

Confirmatory Factor Analysis

The data for this CFA exercise consist of correlations among 9 ‘ability’ variables collected by Holzinger & Swineford in one Chicago junior high school in 1939—the same data we used for EFA. The 9 variables consist of 3 perceptual tests, 3 verbal tests and 3 counting/speed tests, so one might expect a 3-factor solution would be sufficient to account for the correlations, with 3 non-overlapping factors. Here, we will allow them to be correlated. For simplicity, the variables are named V1-V9.

There is a lot of typing for this exercise; you may want to open the Word document, **CFA.doc** (in N:\psy6140\tutorials\) or the SAS file, N:\psy6140\examples\factor\calis16t.sas and copy/paste into SAS.

1. Read the correlation matrix into SAS, creating the data set psych9.

```

title 'Confirmatory Factor Analysis, Holzinger & Swineford';
data psych9(TYPE=CORR);
  _type_ = 'CORR';
  input _name_ $ V1-V9;
  label V1='Visual perception' V2='Cubes'
        V3='Lozenges' V4='Paragraph comprehension'
        V5='Sentence completion' V6='Word meaning'
        V7='Addition' V8='Counting dots'
        V9='Straight-curved capitals';
datalines;
V1      1.      .      .      .      .      .      .      .      .
V2      .318    1.      .      .      .      .      .      .      .
V3      .436    .419    1.      .      .      .      .      .      .
V4      .335    .234    .323    1.      .      .      .      .      .
V5      .304    .157    .283    .722    1.      .      .      .      .
V6      .326    .195    .350    .714    .685    1.      .      .      .
V7      .116    .057    .056    .203    .246    .170    1.      .      .
V8      .314    .145    .229    .095    .181    .113    .585    1.      .
V9      .489    .239    .361    .309    .345    .280    .408    .512    1.
;

```

2. **Using FACTOR notation:** There are several different ways, with different syntax, to specify a CFA model in PROC CALIS. One way is with the FACTOR statement, where you indicate the free and fixed parameters in three matrices, corresponding to the model

$$C = F P F' + U, \quad \text{with } P = P'$$

where F is the matrix of factor loadings, P is the matrix of factor correlations and U is the diagonal matrix of unique variances. (In the lecture notes, I use the notation $S = \Lambda \Phi \Lambda' + \Theta$.) With the FACTOR specification, you assign names to the **free** parameters in each matrix; ‘3 * Lam:’ is a shorthand for ‘Lam1 Lam2 Lam3’. **Fixed** parameters are assigned numbers.

```

proc calis data=psych9 method=max edf=144
  short mod outram=ram1;

```

CFA

```
title2 'Model A: Restricted Oblique Clusters';
FACTOR N=3;
matrix _F_                                /* factor loadings */
    [ ,1] = 3 * Lam: ,
    [ ,2] = 3 * 0 3 * Lam: ,
    [ ,3] = 6 * 0 3 * Lam: ;
matrix _P_                                /* factor correlations */
    [1, ] = 1. ,
    [2, ] = Cor: 1. ,
    [3, ] = 2 * Cor: 1. ;
matrix _U_                                /* unique variances */
    [1,1] = EPS1-EPS9 ;
bounds
    EPS1-EPS9 >= 0.;                    /* must be non-negative */
run;
```

- Examine the goodness of fit of this model; for now, look at the Chi-Square statistic and the Pr > Chi-Square. Does this model fit the data?
- Examine the modification indices for the factor loadings, in the table labeled 'Lagrange Multiplier and Wald Test Indices'. Which loading gives the largest value?

3. The display of goodness of fit statistics is ugly and hard to read. My macro `calisgfi` gives a nicer, formatted display, from the information in the OUTRAM data set:

```
%calisgfi(ram=ram1, nv=9);
```

4. **Using LINEQS notation:** Linear equation notation specifies a linear equation for each observed variable, using F1-F3 to denote factors and E1-E9 to denote the unique factors. The statements below specify the same model:

```
proc calis data=psych9 method=max edf=144 short mod outram=ram1;
title2 'Model A: Restricted Oblique Clusters';
lineqs
    V1 = Lam: F1                + E1,
    V2 = Lam: F1                + E2,
    V3 = Lam: F1                + E3,
    V4 =          Lam: F2        + E4,
    V5 =          Lam: F2        + E5,
    V6 =          Lam: F2        + E6,
    V7 =          Lam: F3        + E7,
    V8 =          Lam: F3        + E8,
    V9 =          Lam: F3        + E9;
std
    E1-E9 = EPS1-EPS9 ,
    F1-F3 = 3 * 1. ;
bounds
    EPS1-EPS9 >= 0.;
cov
    F1-F3 = Cor: ;
run;
```

CFA

5. **Model B:** Now we'll adjust this model by adding one more free parameter, allowing V9 (Straight & curved capitals) to load on both F1 and F3. (Typically, when you fit additional models, you copy/paste/edit the statements from one model to the next.)

```
proc calis data=psych9 method=max edf=144 short outram=ram2;
  title2 'Model B: 23 degrees of freedom (F[9,1] free)';
  lineqs
    V1 = Lam: F1                + E1,
    V2 = Lam: F1                + E2,
    V3 = Lam: F1                + E3,
    V4 =           Lam: F2       + E4,
    V5 =           Lam: F2       + E5,
    V6 =           Lam: F2       + E6,
    V7 =           Lam: F3       + E7,
    V8 =           Lam: F3       + E8,
    V9 = Lam: F1      + Lam: F3  + E9;
  std
    E1-E9 = EPS1-EPS9 ,
    F1-F3 = 3 * 1. ;
  bounds
    EPS1-EPS9 >= 0. ;
  cov
    F1-F3 = Cor: ;
  run;
%calisgfi(ram=ram2, nv=9);
```

- How well does this model fit, compared with Model A?

6. PROC CALIS doesn't make it easy to compare several models. My `caliscmp` macro reads several OUTRAM= data sets and computes model comparison statistics.

```
%caliscmp(ram=ram1 ram2,
  models=Model A/Model B,
  compare = 1 2);
```

CFA in R

In R, CFA models can be fit using the `sem` package (and the newer `lavaan` package). The correlation matrix for the Holzinger & Swineford data is contained in the `psych` package. The statements below are contained in the file `N:\psy6140\tutorials\psych9-sem.R`.

```
library(psych)
library(sem)
data(bifactor)
Holzinger.9
```

In the `sem` package, specify the model using the (RAM) notation below, which essentially describes the arrows in the path diagram. Single-headed arrows are specified as “<-”, or “->”. Double-headed arrows (for variances) are specified as “<->” with any number of dashes in

CFA

between. There is one line for each arrow, giving the path, parameter name, and start value (usually NA). There must be **one blank line** following the last model line.

```
## model A: correlated, non-overlapping factors
# path          parameter start value
model.psych9A <- specify.model()
  vis_perc <-- F1,          lam11, NA    # loadings
  cubes <-- F1,            lam21, NA
  lozenges <-- F1,         lam31, NA
  par_comp <-- F2,         lam42, NA
  sen_comp <-- F2,         lam52, NA
  wordmean <-- F2,         lam62, NA
  addition <-- F3,         lam73, NA
  count_dot <-- F3,        lam83, NA
  s_c_caps <-- F3,         lam93, NA
  vis_perc <--> vis_perc,   theta1, NA  # unique variances
  cubes <--> cubes,        theta2, NA
  lozenges <--> lozenges,   theta3, NA
  par_comp <--> par_comp,   theta4, NA
  sen_comp <--> sen_comp,   theta5, NA
  wordmean <--> wordmean,   theta6, NA
  addition <--> addition,   theta7, NA
  count_dot <--> count_dot, theta8, NA
  s_c_caps <--> s_c_caps,   theta9, NA
  F1 <--> F1,              NA,    1    # factor variances
  F2 <--> F2,              NA,    1
  F3 <--> F3,              NA,    1
  F1 <--> F2,              rho12, NA  # factor correlations
  F1 <--> F3,              rho13, NA
  F2 <--> F3,              rho23, NA
```

Fit the model, and obtain modification indices:

```
sem.psych9A <- sem(model.psych9A, Holzinger.9, N=144)
summary(sem.psych9A)
# get modification indices
mod.indices(sem.psych9A)
```

To fit model B, we could just copy/paste/edit `model.psych9A` above, but it is simpler to create the extra path, and use `rbind()` to join it as one more row in the model matrix.

```
## model B: add loading of s_c_caps on F1
extra <- specify.model()
  s_c_caps <-- F1,          lam91, NA

model.psych9B <- rbind(model.psych9A, extra)
(sem.psych9B <- sem(model.psych9B, Holzinger.9, N=144))
summary(sem.psych9B)
# compare models
anova(sem.psych9A, sem.psych9B)
```