# Structural Equation Modelling & Causal Inference

## Day 1–General Introduction to Structural Equation Modelling (SEM)

Ozan Aksoy

UCL Social Research Institute & NCRM
University College London

November 2, 2023

# Topics

1. Logistics of this module
2. Overview of SEM
3. Univariate regression
4. Multivariate regression
5. Recursive path models
6. Model fit
7. Imposing and testing constraints
8. Complicating models
9. Non-recursive path models
10. Identification

# Course overview



- Day 1
  - Basics of Structural Equation Modelling (SEM)
  - Structural (regression type) models
  - Measurement models (Confirmatory Factor Analysis)
  - Multiple group analysis and measurement invariance
- Day 2
  - Longitudinal SEM (Multi-trait multi-method models, cross-lagged models, latent-curve models)
  - Full Information Maximum Likelihood Estimation
  - SEM versus DAGs
- Day 2
  - Fixed versus random effects models
  - Cross-lagged panel models with fixed effects
  - Instrumental variable models

# Plan



- Online lecture (10am-1pm)–Introduces the topics
- Computer practicals (2pm-5pm)–Hands-on exercises with supervision
- All material are at: `https://github.com/aksoyundan/SEM`
- Main software: `R` and `RStudio` (some example code for Stata and Mplus may be provided)
- Main R packages: `lavaan` and `semTools` and some others
- Install R, RStudio, lavaan, semTools etc. at lunch break if you haven't already: `https://rstudio-education.github.io/hopr/starting.html`
- Initial survey: go to `www.menti.com`

# Structural equation models



- Integration of regression models, path models, simultaneous equations, and factor analysis
- SEM may include observed (manifest) and unobserved (latent) variables
  - E.g., factors are unobserved/latent variables measured with observed indicators
  - E.g., social class is latent, income is observed
  - E.g., religiosity is latent, frequency of prayer, church visit etc. are observed
- Loosely, an (observed or latent) variable is called
  - endogenous if DV (maybe IV in other regressions): it is determined within "the system"
  - exogenous if only IV: it is determined outside "the system"
- Errors (residuals, disturbances) are exogenous latent variables
- SEM may include covariance and mean structures

# Structural equation models (cont)



Equivalent names for rich class of linear models

- structural equation model (SEM)
- covariance structure analysis (CSA)
- linear structural relations (LISREL) ...

Modern extensions allow

- non-normal distributed continuous data
- generalized linear relations to deal with categorical (ordinal, binary) and nominal DVs
- limited forms of nonlinear regression (interactions, quadratic effects, . . . )
- multilevel structures: the relations between variables at level 1 (e.g., individual) vary at level 2 (e.g., context)
- discrete latent variables (model-based clustering)
- Bayesian SEM
- Generalized latent variable modelling

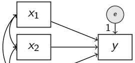
# SEM diagrams



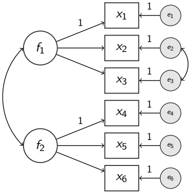
- Models may be denoted by a diagram
  - rectangular box for observed (manifest) variable
  - ellipse for unobserved (latent) variable, including errors
  - causal link: (one-sided) arrow from x to y variable
  - noncausal relation: double pointed arrow between two variables (usually: between two errors or two exogenous variables)
  - sometimes: intercepts represented by triangles (I don't do it)
- Some software (Amos, Stata, ...) allow model specification by drawing a *diagram*
- Nice for novices and simple models
- But . . . infeasible for more complicated model
- Convenient communicatation of models/results
  - use Tikz in LaTeX

# Some examples of SEM diagrams (Tikz)

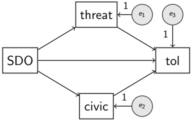




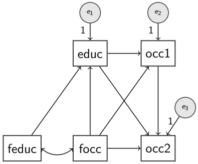

regression model



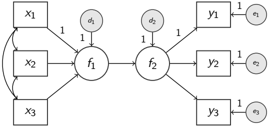

2 factor CFA with correlated factors



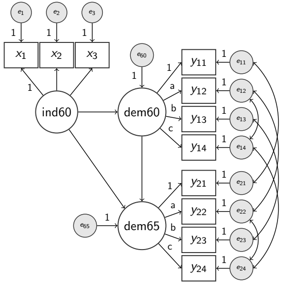

path model




Blau-Duncan path model




MIMIC model




Bollen dynamic model

# Resources to learn about structural equation models



Books

- Kline 2023 5th. Principles and Practice of Structural Equation Modeling.
- Hancock Mueller 2013. Structural Equation Modeling. A Second Course.
- Bollen Curran 2006. Latent Curve Models. A SEM Perspective
- Hox et al 2017 3ed. Multilevel Analysis. Techniques and Applcations.

Websites

- `http://davidakenny.net/kenny.htm`
- `http://www2.gsu.edu/ mkteer/index.html`
- `https://www.guilford.com/kline-materials`

## Linear regression



Regression of continuous $y$ on continuous predictors $x_1$ and $x_2$

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + e_i$$

$$e_i \sim \mathrm{Normal}(0, \sigma^2) \quad \text{(independent homoskedastic residuals)}$$
$$\mathrm{cov}(x_j, e) = 0 \quad \text{(“exogeneity of } x_j\text{”)}$$

In R- lavaan,

- a regression is written as `y ~ xlist;`
- intercept $\beta_0$, coefficients $\beta_j$, and error term $e_i$ are *implicit*

simple regression

```
x1 <- rnorm(100, mean = 0, sd = 2)
x2 <- 0.3*x1 + rnorm(100, mean = 0, sd = 2)
y  <- 0.2 + 0.7*x1 + 1.2*x2 + rnorm(100, mean = 0, sd = 1)
d  <- as.data.frame(cbind(y, x1, x2))
library(lavaan)
mymodel <- 'y ~ x1 + x2
            y ~ 1' # otherwise intercept is suppressed
fit <- sem(mymodel, data=d)
summary(fit, standardized = TRUE)
```

# Regression analysis: Output



```
lavaan (0.5-23.1097) converged normally after  19 iterations
  Number of observations                           100
[omitted for brevity]
Regressions:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
  y ~
    x1                0.695    0.051   13.699    0.000    0.695    0.515
    x2                1.069    0.062   17.197    0.000    1.069    0.646
Intercepts:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
   .y                 0.172    0.104    1.656    0.098    0.172    0.060
Variances:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
   .y                 1.072    0.152    7.071    0.000    1.072    0.130
```

- Thus: `E(Y) = 0.2 + 0.7*X1 + 1.1*X2, var(e) = 1.1`
- Under Ho: $b_x = 0 : \hat{b}_x / \hat{\mathrm{se}}(\hat{b}_x)$ approximately standard normal
- `standardized = TRUE` produces standardized solution; and R2 = 1-standardized residual variance.

# R–lavaan operators



| formula type | operator | mnemonic |
|---|---|---|
| latent variable definition | $=\sim$ | is measured by |
| regression | $\sim$ | is regressed on |
| (residual) (co)variance | $\sim\sim$ | is correlated with |
| intercept | $\sim 1$ | intercept |
| new parameter | $:=$ | defined as |

# Multivariate models



Often need to analyze multiple responses simultaneously:

- multiple dimensions of problem behavior (aggression, depression, . . . )
- interpersonal trust, measured at different time points
- sex role attitudes of husbands and wives
- educational attainment, occupational status, income . . .

Distinguish variables

- variable is only y
- variable is only x
- variable is both y and x (intermediate, mediator, ...)

# Multivariate and seemingly unrelated regression



Models and assumptions applicable if no variable is both y and x

- MVREG/MANOVA: same x variables predicting the y's
- SUREG: separate spec (maybe overlap) of x's predicting the y's
- an y is not x for another y (otherwise: path, simeqns)
- errors across y's may covary
- better approach than fitting separate models per dv

Formal model specification of SUREG (and cov(x,e)=0):

$$
\begin{aligned}
y_{i1} &= \beta_{10} + \beta_{11} x_{i1} + \beta_{12} x_{i2} &&+ e_{i1} \qquad \mathrm{var}(e_{ij}) = \sigma_j^2 \\
y_{i2} &= \beta_{20} + \beta_{21} x_{i1} + \beta_{23} x_{i3} &&+ e_{i2} \qquad \mathrm{cov}(e_{i1}, e_{i2}) = \sigma_{12}
\end{aligned}
$$

In R:

- may combine different types of regression (e.g. binary-continuous)
- `y1 ~~ y2;` specifies that covariance of e.y1 and e.y2. is a free parameter

# MVREG–example



mvreg1

```
library(lavaan)
model2 <- 'y1 ~ x1 + x2
           y2 ~ x1 + x2
           y1 ~~ y2' # adds cov(e.y1,e.y2), default in R
fit2 <- sem(model2, data=d)
summary(fit2)
```

mvreg2

```
model3 <- 'y1 ~ x1 + x2
           y2 ~ x1 + x2
           y1 ~~ 0*y2' # removes cov(e.y1,e.y2) from model
fit3 <- sem(model3, data=d)
summary(fit3)
```

# Recursive path model; direct and indirect effects



Formal model specification with mediator:

$$\begin{aligned} y_{i1} &= \beta_{10} + \beta_{11}x_{i1} + \beta_{12}x_{i2} + \gamma_{12}y_{12} + e_{i1} & \operatorname{var}(e_{ij}) &= \sigma_j^2 \\ y_{i2} &= \beta_{20} + \beta_{21}x_{i1} + \beta_{23}x_{i3} \qquad\qquad + e_{i2} & \operatorname{cov}(e_{i1}, e_{i2}) &= 0 \end{aligned}$$

Reduced form by substitution

$$\begin{aligned} y_{i1} &= \beta_{10} + \beta_{11}x_{i1} + \beta_{12}x_{i2} + \gamma_{12}(\beta_{20} + \beta_{21}x_{i1} + \beta_{23}x_{i3} + e_{i2}) + e_{i1} \\ &= (\beta_{10} + \gamma_{12}\beta_{20}) + (\beta_{11} + \gamma_{12}\beta_{21})x_{i1} + \beta_{12}x_{i2} + \gamma_{12}\beta_{23}x_{i3} + (e_{i1} + \gamma_{12}e_{i2}) \end{aligned}$$

In words:

- x1 affects y1 directly, and indirectly via y2
- total effect of x1 = direct ($\beta_{11}$) + indirect effect ($\gamma_{12}\beta_{21}$)
- General case: indirect effect of x on y is sum over all possible paths from x to y of product of path coefficients
- Below: In (nonrecursive) model with causal cyles, *infinitely* many possible paths, but *sum* mathematically well-defined
- SEs of indirect/total effects by delta method (or bootstrsap)

# Path model in R



Blau & Duncan

```
bd_low <- '
1.0000
0.5160  1.0000
0.4530  0.4380  1.0000
0.3320  0.4170  0.5380  1.0000
0.3220  0.4050  0.5960  0.5410  1.0000'
bd.corr <- getCov(bd_low, names = c("faed", "faocc",
                  "educ", "occ1", "occ2"))
m.bd <- 'educ ~ a*faed + b*faocc
         occ1 ~ c*educ + d*faocc
         ac    := b*c
         total := d + (b*c)'
fit.bd <- sem(m.bd, sample.cov = bd.corr, sample.nobs = 20700)
summary(fit.bd)
```

# Path model in R—selected output



```
lavaan (0.5-23.1097) converged normally after  12 iterations
...
Regressions:
                   Estimate  Std.Err  z-value  P(>|z|)
  educ ~
    faed      (a)     0.309    0.007   44.383    0.000
    faocc     (b)     0.278    0.007   39.937    0.000
  occ1 ~
    educ      (c)     0.440    0.006   69.488    0.000
    faocc     (d)     0.224    0.006   35.463    0.000
Variances:
                   Estimate  Std.Err  z-value  P(>|z|)
   .educ              0.738    0.007  101.735    0.000
   .occ1              0.670    0.007  101.735    0.000

Defined Parameters:
                   Estimate  Std.Err  z-value  P(>|z|)
    ac                0.122    0.004   34.625    0.000
    total             0.347    0.007   53.027    0.000
```

Note: $0.122 = 0.278\,(FAOCC \rightarrow EDUC)\ x\ 0.440\,(EDUC \rightarrow OCC1)$

# Indirect effects vs. mediation: causality caveat!!



- Mediation refers to causal hypothesis
- Mediation involves indirect effect
- But not all indirect effects signal mediation
- Mediation should be used sparingly, more specifically...
- Sequential ignorability assumption has to be met for appropriate tests of mediation:
  - $X -> M -> Y$; $X -> Y$
  - X has to be exogenous to M and Y (no confounding between X, M, and Y)
  - M has to be exogenous to Y (no confounding between M and Y)
  - No interaction between X and M (can be relaxed)
- If interested: read on causal mediation analysis

## Path model: multiple indirect effects



In the specification below, x1 influences y1 in three ways

- directly
- indirectly via y2
- indirectly via y3

total effect = direct effect + sum of indirect effects

```
M <- 'y1 ~ a*y2 + b*y3 + c*x1 + x2
      y2 ~   x3 + d*x1
      y3 ~   x4 + e*x1
      id1 := d*a  # via y2
      id2 := e*b  # via y3
      tot := c + d*a + e*b # total effect
```

# Equivalent models



- Different models with same fit, but very different interpretation
- Often arrows may be reversed, replaced by mutual influence, or freeing covariance among residuals!
- Examples of 3 equivalent models:

$$\begin{array}{ll} (1) & x \rightarrow y \rightarrow z \\ (2) & x \leftarrow y \leftarrow z \\ (3) & x \leftarrow y \rightarrow z \end{array}$$

- Often DOZENS-HUNDREDS of equivalent models; Theory is unavoidable!
- Rarely addressed in application; no good software tools to sensitize researchers

# Assessing Fit: Model Chi-square



- LRtest of model vs saturated model
  - Hope: not significant, test of exact fit hypothesis
  - df = sample moments - number of free parameters
  - senstive to multivariate normality
  - With some estimators (eg MLM, ...), tests adjusted for nonnormality
  - sensitive to sample size: complex (maybe stupid) model not rejected in small samples
  - reasonable but untrue (all!) models rejected in big samples
- LRtest of model vs base model
  - Hope: highly significant
  - common base model: independence of variables
  - compare null model is regression
  - base nearly always fits very badly (why?)
  - base is used to evaluate quality of model
    hardly an accomplishment to improve on the silly, use a more worthy opponent: a more informative baseline

# Example: fit measures of Blau-Duncan model



Blau & Duncan fit (selection)

```
lavaan (0.5-23.1097) converged normally after  12 iterations
  Number of observations                         20700
  Estimator                                         ML
  Minimum Function Test Statistic               13.361
  Degrees of freedom                                 1
  P-value (Chi-square)                           0.000
Model test baseline model:
  Minimum Function Test Statistic            14598.385
  Degrees of freedom                                 5
  P-value                                        0.000
User model versus baseline model:
  Comparative Fit Index (CFI)                    0.999
Root Mean Square Error of Approximation:
  RMSEA                                          0.024
  90 Percent Confidence Interval          0.014  0.037
  P-value RMSEA <= 0.05                          1.000
Standardized Root Mean Square Residual:
  SRMR                                           0.005
```

## Comparative fit indices



- compare fit of model with fit of baseline
- recall: baseline is unworthy opponent
- many indices in literature, $\mathrm{R}$ reports TLI and CFI
- Comparative Fit Index (good CFI $> 0.90$, better:$> 0.95$)
- Do not rely on strict cutoff values!

$$\text{CFI} = 1 - \frac{\max(\chi^2_{\text{model}} - \text{df}_{\text{model}}, 0)}{\max(\chi^2_{\text{base}} - \text{df}_{\text{base}}, 0)}$$

# Absolute fit indices



- Root Mean Square Error Approximation

$$\text{RMSEA} = \sqrt{\frac{\max(\chi^2_{\text{model}} - \text{df}_{\text{model}}, 0)}{\text{df}_{\text{model}}(N-1)}}$$

  - recall: if model fits, $E(\chi^2_{model}) = \text{df}$
  - RMSEA $<$ .08 (reasonable fit); RMSEA $<$ .05 (good fit)
  - $\mathrm{R}$ reports 90CI and PCLOSE = P(RMSEA<0.05).
  - Good fit: hi(90CI)<.05 and PCLOSE close to 1.
- Root Mean of squared standardized Residuals (SRMR $<$ 0.05)

$$\text{SRMR} = \sqrt{\frac{1}{ns}\left(\sum_{ij}\left(\frac{S_{ij}-\hat{\Sigma}_{ij}}{S_{ij}}\right)^2 + \sum_i\left(\frac{m_i-\hat{\mu}_i}{m_i}\right)^2\right)}$$

  Here: ns is number of sample moments, $S_{ij}$ and $m_i$ are sample moments, $\hat{\Sigma}_{ij}$ and $\hat{\mu}_i$ are fitted moments
- If $\chi^2$ test is rejected, check residual moments (obs-fitted covs)

# Checking residual moments: Blau & Duncan example



```
> resid(fit.bd, type="standardized")
$type
[1] "standardized"

$cov
      educ  occ1  faed  faocc
educ  0.000
occ1  0.000 0.000
faed  0.000 2.477 0.000
faocc 0.000 0.000 0.000 0.000

$mean$
 educ  occ1  faed faocc
    0     0     0     0
```

# Information criteria



Used to compare *nonnested* models

- Akaike's Information Criterion

$$\mathrm{AIC} = C + 2q$$

- Bayesian/Schwarz Information Criterion

$$\mathrm{BIC} = C + \ln(N) * q$$

- $C$ = LR-test-statistic comparing *model* with *saturated*
- $q$ = # parameters
- Lower AIC or BIC is better

# Minimum set of fit statsistics



- Model $\chi^2$, df, p-value (exact-fit hypothesis)
- RMSEA with 90% CI (close-fit hypothesis)
- CFI
- SRMR

If exact-fit hypothesis is rejected, analyze the residual correlation matrix, using data (residuals, modification indices etc.) modify the model in a theory guided and transparent way

## Model selection and model improvement



- strike a better balance between fit and complexity:
  - simplify model if possible (model trimming) – little loss in fit
  - complicate model if necessary (model building) – major improvement in fit
- simplifications to consider
  - constrain parameters to 0: remove arrows
  - equate parameters (eg controls have same effects on different yvars or across groups)
  - remove items in scale
- complications to consider (use modification indices)
  - free parameters
  - relax equality constraints

# Fixing, freeing, initializing parameters



Fix parameters to specific values: VALUE*x

- to simplify the model (eg, fix covariance to 0)
- to identify the model
- to avoid inadmissable solution (eg, fix a variance to 0 to circumvent a negative estimated value)
- substantive reasons

Setting a parameter free: NA*x or adding the parameter explicitly

- free a covariance that would otherwise be assumed to be 0
- in CFA: free loading of item (next weeks)
- relax cross-group constraints (next weeks)

Initialize parameters to reasonable values: start(0.8)*x2

- Iterative fitting requires good starting values
- R- lavaan is often ok, sometimes needs help

# Fixing, freeing, and initializing parameters—example



```

model <- 'y1 ~ 1*x1 + x3         #b(x1) fixed to 1
          y2 ~   x1 + x3
          y1 ~ 0*1                #intercept fixed to 0
          y1 ~~ start(0.12)*y1 #initial value error variance
          y1 ~~ y2'               #explicit frees cov(e.y1,e.y2)
fit <- sem(model, data=d)
```

# Imposing equality constraints—2 methods



```
model1 <- 'y1 ~ p*x1 + p*x2 # === METHOD 1 ===
                           #b(x2) and b(x2) labeled p
           y2 ~ x1
           y1 ~~ a*y1      #var(e.y1) and var(e.y2)
           y2 ~~ a*y2'     #labeled a hence equal
```

```
mod1   <- 'y1 ~ a*x1 + b*x2 # === METHOD 2 ===
           y2 ~ x1
           y1~~ c*y1
           y2~~ d*y2
           a == b          #constrains a to be equal to b
           c == d^2'       #non-linear constraints possible
```

Explicit constraints are more general, may be nonlinear: p1==p2/2, p3==p1+p2, p1*p2==1, p1>p3, etc (but: R may have difficulty fitting the model . . . )

Methods may be mixed

## Testing constraints (Wald-style)



In regression of y on x1 x2 x3, two common tests:

- test b(x1)=0 and b(x2)=0
- test b(x1)=b(x2)

In R-lavaan:

- give labels (cannot start with a number) to parameters,
- or extract default labels used by lavaan (`coef(fit)`)
- Wald test: use `lavTestWald()` with (non)linear constraints
  - lavTestWald(fit, constraints = "a == b")
  - lavTestWald(fit, constraints = con)
- Multiple constraints merged in `lavTestWald()`

# Testing constraints—examples



model and constraints

```
mod1 <- 'y1 ~ a*x1 + b*x2
         y2 ~ x1
         y1~~ c*y1
         y2~~ d*y2'
fit3 <- sem(mod1, data=d)
con <- 'a == b
        c == d^2'
```

Wald test edited output

```
> lavTestWald(fit3, constraints = 'a==0')
$stat [1] 196.3631; $df [1] 1
$p.value [1] 0; $se [1] "standard"

> lavTestWald(fit3, constraints = con)
$stat [1] 16.01391; $df [1] 2
$p.value [1] 0.0003331379; $se [1] "standard"
```

# Testing constraints—likelihood ratio test



- Beware, things change if other than ML estimator
- Fit two models: (1) unconstrained, (2) constrained
- Write down log-likelihoods and dfs:
  $\mathrm{LL}_u$ and $\mathrm{df}_u$
  $\mathrm{LL}_c$ and $\mathrm{df}_c$
- Compute

$$\begin{aligned} \mathrm{LR} &= 2 * (\mathrm{LL}_u - \mathrm{LL}_c) \\ \mathrm{df} &= \mathrm{df}_u - \mathrm{df}_c \\ p &= \mathrm{Prob}(\mathrm{Chi2}(\mathrm{df}) > \mathrm{LR}) \end{aligned}$$

  p? R: `pchisq(LR, df, lower.tail = FALSE)`
- Or better use `anova(modu, modc)`

# Testing constraints—robust likelihood ratio test



When estimator is MLM, MLR, or WLSM, the “manual” LR test above is invalid. Use Sattora-Bentler LR test instead:

c0/c1 : scaling correction factor for $M_0$ / $M_1$
d0/d1 : degrees of freedom for $M_0$ / $M_1$
T0/T1 : Satorra-Bentler scaled $\chi^2$ for $M_0$ / $M_1$

cd = (d0 * c0 - d1 * c1)/(d0 - d1)
T = (T0 * c0 - T1 * c1)/cd
T is distributed $\chi^2$ with df = d0 - d1

The `anova()` function still works...

```
fit3 <- sem(mod1, data=d, estimator = 'MLR')
fit2 <- sem(mod2, data=d, estimator = 'MLR')
anova(fit3, fit2)
```

# Testing constraints with LR test–Example



LRtest

```
mod2 <- 'y1 ~ a*x1 + b*x2
         y2 ~ x1 + x2
         y1~~ c*y1
         y2~~ d*y2'
mod3 <- 'y1 ~ a*x1 + a*x2
         y2 ~ x1 + x2
         y1~~ b*y1
         y2~~ b*y2'
fit3 <- sem(mod3, data=d)
fit2 <- sem(mod2, data=d)
anova(fit3, fit2)
```

LR-test output

```
> anova(fit3, fit2)
Chi Square Difference Test

     Df    AIC    BIC  Chisq Chisq diff Df diff Pr(>Chisq)
fit2  0 1377.1 1395.4  0.000
fit3  2 1395.9 1408.9 22.745     22.745       2  1.151e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

# Complicate models using Modification Indices



- Modification Indices: score tests for parameters currently fixed to 0 or otherwise constrained
- Chi2(1df) distributed if constraint is true
- Score tests are approximate LRtests, guestimated without refitting extended model
- Wald tests are approximate LRtests, guestimated without refitting more restricted model
- check modindex $\geq 4$ (ex ante) or $\geq 10$ (ex post)

```
summary(fit3, modindices=TRUE)
mi3 <- modindices(fit3)
mi3[mi3$op == "~~",]
mi3[mi3$mi > 4,]
```

# Modification indices (output)



MODINDICES for (Blau-Duncan)

```
summary(fit.bd, modindices = TRUE)
[...omitted for brevity...]
Modification Indices:

     lhs op   rhs     mi    epc sepc.lv sepc.all sepc.nox
7   faed ~~  faed  0.000  0.000   0.000    0.000    0.000
8   faed ~~ faocc  0.000  0.000   0.000    0.000    0.000
9  faocc ~~ faocc  0.000  0.000   0.000    0.000    0.000
12  educ ~~  occ1 13.357 -0.061  -0.061   -0.061   -0.061
13  educ  ~  occ1 13.357 -0.090  -0.090   -0.090   -0.090
14  occ1  ~  faed 13.357  0.025   0.025    0.025    0.025
15  faed  ~  educ  0.000  0.000   0.000    0.000    0.000
16  faed  ~  occ1 10.053  0.021   0.021    0.021    0.021
17  faed  ~ faocc  0.000  0.000   0.000    0.000    0.000
18 faocc  ~  educ  0.000  0.000   0.000    0.000    0.000
19 faocc  ~  occ1  4.960 -0.020  -0.020   -0.020   -0.020
20 faocc  ~  faed  0.000  0.000   0.000    0.000    0.000
```

Note the insensible modifications

# Non-recursive path models



- Causal direct/indirect feedback
- Mutual influence
- Correlated disturbances with direct effects between DVs
- Example: dyadic analysis – friends, siblings, influence each other...
- Identification is not guaranteed

# Non-recursive path model



- Direct feedback loop
- Infinite number of indirect effects

unidentified non-recursive

```
mod3 <- 'y1 ~ a*x1 + x2
         y2 ~ b*x2 + x1
         y1 ~ c*y2
         y2 ~ d*y1
         y1 ~~ y2
summary(fit3 <- sem(mod3, data=d))
```

identified non-recursive

```
mod4 <- 'y1 ~ a*x1
         y2 ~ b*x2
         y1 ~ c*y2
         y2 ~ d*y1
         y1 ~~ y2
         e := (a*c)/(1-(c*d))'
summary(fit4 <- sem(mod4, data=d))
```

# Identification



- Identification: existence of unique set of parameter estimates
- Possible: some parameters identified, others unidentified
- Necessary but not sufficient conditions:
  1. $Df \geq 0$ (N-parameters $\geq$ N-obs)
  2. Every latent variable (e.g., disturbance, factor) has a scale
- Underidentification ($Df < 0$): multiple parameters lead to same predictions, which ones should be reported?
- Example: EFAs are underidentified, enabling rotation
- Recursive models with (1) and (2) are identified
- For non-recursive models difficult to assess identification
  - order condition: For an equation to be identified, the number of excluded exogenous variables for each endogenous variable must be at least as large as the number of total endogenous variables, minus one.
  - rank condition: definite answer, but complicated to assess
  - empirical unidentification
  - run model with artificial data before data collection