

Structural Equation Modeling

Course material by Dirk Heerwegh, revised by Ahu Alanya

Spring 2017

Lab Session 2

Lab Session 2: Overview

- Generating data, application: acquiescence modeling
- Adding a structural part to the model (“full SEM”)
- MIMIC models
- First steps into measurement equivalence testing

Generating data

- Working with simulated data can be done for a number of reasons:
 - methodological investigation (does the analysis method deliver as advertised?)
 - power analysis (what sample size is needed to achieve statistical significance for a parameter)
- Common steps:
 - generate data according to a population model (the “true” model)
 - analyzing the generated data using an analysis model. The analysis model may be different from the population model. In this case, you are testing the “robustness” of the model.
- these steps can be done a single time or many times, depending on the purpose (e.g. for power analysis these steps would be done a large number of times)

Modeling “acquiescence”

- What is “acquiescence”?
 - “Likert-type scale items, which are commonly used in survey research to measure respondent attitudes, have often been shown to be susceptible to an acquiescent response bias. Acquiescence is commonly defined as the tendency of some respondents to agree with an item irrespective of the content of that item”
- How can we measure it?
 - a common style factor behind a set of agree-disagree items that are semantically balanced
- Reference: Welkenhuysen-Gybels, J., Billiet, J. & Cambré, B. (2003). “Adjustment for acquiescence in the assessment of the construct equivalence of Likert-type score items”. *Journal of Cross-Cultural Research*, Vol. 34, No. 6, pp. 702-722.

Generating a data structure

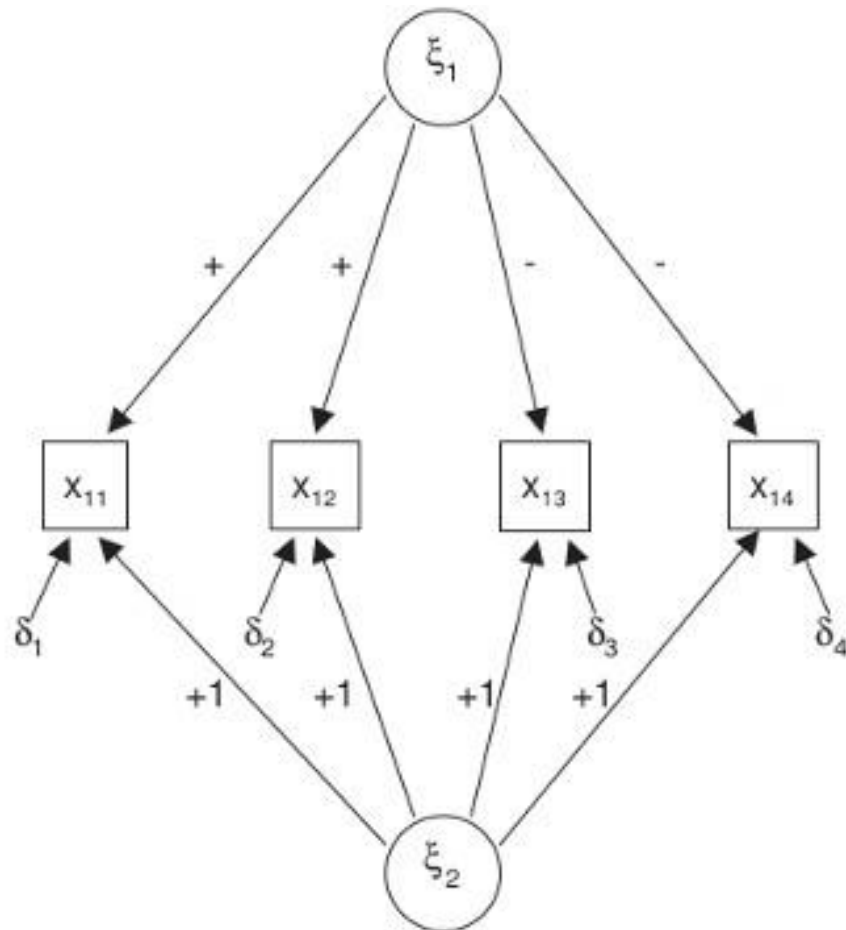


Figure 1: Measurement With One Content (ξ_1) and One Acquiescence Factor (ξ_2)

From Welkenhuysen-Gybels et al. (2003).

Generating a data structure: decisions

- Factor loadings can be chosen freely within a set of restrictions:
 - must be substantial (e.g. $> .4$) on the content factor, and the sign must be as shown in the figure
 - must be smaller on the style factor (e.g. around .2) and must all be equal and positive
- The content and the style factor must have zero-correlation
- Sample size can be chosen freely, start with a reasonable range (e.g. 500-2000)

Generating a data structure

```
library(lavaan)
population.model<-'content =~ 0.5*x1 + 0.6*x2 + -0.8*x3 + -0.9*x4
style =~ 0.2*x1 + 0.2*x2 + 0.2*x3 + 0.2*x4
content~~1*content'
```

```

style    ~~ 1*style
content  ~~ 0*style'

set.seed(5647)
myData<-simulateData(population.model, sample.nobs=500, standardized=TRUE)

```

Ex. 1.1: analyzing generated data

- Analyze the data using the correct analysis model (i.e. including the style factor)
- Review the output: are the parameter values as expected (factor loadings, factor covariances, etc.)?
- Analyze the data using an incorrect analysis model by leaving out the style factor. Compare model fit. Has the model fit significantly deteriorated? Compare factor loadings across models. Is there evidence of bias on the estimates?

Ex. 1.1: solution

```

# analyzing according to the correct model
analysis.model1<-'content =~ NA*x1 + x2 + x3 + x4
style =~ NA*x1 + 11*x1 + 11*x2 + 11*x3 + 11*x4
content =~ 1 * content
style =~ 1 * style
content =~ 0 * style'
acq.fit1 <- cfa(analysis.model1, data=myData)

# analyzing according to an incorrect analysis model
analysis.model2<-'content =~ x1 + x2 + x3 + x4'
acq.fit2 <- cfa(analysis.model2, data=myData)

```

Ex. 1.1: solution (continued)

- Model fit

Criterion	Model 1	Model 2
χ^2	0.01	15.203
df	1	2
p	0.921	0
CFI	1	0
TLI	1.008	0.945
RMSEA	0	0.115

Note: Model 1 includes the style factor; Model 2 does not include the style factor.

Ex. 1.1: solution (continued)

- Standardized factor loadings

Factor	Variable	Model 1	Model 2
Content	x1	-0.548	0.483
	x2	-0.573	0.505
	x3	0.815	-0.844
	x4	0.895	-0.923
Style	x1	0.231	n/a
	x2	0.239	n/a
	x3	0.231	n/a
	x4	0.225	n/a

Note: Model 1 includes the style factor; Model 2 does not include the style factor.

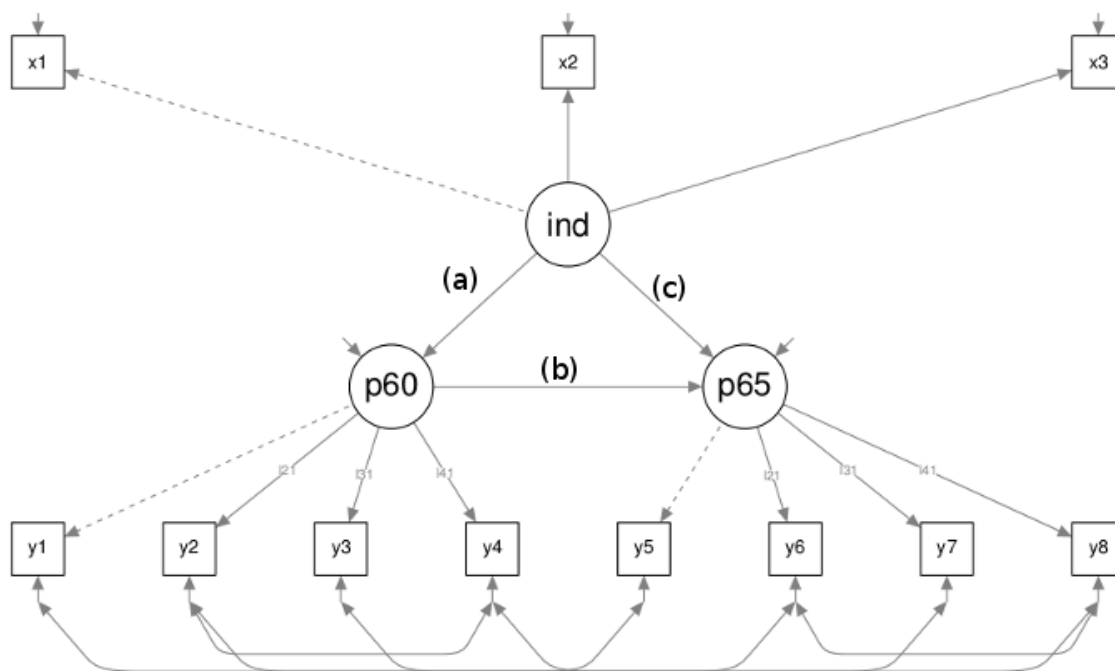
Ex. 1.1: solution (continued)

```
anova(acq.fit2, acq.fit1)
```

```
## Chi Square Difference Test
##
##           Df      AIC      BIC   Chisq Chisq diff Df diff Pr(>Chisq)
## acq.fit1    1 5032.5 5070.5   0.0098
## acq.fit2    2 5045.7 5079.4 15.2030      15.193      1 9.705e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Moving to models with structural parts

- The “Political Democracy model” from Bollen, K. A. (1989): Variables y1 through y4 are intended to be indicators of the latent variable *political democracy in 1960*; y5 through y8 indicators of *political democracy in 1965*; and x1 through x3 indicators of *industrialization in 1960*. Number of observations is 75. We assume that industrialization influences the political democracy variables in 1960 and 1965; and that the 1960 pol. dem. variable influences the 1965 pol. dem. variable:



Ex. 1.2. Political Democracy SEM

- Use the covariance matrix below to fit the model depicted in the figure
- Does the model fit the data well?
- Inspect the regression parameters and the R-squared values of the latent variables
- Manually calculate the indirect effect of Industrialization on Political Democracy in 1965 via Pol. Dem. in 1960 ($a \times b$) and its S.E. using Sobel's formula:
 - Indirect effect = $a \times b$
 - $SE\ a \times b = \sqrt{b^2 \times s_a^2 + a^2 \times s_b^2}$
- Manually calculate the total effect of Industrialization on Political Democracy in 1965
- Ask lavaan to calculate these indirect and total effects. Compare to your own solution

Ex. 1.2. Political Democracy SEM: covariance matrix (N=75)

```
pol.democ.sem.lower <- '
6.89
6.25 15.58
5.84 5.84 10.76
6.09 9.51 6.69 11.22
5.06 5.60 4.94 5.70 6.83
5.75 9.39 4.73 7.44 4.98 11.38
5.81 7.54 7.01 7.49 5.82 6.75 10.80
5.67 7.76 5.64 8.01 5.34 8.25 7.59 10.53
0.73 0.62 0.79 1.15 1.08 0.85 0.94 1.10 0.54
1.27 1.49 1.55 2.24 2.06 1.81 2.00 2.23 0.99 2.28
0.91 1.17 1.04 1.84 1.58 1.57 1.63 1.69 0.82 1.81 1.98'
```

Ex. 1.2.: Solution

```
pol.democ.sd<- c(2.623, 3.947, 3.281, 3.349, 2.613, 3.373, 3.286, 3.246)

pol.democ.sem.cov <- getCov(pol.democ.sem.lower,
                             names=c(paste("y", 1:8, sep=""), paste("x", 1:3, sep="")))

pol.democ.sem.model<-'pd60 =~ y1 + (121)*y2 + (131)*y3 + (141)*y4
pd65 =~ y5 + (121)*y6 + (131)*y7 + (141)*y8
indus =~ x1 + x2 + x3
y1 =~ y5
y2 =~ y6
y3 =~ y7
y4 =~ y8
y2 =~ y4
y6 =~ y8
pd65 ~ (b)*pd60 + (c)*indus
pd60 ~ (a)*indus

indirect := a*b
total := (a*b)+c'

pol.democ.sem.fit <- sem(model=pol.democ.sem.model,
                          sample.cov=pol.democ.sem.cov, sample.nobs=75)
```

Ex. 1.2: Solution

```
a <- inspect(pol.democ.sem.fit, "est")$beta["pd60", "indus"]
b <- inspect(pol.democ.sem.fit, "est")$beta["pd65", "pd60"]
a.se <- inspect(pol.democ.sem.fit, "se")$beta["pd60", "indus"]
b.se <- inspect(pol.democ.sem.fit, "se")$beta["pd65", "pd60"]
```

Manual calculation of indirect effect

```
a * b
```

```
## [1] 1.270565
```

Manual calculation of S.E. of the indirect effect

```
sqrt(a^2*b.se^2 + b^2*a.se^2)
```

```
## [1] 0.3581313
```

Manual calculation of the total effect

```
a * b + inspect(pol.democ.sem.fit, "est")$beta["pd65", "indus"]
```

```
## [1] 1.871494
```

MIMIC models

- MIMIC stands for “Multiple Indicators, Multiple Causes”
- Simple example: one factor measured by 4 indicators, and influenced by 2 causes
- Generate some data according to a population model:

```
population.model<-'f =~ .7*x1 + .6*x2 + - .5*x3 +.4*x4  
f =~ .2*y1 + .3*y2'  
  
set.seed(114254)  
mimic.data<-simulateData(population.model,sample.nobs=500,standardized=TRUE)
```

- Typically, the measurement model is developed first, after which covariates are added.

Ex.1.3

- Fit the measurement model and review the fit statistics. Also review the parameter estimates. How close are they to the population values?
- Add one covariate and watch for changes in fit statistics and parameter estimates.
- Add the second covariate and again see if fit statistics or parameter estimates change.
- Calculate the degrees of freedom for each of these models manually.
- Repeat steps a-c a few times with different data sets. Is there a trend for the fit statistics to change (improve or deteriorate) when including covariates?

Ex. 1.3: Solution (a-c)

```
model1<-'f =~ x1 + x2 + x3 + x4'  
fit1<-cfa(model1,data=mimic.data)  
model2<-'f =~ x1 + x2 + x3 + x4  
f =~ y1'  
fit2<-cfa(model2,data=mimic.data)  
model3<-'f =~ x1 + x2 + x3 + x4  
f =~ y1 + y2'  
fit3<-cfa(model3,data=mimic.data)
```

Ex. 1.3: Solution (a-c)

- Fit Statistics:

Fit Statistic	Model 1	Model 2	Model 3
χ^2	5.239 (df=2, p=0.073)	7.962 (df=5, p=0.158)	10.857 (df=8, p=0.21)
CFI	0.978	0.981	0.985
TLI	0.935	0.961	0.973
RMSEA	0.057	0.034	0.027
AIC	5939.611	7319.034	8711.114
BIC	5973.328	7356.965	8753.26

Ex. 1.3: Solution (d)

- Degrees of freedom for Model 3:
 - pieces of information (with 6 observed variables) = $p(p+1)/2=21$
 - measurement part: 3 factor loadings, 4 residual variances. Total = 7.
 - structural part: 2 regressions, 1 residual factor variance. Total = 3.
 - exogenous variables: 2 variances, 1 covariance. Total = 3.
- $df = 21 - (7 + 3 + 3) = 8$.

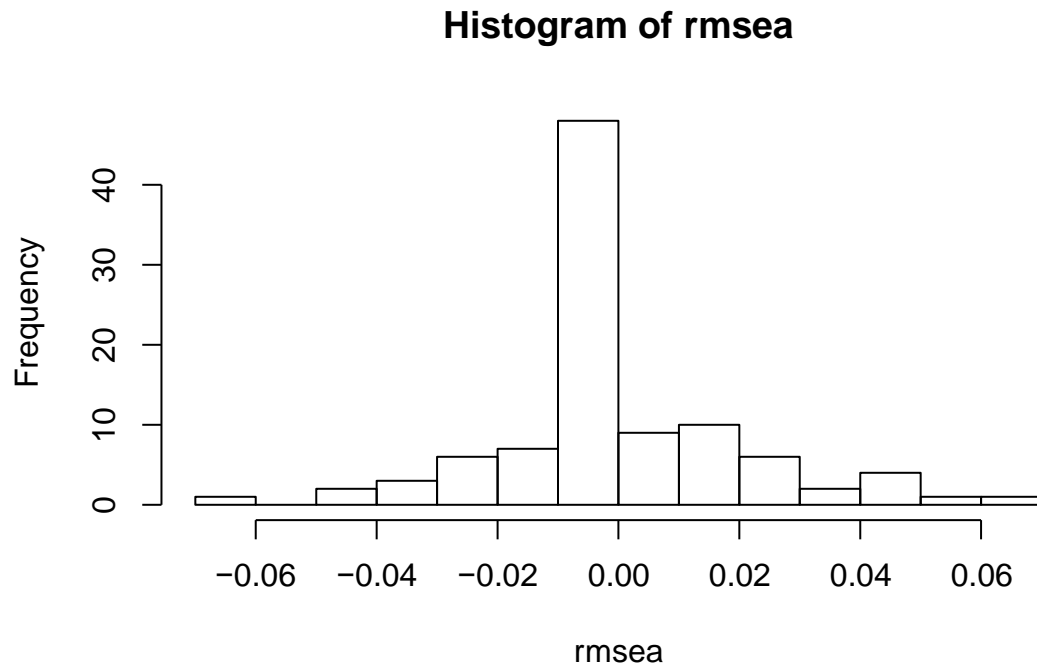
Ex. 1.3: Solution (e)

```
# we will only look at RMSEA values for models 1 and 3
nsim <-100 # set up 100 simulations
rmsea<-rep(NA,nsim)
set.seed(114254)
for (i in 1:nsim){
  mimic.data<-simulateData(population.model,sample.nobs=500,standardized=TRUE)
  fit1<-cfa(modell,data=mimic.data)
  fit3<-cfa(model3,data=mimic.data)
  # calculate difference in RMSEA (positive values indicate a better fit of Model 3)
  rmsea[i]<-inspect(fit1,"fit")["rmsea"]- inspect(fit3,"fit")["rmsea"]
}
mean(rmsea)

## [1] 0.002048203
```

Ex. 1.3: Solution (e)

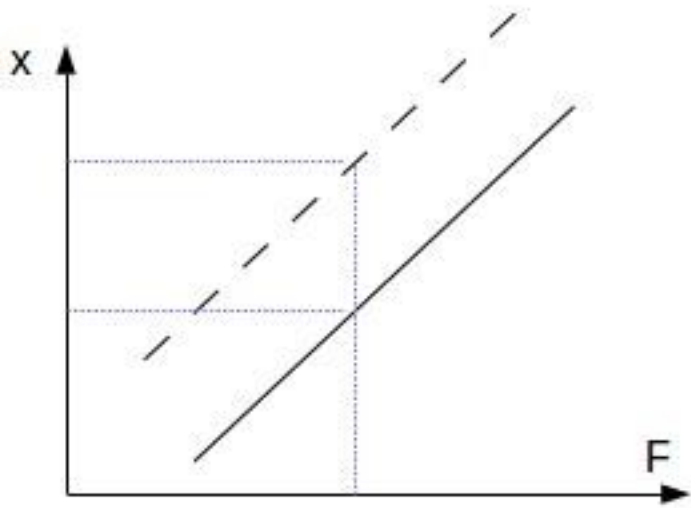
```
hist(rmsea,breaks=10)
```

Measurement Equivalence Testing

- Measurement Equivalence if a measurement instrument produces equivalent results, regardless of some unrelated properties of the test subjects.
- Absence of measurement equivalence would imply some degree of distortion of the results ("bias"). A test instrument could be perceived as "unfair", e.g. an IQ test that favors males by including "gender-biased test items"
- What is the effect of such a bias? Would we draw incorrect conclusions?
- Can we detect whether such bias is present in our data?

One possible form of Measurement Non-Equivalence: “uniform DIF (Differential Item Functioning)”



- Intercept of x is shifted upwards
- This corresponds to a direct effect of the covariate (e.g. gender) on the observed variable

Ex. 1.4

- Generate data using the same model as in Ex. 1.3, but include a direct effect of “y2” to “x3” (magnitude = .2). Simulate a data set that includes 500 subjects.
- Fit a MIMIC model as in Ex. 1.3 (including both y-covariates). Review model fit and parameter estimates.
- Request modification indexes. Is the direct effect of y2 on x3 detected?
- Rewrite the model to include all direct effects of the y-covariates on all x-variables, but fix the regression weights to zero. Re-fit the model and request modification indexes.
- Re-fit the model including the direct effect to account for the uniform DIF. What happens to model fit and model parameters?

Ex. 1.4: Solution

```
population.model<-'f =~ .7*x1 + .6*x2 + -.5*x3 + .4*x4
f =~ .2*y1 + .3*y2
x3 =~ .2*y2'
set.seed(114254)
mimic.data<-simulateData(population.model,sample.nobs=500,standardized=TRUE)
model1<-'f =~ x1 + x2 + x3 + x4
f =~ y1 + y2'
fit1 <- cfa(model=model1,data=mimic.data)
# summary(fit1,fit.measures=TRUE, stand=TRUE, mod=TRUE)
```

Ex. 1.4: Solution

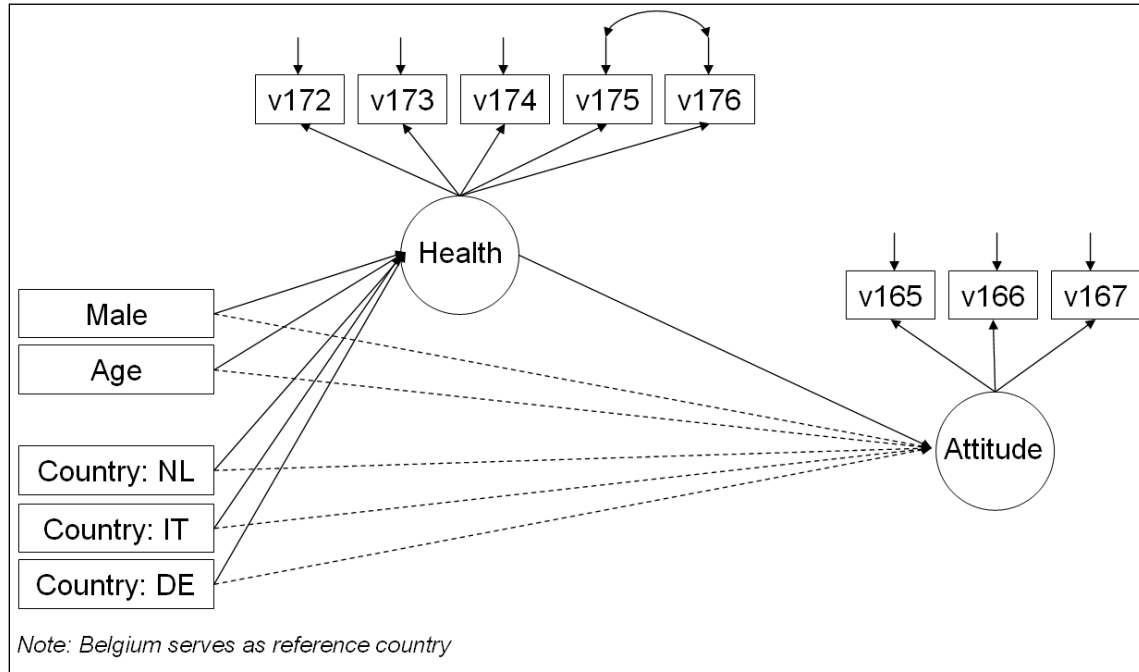
```
modellb <- 'f =~ x1 + x2 + x3 + x4
f =~ y1 + y2
x1 =~ 0*y1 + 0*y2
x2 =~ 0*y1 + 0*y2
x3 =~ 0*y1 + 0*y2
x4 =~ 0*y1 + 0*y2'
fit1b <- cfa(model=modellb, data=mimic.data)
# summary(fit1b, fit.measures=TRUE, stand=TRUE, mod=TRUE)
model2<-'f =~ x1 + x2 + x3 + x4
f =~ y1 + y2
x3 =~ y2'
fit2 <- cfa(model=model2, data=mimic.data)
# summary(fit2, fit.measures=TRUE, stand=TRUE, mod=TRUE)
anova(fit1, fit2)

## Chi Square Difference Test
##
##      Df      AIC      BIC   Chisq Chisq diff Df diff Pr(>Chisq)
## fit2   7 8713.7 8760.1  2.2957
## fit1   8 8719.6 8761.7 10.1220      7.8263      1  0.005149 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Ex. 2 Eurobarometer on Public Health Attitudes, Behavior, and Prevention

Eurobarometer 72.3 (October 2009) contained questions about attitudes towards alcohol use. In this exercise a random sample of 1000 records from the full dataset will be used. The sample is restricted to respondents from Belgium, the Netherlands, Italy and Germany. The dataset is called ZA4977_sem.dat (comma separated values, variable names on the first line).

We want to fit the full SEM model depicted below.



The variable names and additional information is shown below.

Reference in figure	Variable	Values	Remarks
Male	Male	0=female, 1=male	
Age	Age	15 ... 92	
Country: NL	Ctry_nl	0=no, 1=yes	
Country: IT	Ctry_it	0=no, 1=yes	
Country: DE	Ctry_de	0=no, 1=yes	
Ref. country = BE	Ctry_be	0=no, 1=yes	Not used in the model

Reference in figure	Variable	Values	Remarks
v165	n165	1=Totally disagree 2=Tend to disagree 3= Tend to agree 4=Totally agree	Random police alcohol blood checks on EU roads would reduce people alcohol consumption before driving
v166	n166		Alcohol advertising targeting young people should be banned in all EU Member States
v167	n167		Selling and serving alcohol to people under the age of 18 years should be banned in all EU Member States
	n171	1=Totally disagree 2=Tend to disagree 3= Tend to agree 4=Totally agree	Would you agree or disagree that the consumption of alcoholic beverages can increase the risk of the following health conditions? ... Liver diseases (not used in the model)
v172	n172		... Heart diseases
v173	n173		... Cancers
v174	n174		... Asthma
v175	n175		... Depression
v176	n176		... Birth defects

First set up a measurement model involving only the latent variables and their indicators. You may want to do an EFA model to iteratively work your way towards a final measurement model (include variable n171 if you do), or you may start out with the measurement model as shown.

In the final step, introduce the covariates and build the MIMIC model as shown in the graph. You have now built a MIMIC model.

Ex. 2: Solution

MODEL 1

```
model<- 'health =~ NA*n171 + n172 + n173 + n174 + n175 + n176 + n165 + 0*n166 + n167
attitude =~ NA*n171 + n172 + n173 + n174 + 0*n175 + n176 + n165 + n166 + n167
health =~ 1*health
attitude =~ 1*attitude'
fit<-cfa(model,data=s)
```

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

MODEL 2

- removing non-significant factor loadings and low factor loadings that are
- as expected (i.e. loading on the "wrong" factor)

```
model<-'health =~ NA*n171 + n172 + n173 + n174 + n175 + n176 + 0*n165 + 0*n166 +  
0*n167  
attitude =~ 0*n171 + 0*n172 + 0*n173 + 0*n174 + 0*n175 + 0*n176 + n165 + n166 + n167  
health ~~ 1*health  
attitude ~~ 1*attitude'  
fit<-cfa(model,data=s)  
summary(fit,fit.measures=TRUE,standardized=TRUE, modindices=TRUE)
```

MODEL 3

- removing item n171 because it seems like a "trouble maker" (see the MI's, largest MI's involve n171)

```
# n171 ~~ n174 : 77.059 (expected correlation: -.27)  
# n171 ~~ n176 : 71.198 (expected correlation: .25)
```

- an alternative is to allow these error covariances

```
model<-'health =~ NA*n172 + n173 + n174 + n175 + n176 + 0*n165 + 0*n166 + 0*n167  
attitude =~ 0*n172 + 0*n173 + 0*n174 + 0*n175 + 0*n176 + n165 + n166 + n167  
health ~~ 1*health  
attitude ~~ 1*attitude'  
fit<-cfa(model,data=s)  
summary(fit,fit.measures=TRUE,standardized=TRUE, modindices=TRUE)
```

Final measurement model

- # there is one large MI (44) for an error covariance between n173 and n174 (cancers
- # and asthma). The expected correlation between the residuals is a moderately
- # strong one (0.21).
- # I don't see a strong theoretical basis for allowing this error covariance, so I'm not
- # going to introduce it. We didn't see this error covariance when doing the analysis on
- # the original file (which also contained some records from Greece). So maybe this is
- # a country-specific thing?
- # Also note the moderately large MI (24.913) for n166 ~~ n167 and the strong expected
- # correlation (.664). The two items are worded as follows:
- # n166: Alcohol advertising targeting young people should be banned in all EU member states
- # n167: Selling and serving alcohol to people under 18 years should be banned in all EU member states
- # So these items are quite similar (both about banning something for young people), so
- # I will introduce the error covariance

```

model<-
'health =~ NA*n172 + n173 + n174 + n175 + n176 + 0*n165 + 0*n166 + 0*n167
attitude =~ 0*n172 + 0*n173 + 0*n174 + 0*n175 + 0*n176 + n165 + n166 + n167
health ~~ 1*health
attitude ~~ 1*attitude
n166 ~~ n167'
fit<-cfa(model,data=s)
summary(fit,fit.measures=TRUE,standardized=TRUE, modindices=TRUE)

```

The model fit is not really good (we can obtain a better fit if we allow an error covariance between n173 and n174 (RMSEA=0.035) but without theoretical basis and no convincing post-hoc explanation, this seems like overfitting the model

Final step: transform to a path model

```

model<- 'health =~ NA*n172 + n173 + n174 + n175 + n176 + 0*n165 + 0*n166 + 0*n167
attitude =~ 0*n172 + 0*n173 + 0*n174 + 0*n175 + 0*n176 + n165 + n166 + n167
health ~~ 1*health
attitude ~~ 1*attitude
n166 ~~ n167
attitude ~ health'
fit<-sem(model,data=s)
summary(fit,fit.measures=TRUE,standardized=TRUE, modindices=FALSE)

```

Adding control variables

```

model<- 'health =~ NA*n172 + n173 + n174 + n175 + n176 + 0*n165 + 0*n166 + 0*n167
attitude =~ 0*n172 + 0*n173 + 0*n174 + 0*n175 + 0*n176 + n165 + n166 + n167
health ~~ 1*health
attitude ~~ 1*attitude
n166 ~~ n167
attitude ~ health + male + age + ctry_nl + ctry_it + ctry_de
health ~ male + age + ctry_nl + ctry_it + ctry_de'
fit<-sem(model,data=s)
summary(fit,fit.measures=TRUE,standardized=TRUE, modindices=FALSE)

```