Confirmatory factor analysis (practical)

Note updated December 11, 2018

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

Contents

1 Practical **1**

2 Preliminaries **1**
2.1 Load libraries . . . 1
2.2 Load data set . . . 2

3 Confirmatory factor analysis **3**
3.1 Preliminary steps . . . 3
3.2 Step 1 . . . 4
3.3 Step 2 . . . 5
3.4 Step 3 . . . 10

4 Path diagram **13**

5 Results presentation **13**

References **15**

1 Practical

In this practical session, we are going to confirm our model based on the EFA findings. The same data set, "Attitude_Statistics v3.sav" will be used.

We will focus on:

- Model fit
- Factor loadings
- Factor correlations (no multicollinearity)

2 Preliminaries

2.1 Load libraries

In addition to `psych` (Revelle, 2018), we are going to use `lavaan` version 0.6.3 (Rosseel, 2018), `semTools` version 0.5.1 (Jorgensen, Pornprasertmanit, Schoemann, & Rosseel, 2018) and `semPlot` version 1.1 (Epskamp & Simon Stuber, 2017) in our analysis. These packages must be installed from downloaded packages from CRAN at https://cran.r-project.org/.

Again, make sure you already installed all of them before loading the packages.

```
library(foreign)
library(psych)
library(lavaan)  # for CFA
library(semTools)  # for additional functions in SEM
library(semPlot)  # for path diagram
```

2.2 Load data set

We include only good items from **PA1** and **PA2** in `data.cfa` data frame.

```
data = read.spss("Attitude_Statistics v3.sav", F, T)  # Shortform
# Include selected items from PA1 & PA2 in "data.cfa"
data.cfa = data[c("Q4","Q5","Q6","Q7","Q8","Q9","Q10","Q11")]
str(data.cfa)
```

```
## 'data.frame':    150 obs. of  8 variables:
##  $ 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" ...
```

```
dim(data.cfa)
```

```
## [1] 150   8
```

```
names(data.cfa)
```

```
## [1] "Q4"  "Q5"  "Q6"  "Q7"  "Q8"  "Q9"  "Q10" "Q11"
```

```
head(data.cfa)
```

```
##   Q4 Q5 Q6 Q7 Q8 Q9 Q10 Q11
## 1  3  4  4  3  3  3   3   4
## 2  3  4  4  4  3  3   3   4
## 3  1  1  1  1  4  4   5   1
```

```
## 4  4  3  2  2  2  2   2   3
## 5  2  5  1  4  5  5   3   4
## 6  3  4  4  4  3  4   4   4
```

3 Confirmatory factor analysis

3.1 Preliminary steps

Descriptive statistics

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

```
describe(data.cfa)
```

```
##     vars   n mean   sd median trimmed  mad min max range  skew kurtosis   se
## Q4     1 150 2.81 1.17      3    2.77 1.48   1   5     4  0.19    -0.81 0.10
## Q5     2 150 3.31 1.01      3    3.32 1.48   1   5     4 -0.22    -0.48 0.08
## Q6     3 150 3.05 1.09      3    3.05 1.48   1   5     4 -0.04    -0.71 0.09
## Q7     4 150 2.92 1.19      3    2.92 1.48   1   5     4 -0.04    -1.06 0.10
## Q8     5 150 3.33 1.00      3    3.34 1.48   1   5     4 -0.08    -0.12 0.08
## Q9     6 150 3.44 1.05      3    3.48 1.48   1   5     4 -0.21    -0.32 0.09
## Q10    7 150 3.31 1.10      3    3.36 1.48   1   5     4 -0.22    -0.39 0.09
## Q11    8 150 3.35 0.94      3    3.37 1.48   1   5     4 -0.31    -0.33 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,

```
response.frequencies(data.cfa)
```

```
##         1     2    3    4     5 miss
## 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
```

All response options are used, and there are no missing values.

Multivariate normality

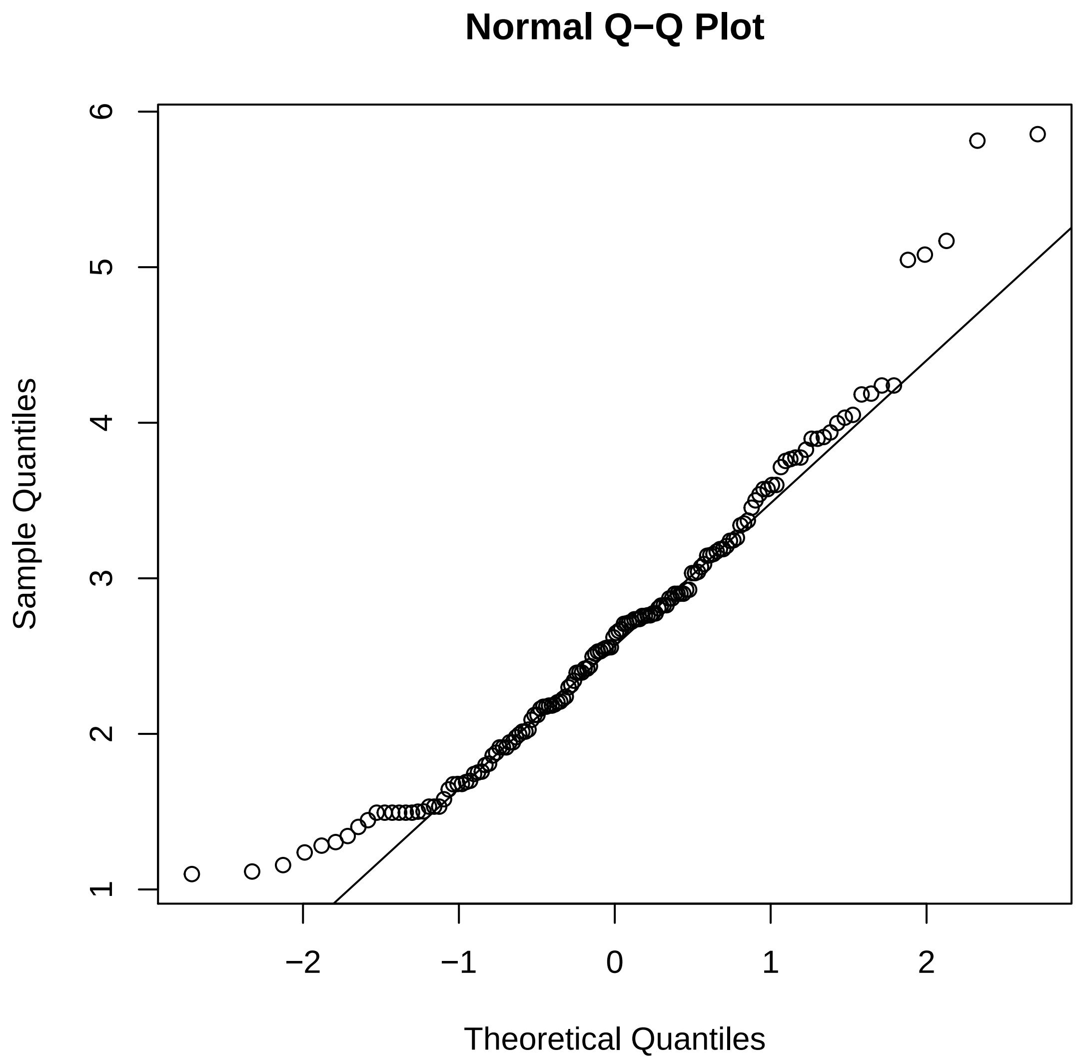
This is done to check the multivariate normality of the data. If the data are normally distributed, we may use maximum likelihood (ML) estimation method for the CFA. In `lavaan`, we have a number of alternative estimation methods (the full list is available at http://lavaan.ugent.be/tutorial/est.html or by typing `?lavOptions`). Two common alternatives are:

1. **MLR** (robust ML), suitable for complete and incomplete, non-normal data (Rosseel, 2018).
2. **WLSMV** (robust weighted least squares), suitable for categorical response options (e.g. dichotomous, polynomous, ordinal (Brown, 2015))

```
mardia(data.cfa)
```

```
## Call: mardia(x = data.cfa)
##
## Mardia tests of multivariate skew and kurtosis
```

```
## Use describe(x) the to get univariate tests
## n.obs = 150   num.vars =  8
## b1p =  11.37   skew =  284.27  with probability =  2.3e-15
##  small sample skew =  291.24  with probability =  3.3e-16
## b2p =  96.9   kurtosis =  8.18  with probability =  2.2e-16
```



the data are not multivariate normal (`kurtosis` > 5, $P < 0.05$). We will use **MLR** in our analysis.

3.2 Step 1

Specify the measurement model

Specify the measurement model according to `lavaan` syntax.

```
model = "
PA1 =~ Q4 + Q5 + Q6 + Q7 + Q11
PA2 =~ Q8 + Q9 + Q10
"
```

`=~` indicates "measured by", thus the items represent the factor.

By default, `lavaan` will correlate PA1 and PA2 (i.e. `PA1 ~~ PA2`), somewhat similar to oblique rotation in EFA. `~~` means "correlation". We will use `~~` when we add correlated errors later.

3.3 Step 2

Fit the model

Here, we fit the specified model. By default, marker indicator variable approach[1] is used in `lavaan` to scale a factor[2]. We use **MLR** as the extimation method.

```
cfa.model = cfa(model, data = data.cfa, estimator = "MLR")
# cfa.model = cfa(model, data = data.cfa, std.lv = 1)  # factor variance = 1
summary(cfa.model, fit.measures = T, standardized = T)
```

```
## lavaan 0.6-3 ended normally after 21 iterations
##
##   Optimization method                           NLMINB
##   Number of free parameters                         17
##
##   Number of observations                           150
##
##   Estimator                                         ML      Robust
##   Model Fit Test Statistic                      37.063      27.373
##   Degrees of freedom                                19          19
##   P-value (Chi-square)                           0.008       0.096
##   Scaling correction factor                                  1.354
##     for the Yuan-Bentler correction (Mplus variant)
##
## Model test baseline model:
##
##   Minimum Function Test Statistic              453.795     325.195
##   Degrees of freedom                                28          28
##   P-value                                        0.000       0.000
##
## User model versus baseline model:
##
##   Comparative Fit Index (CFI)                    0.958       0.972
##   Tucker-Lewis Index (TLI)                       0.937       0.958
##
##   Robust Comparative Fit Index (CFI)                         0.973
##   Robust Tucker-Lewis Index (TLI)                            0.960
##
## Loglikelihood and Information Criteria:
##
##   Loglikelihood user model (H0)              -1566.019   -1566.019
##   Scaling correction factor                                  1.140
##     for the MLR correction
##   Loglikelihood unrestricted model (H1)      -1547.487   -1547.487
##   Scaling correction factor                                  1.253
##     for the MLR correction
```

[1] The regression weight of an item from a factor is fixed to 1. Another approach in CFA is to fix the factor variance to 1 (Brown, 2015).

[2] The latent variable (factor) is an unobserved variable, thus it has to be scaled by a method to define its metric/unit of measurement. This is done by fixing either the item regression weight or the factor variance to 1.

```
##
##   Number of free parameters                         17          17
##   Akaike (AIC)                                3166.037    3166.037
##   Bayesian (BIC)                              3217.218    3217.218
##   Sample-size adjusted Bayesian (BIC)         3163.416    3163.416
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                          0.080       0.054
##   90 Percent Confidence Interval          0.040  0.118       0.000  0.091
##   P-value RMSEA <= 0.05                          0.098       0.397
##
##   Robust RMSEA                                               0.063
##   90 Percent Confidence Interval                             0.000  0.112
##
## Standardized Root Mean Square Residual:
##
##   SRMR                                           0.079       0.079
##
## Parameter Estimates:
##
##   Information                                 Observed
##   Observed information based on                Hessian
##   Standard Errors                   Robust.huber.white
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   PA1 =~
##     Q4                1.000                               0.952    0.814
##     Q5                0.660    0.092    7.218    0.000    0.629    0.624
##     Q6                0.810    0.090    9.010    0.000    0.771    0.708
##     Q7                0.916    0.086   10.641    0.000    0.872    0.735
##     Q11               0.533    0.093    5.719    0.000    0.507    0.544
##   PA2 =~
##     Q8                1.000                               0.653    0.655
##     Q9                1.347    0.156    8.654    0.000    0.880    0.844
##     Q10               1.436    0.199    7.206    0.000    0.938    0.856
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   PA1 ~~
##     PA2               0.077    0.075    1.035    0.301    0.124    0.124
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .Q4                0.460    0.089    5.182    0.000    0.460    0.337
##    .Q5                0.620    0.086    7.178    0.000    0.620    0.611
##    .Q6                0.590    0.101    5.836    0.000    0.590    0.498
##    .Q7                0.647    0.125    5.181    0.000    0.647    0.460
##    .Q11               0.611    0.077    7.934    0.000    0.611    0.704
##    .Q8                0.567    0.101    5.628    0.000    0.567    0.570
##    .Q9                0.312    0.094    3.325    0.001    0.312    0.287
##    .Q10               0.321    0.106    3.046    0.002    0.321    0.267
##     PA1               0.906    0.137    6.587    0.000    1.000    1.000
```

```
##     PA2                 0.427    0.106    4.009    0.000    1.000    1.000
```

Results

Read the results marked as `Robust`. These represent the results of **MLR**.

To interpret the results, we must looks at

1. Overall model fit - by fit indices.
2. Localized areas of misfit
   - Residuals.
   - Modification indices.
3. Parameter estimates
   - Factor loadings (`Std.all` column under `Latent Variables` table).
   - Factor correlations (`Std.all` column under `Covariances` table).

1. Fit indices.

The following are a number of selected fit indices and the recommended cut-off values (Brown, 2015; Schreiber, Nora, Stage, Barlow, & King, 2006),

Category	Fit index	Cut-off
Absolute fit	χ^2	P>0.05
	Standardized root mean square (SRMR)	≤ 0.08
Parsimony correction	Root mean square error of approximation (RMSEA)	and its 90% CI ≤ 0.08, CFit P>0.05
Comparative fit	Comparative fit index (CFI)	≥ 0.95
	Tucker-Lewis index (TLI)	

2. Localized areas of misfit (Brown, 2015)

- Residuals

Residuals are the difference between the values in the sample and model-implied variance-covariance matrices.

Standardized residuals (SRs) $> |2.58|$ indicate the standardized discrepancy between the matrices.

- Modification indices (MIs)

A modification index indicates the expected parameter change if we include a particular specification in the model (i.e. a constrained/fixed parameter is freely estimated, e.g. by correlating between errors of Q1 and Q2).

Specifications with MIs > 3.84 should be investigated.

3. Parameter estimates

- Factor loadings (FLs) (`Std.all` column under `Latent Variables` table).

The guideline for EFA is applicable also to CFA. For example, FLs ≥ 0.5 are practically significant. In addition, the P-values of the FLs must be significant (at $\alpha = 0.05$).

Also look for out-of-range values. FLs should be in range of 0 to 1 (absolute values), thus values > 1 are called *Heywood cases* or *offending estimates* (Brown, 2015)

- Factor correlations (`Std.all` column under `Covariances` table).

Similar to EFA, a factor correlation must be < 0.85, which indicates that the factors are distinct. A correlation > 0.85 indicates multicollinearity problem. Also look for out-of-range values. Factor correlations should be in range of 0 to 1 (absolute values).

In addition, when a model has Heywood cases, the solution is not acceptable. The variance-covariance matrix (of our data) could be *non-positive definite* i.e. the matrix is not invertible for the analysis.

In our output:

Fit indices,

```
##
##   Number of observations                           150
##
##   Estimator                                         ML      Robust
##   Model Fit Test Statistic                      37.063      27.373
##   Degrees of freedom                                19          19
##   Comparative Fit Index (CFI)                    0.958       0.972
##   Tucker-Lewis Index (TLI)                       0.937       0.958
##
##   90 Percent Confidence Interval          0.040  0.118       0.000  0.091
##   P-value RMSEA <= 0.05                          0.098       0.397
##
```

The model has good model fit based on all indices, with the exception of the upper 90% CI of robust RMSEA = 0.112. Please note there is no CFit P-value for robust RMSEA.

FLs and factor correlation,

```
##   Information                                 Observed
##   Observed information based on                Hessian
##   Standard Errors                   Robust.huber.white
##  Latent Variables:
##                      Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   PA1 =~
##     Q4                  1.000                               0.952    0.814
##     Q5                  0.660    0.092    7.218    0.000    0.629    0.624
##     Q6                  0.810    0.090    9.010    0.000    0.771    0.708
##     Q7                  0.916    0.086   10.641    0.000    0.872    0.735
##     Q11                 0.533    0.093    5.719    0.000    0.507    0.544
##   PA2 =~
##     Q8                  1.000                               0.653    0.655
##     Q9                  1.347    0.156    8.654    0.000    0.880    0.844
##     Q10                 1.436    0.199    7.206    0.000    0.938    0.856
```

Remember to read the results down the `Std.all` column. All FLs > 0.5 and the factor correlation < 0.85. There is no problem with the item quality and the factors are distinct.

In addition, to obtain the standardized results with SE,

```r
standardizedSolution(cfa.model)  # standardized, to view the SE of FL
```

```
##    lhs op rhs est.std    se      z pvalue ci.lower ci.upper
## 1  PA1 =~  Q4   0.814 0.041 20.017  0.000    0.735    0.894
## 2  PA1 =~  Q5   0.624 0.068  9.117  0.000    0.490    0.758
## 3  PA1 =~  Q6   0.708 0.060 11.867  0.000    0.591    0.825
## 4  PA1 =~  Q7   0.735 0.059 12.523  0.000    0.620    0.850
## 5  PA1 =~ Q11   0.544 0.080  6.781  0.000    0.387    0.702
## 6  PA2 =~  Q8   0.655 0.067  9.712  0.000    0.523    0.788
## 7  PA2 =~  Q9   0.844 0.048 17.646  0.000    0.751    0.938
## 8  PA2 =~ Q10   0.856 0.048 17.670  0.000    0.761    0.951
## 9   Q4 ~~  Q4   0.337 0.066  5.078  0.000    0.207    0.467
## 10  Q5 ~~  Q5   0.611 0.085  7.156  0.000    0.444    0.778
## 11  Q6 ~~  Q6   0.498 0.085  5.894  0.000    0.333    0.664
## 12  Q7 ~~  Q7   0.460 0.086  5.324  0.000    0.290    0.629
```

```
## 13 Q11 ~~ Q11   0.704 0.087  8.049  0.000    0.532    0.875
## 14  Q8 ~~  Q8   0.570 0.088  6.446  0.000    0.397    0.744
## 15  Q9 ~~  Q9   0.287 0.081  3.550  0.000    0.128    0.445
## 16 Q10 ~~ Q10   0.267 0.083  3.226  0.001    0.105    0.430
## 17 PA1 ~~ PA1   1.000 0.000     NA     NA    1.000    1.000
## 18 PA2 ~~ PA2   1.000 0.000     NA     NA    1.000    1.000
## 19 PA1 ~~ PA2   0.124 0.113  1.099  0.272   -0.098    0.346
```

and, to obtain the unstandardized results with 95% CI,

```
parameterEstimates(cfa.model)  # unstandardized, to view the 95% CI
```

```
##    lhs op rhs   est    se      z pvalue ci.lower ci.upper
## 1  PA1 =~  Q4 1.000 0.000     NA     NA    1.000    1.000
## 2  PA1 =~  Q5 0.660 0.092  7.218  0.000    0.481    0.840
## 3  PA1 =~  Q6 0.810 0.090  9.010  0.000    0.634    0.986
## 4  PA1 =~  Q7 0.916 0.086 10.641  0.000    0.748    1.085
## 5  PA1 =~ Q11 0.533 0.093  5.719  0.000    0.350    0.716
## 6  PA2 =~  Q8 1.000 0.000     NA     NA    1.000    1.000
## 7  PA2 =~  Q9 1.347 0.156  8.654  0.000    1.042    1.652
## 8  PA2 =~ Q10 1.436 0.199  7.206  0.000    1.046    1.827
## 9   Q4 ~~  Q4 0.460 0.089  5.182  0.000    0.286    0.633
## 10  Q5 ~~  Q5 0.620 0.086  7.178  0.000    0.451    0.789
## 11  Q6 ~~  Q6 0.590 0.101  5.836  0.000    0.392    0.788
## 12  Q7 ~~  Q7 0.647 0.125  5.181  0.000    0.402    0.891
## 13 Q11 ~~ Q11 0.611 0.077  7.934  0.000    0.460    0.762
## 14  Q8 ~~  Q8 0.567 0.101  5.628  0.000    0.369    0.764
## 15  Q9 ~~  Q9 0.312 0.094  3.325  0.001    0.128    0.495
## 16 Q10 ~~ Q10 0.321 0.106  3.046  0.002    0.115    0.528
## 17 PA1 ~~ PA1 0.906 0.137  6.587  0.000    0.636    1.175
## 18 PA2 ~~ PA2 0.427 0.106  4.009  0.000    0.218    0.635
## 19 PA1 ~~ PA2 0.077 0.075  1.035  0.301   -0.069    0.224
```

Localized areas of misfit,

```
mi = modificationIndices(cfa.model)
subset(mi, mi > 3.84)  # since we are using MLR, look at 'mi'
```

```
##    lhs op rhs     mi   epc sepc.lv sepc.all sepc.nox
## 20 PA1 =~  Q8 10.264 0.244   0.232    0.233    0.233
## 24 PA2 =~  Q5  8.359 0.332   0.217    0.215    0.215
## 37  Q5 ~~ Q11  6.301 0.144   0.144    0.234    0.234
## 55  Q9 ~~ Q10 10.264 2.325   2.325    7.346    7.346
```

```
sr = residuals(cfa.model, type="standardized"); sr
```

```
## $type
## [1] "standardized"
##
## $cov
##         Q4     Q5     Q6     Q7    Q11     Q8     Q9    Q10
## Q4   0.000
## Q5   0.007  0.000
## Q6  -0.159 -0.361  0.000
## Q7   0.214 -1.113  0.594  0.000
## Q11 -0.075  1.568 -0.878 -0.657  0.000
## Q8   1.599  3.397  1.464  1.268  1.139  0.000
```

```
## Q9   -0.940  2.065  0.131 -1.397  0.936 -0.116  0.000
## Q10 -1.092  1.495 -1.121 -1.319  0.354 -0.042  0.065  0.000
```

There are four suggested specifications with MIs > 3.84. We may ignore `PA1 =~  Q8` and `PA2 =~  Q5` based on content, because it is not justifiable to allow these two items specified under other factors. `Q9 ~~ Q10` is justifiable, based on the wording "is important". But `Q5 ~~ Q11` is not justifiable.

Q5 has one problematic SR with Q8 (SR = 3.397). So we may need to handle Q5 (or its pair Q8) too.

3.4 Step 3

Whenever the model do not fit well, we must revise the model. To do so, we must look for the causes of the poor fit to the data. The causes in CFA could be:

1. Item – the item has low FL (< 0.3), is specified to load on wrong factor or has cross-loading issue.
2. Factor – the factors have multicollinearity problem (correlation > 0.85), or the presence of redundant factors in a model. This can detected by residuals and MIs.
3. Correlated error (method effect) – some items are similarly worded (e.g. "I like …", "I believe…") or have almost similar meaning/content. This is usually detected by residuals and MIs.
4. Improper solution – the solution with Heywood cases. It could be because the specified model is not supported by the data and the misspecification could be a combination of all the first three causes listed above. A small sample may also lead to improper solution.

The problems might not surface if a proper EFA is done in the first place and the model is theoretically sound.

Model-to-model comparison following revision is done based on:

1. χ^2 difference
   - for nested[3] models only.
2. AIC (Akaike Information Criterion) and BIC (Bayesian Information Criterion)
   - for nested and unnested models.
   - an improvement in the model is shown as a reduction in AIC and BIC values (Brown, 2015). Better model = Smaller AIC/BIC.

Model revision

Revision 1: Based on MI, `Q9 ~~ Q10`?

Both from PA2, reasonable by the wording of the questions.

```
model1 = "
PA1 =~ Q4 + Q5 + Q6 + Q7 + Q11
PA2 =~ Q8 + Q9 + Q10
Q9 ~~ Q10
"
cfa.model1 = cfa(model1, data = data.cfa, estimator = "MLR")
```

```
## Warning in lav_object_post_check(object): lavaan WARNING: some estimated ov variances are
## negative
```

```
summary(cfa.model1, fit.measures=T, standardized=T)
```

Take note of the warning message above! It points out to problem(s) in our model. Here we have *negative variance*. Remember that variance is the square of standard deviation, thus it is impossible to have a negative variance!

[3] model with same number of items, but with different model specifications e.g. number of factors

```
##
##   Number of observations                           150
##
##   Estimator                                         ML      Robust
##   Model Fit Test Statistic                      26.487      19.771
##   Degrees of freedom                                18          18
##   Comparative Fit Index (CFI)                    0.980       0.994
##   Tucker-Lewis Index (TLI)                       0.969       0.991
##
##   90 Percent Confidence Interval          0.000  0.099       0.000  0.073
##   P-value RMSEA <= 0.05                          0.376       0.754
##
```

The upper 90% CI of RMSEA is smaller, but

```
##   Information                                 Observed
##   Observed information based on                Hessian
##   Standard Errors                   Robust.huber.white
##  Latent Variables:
##                      Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   PA1 =~
##     Q4                  1.000                                0.950    0.813
##     Q5                  0.663    0.089    7.470    0.000    0.630    0.626
##     Q6                  0.811    0.091    8.876    0.000    0.771    0.708
##     Q7                  0.923    0.089   10.409    0.000    0.877    0.739
##     Q11                 0.528    0.092    5.729    0.000    0.502    0.538
##   PA2 =~
##     Q8                  1.000                                1.517    1.522
##     Q9                  0.247    0.314    0.786    0.432    0.374    0.359
##     Q10                 0.267    0.332    0.803    0.422    0.404    0.369
##  Covariances:
##                      Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
```

we have a serious Heywood case here! Q8 FL = 1.522. Thus this solution is not acceptable.

Revision 2: Remove Q5? Because Q5 has problematic SR with Q8. (You may also try removing Q8 instead).

```
model2 = "
PA1 =~ Q4 + Q6 + Q7 + Q11
PA2 =~ Q8 + Q9 + Q10
"
cfa.model2 = cfa(model2, data = data.cfa, estimator = "MLR")
summary(cfa.model2, fit.measures=T, standardized=T)
```

```
##
##   Number of observations                           150
##
##   Estimator                                         ML      Robust
##   Model Fit Test Statistic                      20.451      14.467
##   Degrees of freedom                                13          13
##   Comparative Fit Index (CFI)                    0.979       0.994
##   Tucker-Lewis Index (TLI)                       0.966       0.990
##
##   90 Percent Confidence Interval          0.000  0.111       0.000  0.079
##   P-value RMSEA <= 0.05                          0.314       0.704
##
```

The upper 90% CI of RMSEA has reduced from 0.112 to 0.104.

```
##   Information                                      Observed
##   Observed information based on                     Hessian
##   Standard Errors                        Robust.huber.white
##  Latent Variables:
##                      Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   PA1 =~
##     Q4                  1.000                               0.941    0.806
##     Q6                  0.830    0.099    8.391    0.000    0.781    0.718
##     Q7                  0.960    0.100    9.597    0.000    0.904    0.762
##     Q11                 0.504    0.091    5.547    0.000    0.474    0.509
##   PA2 =~
##     Q8                  1.000                               0.651    0.653
##     Q9                  1.351    0.155    8.692    0.000    0.880    0.844
##     Q10                 1.444    0.202    7.143    0.000    0.940    0.858
##  Covariances:
```

The FLs and factor correlation are acceptable. No Heywood's case.

```
mi2 = modificationIndices(cfa.model2)
subset(mi2, mi > 3.84)
```

```
##    lhs op rhs    mi   epc sepc.lv sepc.all sepc.nox
## 18 PA1 =~  Q8 9.707 0.241   0.227    0.228    0.228
## 45  Q9 ~~ Q10 9.707 3.661   3.661   11.615   11.615
```

```
sr2 = residuals(cfa.model2, type="standardized"); sr2
```

```
## $type
## [1] "standardized"
##
## $cov
##     Q4     Q6     Q7     Q11    Q8     Q9     Q10
## Q4   0.000
## Q6  -0.231  0.000
## Q7  -0.089  0.390  0.000
## Q11  0.594 -0.408 -0.461  0.000
## Q8   1.970  1.759  1.579  1.407  0.000
## Q9  -0.495  0.488 -1.183  1.325 -0.115  0.000
## Q10 -0.524 -0.771 -1.043  0.650 -0.051  0.077  0.000
```

There is no more SR > 2.58.

So we may stop at **model2**, although the upper 90% CI of RMSEA is still > 0.08, but there is no more localized areas of misfit by SR.

Model-to-model comparison

Because **model2** is not nested in **model**, we compare mainly by AIC and BIC, and additionally by χ^2 difference (in our case scaled χ^2 difference),

```
anova(cfa.model, cfa.model2, method = "satorra.bentler.2010")
```

```
## Scaled Chi Square Difference Test (method = "satorra.bentler.2001")
##
##            Df  AIC    BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## cfa.model2 13 2791 2836.2 20.451
## cfa.model  19 3166 3217.2 37.063     13.562       6    0.03493 *
```

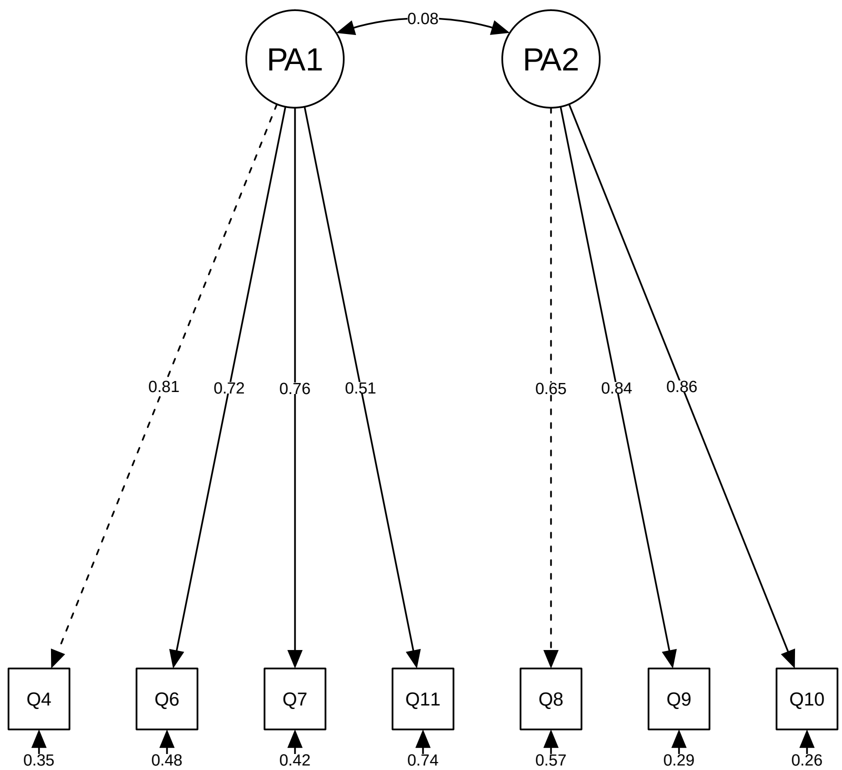
```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Clearly, the AIC and BIC are reduced (`model2` [without Q5] vs `model` [with Q5]). The χ^2 difference is significant, which indicates an improvement in model fit.

4 Path diagram

A CFA model can be nicely presented in the form of path diagram.

```
semPaths(cfa.model2, 'path', 'std', style = 'lisrel',
         edge.color = 'black', intercepts = F)
```



5 Results presentation

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

1. The estimation method e.g. ML, MLR, WLSMV etc.
2. The model specification and the theoretical background supporting the model.
3. Details about the selected fit indices, residuals, MIs, FLs and factor correlations and the accepted cut-off values.
4. Detailed comments on the fit and parameters of the tested models. This is usually done in reference to summary tables.
5. Details about the revision process, i.e. item deletion, addition of correlated errors or any other modifications and the effects on the model fit. Also mention the reasons e.g. high SRs, low Fls etc.
6. Summary tables, which outlines the model fit indices, model comparison, FLs, communalities and factor correlations.
7. The path diagram (most of the time, of the final model). This may be requested by some journals.

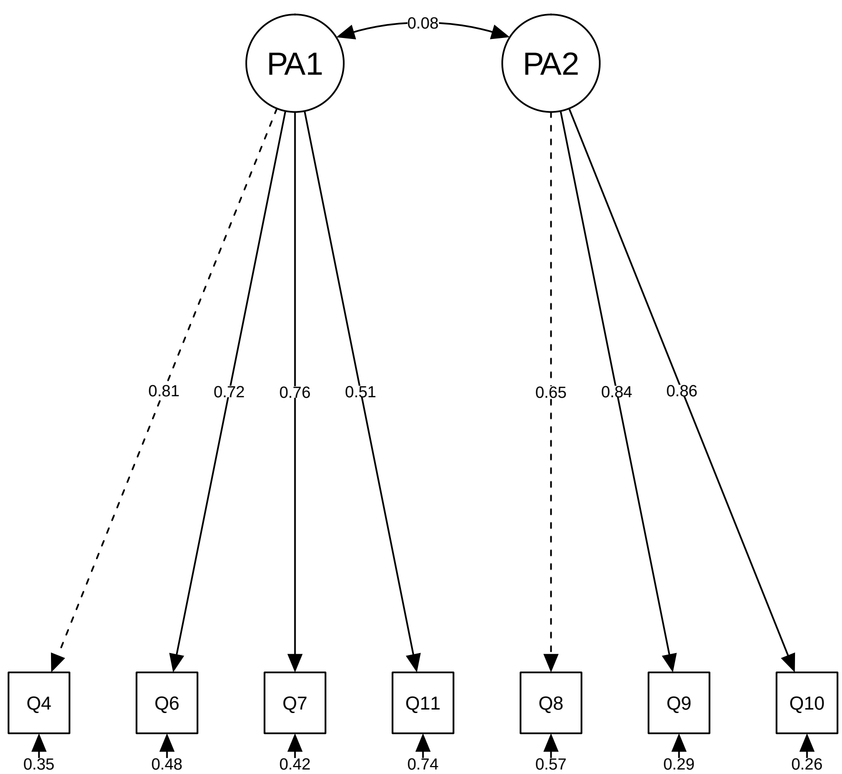
Fit indices of the models.

Model	χ^2(df)	P	χ^2_{diff}(df)	P	SRMR	RMSEA	90% CI	CFI	TLI	AIC	BIC
Model	27.4(19)	0.096	-		0.072	0.063	0.000, 0.112	0.973	0.960	3182	3257
Model 2	14.5(13)	0.342	13.6 (6)	0.035	0.057	0.033	0.000, 0.104	0.994	0.990	2805	2871

Factor loadings of Model 2.

Factor	Item	Factor loading
Affinity	Q4	0.806
	Q6	0.718
	Q7	0.762
	Q11	0.509
Importance	Q8	0.653
	Q9	0.844
	Q10	0.858
Factor correlation: Affinity ↔ Importance r = 0.078.		

The path diagram of Model 2.



References

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

Epskamp, S., & Simon Stuber. (2017). *SemPlot: Path diagrams and visual analysis of various sem packages' output*. Retrieved from https://CRAN.R-project.org/package=semPlot

Jorgensen, T. D., Pornprasertmanit, S., Schoemann, A., & Rosseel, Y. (2018). *SemTools: Useful tools for structural equation modeling*. Retrieved from https://CRAN.R-project.org/package=semTools

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

Rosseel, Y. (2018). *Lavaan: Latent variable analysis*. Retrieved from https://CRAN.R-project.org/package=lavaan

Schreiber, J. B., Nora, A., Stage, F. K., Barlow, E. A., & King, J. (2006). Reporting structural equation modeling and confirmatory factor analysis results: A review. *The Journal of Educational Research*, *99*(6), 323–338.