Structural Equation Modeling

Course material by Dirk Heerwegh, revised by Ahu Alanya
Spring 2017

Lab sessions: why and how?

- Aims:
  1. Apply theoretical knowledge
  2. Increase understanding by interacting with data
  3. Learn to use R and some packages within R
- How:
  - Relatively unstructured
  - Go at your own pace, try to do the exercises yourself (do yourself a favor and don't just copy paste and run the solutions)

"There's never time to do it right, but there is always time to do it over"

4 lab sessions:

1. Basics of model fitting (CFA)
2. Models with a structural part
3. Multiple group CFA
4. Categorical data

Lab Session 1

Lab Session 1: Overview

- A one-factor CFA model
  - Model fitting
  - Reviewing the output
  - Model modifications
- A two-factor CFA model
- Second order CFA model
- The Holzinger-Swineford data

Software used: R

- This is not an R course!
- We will learn some R as we go along
- I will use RStudio
- Many packages or libraries exist to do specific analyses
  - We will use packages "lavaan", "semPlot" and "lavaan.survey"

The data

```r
ds<-read.table(file="~/Documents/KUL/SEM/2015/session1.csv",
               sep=",",header=TRUE)
nrow(ds) # number of subjects
ncol(ds) # number of variables
names(ds) # names of variables
```

This dataset contains 472 rows (subjects), and 5 variables (x1, x2, x3, x4, x5).

```r
cov(ds) # covariance matrix
```

```
##            x1         x2         x3         x4        x5
## x1 0.93138775 0.06195157 0.08777868 0.06921163 0.1137630
## x2 0.06195157 1.00296718 0.57109302 0.22003987 0.2020697
## x3 0.08777868 0.57109302 0.94664354 0.26113724 0.2572928
## x4 0.06921163 0.22003987 0.26113724 0.97074379 0.2274071
## x5 0.11376300 0.20206969 0.25729278 0.22740707 0.9979642
```

You can also read in the data with the filename if you change your working directory.

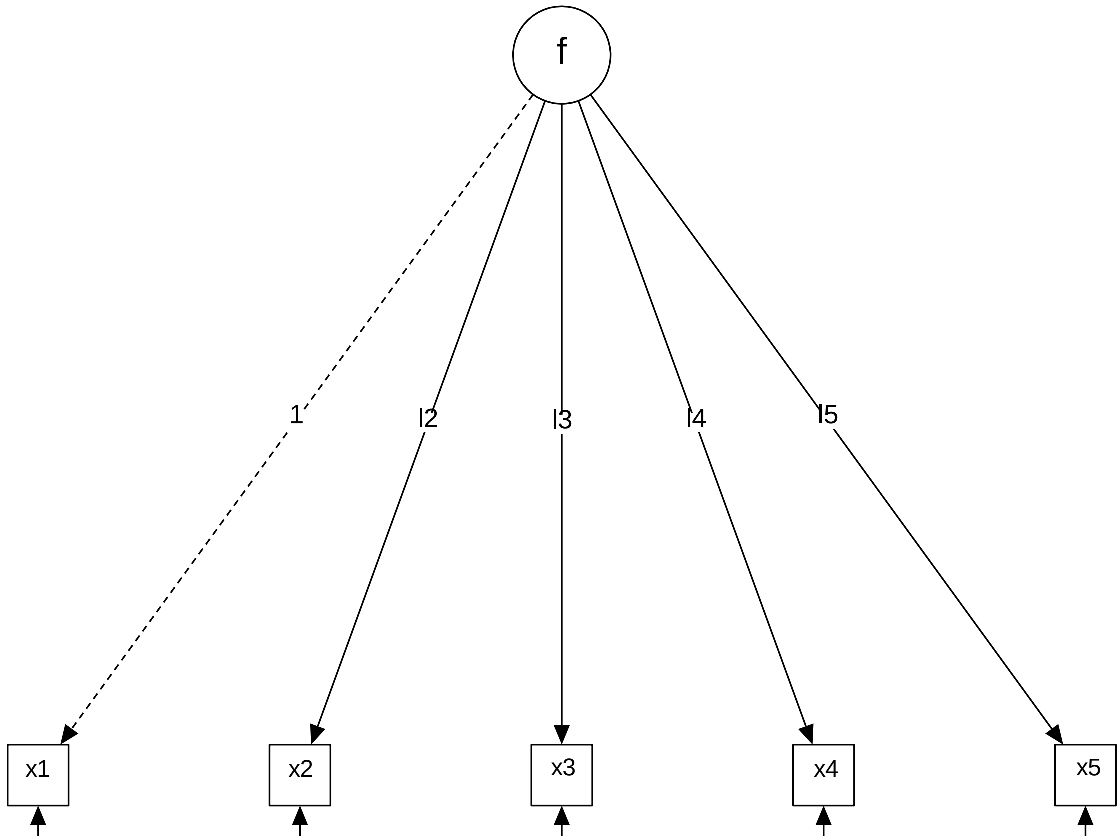
```r
getwd()
setwd("C:/Users/Ahu.Alanya/Desktop/SEM_AA_2016/week1/CFA")
list.files()
ds <- read.table(file="session1.csv", sep=",",header=TRUE)
```

To see the available datasets in the Lavaan package:

```r
install.packages("lavaan")
install.packages("lavaan.survey")

data(package = "lavaan")
ds <-data.frame(HolzingerSwineford1939)
example(cfa)
```

A one-factor CFA model



Statistical identification

- pieces of information: $\frac{p(p+1)}{2} = \frac{5(6)}{2} = 15$
- number of parameters to estimate: 1 factor variance, 4 factor loadings, 5 residual variances
- df = 15 - 10 = 5
- A positive df is a *necessary*, but not a *sufficient* condition for statistical identification

CFA using the lavaan package

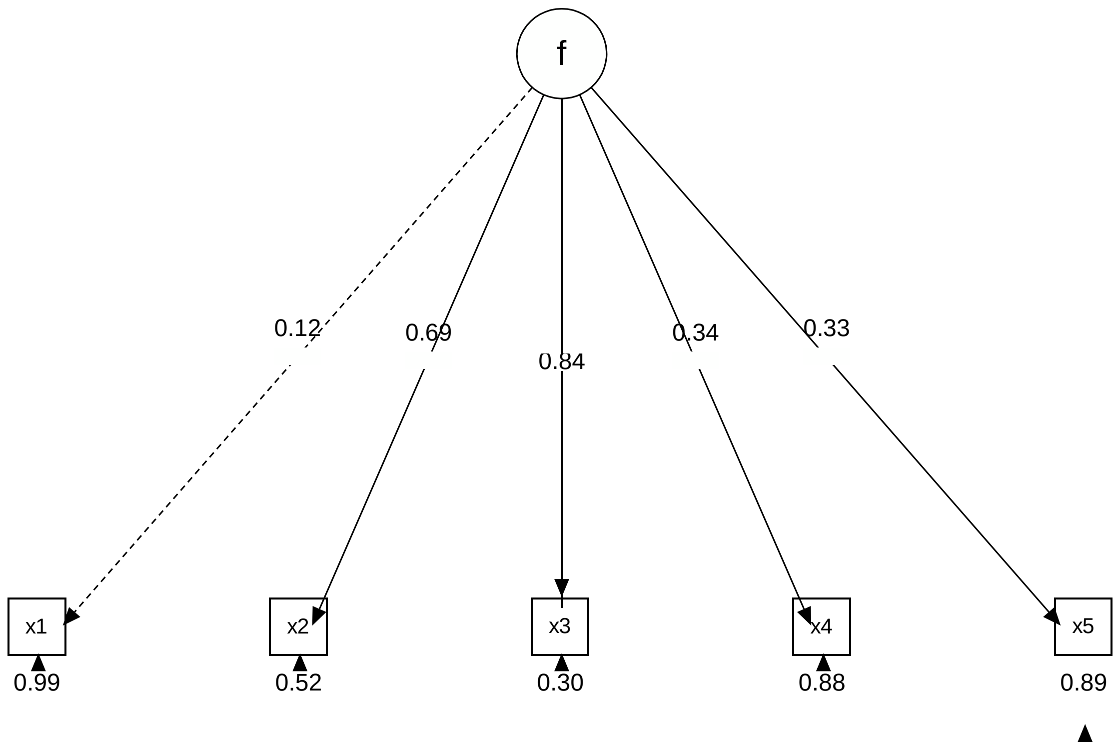
```
library(lavaan)
model<-'f =~ x1 + x2 + x3 + x4 + x5'
```

```r
# fit the model
fit <- cfa(model, data=ds)
```

- =~ means that some latent variable is "measured by" a set of manifest variables
- lavaan by default assumes the first factor loading is fixed to 1 and the variance of the latent variable is freely estimated
  - if necessary, you can freely estimate the first factor loading and fix the factor variance to 1:

```r
model.alternative<-'f =~ NA*x1 + x2 + x3 + x4 + x5
f ~~ 1*f'
```

```r
library(semPlot)
semPaths(fit,"model","stand",style="LISREL",rotation=1,
         edge.color="black",edge.label.cex=1,mar=c(10,1,2,1))
```



Important things to check after running the model:

- Estimates
  - Did indicator load well on the factors?
- Model fit
  - Are the fit indices good?
  -
- Heywood cases/ reasonable solution
  - Are variances positive + r-squared is below 1?
  - Are there any extreme SEs?

Some functions to print parameter vectors or matrices, or fit measures

```
lavInspect(fit, "sampstat")
lavTech(fit,"se")
lavTech(fit," dx.all")
lavTech(fit,"fit")
fitMeasures()
```

- Gives the fit indices separately.

```
modificationIndices()
```

- Get the modification indices separately from all the output.

```
residuals()
```

- Gives you the difference between actual and reproduced correlation table.

```
fitted() function
```

- Gives you the recreated covariance table .

```
parameterEstimates(fit,standardized=TRUE)
```

- Gives parameter estimates with Confidence Intervals .

Parameter Estimates

```
summary(fit,fit.measures=TRUE,standardized=TRUE)
```

Factor loadings

Variable	Estimate (s.e.)	Z-value	Prob.	Stand.
x1	1 (n/a)	(n/a)	(n/a)	0.119
x2	6.056 (2.668)	2.27	0.023	0.695
x3	7.073 (3.139)	2.253	0.024	0.835
x4	2.911 (1.338)	2.176	0.03	0.339
x5	2.84 (1.311)	2.167	0.03	0.327

Residual variances and R-squared

```
theta <- round(inspect(fit,"est")$theta,3) theta.std
<- round(inspect(fit,"std")$theta,3) r2 <-
round(inspect(fit,"r2"),3)
```

OR

```
summary(fit, standardized=TRUE, rsquare=TRUE)
residuals(fit, type="cor")
```

Variable	Res. var.	Stand. res. var.	R-squared
x1	0.916	0.986	0.014
x2	0.518	0.517	0.483
x3	0.286	0.302	0.698
x4	0.857	0.885	0.115
x5	0.89	0.893	0.107

Global Model Fit

```
summary(fit,fit.measures=TRUE,standardized=TRUE)
```

Fit Statistic	Value
loglik H0	-3168.206
loglik H1	-3161.341
χ^2	13.73 (df=5, p=0.017)
CFI	0.97
TLI	0.939
RMSEA	0.061 [0.023;0.1], p(RMSEA<=0.05) = 0.272
AIC	6356.412
BIC	6397.982

Local Model Fit

```
summary(fit,stand=TRUE,mod=TRUE)
```

- lavaan prints all Modification Indexes, even very small ones or those associated with parameters which are already estimated freely

```
mi<-inspect(fit,"mi")
mi.sorted<- mi[order(-mi$mi),]   # sort from high to low
mi.sorted[1:5,]  # only display some large MI values
```

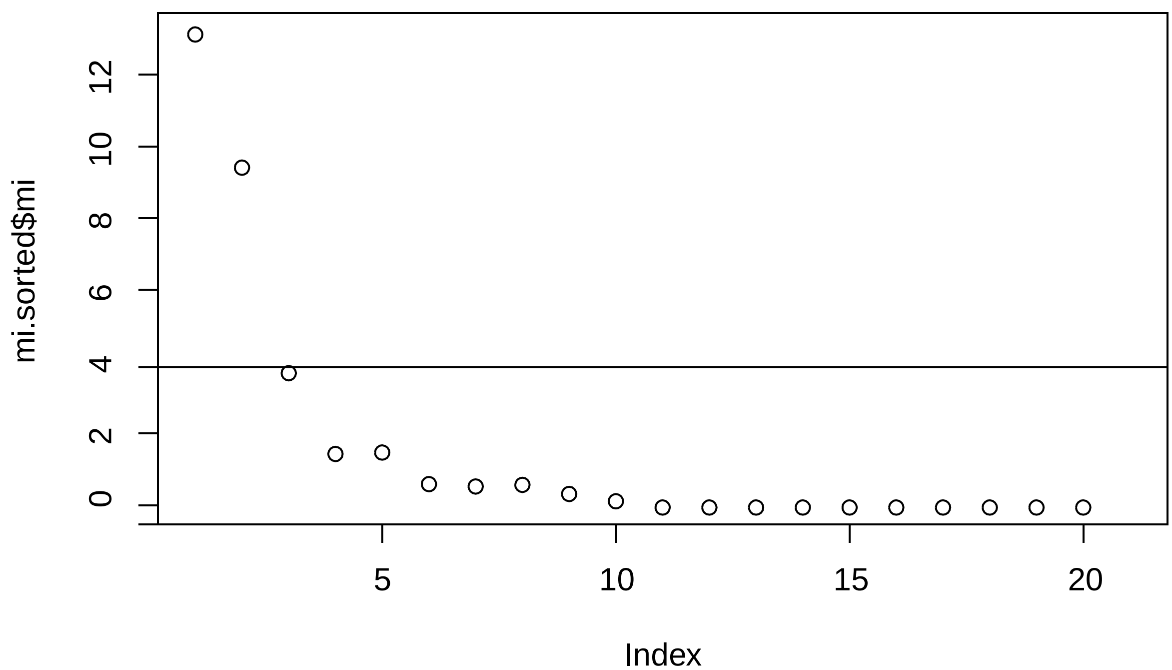
OR

```
modificationIndices(fit)
```

```
##   lhs op rhs     mi    epc  sepc.lv  sepc.all  sepc.nox
## 1  x2 ~~  x3 13.195  0.722   0.722     0.742     0.742
## 2  x4 ~~  x5  9.483  0.130   0.130     0.132     0.132
## 3  x1 ~~  x5  3.496  0.079   0.079     0.082     0.082
## 4  x2 ~~  x5  1.500 -0.059  -0.059    -0.059    -0.059
## 5  x3 ~~  x4  1.360 -0.067  -0.067    -0.070    -0.070
```

Local Model Fit

```
plot(mi.sorted$mi)  # plot the MI values
abline(h=3.84)  # add a horizontal reference line (chisq value for 1 df where p=0.05)
```



Local Model Fit

- Modify model based on a review of:
  - MI's in combination with EPC's (Expected Value Change) --both need to be "substantial"
  - Theory or the source of the data (e.g. review the content of the test items)
- Modifying a CFA moves it away from a strictly confirmatory model
  - The more modifications, the more exploratory the model becomes
  - Maybe this model was not ready for a confirmatory modeling strategy?

Local Model Fit: ex. 1.1.

- Modify the model by including the error covariance, refit the model
- Review the parameter estimates and compare to the first model
- Review the fit statistics (global and local)
- Verify that the new model indeed fits the data better (perform a chi-squared difference test)

Local Model Fit: solution ex. 1.1.

```
model.revised <- 'f =~ x1 + x2 + x3 + x4 + x5
x2 ~~ x3'
fit.revised <- cfa(model.revised, data=ds)
anova(fit,fit.revised)
```

```
## Chi Square Difference Test
##
##             Df    AIC    BIC   Chisq Chisq diff Df diff Pr(>Chisq)
## fit.revised  4 6345.7 6391.4  1.0242
## fit          5 6356.4 6398.0 13.7302     12.706       1  0.0003645 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Local Model Fit: ex. 1.2.

- Consider modifying the model further based on the MIs, revise the model further.
- Review the parameter estimates and compare to the two revised models.

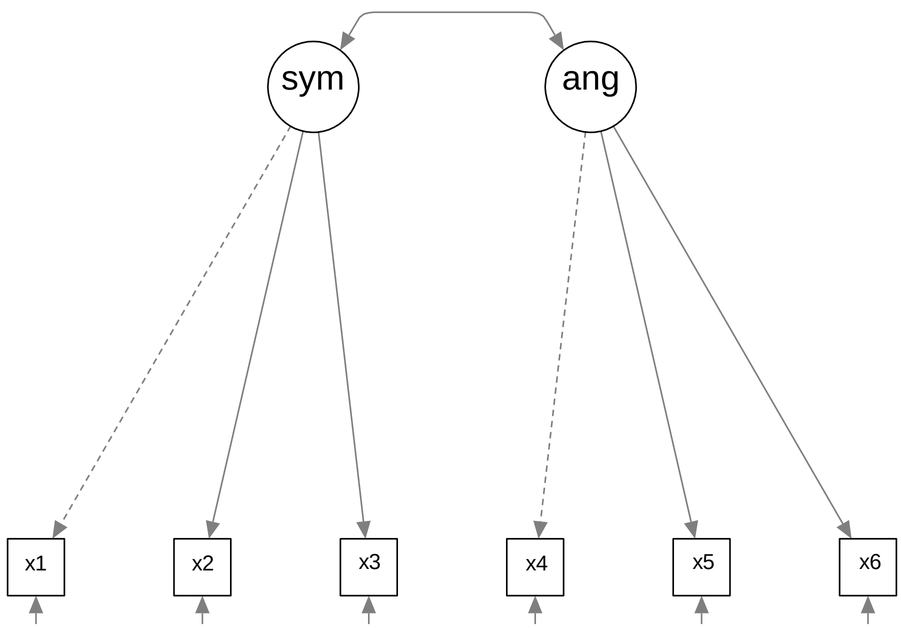
Include the fit indices in the table below.

	X2	Df	RMSEA	SRMR	CFI	TLI	AIC
Revised m1							
Revised m2							

A two-factor CFA model

- Social psychological experiment by Reisenzein (1986) on “helping behavior”
  - Hypothetical story about a person collapsing and lying on a subway floor
  - Half the subjects are told the person was drunk, the other half that the person was ill
  - Do feelings of sympathy and anger mediate the likelihood of helping the victim?
- Focus is on latent variables “sympathy” and “anger”
  - x1 “How much sympathy would you feel for that person?” (1=none at all, 9=very much)
  - x2 “I would feel pity for this person” (1=none at all, 9=very much)
  - x3 “How much concern would you feel for this person?” (1=none at all, 9=very much)
  - x4 “How angry would you feel at that person?” (1=not at all, 9=very much)
  - x5 “How irritated would you feel by that person?” (1=not at all, 9=very much)
  - x6 “I would feel aggravated by that person” (1=not at all, 9=very much so)

A two-factor CFA model



Sample covariance

- Lavaan can analyze summary data in the form of a covariance matrix, instead of raw data. In this case, we must specify the sample size since it is not inferable from the covariance matrix.

```
reis.lower<-'
 6.982
 4.686  6.047
 4.335  3.307  5.037
-2.294 -1.453 -1.979 5.569
-2.209 -1.262 -1.738 3.931 5.328
-1.671 -1.401 -1.564 3.915 3.601 4.977'

reis.cov<-getCov(reis.lower,names=c("x1","x2","x3","x4","x5","x6"))

reis.model<-'sympathy  =~ x1 + x2 + x3
anger =~ x4 + x5 + x6'

reis.fit<-cfa(reis.model,sample.cov=reis.cov,sample.nobs=138)
```

Ex. 1.3

- Review the model parameters and model fit.
- Request the model-implied covariance matrix using lavaan's inspect function.
- Calculate the model-implied covariance matrix using the formula: $\Lambda_x \Phi \Lambda_x^{0} + \Theta_\delta$

Ex. 1.3: solution

```
inspect(reis.fit,"cov.ov")  # model-implied var-covariance matrix for observed variables
```

```
##      x1     x2     x3     x4     x5     x6
## x1  6.931
## x2  4.624  6.003
## x3  4.320  3.310  5.001
## x4 -2.204 -1.689 -1.578  5.529
## x5 -2.023 -1.550 -1.448  3.933  5.289
## x6 -1.993 -1.527 -1.426  3.874  3.555  4.941
```

```
inspect(reis.fit,"est")$lambda  %*% inspect(reis.fit,"est")$psi  %*%
  t(inspect(reis.fit,"est")$lambda) +
  inspect(reis.fit,"est")$theta  # same thing, but calculated from model parameters
```

```
##      x1     x2     x3     x4     x5     x6
## x1  6.931
## x2  4.624  6.003
## x3  4.320  3.310  5.001
## x4 -2.204 -1.689 -1.578  5.529
## x5 -2.023 -1.550 -1.448  3.933  5.289
## x6 -1.993 -1.527 -1.426  3.874  3.555  4.941
```

Note:

"cov.ov": The model-implied variance-covariance matrix of the observed variables. Aliases: "sigma","sigma.hat".

Ex. 1.4

- Do we really need two factors? Fit one factor model and verify that two factor model is appropriate.
- Review the model parameters and model fit.

Ex. 1.4: solution

	X2	Df	RMSEA	SRMR	CFI	TLI	AIC
One factor model							
Two factor model							

Second-order CFA

- Factor analysis assumes that relatively few underlying latent variables may underlie a large number of indicators
- This idea can be extended: more general and abstract latent variables may determine the "first-order" latent variables
- We will analyze data from Marsh and Hocevar (1985) on "Self-concept" for 251 fifth-graders in Sydney, Australia. Their publication contains summary data (means, standard deviations and correlations) which we can use to replicate (parts of) their analysis.

Original hierarchical self-concept CFA model:

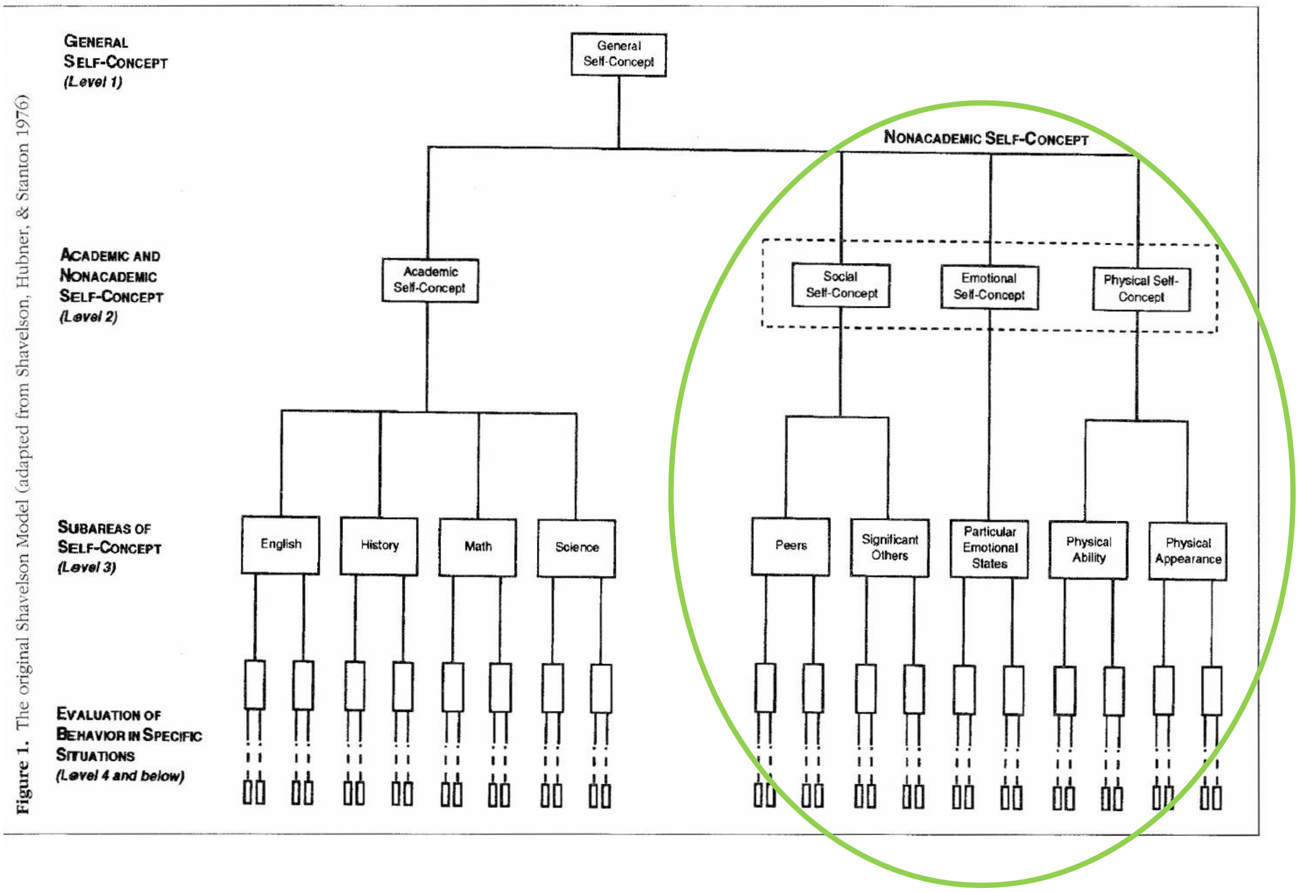


Figure 1. The original Shavelson Model (adapted from Shavelson, Hubner, & Stanton 1976)

The data

- Self-Description Questionnaire (SDQ), designed to measure four non-academic aspects:
  - Physical Ability
  - Physical Appearance
  - Relations with Peers
  - Relations with Parents
- and three academic aspects:
  - Reading
  - Mathematics
  - General School
- Each aspect is represented by four variables, each being the total response to 2 items designed to measure the same SDQ dimension.
- We will focus on the non-academic aspects for fifth graders. The question is whether these four aspects are four dimensions of a more general, "non-academic self-concept" factor?

Reading in summary data

- Lavaan can analyze a covariance matrix. We will transform the correlation matrix + standard deviations to a covariance matrix.

```
lower<-'
1.00
.31 1.00
.52 .45 1.00
.54 .46 .70 1.00
.15 .33 .22 .21 1.00
.14 .28 .21 .13 .72 1.00
.16 .32 .35 .31 .59 .56 1.00
.23 .29 .43 .36 .55 .51 .65 1.00
.24 .13 .24 .23 .25 .24 .24 .30 1.00
```

```
.19 .26 .22 .18 .34 .37 .36 .32 .38 1.00
.16 .24 .36 .30 .33 .29 .44 .51 .47 .50 1.00
.16 .21 .35 .24 .31 .33 .41 .39 .47 .47 .55 1.00
.08 .18 .09 .12 .19 .24 .08 .21 .21 .19 .19 .20 1.00
.01 -.01 .03 .02 .10 .13 .03 .05 .26 .17 .23 .26 .33 1.00
.06 .19 .22 .22 .23 .24 .20 .26 .16 .23 .38 .24 .42 .40 1.00
.04 .17 .10 .07 .26 .24 .12 .26 .16 .22 .32 .17 .42 .42 .65 1.00'
sd<-c(1.84,1.94,2.07,1.82,2.34,2.61,2.48,2.34,1.71,1.93,2.18,1.94,1.31,1.57,1.77,1.47)
```

Reading in summary data

- Transform the correlation matrix to a covariance matrix using the getCov function, and supply variable names.

```
marsh.cov<-getCov(lower,sds=sd,names=c("phyab1","phyab2","phyab3","phyab4","appear1",
                                        "appear2","appear3","appear4","peerrel1","peerrel2",
                                        "peerrel3","peerrel4","parrel1","parrel2","parrel3",
                                        "parrel4"))
```

Analyzing the data: fitting a second-order CFA

- Make sure you specify the correct number of observations!

```
marsh.model<-'phys =~ phyab1 + phyab2 + phyab3 + phyab4
appear =~ appear1 + appear2 + appear3 + appear4
peerrel =~ peerrel1 + peerrel2 + peerrel3 + peerrel4
parrel =~ parrel1 + parrel2 + parrel3 + parrel4
selfConcept =~ phys + appear + peerrel + parrel'
marsh.fit<-cfa(model=marsh.model,sample.cov=marsh.cov,sample.nobs=251)
```

Ex. 1.5: Results from the second order CFA

a. Review the model output using the summary function. Does the model fit well?
b. Review the R^2 values of the first-order latent variables. Which first-order factor is explained best by the second-order factor?
c. Request model modification indices and explore model modifications.

Ex. 1.6: Comparison to a first-order CFA model

a. Omit the second-order latent variable “selfConcept” from the initial second-order CFA model and re-estimate it.
b. How many degrees of freedom are lost? Why?
c. Statistically compare model fit: does the second-order factor model fit significantly worse than the first-order factor model?

Solutions: Ex. 1.5

```
inspect(marsh.fit,"r2")
```

```
##   phyab1   phyab2   phyab3   phyab4  appear1  appear2  appear3  appear4
##    0.385    0.301    0.712    0.691    0.636    0.587    0.606    0.565
## peerrel1 peerrel2 peerrel3 peerrel4  parrel1  parrel2  parrel3  parrel4
##    0.359    0.416    0.615    0.519    0.290    0.271    0.657    0.628
##     phys   appear  peerrel   parrel
##    0.297    0.540    0.774    0.253
```

```
mi<-inspect(marsh.fit,"mi")
mi.sorted<-mi[order(-mi$mi),]  # sort from high to low
mi.sorted[1:5,]  # only display some large MI values
```

```
##           lhs op      rhs     mi   epc sepc.lv sepc.all sepc.nox
## 1     appear1 ~~  appear2 51.455 1.719   1.719    0.283    0.283
## 2        phys =~  appear4 16.174 0.467   0.533    0.228    0.228
## 3 selfConcept =~  appear4 14.798 1.524   0.947    0.405    0.405
## 4     appear4 ~~ peerrel3 14.124 0.644   0.644    0.127    0.127
## 5     appear3 ~~  appear4 13.597 0.828   0.828    0.143    0.143
```

- there is a large MI for appear1~~appear2. This might be a justifiable model modification.

Solutions: Ex. 1.6

```
marsh.model2<-'phys  =~ phyab1 + phyab2 + phyab3 + phyab4
appear  =~ appear1 + appear2 + appear3 + appear4
peerrel  =~ peerrel1 + peerrel2 + peerrel3 + peerrel4
parrel  =~ parrel1 + parrel2 + parrel3 + parrel4'
marsh.fit2<-cfa(model=marsh.model2,sample.cov=marsh.cov,sample.nobs=251)
anova(marsh.fit,marsh.fit2)
```

```
## Chi Square Difference Test
##
##             Df   AIC   BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## marsh.fit2  98 15243 15377 217.31
## marsh.fit  100 15242 15368 219.48     2.1642       2     0.3389
```

- 2 df difference because 6 factor covariances instead of 4 factor loadings.
- Based on the χ^2 difference test, the second order model does not fit statistically significantly worse than the first-order factor model.

Holzinger-Swineford (1939)

- A “classic” dataset, containing data from 26 tests that intended to measure several ability dimensions in children.
- Lavaan contains a dataset that comprises 19 of these tests, intended to measure four factors:
  - Spatial ability

- Verbal ability
- Speed
- Memory

- Two schools were included in the study: Grant-White (n=145) and Pasteur (n=156). Children in these two schools differed on socio-economic background (Grant-White: American-born parents, Pasteur: children from homes of factory workers who were foreign-born). We will take this into account in Lab Session 3.

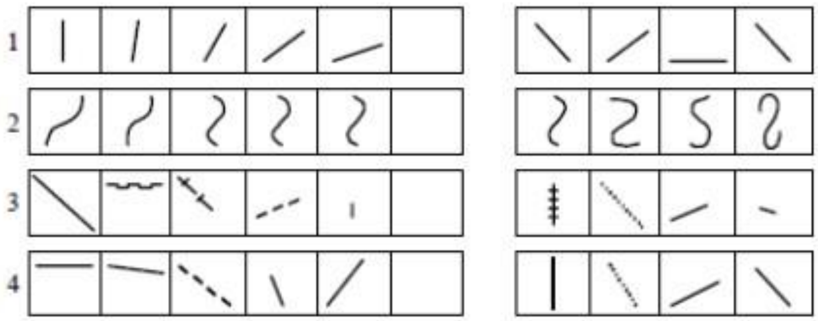
HolzingerSwineford1939 dataset

Sample size: 301, 15 variables:

- id : identifier
- sex : gender
- ageyr : age (year part)
- agemo : age (months part)
- school : school (Pasteur / Grant-White)
- grade : grade
- x1 : visual perception
- x2 : cubes
- x3 : lozenges
- x4 : paragraph comprehension
- x5 : sentence completion
- x6 : word meaning
- x7 : speeded addition
- x8 : speeded counting of dots
- x9 : speeded discrimination straight and curved capitals

Holzinger-Swineford (1939): test item examples

Test 1 Visual-Perception Test



Test 5 General Information

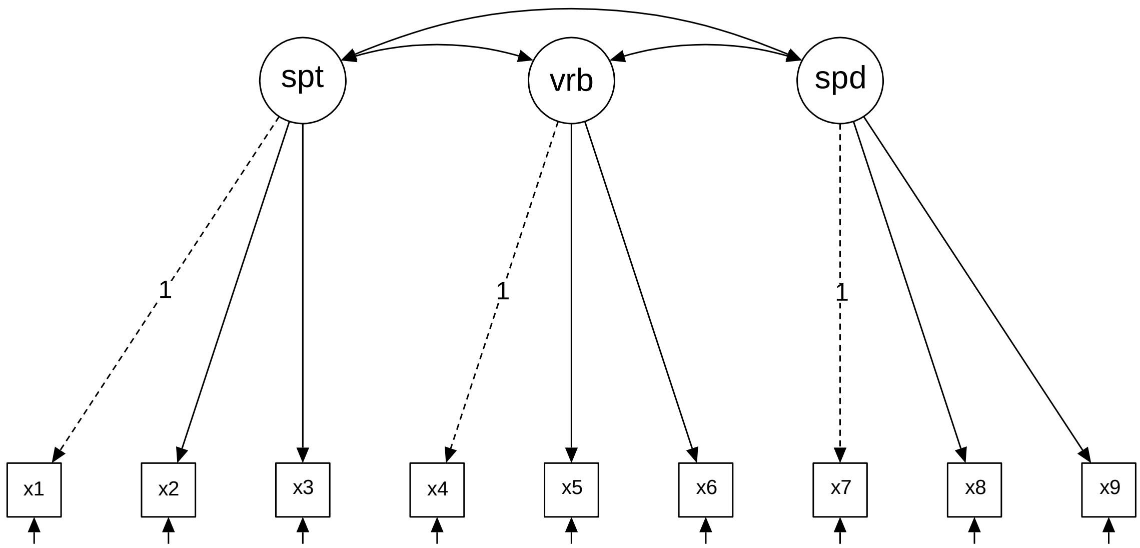
In each sentence below you have four choices for the last word, but only one is right. From the last four words of each sentence, select the right one and underline it.
EXAMPLE: Men see with their ears, nose, eyes, mouths.

---

1. Pumpkins grow on bushes, trees, vines, shrubs.
2. Coral comes from reefs, mines, trees, tusks.
3. Sugar cane grows mostly in Montana, Texas, Illinois, New York

Ex. 1.7: Holzinger-Swineford model

- Fit the model as depicted below. Fix the factor loading of variables x1, x4, and x7 to 1 so as to scale the latent variables *spt* ("spatial"), *vrb* ("verbal"), and *spd* ("speed").



Ex. 1.8: Holzinger-Swineford model

a. Fit the same model, but fix the factor variances to 1 and freely estimate all factor loadings. Are there any differences in model fit? Are there any differences in factor loadings that were also freely estimated in 1.7?
b. Review the modification indices. Can the model fit be improved?
c. Allow variable x9 to load on "spatial" and refit the model. Has the model fit been improved? Is there a theoretical basis to defend this particular model change?
d. Fit a second order CFA. Set the second factor variance equal to 1 and freely estimate the factor loadings from the first order latent variables on the second order latent variable. What has happened to the model fit? Why?

Solutions: Ex. 1.7

```
hs.model<-'spatial  =~ x1 + x2 + x3
verbal  =~ x4 + x5 + x6
speed  =~ x7 + x8 + x9'
hs.fit<-cfa(model=hs.model,data=HolzingerSwineford1939)
```

Solutions: Ex. 1.8 a

```
hs.model2<-'spatial  =~ NA*x1 + x2 + x3
verbal  =~ NA*x4 + x5 + x6
speed =~ NA*x7 + x8 + x9
spatial ~~ 1*spatial
verbal ~~ 1*verbal
speed ~~ 1*speed'
hs.fit2<-cfa(model=hs.model2,data=HolzingerSwineford1939)
```

Solutions: Ex. 1.8 b

```
mi<-inspect(hs.fit,"mi")
mi.sorted<-mi[order(-mi$mi),]  # sort from high to low
mi.sorted[1:5,]  # only display some large MI values
```

```
##       lhs op rhs     mi    epc sepc.lv sepc.all sepc.nox
## 1 spatial =~  x9 36.411  0.577   0.519    0.515    0.515
## 2      x7 ~~  x8 34.145  0.536   0.536    0.488    0.488
## 3 spatial =~  x7 18.631 -0.422  -0.380   -0.349   -0.349
## 4      x8 ~~  x9 14.946 -0.423  -0.423   -0.415   -0.415
## 5  verbal =~  x3  9.151 -0.272  -0.269   -0.238   -0.238
```

- x9 has a visual component (straight vs. curved capitals), so it makes sense to allow x9 to also load on the spatial (visual) factor

Solutions: Ex. 1.8 c

```
hs.model3<-'spatial  =~ x1 + x2 + x3 + x9
verbal  =~ x4 + x5 + x6
speed =~ x7 + x8 + x9'
hs.fit3<-cfa(model=hs.model3,data=HolzingerSwineford1939)
anova(hs.fit,hs.fit3)
```

```
## Chi Square Difference Test
##
##         Df    AIC    BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## hs.fit3 23 7486.6 7568.1 52.382
## hs.fit  24 7517.5 7595.3 85.305     32.923       1  9.586e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- χ^2 has decreased significantly.

Solutions: Ex. 1.8 d

```
hs.model4<-'spatial  =~ x1 + x2 + x3 + x9
verbal =~ x4 + x5 + x6
speed =~ x7 + x8 + x9
general =~ NA*spatial + verbal + speed
general ~~ 1*general'
hs.fit4<-cfa(model=hs.model4,data=HolzingerSwineford1939)
```

- Model fit is identical to that of the first-order factor model (hs.model3). Note that the df has not changed because all we have done is replaced 3 factor covariances by 3 factor loadings.