Exploratory factor analysis (practical)

Note updated March 12, 2019

Wan Nor Arifin (wnarifin@usm.my), Universiti Sains Malaysia
Website: wnarifin.github.io

Contents

1 Practical

In this practical session, we are going to explore the validity of a new questionnaire of attitude towards statistics by exploratory factor analysis.

We will focus on:

- The number of extracted factors.
- Factor loadings.
- Factor correlations (no multicollinearity).

2 Preliminaries

2.1 Load libraries

Our analysis will involve `psych` (Revelle, 2018) and `lattice` (Sarkar, 2018) packages. Make sure you already installed all of them. Load the libraries,

```
library(foreign)  # for importing SPSS data
library(psych)  # for psychometrics
library(lattice)  # easy to plot multivariate plots
```

2.2 Load data set

Download data set "Attitude_Statistics v3.sav".

Read the data set as `data` and view the basic information. Replace `data` again after removing `ID` variable. This will make our analysis easier to code.

```
data = read.spss("Attitude_Statistics v3.sav", use.value.labels = F, to.data.frame = T)
str(data)
```

```
## 'data.frame':    150 obs. of  13 variables:
##  $ ID : num  1 2 3 4 5 6 8 9 10 11 ...
##  $ Q1 : num  2 3 5 2 4 4 4 3 4 2 ...
##   ..- attr(*, "value.labels")= Named num  5 4 3 2 1
##   .. ..- attr(*, "names")= chr  "STRONGLY AGREE" "AGREE" "NEUTRAL" "DISAGREE" ...
##  $ Q2 : num  3 2 4 2 1 4 2 4 2 5 ...
##   ..- attr(*, "value.labels")= Named num  5 4 3 2 1
##   .. ..- attr(*, "names")= chr  "STRONGLY AGREE" "AGREE" "NEUTRAL" "DISAGREE" ...
##  $ Q3 : num  3 3 5 2 4 4 5 3 3 2 ...
##   ..- attr(*, "value.labels")= Named num  5 4 3 2 1
##   .. ..- attr(*, "names")= chr  "STRONGLY AGREE" "AGREE" "NEUTRAL" "DISAGREE" ...
##  $ Q4 : num  3 3 1 4 2 3 3 2 4 4 ...
##   ..- attr(*, "value.labels")= Named num  5 4 3 2 1
##   .. ..- attr(*, "names")= chr  "STRONGLY AGREE" "AGREE" "NEUTRAL" "DISAGREE" ...
##  $ Q5 : num  4 4 1 3 5 4 4 3 3 4 ...
##   ..- attr(*, "value.labels")= Named num  5 4 3 2 1
##   .. ..- attr(*, "names")= chr  "STRONGLY AGREE" "AGREE" "NEUTRAL" "DISAGREE" ...
##  $ Q6 : num  4 4 1 2 1 4 3 3 4 5 ...
##   ..- attr(*, "value.labels")= Named num  5 4 3 2 1
##   .. ..- attr(*, "names")= chr  "STRONGLY AGREE" "AGREE" "NEUTRAL" "DISAGREE" ...
##  $ Q7 : num  3 4 1 2 4 4 4 2 3 4 ...
##   ..- attr(*, "value.labels")= Named num  5 4 3 2 1
##   .. ..- attr(*, "names")= chr  "STRONGLY AGREE" "AGREE" "NEUTRAL" "DISAGREE" ...
##  $ Q8 : num  3 3 4 2 5 3 3 3 3 4 ...
##   ..- attr(*, "value.labels")= Named num  5 4 3 2 1
##   .. ..- attr(*, "names")= chr  "STRONGLY AGREE" "AGREE" "NEUTRAL" "DISAGREE" ...
##  $ Q9 : num  3 3 4 2 5 4 3 4 5 4 ...
##   ..- attr(*, "value.labels")= Named num  5 4 3 2 1
##   .. ..- attr(*, "names")= chr  "STRONGLY AGREE" "AGREE" "NEUTRAL" "DISAGREE" ...
##  $ Q10: num  3 3 5 2 3 4 3 4 4 4 ...
##   ..- attr(*, "value.labels")= Named num  5 4 3 2 1
##   .. ..- attr(*, "names")= chr  "STRONGLY AGREE" "AGREE" "NEUTRAL" "DISAGREE" ...
##  $ Q11: num  4 4 1 3 4 4 3 3 4 4 ...
##   ..- attr(*, "value.labels")= Named num  5 4 3 2 1
##   .. ..- attr(*, "names")= chr  "STRONGLY AGREE" "AGREE" "NEUTRAL" "DISAGREE" ...
##  $ Q12: num  2 2 4 3 4 4 3 3 3 2 ...
##   ..- attr(*, "value.labels")= Named num  5 4 3 2 1
##   .. ..- attr(*, "names")= chr  "STRONGLY AGREE" "AGREE" "NEUTRAL" "DISAGREE" ...
##  - attr(*, "variable.labels")= Named chr  "Participant's ID" "I love all analysis." "I dream of norma
##   ..- attr(*, "names")= chr  "ID" "Q1" "Q2" "Q3" ...
##  - attr(*, "codepage")= int 65001
```

```r
data = data[-1]  # exclude ID, we are not going to use the variable
dim(data)  # 12 variables
```

```
## [1] 150  12
```

```r
names(data)  # list variable names
```

```
##  [1] "Q1"  "Q2"  "Q3"  "Q4"  "Q5"  "Q6"  "Q7"  "Q8"  "Q9"  "Q10" "Q11" "Q12"
```

```r
head(data)  # the first 6 observations
```

```
##   Q1 Q2 Q3 Q4 Q5 Q6 Q7 Q8 Q9 Q10 Q11 Q12
## 1  2  3  3  3  4  4  3  3  3   3   4   2
## 2  3  2  3  3  4  4  4  3  3   3   4   2
## 3  5  4  5  1  1  1  1  4  4   5   1   4
## 4  2  2  2  4  3  2  2  2  2   2   3   3
## 5  4  1  4  2  5  1  4  5  5   3   4   4
## 6  4  4  4  3  4  4  4  3  4   4   4   4
```

3 Exploratory factor analysis

3.1 Preliminary steps

Descriptive statistics

Check minimum/maximum values per item, and screen for missing values,

```r
describe(data)
```

```
##     vars   n mean   sd median trimmed  mad min max range  skew kurtosis   se
## Q1     1 150 3.13 1.10      3    3.12 1.48   1   5     4 -0.10    -0.73 0.09
## Q2     2 150 3.51 1.03      3    3.55 1.48   1   5     4 -0.14    -0.47 0.08
## Q3     3 150 3.18 1.03      3    3.17 1.48   1   5     4 -0.03    -0.42 0.08
## Q4     4 150 2.81 1.17      3    2.77 1.48   1   5     4  0.19    -0.81 0.10
## Q5     5 150 3.31 1.01      3    3.32 1.48   1   5     4 -0.22    -0.48 0.08
## Q6     6 150 3.05 1.09      3    3.05 1.48   1   5     4 -0.04    -0.71 0.09
## Q7     7 150 2.92 1.19      3    2.92 1.48   1   5     4 -0.04    -1.06 0.10
## Q8     8 150 3.33 1.00      3    3.34 1.48   1   5     4 -0.08    -0.12 0.08
## Q9     9 150 3.44 1.05      3    3.48 1.48   1   5     4 -0.21    -0.32 0.09
## Q10   10 150 3.31 1.10      3    3.36 1.48   1   5     4 -0.22    -0.39 0.09
## Q11   11 150 3.35 0.94      3    3.37 1.48   1   5     4 -0.31    -0.33 0.08
## Q12   12 150 2.83 0.98      3    2.83 1.48   1   5     4  0.09    -0.68 0.08
```

Note that all $n = 150$, no missing values. `min`–`max` cover the whole range of response options.

% of response to options per item,

```r
response.frequencies(data)
```

```
##         1     2    3    4     5 miss
## Q1  0.073 0.220 0.32 0.28 0.107    0
## Q2  0.033 0.093 0.42 0.24 0.213    0
## Q3  0.053 0.180 0.41 0.24 0.113    0
## Q4  0.140 0.280 0.30 0.19 0.093    0
## Q5  0.040 0.167 0.35 0.33 0.113    0
## Q6  0.080 0.233 0.33 0.26 0.093    0
## Q7  0.133 0.267 0.23 0.29 0.080    0
```

```
## Q8  0.047 0.100 0.48 0.23 0.147    0
## Q9  0.047 0.093 0.42 0.25 0.187    0
## Q10 0.073 0.107 0.42 0.23 0.167    0
## Q11 0.027 0.153 0.35 0.39 0.087    0
## Q12 0.073 0.327 0.33 0.23 0.033    0
```

All response options are used with no missing values.

Normality of data

This is done to check for the normality of the data. If the data are normally distributed, we may use maximum likelihood (ML) for the EFA, which will allow more detailed analysis. Otherwise, the extraction method of choice is **principal axis factoring (PAF)**, because it does not require normally distributed data (Brown, 2015).

Univariate normality

1. Histograms

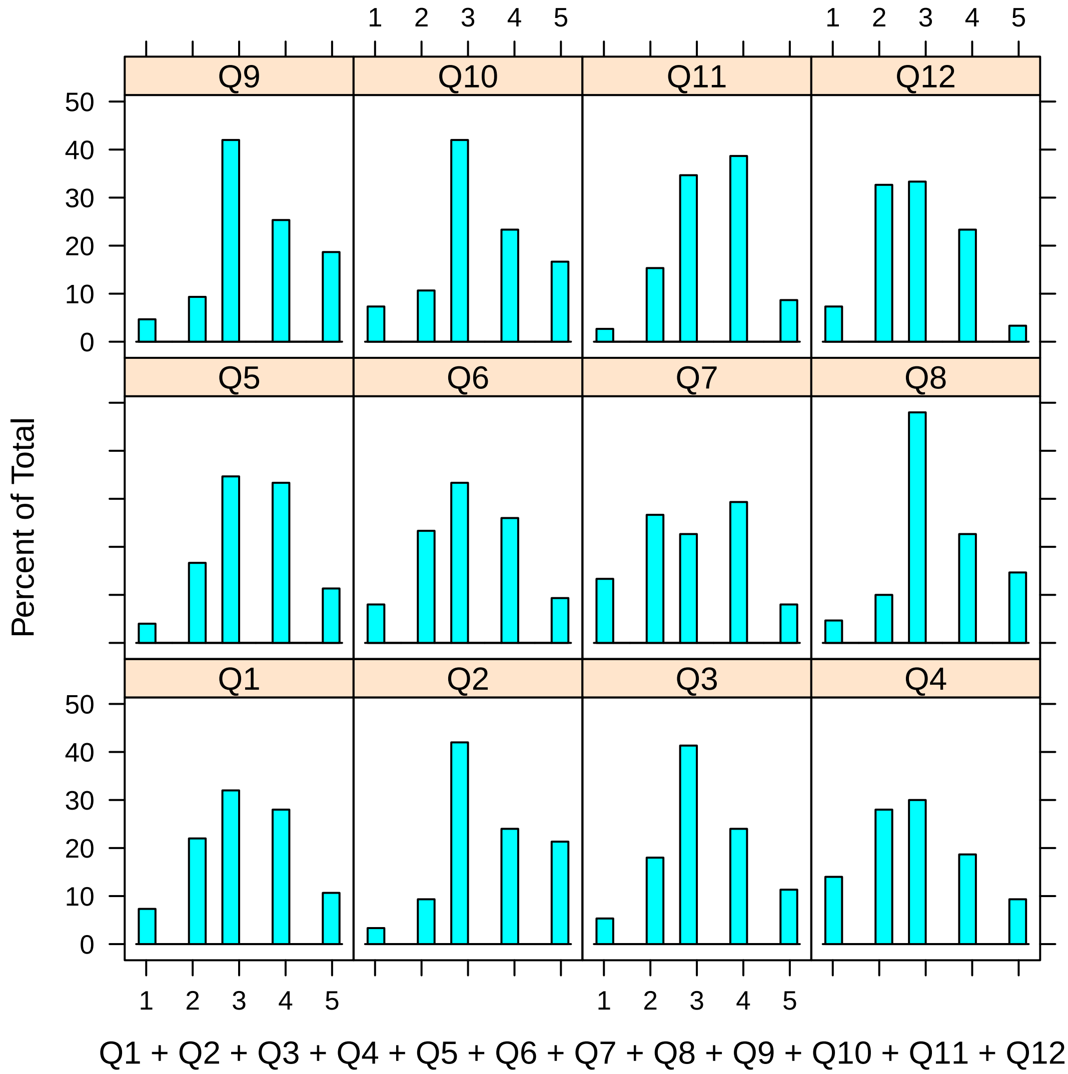
Prepare the list of variable we want to copy-paste in `histogram()`,

```
cat(names(data), sep = " + ")
```

```
## Q1 + Q2 + Q3 + Q4 + Q5 + Q6 + Q7 + Q8 + Q9 + Q10 + Q11 + Q12
```

Plot the histograms,

```
histogram(~Q1 + Q2 + Q3 + Q4 + Q5 + Q6 + Q7 + Q8 + Q9 + Q10 + Q11 + Q12, data = data)
# graphically looks normal
```



all of which look quite normal.

2. Shapiro Wilk's test

```
mapply(shapiro.test, data)
```

```
##           Q1                            Q2
## statistic 0.9153467                     0.8832131
## p.value   1.075493e-07                  1.656343e-09
## method    "Shapiro-Wilk normality test" "Shapiro-Wilk normality test"
## data.name "dots[[1L]][[1L]]"            "dots[[1L]][[2L]]"
##           Q3                            Q4
## statistic 0.9078453                     0.9134687
## p.value   3.76009e-08                   8.22525e-08
## method    "Shapiro-Wilk normality test" "Shapiro-Wilk normality test"
## data.name "dots[[1L]][[3L]]"            "dots[[1L]][[4L]]"
##           Q5                            Q6
## statistic 0.9061485                     0.9161874
## p.value   2.985958e-08                  1.21404e-07
## method    "Shapiro-Wilk normality test" "Shapiro-Wilk normality test"
## data.name "dots[[1L]][[5L]]"            "dots[[1L]][[6L]]"
```

```
##           Q7                             Q8
## statistic 0.9055861                      0.8811527
## p.value   2.767883e-08                   1.300764e-09
## method    "Shapiro-Wilk normality test" "Shapiro-Wilk normality test"
## data.name "dots[[1L]][[7L]]"             "dots[[1L]][[8L]]"
##           Q9                             Q10
## statistic 0.8893183                      0.8957393
## p.value   3.445127e-09                   7.653311e-09
## method    "Shapiro-Wilk normality test" "Shapiro-Wilk normality test"
## data.name "dots[[1L]][[9L]]"             "dots[[1L]][[10L]]"
##           Q11                            Q12
## statistic 0.8919437                      0.9009719
## p.value   4.75751e-09                    1.500687e-08
## method    "Shapiro-Wilk normality test" "Shapiro-Wilk normality test"
## data.name "dots[[1L]][[11L]]"            "dots[[1L]][[12L]]"
```

all P-values < 0.05, i.e. not normal.

Multivariate normality

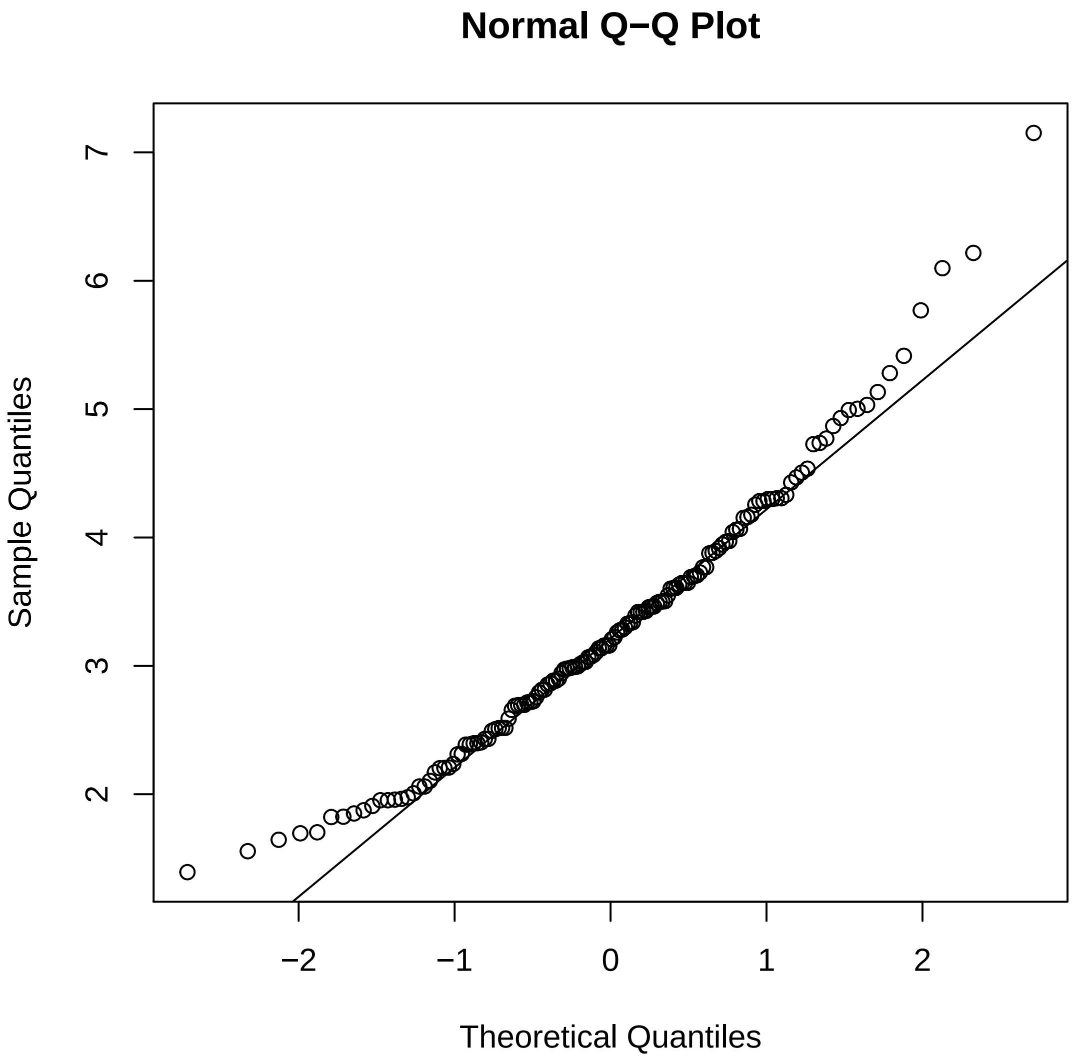
To say the data are multivariate normal:

- z-kurtosis < 5 (Bentler, 2006) and the P-value should be ≥ 0.05.
- The plot should also form a straight line (Arifin, 2015).

Run Mardia's multivariate normality test,

```r
mardia(data)
```

```
## Call: mardia(x = data)
##
## Mardia tests of multivariate skew and kurtosis
## Use describe(x) the to get univariate tests
## n.obs = 150   num.vars =  12
## b1p =  29.4   skew =  735.01  with probability =  0
##  small sample skew =  752.01  with probability =  0
## b2p =  200.33   kurtosis =  10.8  with probability =  0
```



In our case, `kurtosis` = 10.8 ($P < 0.05$). At the start and the end, the dots are away from the line. Thus, the data are not normally distributed at multivariate level. Our extraction method PAF can deal with this non-normality.

3.2 Step 1

Check suitability of data for analysis

1. Kaiser-Meyer-Olkin (KMO) Measure of Sampling Adequacy (MSA)

MSA is a relative measure of the amount of correlation (Kaiser, 1970). It indicates whether it is worthwhile to analyze a correlation matrix or not. KMO is an overall measure of MSA for a set of items. The following is the guideline in interpreting KMO values (Kaiser & Rice, 1974):

Value	Interpretation
< 0.5	Unacceptable
0.5 – 0.59	Miserable
0.6 – 0.69	Mediocre
0.7 – 0.79	Middling
0.8 – 0.89	Meritorious

Value	Interpretation
0.9 – 1.00	Marvelous

KMO of our data,

```
KMO(data)
```

```
## Kaiser-Meyer-Olkin factor adequacy
## Call: KMO(r = data)
## Overall MSA =  0.76
## MSA for each item =
##   Q1   Q2   Q3   Q4   Q5   Q6   Q7   Q8   Q9  Q10  Q11  Q12
## 0.34 0.83 0.64 0.75 0.83 0.81 0.82 0.81 0.68 0.70 0.82 0.68
```

KMO = 0.76, i.e. middling. In general, > 0.7 is acceptable.

2. Bartlet's test of sphericity

Basically it tests whether the correlation matrix is an identity matrix[1] (Bartlett, 1951; Gorsuch, 2014).

A significant test indicates worthwhile correlations between the items (i.e. off-diagonal values are not 0).

Test our data,

```
cortest.bartlett(data)
```

```
## R was not square, finding R from data

## $chisq
## [1] 562.3065
##
## $p.value
## [1] 7.851736e-80
##
## $df
## [1] 66
```

P-value < 0.05, significant. Items are correlated.

Determine the number of factors

There are several ways in determining the number of factors, among them are (Courtney, 2013):

1. Kaiser's eigenvalue > 1 rule.
2. Cattell's scree test.
3. Parallel analysis.
4. Very simple structure (VSS).
5. Velicer's minimum average partial (MAP).

The details:

1. Kaiser's eigenvalue > 1 rule.

[1] a matrix, for example, in case of three variables:

$$\begin{matrix} V1 \\ V2 \\ V3 \end{matrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Take note of the zero correlations with other variables.

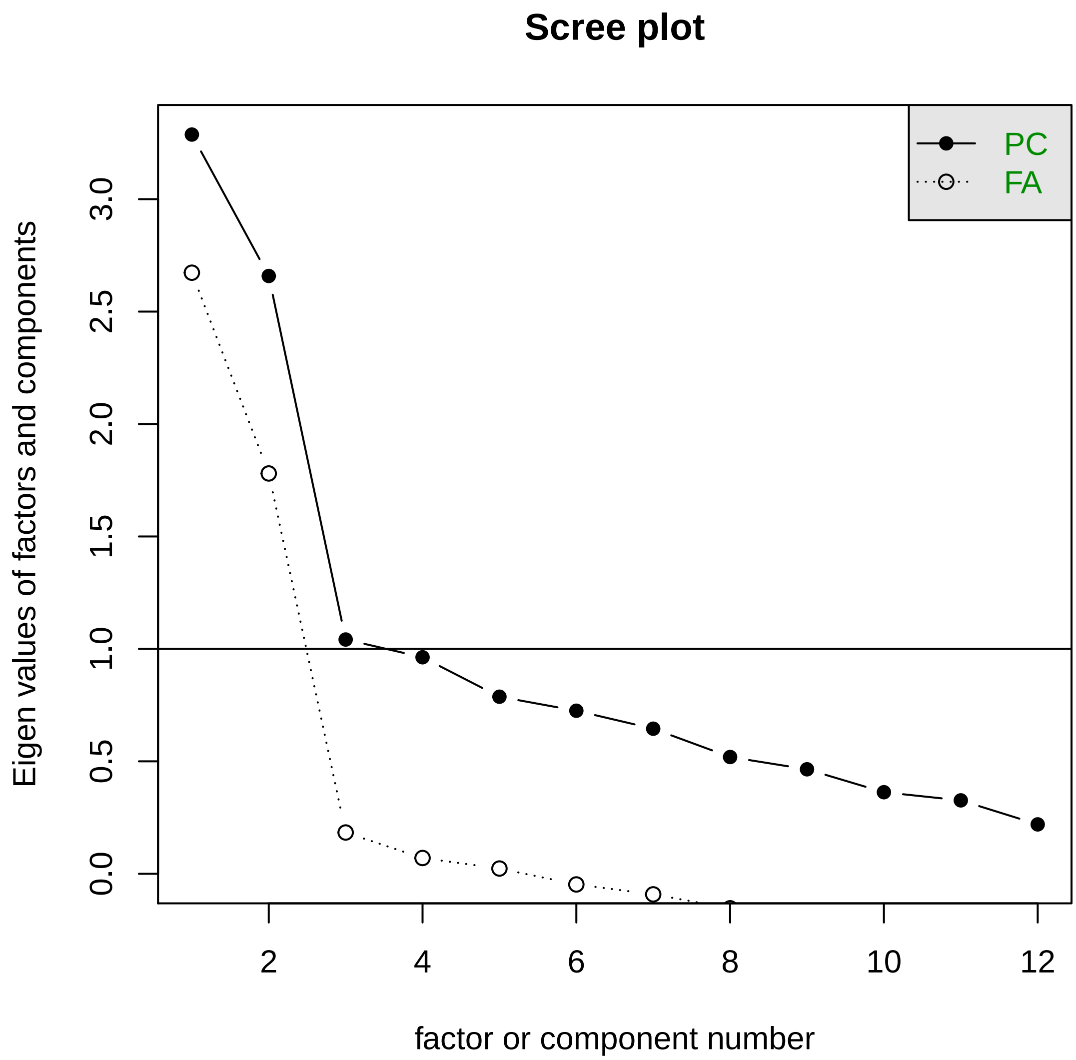
Factors with eigenvalues > 1 are retained. Eigenvalue can be interpreted as the proportion of the information in a factor. The cut-off of 1 means the factor contains information = 1 item. Thus it is not worthwhile keeping factor with information < 1 item.

2. Cattell's scree test.

"Scree" is a collection of loose stones at the base of a hill. This test is based on eye-ball judgement of an eigenvalues vs number of factors plot. Look for the number of eigenvalue points/factors before we reach the "scree", i.e. at the elbow of the plot.

Obtain the eigenvalues and scree plot,

```
scree = scree(data)
print(scree)
```




```
## Scree of eigen values
## Call: NULL
## Eigen values of factors  [1]  2.67  1.78  0.18  0.07  0.02 -0.05 -0.09 -0.15 -0.20 -0.41
## -0.46 -0.70
## Eigen values of Principal Components [1] 3.29 2.66 1.04 0.96 0.79 0.73 0.65 0.52 0.46 0.36
## 0.33 0.22
```

Eigen values of factors > 1 = 2, and 2 dots (i.e. for –o– FA) are above the Eigen values = 1 horizontal line.

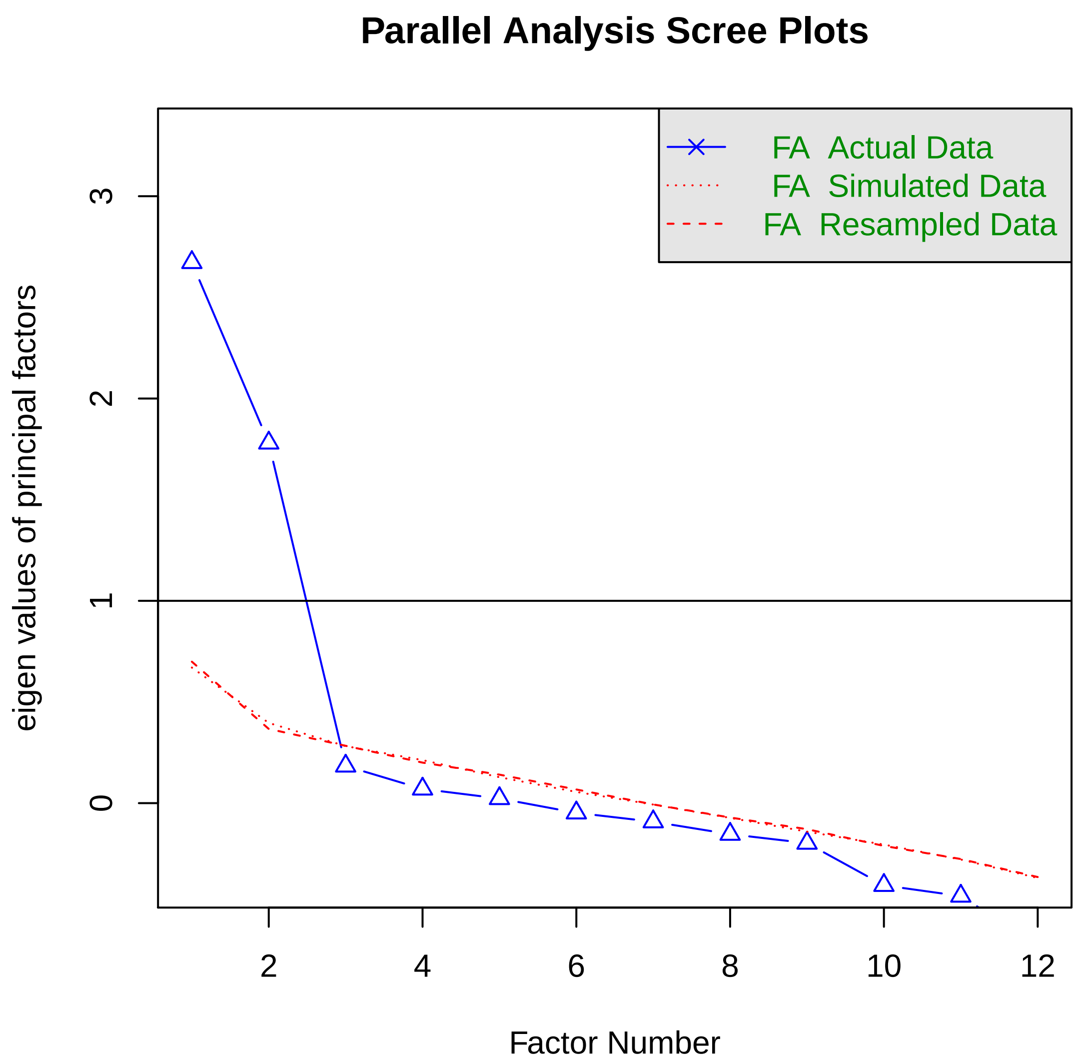
Thus, based on our judgement on the scree plot and eigenvalues (of factor analysis), the suitable number of factors = 2.

3. Parallel analysis.

The scree plot based on the data is compared to the scree plot based on the randomly generated data (Brown, 2015). The number of factors is the number of points above the intersection between the plots.

```r
parallel = fa.parallel(data, fm = "pa", fa = "fa")
print(parallel)
```

```
## Parallel analysis suggests that the number of factors =  2  and the number of components =  NA
## Call: fa.parallel(x = data, fm = "pa", fa = "fa")
## Parallel analysis suggests that the number of factors =  2  and the number of components =  NA
##
##  Eigen Values of
##
##  eigen values of factors
##  [1]  2.67  1.78  0.18  0.07  0.02 -0.05 -0.09 -0.15 -0.20 -0.41 -0.46 -0.70
##
##  eigen values of simulated factors
##  [1]  0.67  0.40  0.28  0.21  0.13  0.06 -0.01 -0.07 -0.14 -0.20 -0.28 -0.37
##
##  eigen values of components
##  [1] 3.29 2.66 1.04 0.96 0.79 0.73 0.65 0.52 0.46 0.36 0.33 0.22
##
##  eigen values of simulated components
## [1] NA
```



From the parallel-analysis scree plot, there are two dots above the dashed lines. This is suggestive of 2 factors.

4. Very simple structure (VSS) criterion.

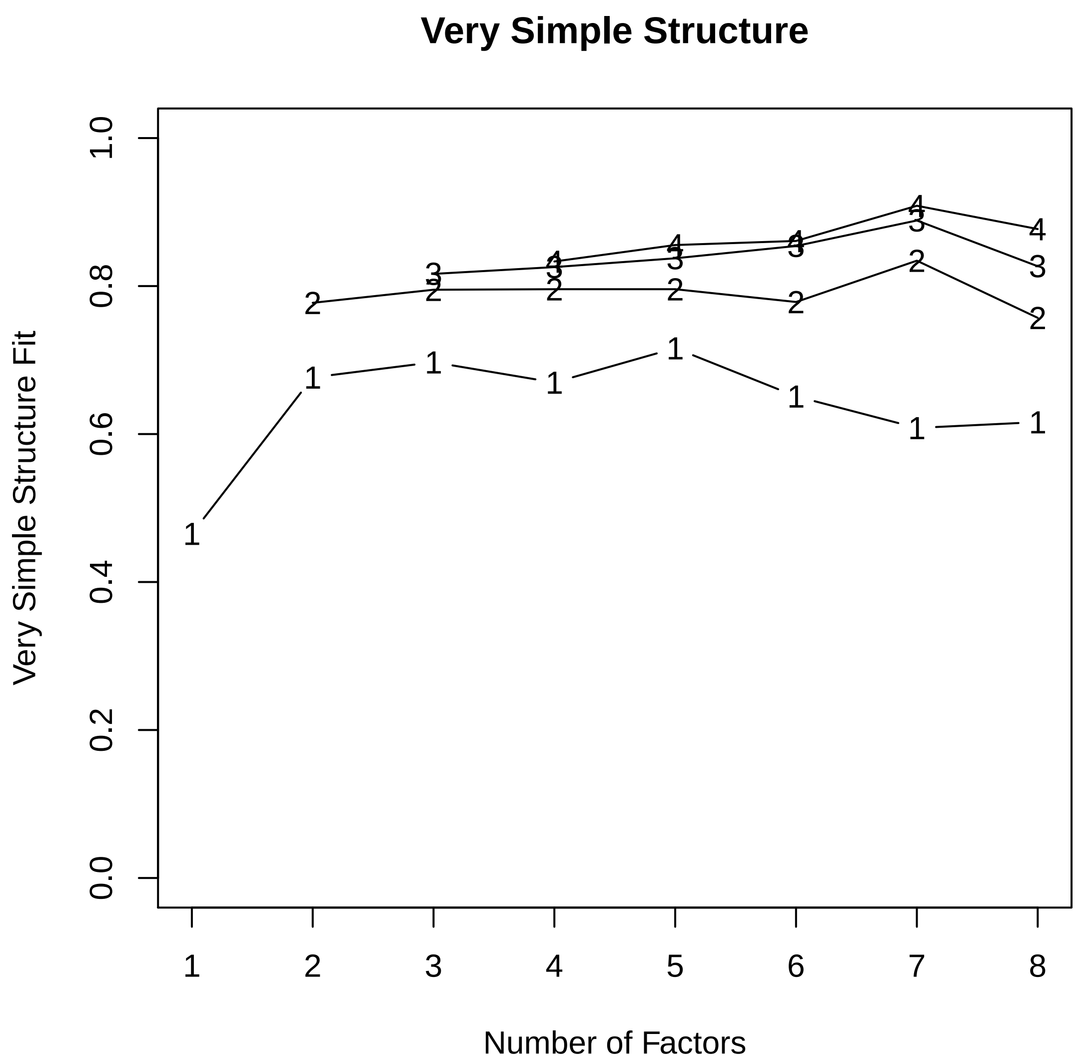
VSS compares the original correlation matrix to a simplified correlation matrix (Revelle, 2018). Look for the highest VSS value at complexity 1 (`vss1`) i.e. an item loads only on one factor.

5. Velicer's minimum average partial (MAP) criterion.

MAP criterion indicates the optimum number of factors that minimizes the MAP value. The procedure extracts the correlations explained by the factors, leaving only minimum correlations unrelated to the factors.

Obtain these two criteria,

```
vss(data)
```




```
##
## Very Simple Structure
## Call: vss(x = data)
## Although the VSS complexity 1 shows  5  factors, it is probably more reasonable to think
## about  3  factors
## VSS complexity 2 achieves a maximimum of 0.83  with  7  factors
##
## The Velicer MAP achieves a minimum of 0.03  with  2  factors
## BIC achieves a minimum of  NA  with  2  factors
## Sample Size adjusted BIC achieves a minimum of  NA  with  2  factors
##
## Statistics by number of factors
##   vss1 vss2   map dof   chisq    prob sqresid  fit  RMSEA  BIC SABIC complex  eChisq
## 1 0.47 0.00 0.065  54 306.874 4.7e-37    11.9 0.47 0.1814   36 207.2     1.0 622.420
## 2 0.68 0.78 0.029  43  62.250 2.9e-02     4.9 0.78 0.0585 -153 -17.1     1.3  41.527
## 3 0.70 0.80 0.048  33  46.613 5.8e-02     4.1 0.82 0.0568 -119 -14.3     1.3  25.305
## 4 0.67 0.80 0.067  24  27.823 2.7e-01     3.7 0.83 0.0385  -92 -16.5     1.4  14.039
## 5 0.72 0.80 0.089  16  19.445 2.5e-01     3.0 0.86 0.0438  -61 -10.1     1.3   8.727
## 6 0.65 0.78 0.119   9   8.585 4.8e-01     2.8 0.87 0.0097  -37  -8.0     1.5   3.975
## 7 0.61 0.83 0.159   3   3.094 3.8e-01     1.9 0.92 0.0261  -12  -2.4     1.5   1.281
```

```
## 8 0.62 0.76 0.239  -2   0.082      NA      2.2 0.90      NA    NA     NA      1.6   0.038
##      SRMR eCRMS eBIC
## 1 0.1773 0.196  352
## 2 0.0458 0.057 -174
## 3 0.0357 0.051 -140
## 4 0.0266 0.044 -106
## 5 0.0210 0.043  -71
## 6 0.0142 0.038  -41
## 7 0.0080 0.038  -14
## 8 0.0014    NA   NA
```

VSS complexity of 1 indicates 3/5 factors (`vss1` largest at 3 and 5 factors), while MAP indicates 2 factors (`map` smallest at 2 factors).

3.3 Step 2

Run EFA

Our data are not normally distributed, hence the extraction method of choice is **principal axis factoring (PAF)**, because it does not assume normality of data (Brown, 2015). The recommended rotation method is **oblimin** (Fabrigar & Wegener, 2012).

We run EFA by

1. fixing the number of factors as decided from previous step. Two factors are reasonable.
2. choosing an appropriate extraction method. We use PAF, `fm = "pa"`.
3. choosing an appropriate oblique rotation method. We use oblimin, `rotate = "oblimin"`.

```r
fa = fa(data, nfactors = 2, fm = "pa", rotate = "oblimin")
print(fa, cut = 0.3, digits = 3)  # cut = .3 to view only FLs > 0.3
# print(fa, digits = 3) # uncomment to view FLs < .3
```

```
## Factor Analysis using method =  pa
## Call: fa(r = data, nfactors = 2, rotate = "oblimin", fm = "pa")
## Standardized loadings (pattern matrix) based upon correlation matrix
##        PA1    PA2      h2    u2  com
## Q1                0.00366 0.996 1.77
## Q2           0.413 0.20708 0.793 1.29
## Q3  -0.339  0.439 0.28192 0.718 1.88
## Q4   0.813        0.65855 0.341 1.00
## Q5   0.584        0.41688 0.583 1.30
## Q6   0.725        0.52512 0.475 1.00
## Q7   0.732        0.53270 0.467 1.01
## Q8           0.655 0.50124 0.499 1.22
## Q9           0.773 0.59830 0.402 1.00
## Q10          0.883 0.77491 0.225 1.01
## Q11  0.528        0.29771 0.702 1.07
## Q12 -0.326        0.17665 0.823 1.98
##
##                         PA1   PA2
## SS loadings           2.646 2.329
## Proportion Var        0.220 0.194
## Cumulative Var        0.220 0.415
## Proportion Explained  0.532 0.468
## Cumulative Proportion 0.532 1.000
##
```

```
##  With factor correlations of
##       PA1   PA2
## PA1 1.000 0.087
## PA2 0.087 1.000
```

Results

Judge the quality of items.

We must looks at

1. Factor loadings (FL).
2. Communalities.
3. Factor correlations.

1. Factor loadings (pattern coefficients).

Factor loadings (FLs) / pattern coefficients are partial correlation coefficients of factors to items. FLs can be interpreted as follows (Hair, Black, Babin, & Anderson, 2010):

Value	Interpretation
0.3 to 0.4	Minimally acceptable
≥ 0.5	Practically significant
≥ 0.7	Well-defined structure

The FLs are interpreted based on absolute values, ignoring the +/- signs. We may need to remove items based on this assessment. Usually we may remove items with FLs < 0.3 (or < 0.4, or < 0.5). But the decision depends on whether we want to set a strict or lenient cut-off value.

In our output[2]:

```
## Factor Analysis using method =  pa
## Call: fa(r = data, nfactors = 2, rotate = "oblimin", fm = "pa")
## Standardized loadings (pattern matrix) based upon correlation matrix
##        PA1    PA2      h2    u2  com
## Q1                0.00366 0.996 1.77
## Q2           0.413 0.20708 0.793 1.29
## Q3  -0.339  0.439 0.28192 0.718 1.88
## Q4   0.813        0.65855 0.341 1.00
## Q5   0.584        0.41688 0.583 1.30
## Q6   0.725        0.52512 0.475 1.00
## Q7   0.732        0.53270 0.467 1.01
## Q8           0.655 0.50124 0.499 1.22
## Q9           0.773 0.59830 0.402 1.00
## Q10          0.883 0.77491 0.225 1.01
## Q11  0.528        0.29771 0.702 1.07
## Q12 -0.326        0.17665 0.823 1.98
```

Low FLs? Q1 $< .3$, Q12 $< .4$, Q2 & Q3 $< .5$

Also check for item cross-loading across factors (run the command again as `print(fa, digits = 3)` without `cut = .3`). A cross-loading is when an item has ≥ 2 significant loading (i.e. $> .3/.4/.5$) It indicates the item is not specific to a factor, thus should be removed. The cross-loading can also be judged based on item complexity (`com`). An item specific to a factor should have an item complexity close to one (Pettersson & Turkheimer, 2010).

In our output:

[2] `h2` = communality; `u2` = error; `com` = item complexity.

```
## Factor Analysis using method =  pa
## Call: fa(r = data, nfactors = 2, rotate = "oblimin", fm = "pa")
## Standardized loadings (pattern matrix) based upon correlation matrix
##        PA1    PA2      h2    u2  com
## Q1  -0.036  0.052 0.00366 0.996 1.77
## Q2   0.159  0.413 0.20708 0.793 1.29
## Q3  -0.339  0.439 0.28192 0.718 1.88
## Q4   0.813 -0.024 0.65855 0.341 1.00
## Q5   0.584  0.229 0.41688 0.583 1.30
## Q6   0.725 -0.005 0.52512 0.475 1.00
## Q7   0.732 -0.048 0.53270 0.467 1.01
## Q8   0.217  0.655 0.50124 0.499 1.22
## Q9   0.010  0.773 0.59830 0.402 1.00
## Q10 -0.058  0.883 0.77491 0.225 1.01
## Q11  0.528  0.097 0.29771 0.702 1.07
## Q12 -0.326  0.295 0.17665 0.823 1.98
```

Cross-loadings? Q3 and Q12, indicated by relatively high FLs across factors and high complexity, close to 2.

2. Communalities (`h2`).

An item communality[3] (IC) is the % of item variance explained by the extracted factors (i.e. by both `PA1` and `PA2` here). It may be considered as R^2 in linear regression.

The cut-off value of what is considered acceptable depends on the researcher; it depends on the amount of explained variance that is acceptable to him/her.

A cut-off of 0.5 is practical (Hair et al., 2010), i.e. 50% of item variance is explained by all extracted factors. However, in my practice, it depends on the minimum FL I am willing to accept. Because

$$Item\ variance\ (approximately) = FL^2$$

as if FL = 0 for other factors.

For practical purpose, > 0.25 is acceptable whenever I consider FL > 0.5 as acceptable (because communality $= 0.5^2 = 0.25$).

In our output:

```
## Factor Analysis using method =  pa
## Call: fa(r = data, nfactors = 2, rotate = "oblimin", fm = "pa")
## Standardized loadings (pattern matrix) based upon correlation matrix
##        PA1    PA2      h2    u2  com
## Q1                0.00366 0.996 1.77
## Q2          0.413 0.20708 0.793 1.29
## Q3  -0.339  0.439 0.28192 0.718 1.88
## Q4   0.813        0.65855 0.341 1.00
## Q5   0.584        0.41688 0.583 1.30
## Q6   0.725        0.52512 0.475 1.00
## Q7   0.732        0.53270 0.467 1.01
## Q8          0.655 0.50124 0.499 1.22
```

[3] The simple formula is,

$$IC = FL_{PA1} + FL_{PA2}$$

for the orthogonally-rotated solution.

For the calculation of communality for the obliquely-rotated solution, we have to include the factor correlation (FC) (Brown, 2015),

$$IC = FL^2_{PA1} + FL^2_{PA2} + 2(FL_{PA1} \times FC \times FL_{PA2})$$

For example, communality of **Q2** = `.16^2 + .41^2 + 2*.16*.09*.41` = `0.205508` ≈ `0.2071` in the output. Use `print(fa)` instead to view the full results.

```
## Q9          0.773 0.59830 0.402 1.00
## Q10         0.883 0.77491 0.225 1.01
## Q11  0.528        0.29771 0.702 1.07
## Q12 -0.326        0.17665 0.823 1.98
```

Low communalities? Q1 < Q12 < Q2 < .25 (.004 / .177 / .207 respectively)

3. Factor correlations.

In general, correlations of < 0.85 between factors are expectable in health sciences. If the correlations are > 0.85, the factors are not distinct from each other (factor overlap, or *multicollinearity*), thus they can be combined (Brown, 2015). In EFA context, this can be done by reducing the number of extracted factors.

In our output:

```
##  With factor correlations of
##       PA1   PA2
## PA1 1.000 0.087
## PA2 0.087 1.000
```

PA1 ↔ PA2 = .087 < .85

3.4 Step 3

In Step 2, we found a number of poor quality items. These must be removed from the item pool.

Repeat

Repeat **Step 2** every time an item is removed. Make sure that you remove only **ONE** item at each repeat analysis. Make decisions based on the results.

Stop

We may stop once we have

- satisfactory number of factors.
- satisfactory item quality.

We proceed as follows,

Remove Q1? Low communality, low FL.

```
fa1 = fa(subset(data, select = -Q1), nfactors = 2, fm = "pa", rotate = "oblimin")
print(fa1, cut = 0.3, digits = 3)
```

```
## Factor Analysis using method =  pa
## Call: fa(r = subset(data, select = -Q1), nfactors = 2, rotate = "oblimin",
##     fm = "pa")
## Standardized loadings (pattern matrix) based upon correlation matrix
##        PA1    PA2    h2    u2  com
## Q2          0.412 0.207 0.793 1.29
## Q3  -0.337  0.438 0.280 0.720 1.88
## Q4   0.813        0.658 0.342 1.00
## Q5   0.584        0.417 0.583 1.30
## Q6   0.726        0.526 0.474 1.00
## Q7   0.733        0.534 0.466 1.01
## Q8          0.653 0.499 0.501 1.22
## Q9          0.774 0.601 0.399 1.00
## Q10         0.886 0.779 0.221 1.01
## Q11  0.529        0.298 0.702 1.07
```

```
## Q12 -0.325        0.175 0.825 1.98
##
##                        PA1   PA2
## SS loadings           2.645 2.327
## Proportion Var        0.240 0.212
## Cumulative Var        0.240 0.452
## Proportion Explained  0.532 0.468
## Cumulative Proportion 0.532 1.000
##
##  With factor correlations of
##       PA1   PA2
## PA1 1.000 0.086
## PA2 0.086 1.000
```

Remove Q12? Low communality, (relatively) low FL, high complexity (cross loading).

```r
fa2 = fa(subset(data, select = -c(Q1, Q12)), nfactors = 2, fm = "pa", rotate = "oblimin")
print(fa2, cut = 0.3, digits = 3)
```

```
## Factor Analysis using method =  pa
## Call: fa(r = subset(data, select = -c(Q1, Q12)), nfactors = 2, rotate = "oblimin",
##     fm = "pa")
## Standardized loadings (pattern matrix) based upon correlation matrix
##        PA1    PA2    h2    u2  com
## Q2          0.420 0.211 0.789 1.24
## Q3  -0.313  0.409 0.239 0.761 1.87
## Q4   0.841        0.702 0.298 1.01
## Q5   0.595        0.426 0.574 1.26
## Q6   0.707        0.501 0.499 1.00
## Q7   0.732        0.531 0.469 1.01
## Q8          0.661 0.507 0.493 1.19
## Q9          0.780 0.608 0.392 1.00
## Q10         0.892 0.789 0.211 1.01
## Q11  0.529        0.298 0.702 1.06
##
##                        PA1   PA2
## SS loadings           2.554 2.256
## Proportion Var        0.255 0.226
## Cumulative Var        0.255 0.481
## Proportion Explained  0.531 0.469
## Cumulative Proportion 0.531 1.000
##
##  With factor correlations of
##     PA1 PA2
## PA1 1.0 0.1
## PA2 0.1 1.0
```

Remove Q3? Low communality, (relatively) low FL, high complexity (cross loading).

```r
fa3 = fa(subset(data, select = -c(Q1, Q12, Q3)), nfactors = 2, fm = "pa", rotate = "oblimin")
print(fa3, cut = 0.3, digits = 3)
```

```
## Factor Analysis using method =  pa
## Call: fa(r = subset(data, select = -c(Q1, Q12, Q3)), nfactors = 2,
##     rotate = "oblimin", fm = "pa")
## Standardized loadings (pattern matrix) based upon correlation matrix
```

```
##        PA1    PA2    h2    u2  com
## Q2          0.404 0.202 0.798 1.27
## Q4   0.816        0.660 0.340 1.00
## Q5   0.606        0.435 0.565 1.20
## Q6   0.706        0.497 0.503 1.00
## Q7   0.749        0.551 0.449 1.02
## Q8          0.669 0.518 0.482 1.16
## Q9          0.822 0.670 0.330 1.00
## Q10         0.863 0.733 0.267 1.01
## Q11  0.533        0.301 0.699 1.05
##
##                          PA1   PA2
## SS loadings           2.463 2.104
## Proportion Var        0.274 0.234
## Cumulative Var        0.274 0.507
## Proportion Explained  0.539 0.461
## Cumulative Proportion 0.539 1.000
##
##  With factor correlations of
##       PA1   PA2
## PA1 1.000 0.133
## PA2 0.133 1.000
```

Remove Q2? Low communality, (relatively) low FL.

```
fa4 = fa(subset(data, select = -c(Q1, Q12, Q3, Q2)), nfactors = 2, fm = "pa", rotate = "oblimin")
print(fa4, cut = 0.3, digits = 3)
```

```
## Factor Analysis using method =  pa
## Call: fa(r = subset(data, select = -c(Q1, Q12, Q3, Q2)), nfactors = 2,
##     rotate = "oblimin", fm = "pa")
## Standardized loadings (pattern matrix) based upon correlation matrix
##        PA1    PA2    h2    u2  com
## Q4   0.818        0.664 0.336 1.00
## Q5   0.614        0.447 0.553 1.21
## Q6   0.704        0.494 0.506 1.00
## Q7   0.747        0.549 0.451 1.02
## Q8          0.634 0.471 0.529 1.19
## Q9          0.849 0.717 0.283 1.00
## Q10         0.861 0.733 0.267 1.01
## Q11  0.533        0.299 0.701 1.04
##
##                          PA1   PA2
## SS loadings           2.441 1.933
## Proportion Var        0.305 0.242
## Cumulative Var        0.305 0.547
## Proportion Explained  0.558 0.442
## Cumulative Proportion 0.558 1.000
##
##  With factor correlations of
##       PA1   PA2
## PA1 1.000 0.121
## PA2 0.121 1.000
```

We are now satisfied with the item quality (at FL > 0.5) and factor correlation. Please also note the `Proportion Var` row; the values indicate the amount of variance explained by each factor (i.e. remember R^2

multiple linear regression?). **PA1** explains 30.5%, and **PA2** explains 24.2% of the variance in the items. In total, the extracted factors explain 54.7% of the variance.

3.5 Summary

PA1: Q4, Q5, Q6, Q7, Q11

PA2: Q8, Q9, Q10

Name the factor based on the content of the remaining items/factor (look at the PDF file/printed copy of the questionnaire).

PA1 – Affinity

PA2 – Importance

4 Results presentation

In the report, you must include a number of important statements and results pertaining to the EFA,

1. The extraction and rotation methods.
2. The KMO and Bartlett's test of sphericity results.
3. The number of extracted factors, based on the applied methods e.g. scree plot, parallel analysis, MAP etc.
4. Details about the cut-off values of the FLs, communalities and factor correlations.
5. Details about the repeat EFA, i.e. item removed, reduction/increase in the number of factors etc.
6. The percentage of variance explained (in the final solution).
7. Summary table, which includes FLs, communalities and factor correlations.

Factor loadings of EFA.

Factor	Item	Factor loading	Communality
Affinity	Q4	0.818	0.664
	Q5	0.614	0.447
	Q6	0.704	0.494
	Q7	0.747	0.549
	Q11	0.533	0.299
Importance	Q8	0.634	0.471
	Q9	0.849	0.717
	Q10	0.861	0.733
Factor correlation: - Affinity ↔ Importance r = 0.121.			

References

Arifin, W. N. (2015). The graphical assessment of multivariate normality using SPSS. *Education in Medicine Journal*, *7*(2), e71–e75.

Bartlett, M. S. (1951). The effect of standardization on a χ 2 approximation in factor analysis. *Biometrika*, *38*(3/4), 337–344.

Bentler, P. M. (2006). *EQS 6 structural equations program manual.* Encino, CA: Multivariate Software, Inc.

Brown, T. A. (2015). *Confirmatory factor analysis for applied research.* New York: The Guilford Press.

Courtney, M. G. R. (2013). Determining the number of factors to retain in efa: Using the spss r-menu v2. 0 to make more judicious estimations. *Practical Assessment, Research & Evaluation*, *18*(8), 1–14.

Fabrigar, L., & Wegener, D. (2012). *Exploratory factor analysis.* New York: Oxford University Press.

Gorsuch, R. L. (2014). *Exploratory factor analysis.* New York: Routledge.

Hair, J. F., Black, W. C., Babin, B. J., & Anderson, R. E. (2010). *Multivariate data analysis.* New Jersey: Prentice Hall.

Kaiser, H. F. (1970). A second generation little jiffy. *Psychometrika*, *35*(4), 401–415.

Kaiser, H. F., & Rice, J. (1974). Little jiffy, mark iv. *Educational and Psychological Measurement*, *34*(1), 111–117.

Pettersson, E., & Turkheimer, E. (2010). Item selection, evaluation, and simple structure in personality data. *Journal of Research in Personality*, *44*(4), 407–420.

Revelle, W. (2018). *Psych: Procedures for psychological, psychometric, and personality research.* Retrieved from https://CRAN.R-project.org/package=psych

Sarkar, D. (2018). *Lattice: Trellis graphics for r.* Retrieved from https://CRAN.R-project.org/package=lattice