Confirmatory Factor Analysis

Theory Construction and Statistical Modeling

Kyle M. Lang

Department of Methodology & Statistics
Utrecht University

Outline

SAPI

EFA and CFA

Confirmatory or Exploratory?

CFA in R

Scaling

Extra

South African Personality Inventory Project

SAPI©

Carin Hill
Leon Jackson
Deon Meiring
J. Aleweyn Nel

Ian Rothmann
Michael Temane
Velichko H. Valchev
Fons J. R. van de Vijver

Nel, J. A., Valchev, V. H., Rothmann, S., van de Vijver, F. J. R., Meiring, D., & de Bruin, G. P. (2012). Exploring the personality structure in the 11 languages of South Africa. *Journal of Personality*, *80*, 915–948.

SAPI details

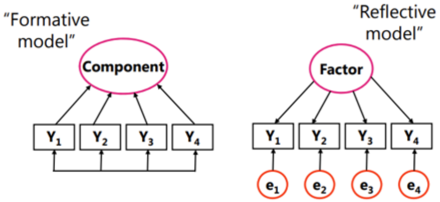
- 1216 participants from 11 official language groups
- From about 50,000 descriptive responses to 262 personality items
- Nine personality clusters:
  - Conscientiousness
  - Emotional Stability
  - Extraversion
  - Facilitating
  - Integrity
  - Intellect
  - Openness
  - Relationship Harmony
  - Soft-Heartedness (Ubuntu)
- Our data: selection of 1000 participants

Factor Analysis

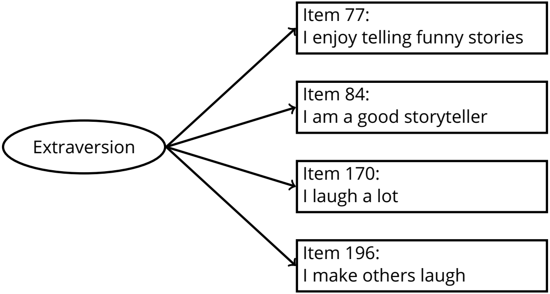
Factor Analysis: Modeling measurement of a latent variable

- EFA: Exploratory Factor Analysis.
- CFA: Confirmatory Factor Analysis.

Both EFA and CFA use a ”reflective” measurement model, not a ”formative” model.

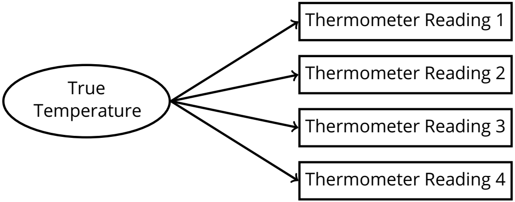


Reflective Constructs



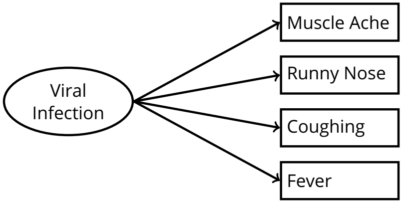
- Items are dependent variables, caused by the factor!
- Latent variable ‘extraversion’ explains item correlations:
  The factor is the reason for the covariances/correlations.

Reflective Constructs



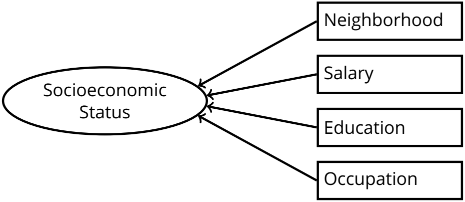
Thermometer readings are the dependent variables, caused by the temperature!

Reflective Constructs



Symptoms are the dependent variables, caused by the viral infection!

Formative Constructs



SES is an *index* defined as a (weighted) sum of the observed items.

- SES is the (latent) dependent variable, predicted by the items.
- This model is not empirically testable.

Interesting read

Interesting read on theory & latent variables:

Borsboom, D., Mellenbergh, G.J., & Van Heerden, J. (2003). The theoretical status of latent variables. *Psychological review, 110*(2), 203.

CONFIRMATORY OR EXPLORATORY?

Two Subscales of Extraversion

Having Fun

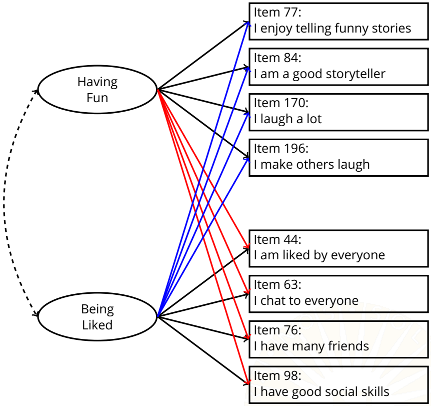
- Item 77: I enjoy telling funny stories
- Item 84: I am a good storyteller
- Item 170: I laugh a lot
- Item 196: I make others laugh

Being Liked

- Item 44: I am liked by everyone
- Item 63: I chat to everyone
- Item 76: I have many friends
- Item 98: I have good social skills

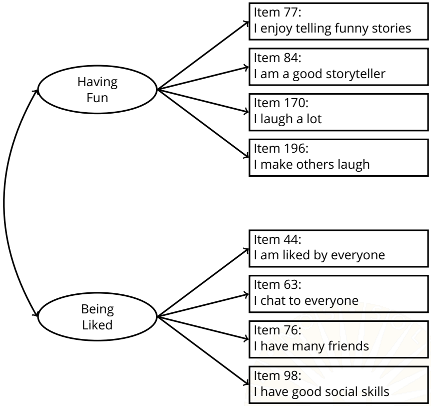
EFA

- All items load onto all factors
- No hypothesized measurement model
- Estimating latent covariances is optional
  - Oblique factors → Estimated
  - Orthogonal factors → Fixed
- Solution is not unique
- Use rotation to improve interpretability



CFA

- The statistical model represents the hypothesized measurement model
- No cross-loadings unless they're predicted by theory
- Almost always estimate the latent covariances
- A unique solution exists



CFA IN R

Example: Estimate a CFA Model

Load the SAPI data.

```
dataDir <- "../data/"
sapi <- read.table(paste0(dataDir, "sapi.txt"),
                   header = TRUE,
                   na.strings = "-999")
```

Specify the **lavaan** model syntax for the SAPI extraversion CFA.

```
mod1 <- '
fun   =~ Q77 + Q84 + Q170 + Q196
liked =~ Q44 + Q63 + Q76  + Q98
'
```

Use the `cfa()` function to estimate the model.

```
library(lavaan)
out1 <- cfa(mod1, data = sapi)
```

Example: Summarize the Fitted CFA

```
partSummary(out1, 1:4)

lavaan 0.6-18 ended normally after 30 iterations

  Estimator                                         ML
  Optimization method                           NLMINB
  Number of model parameters                        17

                                                  Used       Total
  Number of observations                           959        1000

Model Test User Model:

  Test statistic                               130.193
  Degrees of freedom                                19
  P-value (Chi-square)                           0.000
```

Example: Summarize the Fitted CFA

```
partSummary(out1, 5:7)

Parameter Estimates:

  Standard errors                              Standard
  Information                                  Expected
  Information saturated (h1) model           Structured

Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)
  fun =~
    Q77               1.000
    Q84               0.761    0.051   14.902    0.000
    Q170              0.634    0.047   13.558    0.000
    Q196              0.795    0.046   17.381    0.000
  liked =~
    Q44               1.000
    Q63               1.512    0.147   10.278    0.000
    Q76               1.483    0.149    9.955    0.000
    Q98               1.243    0.119   10.462    0.000
```

Example: Summarize the Fitted CFA

```
partSummary(out1, 8:9)

Covariances:
                   Estimate  Std.Err  z-value  P(>|z|)
  fun ~~
    liked             0.231    0.025    9.234    0.000

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)
   .Q77               0.548    0.038   14.389    0.000
   .Q84               0.727    0.039   18.703    0.000
   .Q170              0.687    0.035   19.572    0.000
   .Q196              0.364    0.025   14.731    0.000
   .Q44               0.662    0.034   19.291    0.000
   .Q63               0.807    0.048   16.943    0.000
   .Q76               0.966    0.054   17.931    0.000
   .Q98               0.469    0.029   16.121    0.000
    fun               0.627    0.056   11.303    0.000
    liked             0.182    0.029    6.290    0.000
```

Example: Model Fit Statistics

Example: Model Fit Statistics

```
fitMeasures(out1)

               npar                fmin               chisq
             17.000               0.068             130.193
                 df              pvalue      baseline.chisq
             19.000               0.000            1574.886
        baseline.df     baseline.pvalue                 cfi
             28.000               0.000               0.928
                tli                nnfi                 rfi
              0.894               0.894               0.878
                nfi                pnfi                 ifi
              0.917               0.622               0.929
                rni                logl   unrestricted.logl
              0.928          -10147.587          -10082.491
                aic                 bic              ntotal
          20329.175           20411.895             959.000
               bic2               rmsea      rmsea.ci.lower
          20357.903               0.078               0.066
     rmsea.ci.upper      rmsea.ci.level        rmsea.pvalue
              0.091               0.900               0.000
     rmsea.close.h0 rmsea.notclose.pvalue  rmsea.notclose.h0
              0.050               0.421               0.080
```

Example: Visualize the Fitted CFA

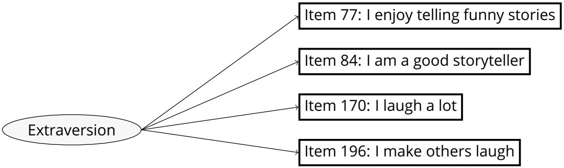
```r
library(lavaanPlot)
lavaanPlot(model = out1,
           node_options = list(shape = "box",
                               fontname = "Helvetica"),
           edge_options = list(color = "grey"),
           coefs = TRUE,
           stand = TRUE,
           covs = TRUE)
```

Example: Visualize the Fitted CFA

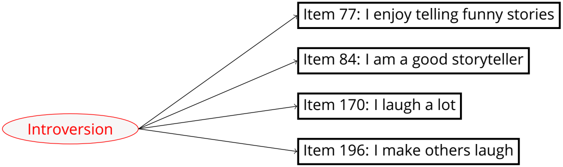
```
Error in path.expand(path):  invalid 'path' argument
```

Latent variable scaling

Latent variables are not observed, thus no inherent scale.



Latent variable scaling Ctd.



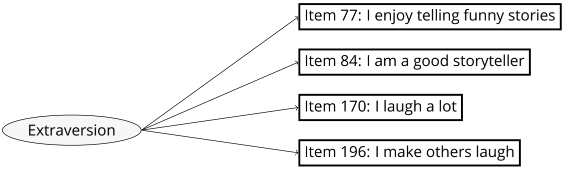
Therefore, set up model such that scale of latent variable is clear.

Three common ways

1. Marker-variable method
Constrain one of the factor loadings (default).

2. Reference group method:
Constrain the factor variance.

3. Effect coding:
Constrain the average of the loadings.



1. Marker-variable method (default)

Default parameterization:

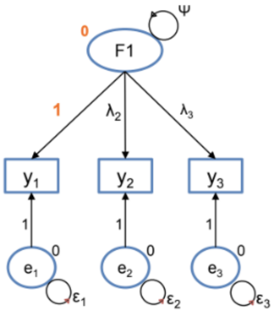
- First factor loading constrained at 1.
- Factor mean constrained at 0.

Other defaults:

- Mean of residuals is by definition 0.
- Residuals have a loading of 1.

Estimated:

- factor variance (Ψ),
- 'other' factor loadings (λ_2, λ_3),
- all item intercepts (ν_1, ν_2, ν_3),
- all residual variances (ϵ_1, ϵ_2, ϵ_3).



1. Default marker-variable method - lavaan

```
# Model
model.1CFA <- '
 Extraversion =~ Q77 + Q84 + Q170 + Q196
'

# Fit model
fit_1CFA <- cfa(model.1CFA, data=data_sapi,
                missing='fiml', fixed.x=F)  # use FIML

Error in eval(sc, parent.frame()):  object 'data_sapi' not found
```

- First factor loading constrained at 1:

```
Extraversion =~
  Q77                   1.000
```

- Factor mean constrained at 0:

```
Extraversion      0.000
```

1. Default marker-variable method - lavaan Ctd

```
parameterEstimates(fit_1CFA)[1:4,-c(5,6,7)]

Error in eval(expr, envir, enclos):  object 'fit_1CFA' not found
```

Factor loading of first indicator fixed to 1.
all other loadings are relative to that.

If reference category changed, other loadings also change.

2. Reference-group method

Parameterization:

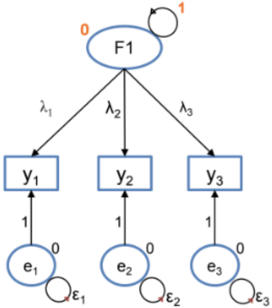
- Factor variance constrained at 1.
- Factor mean constrained at 0.

Defaults:

- Mean of residuals is by definition 0.
- Residuals have a loading of 1.

Estimated:

- all factor loadings (λ_1, λ_2, λ_3),
- all item intercepts (ν_1, ν_2, ν_3),
- all residual variances (ϵ_1, ϵ_2, ϵ_3).



2. Reference-group method - lavaan

```
# Model
model.1CFA_RefGr <- '
  # Free first factor loading, using: NA*
  Extraversion =~ NA*Q77 + Q84 + Q170 + Q196

  # Set factor variance to 1, using: 1*
  Extraversion ~~ 1*Extraversion
 '


# Fit model
fit_1CFA_RefGr <- cfa(model.1CFA_RefGr, data=data_sapi,
                      missing='fiml', fixed.x=F)  # use FIML

Error in eval(sc, parent.frame()):  object 'data_sapi' not found
```

- Factor variance constrained at 1:

```
Extraversion        1.000
```

- Factor mean constrained at 0:

```
Extraversion        0.000
```

2. Reference-group method - lavaan Ctd

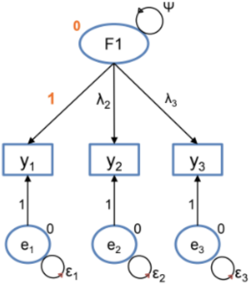
```
parameterEstimates(fit_1CFA_RefGr)[1:4,-c(5,6,7)]

Error in eval(expr, envir, enclos):  object 'fit_1CFA_RefGr' not found
```

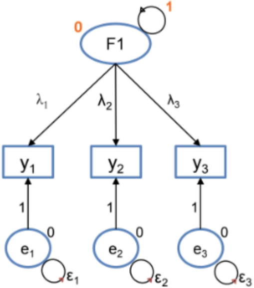
Advantage:
All factor loadings and scores on standardized metric.

Which method to choose?

1. Marker-variable method



2. Reference-group method



Does not matter for substantive conclusions.
Sometimes, pragmatic reasons.

3. Effects-coding method

Parameterization:

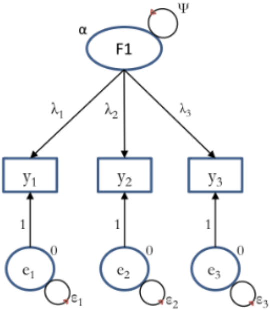
- Constrain the average of the factor loadings to 1: $\frac{1}{3}\sum_{i=1}^{3}\lambda_i = 1$.
- Constrain the average of the item intercepts to 0: $\frac{1}{3}\sum_{i=1}^{3}\nu_i = 0$.

Defaults:

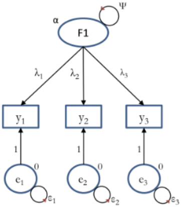
- Mean of residuals is by definition 0.
- Residuals have a loading of 1.

Estimated (subject to the constraints):

- factor variance (Ψ),
- factor mean (α),
- all factor loadings (λ_1, λ_2, λ_3),
- all item intercepts (ν_1, ν_2, ν_3),
- all residual variances (ϵ_1, ϵ_2, ϵ_3).



3. Effects-coding method Ctd



Interpretations can be intuitive:

- Factor on similar scale as the indicators.
- Factor variance (Ψ): average variance of each indicator that can be explained by the factor.
- Factor mean (α): weighted mean of the indicator means

3. Effects-coding method - lavaan model

```
# Model
model.1CFA_EffC <- '
  # Label parameters, such that they can be constrained
  Extraversion =~ lambda1*Q77 + lambda2*Q84 +
                  lambda3*Q170 + lambda4*Q196
  # intercepts
  Q77  ~ nu1*1
  Q84  ~ nu2*1
  Q170 ~ nu3*1
  Q196 ~ nu4*1

  # Constrain average of loadings to 1, i.e., set sum to 4
  lambda1 == 4 - lambda2 - lambda3 - lambda4
  # Constrain average of item intercepts to 0,
  # i.e., set sum to 0
  nu1 == 0 - nu2 - nu3 - nu4
 '
```

3. Effects-coding method - fit lavaan model

Now, use the lavaan() function:

```
# Fit model: Now, use the lavaan() function!
fit_1CFA_EffC <- lavaan(model.1CFA_EffC, data=data_sapi,
                        missing='fiml', fixed.x=F,
                        auto.var = TRUE,
                        auto.fix.first = FALSE,
                        auto.cov.lv.x = TRUE,
                        int.ov.free = TRUE)

Error in eval(expr, envir, enclos):  object 'data_sapi' not found
```

3. Effects-coding method - lavaan outpu

- Constrain the average of the factor loadings to 1: $\frac{1}{4}\sum_{i=1}^{4}\lambda_i = 1$.

```
parameterEstimates(fit_1CFA_EffC)[1:4,1:5]
Error in eval(expr, envir, enclos):  object 'fit_1CFA_EffC' not found
```

- Constrain the average of the item intercepts to 0: $\frac{1}{4}\sum_{i=1}^{4}\nu_i = 0$.

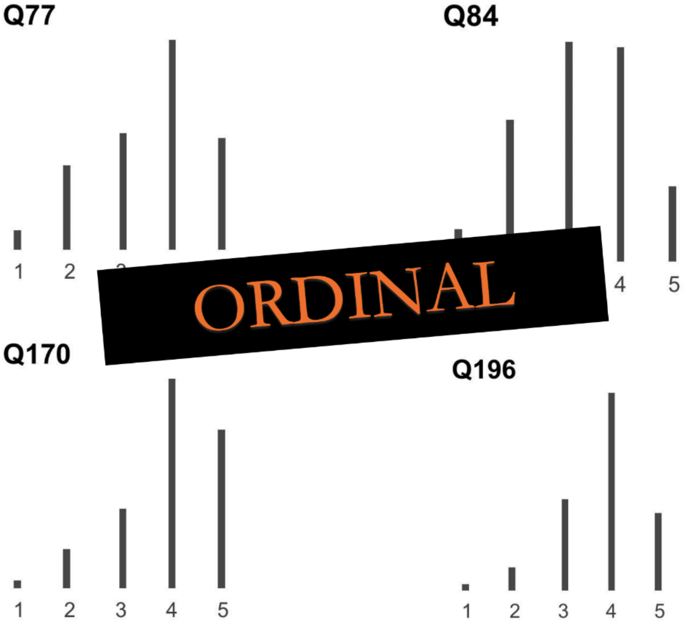
```
parameterEstimates(fit_1CFA_EffC)[5:8,1:5]
Error in eval(expr, envir, enclos):  object 'fit_1CFA_EffC' not found
```

Extra:

Categorical/Ordinal or continuous indicators?

Note: in cfa() you can, for example use ‘ordered = TRUE’ for endogenous variable.
Default then: estimator = ”WLSMV”.

More information on: https://lavaan.ugent.be/tutorial/cat.html


Q77
1 2
Q84
4 5
ORDINAL
Q170
1 2 3 4 5
Q196
1 2 3 4 5

Remark!

Do NOT use a χ^2 test or IC (AIC or BIC)
to compare categorical and continuous models:

- Obviously not nested (so, no χ^2 test anyway).
- AND likelihoods of categorical and continuous indicator models are incomparable!

Note: χ^2 test and IC are based on (log) likelihood (= fit).

Interesting Reading

https://lavaan.ugent.be/tutorial/cat.html

Muthén, B. (1984). A general structural equation model with dichotomous, ordered categorical, and continuous latent variable indicators. Psychometrica, 49(1), 115-132.

Rhemtulla, M., Brosseau-Liard, P.É., & Savalei, V. (2012). When can categorical variables be treated as continuous? A comparison of robust continuous and categorical SEM estimation methods under suboptimal conditions. Psychological Methods, 17(3), 354-373.

Sventina, D., Rutkowski, D. (2020). Multiple group invariance with categorical outcomes using updated guidelines: an illustration using Mplus and the lavaan/semtools packages. Structural Equational Modelling: A Multidisciplinary Journal, 27(1), 111-130