

Confirmatory Factor Analysis (CFA) - School Trouble Example

Adam Garber

Norwegian University of Science and Technology - A Course in `MplusAutomation`

May 30, 2021

Outline

1. One more quick EFA: change rotation method (override the default)
 2. Specify a confirmatory factor analysis (CFA)
 3. Generate a path diagram of your CFA model
-

Getting started - following the routine...

1. Create an R-Project
2. Install packages
3. Load packages

R-Project instructions:

1. click “NEW PROJECT” (upper right corner of window)
2. choose option “NEW DIRECTORY”
3. choose location of project (on desktop OR in a designated folder)

Within R-studio under the files pane (bottom right):

1. click “New Folder” and name folder “data”
 2. click “New Folder” and name folder “efa_mplus”
 3. click “New Folder” and name folder “cfa_mplus”
-

DATA SOURCE: This lab exercise utilizes the NCES public-use dataset: Education Longitudinal Study of 2002 (Lauff & Ingels, 2014) [See website: nces.ed.gov](https://nces.ed.gov/ipeds/data/elsa/)

Begin

loading packages...

```
library(tidyverse)
library(MplusAutomation)
library(here)
library(semPlot)
library(stargazer)
```

Change starting location to folder 05-cfa

```
source("rep_functions.R")

change_here(glue("{project_location}/05-cfa"))

here()
```

```
## [1] "/Users/agarber/github/NTNU-workshop/05-cfa"
```

read in data

```
trouble_data <- read_csv("https://garberadamc.github.io/project-site/data/els_sub3_school_trouble.csv")
```

take a look at the EFA data (same indicators used for lab 4)

```
stargazer(as.data.frame(trouble_data), type="text", digits=1)
```

```
##
## =====
## Statistic   N   Mean St. Dev. Min Pctl(25) Pctl(75) Max
## -----
## BYS22A      718 1.5    0.6    1.0    1.0    2.0    3.0
## BYS22B      717 1.3    0.6    1.0    1.0    1.0    3.0
## BYS22C      714 1.3    0.6    1.0    1.0    1.0    3.0
## BYS22D      715 1.2    0.4    1.0    1.0    1.0    3.0
## BYS22E      711 1.3    0.6    1.0    1.0    1.0    3.0
## BYS22F      716 1.0    0.2    1.0    1.0    1.0    3.0
## BYS22G      716 1.2    0.4    1.0    1.0    1.0    3.0
## BYS22H      713 1.3    0.6    1.0    1.0    1.0    3.0
## BYS24A      719 2.3    1.2    1.0    1.0    3.0    5.0
## BYS24B      715 1.5    0.9    1.0    1.0    2.0    5.0
## BYS24C      712 2.6    1.1    1.0    2.0    3.0    5.0
## BYS24D      711 1.7    0.9    1.0    1.0    2.0    5.0
## BYS24E      719 1.2    0.6    1.0    1.0    1.0    5.0
## BYS24F      713 1.1    0.4    1.0    1.0    1.0    5.0
## BYS24G      719 1.0    0.2    1.0    1.0    1.0    5.0
## -----
```

replicate the SAME random split as last week (do not change the seed)

- get the size of half of original sample
- `floor()` helps with rounding

```
smp_size <- floor(0.50 * nrow(trouble_data))
```

set the seed to make partition reproducible

```
set.seed(20210530)
```

`sample()` randomly selects rows of size = "smp_size"

```
calibrate_smp <- sample(seq_len(nrow(trouble_data)), size = smp_size)
```

create two samples called "calibrate" & "validate"

```
calibrate <- trouble_data[calibrate_smp, ]  
validate <- trouble_data[-calibrate_smp, ]
```

reorder variables & remove column number 15

```
# removed BYS24G (due to low loading < .2)  
calibrate <- calibrate %>%  
  select(1,3:14,2)  
  
# do the same for the validate sample  
validate <- validate %>%  
  select(1,3:14,2)
```

Default rotation: Geomin Oblique

run an EFA with the "calibrate" sample

```
efa_geomin <- mplusObject(  
  TITLE = "Geomin Oblique EFA - LAB 5 DEMO",  
  VARIABLE =  
    "! removed BYS24G (due to low loading < .2)  
    usevar = BYS22A-BYS22B;",  
  
  ANALYSIS =  
    "type = efa 1 3;  
    estimator = mlr;  
    rotation = geomin; ! this is the default (added to be explicit)  
    parallel=50; ! run parallel analysis",  
  
  MODEL = "" ,  
  
  PLOT = "type = plot3;",  
  OUTPUT = "sampstat;",
```

```

usevariables = colnames(calibrate),
rdata = calibrate)

efa_geomin_fit <- mplusModeler(efa_geomin,
                             dataout=here("efa_mplus", "efa_geomin_oblique.dat"),
                             modelout=here("efa_mplus", "efa_geomin_oblique.inp"),
                             check=TRUE, run = TRUE, hashfilename = FALSE)

```

Rotation: Varimax Orthogonal

```

efa_varimax <- mplusObject(
  TITLE = "Varimax Orthogonal EFA - LAB 5 DEMO",
  VARIABLE =
    " ! removed BYS24G (due to low loading < .2)
    usevar = BYS22A-BYS22B;",

  ANALYSIS =
    "type = efa 1 3;
    estimator = mlr;
    rotation = varimax; ! orthogonal (no factor correlations)
    parallel=50; ! run parallel analysis",

  MODEL = "" ,

  PLOT = "type = plot3;",
  OUTPUT = "sampstat;",

  usevariables = colnames(calibrate),
  rdata = calibrate)

efa_varimax_fit <- mplusModeler(efa_varimax,
                                dataout=here("efa_mplus", "efa_varimax_orthogonal.dat"),
                                modelout=here("efa_mplus", "efa_varimax_orthogonal.inp"),
                                check=TRUE, run = TRUE, hashfilename = FALSE)

```

Estimate a Confirmatory Factor Analysis (CFA) model

```

cfa_validate <- mplusObject(

  TITLE = "CFA Validate",

  VARIABLE =
    "usevar = BYS22A-BYS22B;",

  ANALYSIS =
    "estimator = mlr;",

  MODEL =
    "FACTOR_1 by BYS22A BYS22C BYS22D BYS22E BYS22F BYS22G BYS22H;

```

```

    FACTOR_2 BY BYS24A BYS24B BYS24C BYS24D BYS24E BYS24F BYS22B;" ,

PLOT = "type = plot3;",

OUTPUT = "sampstat standardized residual modindices (3.84);",

usevariables = colnames(validate),
rdata = validate)

cfa_val_fit <- mplusModeler(cfa_validate,
                           dataout=here("cfa_mplus", "lab5_cfa_validate.dat"),
                           modelout=here("cfa_mplus", "lab5_cfa_validate.inp"),
                           check=TRUE, run = TRUE, hashfilename = FALSE)

```

create a path diagram of the CFA model Read in the model to R within the cfa_mplus folder

```

cfa_output <- readModels(here("cfa_mplus"), quiet = TRUE)

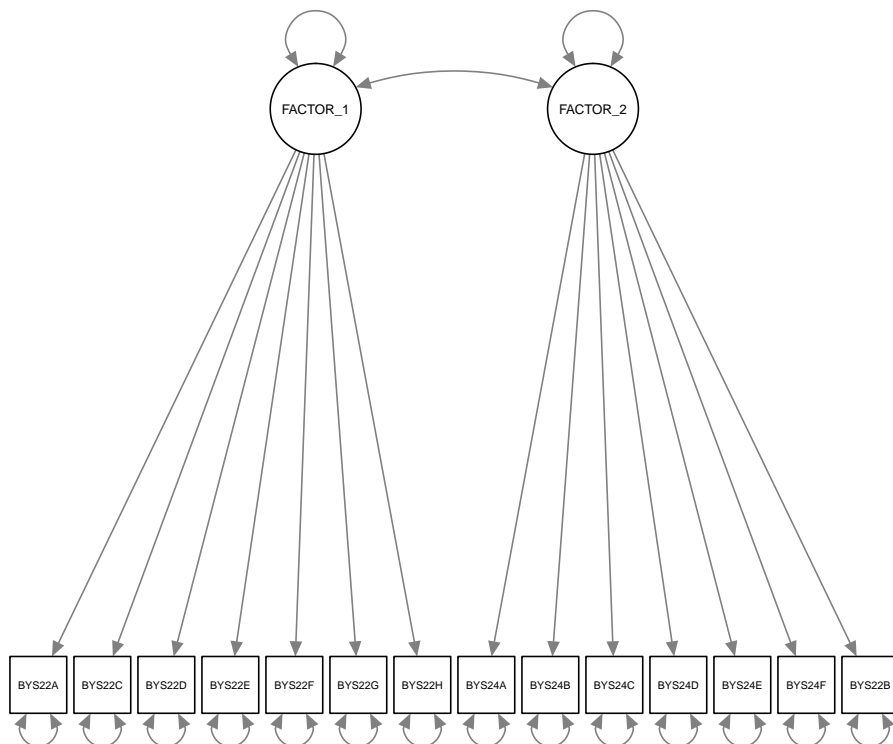
```

Plot model

```

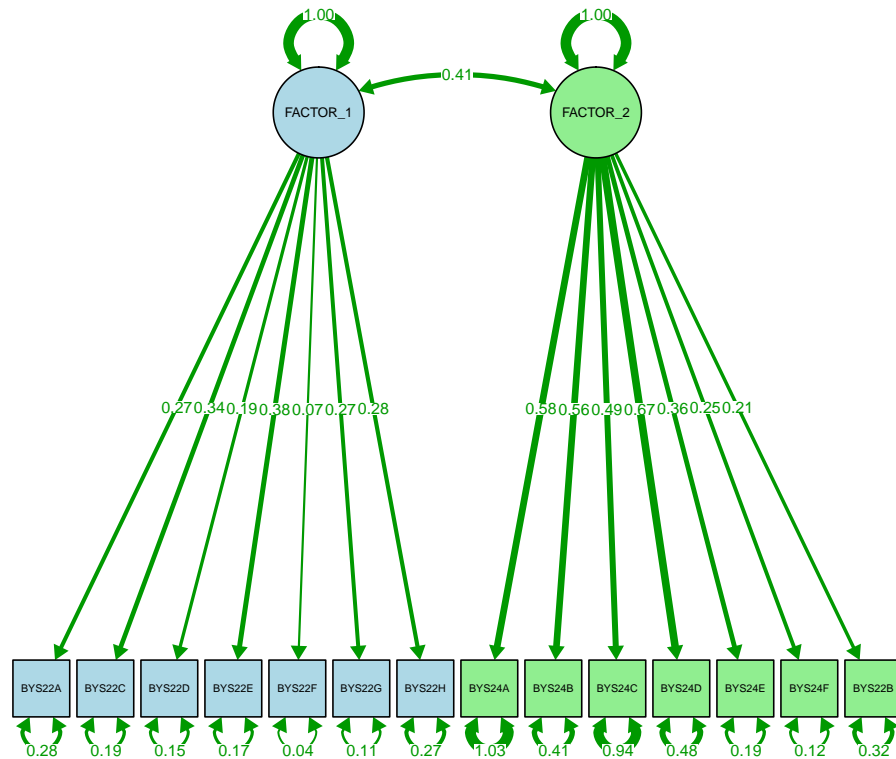
semPaths(cfa_output,
         intercepts=FALSE,
         fixedStyle = c(1)
         )

```



play with some formatting

```
semPaths(cfa_output, "std",
         intercepts=FALSE,
         fixedStyle = c(1),
         fade = FALSE,
         color= list(lat = c("light blue"," light green")),
         )
```



End

References

- Hallquist, M. N., & Wiley, J. F. (2018). MplusAutomation: An R Package for Facilitating Large-Scale Latent Variable Analyses in Mplus. *Structural equation modeling: a multidisciplinary journal*, 25(4), 621-638.
- Horst, A. (2020). Course & Workshop Materials. GitHub Repositories, <https://allisonhorst.github.io/>
- Muthén, L.K. and Muthén, B.O. (1998-2017). *Mplus User's Guide*. Eighth Edition. Los Angeles, CA: Muthén & Muthén
- R Core Team (2017). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, Austria. URL <http://www.R-project.org/>
- Wickham et al., (2019). Welcome to the tidyverse. *Journal of Open Source Software*, 4(43), 1686, <https://doi.org/10.21105/joss.01686>