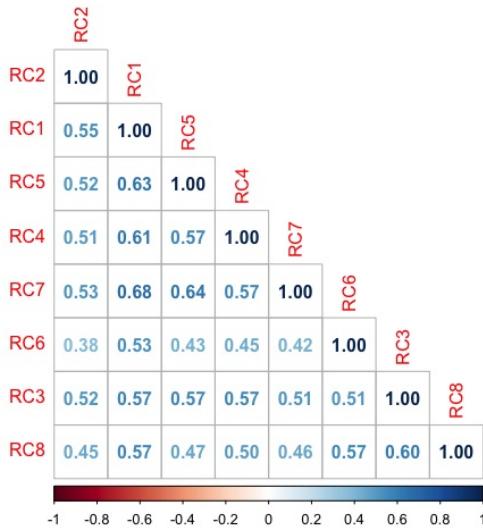
```

We no longer encounter the issue with qd8. It now loads exclusively on one factor. Furthermore, almost all loadings are higher than 0.7 and are better than with the "varimax" rotation.

We can take a look at the correlation between the factors to understand the patterns.

Correlation between factors

```
corrplot(PA8_ob$r.scores,type = "lower", method="number")
```



A positive correlation indicates a relationship between the factors. Thankfully, our correlations are not overly high, but they still represent favorable values. Considering that strong correlations between components may indicate that they are measuring related dimensions, our results are interesting because we want to capture different aspects of the same idea (Quality).

The highest correlation between factors is between factor 7 (Features/Versatility) and factor 1 (Performance) and the correlation is of 68%. This suggests that as Factor 1 increases, Factor 7 tends to increase as well, and vice versa. This also suggests that Factor 7 and Factor 1 are measuring quite similar dimensions. This is not so surprising as performance may depend of the features.

If we want more information for each customer, we can look at the factor scores. Factor value reflects the characteristic of a certain object (case) on a certain factor. They are derived based on the factor loadings obtained during the factor analysis process. If the score is positive, in comparison with all other objects, the object score is above average on a factor. The opposite for a negative score.

Factor scores

```
#PA8_ob$scores
Fscores = as.data.frame(PA8_ob$scores)
dim(Fscores)

## [1] 969    8

colnames(Fscores) = c("Serviceability", "Performance", "Aesthetics", "Ease_of_Use", "Features", "Reliability", "Conformance", "Durability")
head(Fscores, n=7)

##   Serviceability Performance Aesthetics Ease_of_Use Features Reliability Conformance Durability
## 1 -1.0872261  -0.4752005 -1.4514066 -0.7325736 -0.9074135 -1.074354896
## 2  0.1590142  -0.1463224 -0.1067707 -0.7318198  0.3954958 -1.087853689
## 3 -1.1074724  -0.3674655 -1.4479865 -0.7230417 -0.6919136 -0.339256218
## 4  0.3886359  0.9396617  1.2173171  0.9785910  1.2807178  1.185984442
## 5 -1.1090268  -0.6454847 -0.1207430  0.7028242 -0.6890459  0.004331103
## 6 -1.1040097  -0.4005460 -1.5027112 -1.3892332 -0.9045229  0.024019855
## 7  0.1547408  0.9170508  0.1633157  0.6026066 -0.9047279 -0.704853317
```

The items are represented by the same factors as before. However, the factors importance have change. (see Proportion Explained)

- RC1 (qd2,5,7,12,16): Performance
- RC2 (qd9,14,19,21,24): Serviceability (Customer service)
- RC3 (qd15,17,32): Conformance (materials)
- RC4 (qd3,11,13,30): Ease of Use (user experience)
- RC5 (qd1,10,20,27): Aesthetics/Appearance (attractiveness/look)
- RC6 (qd22,29,33): Reliability/Flawlessness (defects)
- RC7 (qd6,8,18,25): Features/Versatility (extra features)
- RC8 (qd26,28,31): Durability

Mean scores for willingness to pay and repurchase intention

We want to know the overall willingness to pay and intention to repurchase for each customer. Thus, we take the average of the questions regarding willingness to pay and the average of the questions regarding repurchase intention.

```
QI2 = cbind(survey[, paste0("qd", seq(1, 33))], survey[, paste0("wtp", seq(1, 3))], survey[, p
colnames(QI2)[ncol(QI2)] = "brandrec"
QI2 = na.omit(QI2)

WTP = QI2[, paste0("wtp", seq(1, 3))]
RI = QI2[, paste0("ri", seq(1, 2))]
WTP_mean = as.data.frame(rowMeans(WTP))
colnames(WTP_mean)[ncol(WTP_mean)] = "WTP"
RI_mean = as.data.frame(rowMeans(RI))
colnames(RI_mean)[ncol(RI_mean)] = "RI"

Mean_scores = cbind(WTP_mean, RI_mean)
head(Mean_scores)
```

```
##          WTP   RI
## 1 4.666667 6.0
## 2 4.333333 5.5
## 3 4.000000 5.5
## 4 1.000000 4.0
## 5 2.666667 5.5
## 6 2.666667 5.0
```

Regression analysis

We can run a regression with the factor scores of the quality dimensions as independent and both the mean score of willingness to pay premium and repurchase intention as dependent variables to understand what drives WTP and RI.

WTP

```
regr_WTP = lm(rowMeans(WTP) ~ Fscores$Serviceability + Fscores$Performance + Fscores$Aesthet
summary(lm.beta(regr_WTP))
```

```
##
## Call:
## lm(formula = rowMeans(WTP) ~ Fscores$Serviceability + Fscores$Performance +
##     Fscores$Aesthetics + Fscores$Ease_of_Use + Fscores$Features +
##     Fscores$Reliability + Fscores$Conformance + Fscores$Durability,
##     data = Fscores)
##
## Residuals:
##      Min        1Q    Median        3Q       Max
## -4.6189 -0.7313  0.1069  0.8480  2.8069
##
## Coefficients:
##                               Estimate Standardized Std. Error t value Pr(>|t|)
## (Intercept)             4.57585           NA  0.03682 124.291 < 2e-16 ***
## Fscores$Serviceability  0.16234         0.11286  0.04802   3.381 0.000751 ***
## Fscores$Performance     0.17457         0.12137  0.05990   2.915 0.003645 **
## Fscores$Aesthetics      0.09889         0.06875  0.05410   1.828 0.067861 .
## Fscores$Ease_of_Use     0.26824         0.18649  0.05179   5.179 2.71e-07 ***
## Fscores$Features        0.20669         0.14370  0.05516   3.747 0.000190 ***
## Fscores$Reliability     -0.02112        -0.01468  0.04792  -0.441 0.659579
## Fscores$Conformance     0.16404         0.11404  0.05372   3.053 0.002325 **
## Fscores$Durability      0.03710         0.02579  0.05168   0.718 0.472999
## ---
## Signif. codes:  0 '****' 0.001 '***' 0.01 '**' 0.05 '*' 0.1 '.' 1
##
## Residual standard error: 1.146 on 960 degrees of freedom
## Multiple R-squared:  0.3704, Adjusted R-squared:  0.3652
## F-statistic: 70.6 on 8 and 960 DF,  p-value: < 2.2e-16
```

For the willingness to pay, the statistically significant factors are serviceability, performance, ease of use, features and conformance. Depending on the level of statistical significance, we can add aesthetics. The most important factor for the willingness to pay is the ease of use followed by features, performance, conformance and serviceability. From a managerial perspective, we should put more emphasis on the ease of use and the features if we want to encourage the willingness to pay of the customer.

RI

```

regr_RI = lm(rowMeans(RI) ~ Fscores$Serviceability + Fscores$Performance + Fscores$Aesthetic
summary(lm.beta(regr_RI))

## 
## Call:
## lm(formula = rowMeans(RI) ~ Fscores$Serviceability + Fscores$Performance +
##     Fscores$Aesthetics + Fscores$Ease_of_Use + Fscores$Features +
##     Fscores$Reliability + Fscores$Conformance + Fscores$Durability,
##     data = Fscores)
##
## Residuals:
##      Min    1Q   Median    3Q   Max 
## -4.3277 -0.6377  0.1806  0.7287  3.3016 
##
## Coefficients:
##             Estimate Standardized Std. Error t value Pr(>|t|)    
## (Intercept) 5.34056       NA     0.03433 155.551 < 2e-16 ***
## Fscores$Serviceability 0.11411       0.08095  0.04478  2.548   0.0110 *  
## Fscores$Performance  0.33146       0.23513  0.05586  5.934  4.13e-09 *** 
## Fscores$Aesthetics   0.09158       0.06497  0.05045  1.815   0.0698 .  
## Fscores$Ease_of_Use  0.28777       0.20414  0.04830  5.958  3.57e-09 *** 
## Fscores$Features    0.09239       0.06554  0.05144  1.796   0.0728 .  
## Fscores$Reliability 0.04033       0.02861  0.04469  0.903   0.3670  
## Fscores$Conformance 0.08613       0.06110  0.05010  1.719   0.0859 .  
## Fscores$Durability  0.11120       0.07888  0.04819  2.307   0.0212 *  
## --- 
## Signif. codes:  0 '****' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1 
##
## Residual standard error: 1.069 on 960 degrees of freedom
## Multiple R-squared:  0.43, Adjusted R-squared:  0.4252 
## F-statistic: 90.51 on 8 and 960 DF, p-value: < 2.2e-16

```

For the repurchase intention, the statistically significant factors are serviceability, performance, ease of use and durability. Depending on the level of statistical significance, we can add aesthetics, features and conformance. The most important factor is performance followed by ease of use,serviceability and durability. From a managerial perspective, we should put more emphasis on the performance and the ease of use if we want to encourage the willingness to pay of the customer.

So WTP and RI are not completely driven by the same factors. However, they share common influences such as the ease of use, the performance and the serviceability. Therefore, from a managerial standpoint, allocating emphasis to these factors is crucial as they impact both WTP and RI.

We have a broad idea of the qualities that drive WTP and RI. It could be interesting to see if the variables that drive RI differ across brands.

Across brands (1-Apple 2-Samsung 3- LG 4- Motorola 5- Other)

Lets compare the repurchase intention across brands.

```
Data = cbind(Fscores, QI2["brandrec"])
```

Apple

First, we will take a look at Apple.

```

Apple = subset(Data, brandrec == 1 )

RI = rowMeans(QI2[QI2$brandrec == 1, paste0("ri", seq(1, 2))])

regr_Apple = lm(RI ~ Serviceability + Performance + Aesthetics + Ease_of_Use + Features + Reliability + Conformance + Durability)

summary(lm.beta(regr_Apple))

```

```

## 
## Call:
## lm(formula = RI ~ Serviceability + Performance + Aesthetics +
##     Ease_of_Use + Features + Reliability + Conformance + Durability,
##     data = Apple)
##
## Residuals:
##      Min    1Q   Median    3Q   Max 
## -3.4074 -0.4342  0.1204  0.5424  2.4596 
##
## Coefficients:
##             Estimate Standardized Std. Error t value Pr(>|t|)    
## (Intercept) 5.71245       NA     0.03981 143.502 < 2e-16 ***
## Serviceability 0.15828       0.13245  0.05309  2.981   0.0030 ** 
## 
```

```

## Performance    0.17954    0.13551    0.06967    2.577    0.0102 *
## Aesthetics    0.14237    0.11997    0.05870    2.426    0.0156 *
## Ease_of_Use   0.25917    0.21768    0.05575    4.649    4.23e-06 ***
## Features      0.02862    0.02371    0.05781    0.495    0.6208
## Reliability   0.03002    0.02599    0.05051    0.594    0.5525
## Conformance   0.09687    0.08261    0.05778    1.677    0.0942 .
## Durability    0.07129    0.06131    0.05433    1.312    0.1901
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8906 on 525 degrees of freedom
## Multiple R-squared:  0.3881, Adjusted R-squared:  0.3788
## F-statistic: 41.62 on 8 and 525 DF,  p-value: < 2.2e-16

```

The variables that are statistically significant are serviceability, performance, aesthetics and ease of use. Depending on the chosen significant level, there is also conformance. This means that these variables are driving the repurchase intention of their clients. Looking at the coefficients, we can say that ease of use is the most important quality followed by performance, serviceability and aesthetics.

Samsung

```

Samsung = subset(Data, brandrec == 2 )

RI = rowMeans(QI2[QI2$brandrec == 2, paste0("ri", seq(1, 2))])

regr_Samsung = lm(RI ~ Serviceability + Performance + Aesthetics + Ease_of_Use + Features +
summary(lm.beta(regr_Samsung))

## 
## Call:
## lm(formula = RI ~ Serviceability + Performance + Aesthetics +
##     Ease_of_Use + Features + Reliability + Conformance + Durability,
##     data = Samsung)
## 
## Residuals:
##      Min        1Q        Median       3Q        Max 
## -3.3745 -0.6283  0.0469  0.8202  2.1637 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 5.11206     NA     0.07291 70.113 < 2e-16 ***
## Serviceability 0.04772     0.03390  0.09843  0.485  0.62829  
## Performance 0.28323     0.22043  0.10587  2.675  0.00802 **  
## Aesthetics 0.04980     0.03638  0.10809  0.461  0.64545  
## Ease_of_Use 0.23899     0.16538  0.10631  2.248  0.02554 *   
## Features 0.03901     0.02889  0.10968  0.356  0.72242  
## Reliability -0.03254    -0.02263  0.10059  -0.324  0.74659  
## Conformance 0.12179     0.08807  0.10895  1.118  0.26482  
## Durability 0.33294     0.23429  0.10732  3.102  0.00217 **  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 1.088 on 225 degrees of freedom
## Multiple R-squared:  0.4246, Adjusted R-squared:  0.4041
## F-statistic: 20.75 on 8 and 225 DF,  p-value: < 2.2e-16

```

For Samsung, the variables that are statistically significant are performance, ease of use and durability. These qualities are important for their customer and the most important is durability followed by performance and ease of use.

LG

```

LG = subset(Data, brandrec == 3 )

RI = rowMeans(QI2[QI2$brandrec == 3, paste0("ri", seq(1, 2))])

regr_LG = lm(RI ~ Serviceability + Performance + Aesthetics + Ease_of_Use + Features + Relia
summary(lm.beta(regr_LG))

## 
## Call:
## lm(formula = RI ~ Serviceability + Performance + Aesthetics +
##     Ease_of_Use + Features + Reliability + Conformance + Durability,
##     data = LG)
## 
## Residuals:
##      Min        1Q        Median       3Q        Max 
## -3.3745 -0.6283  0.0469  0.8202  2.1637 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 5.11206     NA     0.07291 70.113 < 2e-16 ***
## Serviceability 0.04772     0.03390  0.09843  0.485  0.62829  
## Performance 0.28323     0.22043  0.10587  2.675  0.00802 **  
## Aesthetics 0.04980     0.03638  0.10809  0.461  0.64545  
## Ease_of_Use 0.23899     0.16538  0.10631  2.248  0.02554 *   
## Features 0.03901     0.02889  0.10968  0.356  0.72242  
## Reliability -0.03254    -0.02263  0.10059  -0.324  0.74659  
## Conformance 0.12179     0.08807  0.10895  1.118  0.26482  
## Durability 0.33294     0.23429  0.10732  3.102  0.00217 **  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 1.088 on 225 degrees of freedom
## Multiple R-squared:  0.4246, Adjusted R-squared:  0.4041
## F-statistic: 20.75 on 8 and 225 DF,  p-value: < 2.2e-16

```

```

##      Min     1Q Median     3Q    Max
## -3.2214 -0.7352 -0.0259  0.6507  2.3662
##
## Coefficients:
##                               Estimate Standardized Std. Error t value Pr(>|t|)
## (Intercept)            4.71448          NA   0.15786  29.866 <2e-16 ***
## Serviceability       -0.10561        -0.07850  0.18319 -0.576  0.5667
## Performance           0.38755        0.30501  0.23106  1.677  0.0993 .
## Aesthetics            -0.05733       -0.04133  0.20544 -0.279  0.7813
## Ease_of_Use           0.08378        0.06579  0.18623  0.450  0.6546
## Features              0.19480        0.13239  0.23869  0.816  0.4180
## Reliability           0.41497        0.31781  0.17884  2.320  0.0241 *
## Conformance           0.08576        0.06858  0.20129  0.426  0.6718
## Durability            -0.06163      -0.04928  0.18153 -0.339  0.7356
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.162 on 54 degrees of freedom
## Multiple R-squared:  0.3907, Adjusted R-squared:  0.3004
## F-statistic: 4.328 on 8 and 54 DF,  p-value: 0.0004458

```

For LG, it appears that reliability is the primary determinant influencing repeat purchases. Depending on the chosen significant level, there is also Performance.

In conclusion, each brand has different qualities that drives repurchase intention. We can say that performance seems to be an important one as it's statistically significant in all companies depending on the significant level.

Comparing factor scores between brands

To compare the brands on the factor scores. We decided to merge the dataset with the 33 questions and the brandrec with the factor scores.

```
QI3 = cbind(QI2, Fscores)
```

First, I do the mean of the factors scores for each factor across all Apple customers.

Apple

```

Apple2 = QI3[QI3$brandrec==1,]

serviceability_apple_mean = mean(Apple2$Serviceability)
performance_apple_mean = mean(Apple2$Performance)
aesthetics_apple_mean = mean(Apple2$Aesthetics)
ease_apple_mean = mean(Apple2$Ease_of_Use)
features_apple_mean = mean(Apple2$Features)
reliability_apple_mean = mean(Apple2$Reliability)
conformance_apple_mean = mean(Apple2$Conformance)
durability_apple_mean = mean(Apple2$Durability)

apple_means = data.frame(
  Apple = c("Serviceability", "Performance", "Aesthetics",
           "Ease of Use", "Features", "Reliability",
           "Conformance", "Durability"),
  Mean = c(serviceability_apple_mean, performance_apple_mean,
           aesthetics_apple_mean, ease_apple_mean,
           features_apple_mean, reliability_apple_mean,
           conformance_apple_mean, durability_apple_mean)
)

print(apple_means)

```

```

##             Apple      Mean
## 1 Serviceability 0.16058047
## 2 Performance 0.20226403
## 3 Aesthetics 0.13528077
## 4 Ease of Use 0.15293781
## 5 Features 0.18515755
## 6 Reliability 0.08882351
## 7 Conformance 0.14728974
## 8 Durability 0.11200655

```

Samsung

```

Samsung2 = QI3[QI3$brandrec==2,]

serviceability_samsung_mean = mean(Samsung2$Serviceability)
performance_samsung_mean = mean(Samsung2$Performance)
aesthetics_samsung_mean = mean(Samsung2$Aesthetics)

```

```

ease_samsung_mean = mean(Samsung2$Ease_of_Use)
features_samsung_mean = mean(Samsung2$Features)
reliability_samsung_mean = mean(Samsung2$Reliability)
conformance_samsung_mean = mean(Samsung2$Conformance)
durability_samsung_mean = mean(Samsung2$Durability)

samsung_means = data.frame(
  Samsung = c("Serviceability", "Performance", "Aesthetics",
             "Ease of Use", "Features", "Reliability",
             "Conformance", "Durability"),
  Mean = c(serviceability_samsung_mean, performance_samsung_mean,
           aesthetics_samsung_mean, ease_samsung_mean,
           features_samsung_mean, reliability_samsung_mean,
           conformance_samsung_mean, durability_samsung_mean)
)

print(samsung_means)

```

```

##          Samsung      Mean
## 1 Serviceability -0.11183008
## 2 Performance   -0.11850206
## 3 Aesthetics     -0.05263947
## 4 Ease of Use    -0.09002422
## 5 Features       -0.07653561
## 6 Reliability    0.03228215
## 7 Conformance    -0.13306783
## 8 Durability     -0.09287278

```

Now, we can do a t-test to compare the mean for each factors between Apple and Samsung.

T-test

$H_0: \mu_{\text{serviceability, Apple}} = \mu_{\text{serviceability, Samsung}}$ (The means of the two groups are equal) $H_1: \mu_{\text{serviceability, Apple}} \neq \mu_{\text{serviceability, Samsung}}$ (The means of the two groups are not equal)

```

# SERVICEABILITY
t_test_serviceability= t.test(Samsung2$Serviceability, Apple2$Serviceability)
print(t_test_serviceability)

```

```

##
## Welch Two Sample t-test
##
## data: Samsung2$Serviceability and Apple2$Serviceability
## t = -3.5278, df = 422.28, p-value = 0.000465
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -0.4241907 -0.1206304
## sample estimates:
## mean of x mean of y
## -0.1118301 0.1605805

```

#there is a statistically significant difference between the means of the two groups.

$H_0: \mu_{\text{performance, Apple}} = \mu_{\text{performance, Samsung}}$ $H_1: \mu_{\text{performance, Apple}} \neq \mu_{\text{performance, Samsung}}$

```

# PERFORMANCE
t_test_Performance= t.test(Samsung2$Performance, Apple2$Performance)
print(t_test_Performance)

```

```

##
## Welch Two Sample t-test
##
## data: Samsung2$Performance and Apple2$Performance
## t = -3.976, df = 361.59, p-value = 8.464e-05
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -0.4794175 -0.1621147
## sample estimates:
## mean of x mean of y
## -0.1185021 0.2022640

```

#there is a statistically significant difference between the means of the two groups.

```

# AESTHETICS
t_test_Aesthetics= t.test(Samsung2$Aesthetics, Apple2$Aesthetics)
print(t_test_Aesthetics)

```

```
##  
## Welch Two Sample t-test  
##  
## data: Samsung2$Aesthetics and Apple2$Aesthetics  
## t = -2.3802, df = 414.67, p-value = 0.01776  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -0.34311742 -0.03272306  
## sample estimates:  
## mean of x mean of y  
## -0.05263947 0.13528077
```

#there is a statistically significant difference between the means of the two groups.

```
# EASE OF USE  
t_test_Ease_of_Use= t.test(Samsung2$Ease_of_Use, Apple2$Ease_of_Use)  
print(t_test_Ease_of_Use)
```

```
##  
## Welch Two Sample t-test  
##  
## data: Samsung2$Ease_of_Use and Apple2$Ease_of_Use  
## t = -3.2027, df = 433.67, p-value = 0.001462  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -0.39206501 -0.09385906  
## sample estimates:  
## mean of x mean of y  
## -0.09002422 0.15293781
```

#there is a statistically significant difference between the means of the two groups.

```
# FEATURES  
t_test_Features= t.test(Samsung2$Features, Apple2$Features)  
print(t_test_Features)
```

```
##  
## Welch Two Sample t-test  
##  
## data: Samsung2$Features and Apple2$Features  
## t = -3.2971, df = 404.03, p-value = 0.001063  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -0.4177228 -0.1056635  
## sample estimates:  
## mean of x mean of y  
## -0.07653561 0.18515755
```

#there is a statistically significant difference between the means of the two groups.

```
# RELIABILITY  
t_test_Reliability= t.test(Samsung2$Reliability, Apple2$Reliability)  
print(t_test_Reliability)
```

```
##  
## Welch Two Sample t-test  
##  
## data: Samsung2$Reliability and Apple2$Reliability  
## t = -0.73624, df = 443.65, p-value = 0.462  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -0.20747328 0.09439055  
## sample estimates:  
## mean of x mean of y  
## 0.03228215 0.08882351
```

#there is NOT a statistically significant difference between the means of the two groups.

```
# CONFORMANCE  
t_test_Conformance= t.test(Samsung2$Conformance, Apple2$Conformance)  
print(t_test_Conformance)
```

```

## 
## Welch Two Sample t-test
##
## data: Samsung2$Conformance and Apple2$Conformance
## t = -3.5658, df = 422.73, p-value = 0.0004042
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -0.4348997 -0.1258154
## sample estimates:
## mean of x mean of y
## -0.1330678 0.1472897

```

#there is a statistically significant difference between the means of the two groups.

```

# DURABILITY
t_test_Durability= t.test(Samsung2$Durability, Apple2$Durability)
print(t_test_Durability)

```

```

## 
## Welch Two Sample t-test
##
## data: Samsung2$Durability and Apple2$Durability
## t = -2.6504, df = 436.32, p-value = 0.008333
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -0.35681104 -0.05294764
## sample estimates:
## mean of x mean of y
## -0.09287278 0.11200655

```

#there is a statistically significant difference between the means of the two groups.

With the p-value exceeding 0.05 only for Reliability, we possess enough evidence to dismiss the null hypothesis (H_0) for all factors except Reliability. This suggests that there are differences in the average scores for each factor between the two brands. Both Apple and Samsung users perceive reliability similarly. However, they differ in their perceptions of serviceability, performance, aesthetics, ease of use, features, conformance, and durability.

Examining the average values for both sets is informative:

```

combined_table = cbind(apple_means, samsung_means[,2])
colnames(combined_table) = c("Attribute", "Apple means", "Samsung means")
print(combined_table)

```

Attribute	Apple means	Samsung means
1 Serviceability	0.16058047	-0.11183008
2 Performance	0.20226403	-0.11850206
3 Aesthetics	0.13528077	-0.05263947
4 Ease of Use	0.15293781	-0.09002422
5 Features	0.18515755	-0.07653561
6 Reliability	0.08882351	0.03228215
7 Conformance	0.14728974	-0.13306783
8 Durability	0.11200655	-0.09287278

(Here we don't take into account the row "Reliability" as the difference is not significant)

According to the mean factor scores, Apple consumers generally have higher opinions about their devices than Samsung customers do about serviceability, performance, aesthetics, simplicity of use, features, conformity, and durability. For example, With the mean factor score for serviceability being 0.16058047 for Apple customers and -0.11183008 for Samsung customers, we can observe that, on average, Apple customers perceive serviceability more positively than Samsung customers do.

Conclusion

We found that the use of both Principal Axis Factoring and Principal Component Analysis with oblique rotation ("promax") enriches the analysis by considering the possibility of correlated factors, which is realistic given the multifaceted nature of product quality. Additional, regression analyses reveal that different quality dimensions have varying degrees of influence on willingness to pay premium and their intentions to repurchase. Factors such as Ease of Use and Features emerge as critical drivers for WTPP, while Performance and Ease of Use for RI. Finally, brand-specific dynamics were found, for example, Apple and Samsung customers show distinct priorities and perceptions regarding quality dimensions, offering valuable guidance for future targeted marketing and product development strategies.