Introduction to SEM

Tarek Carls

2024-09-13

Introduction to SEM

Introduction to SEM

Purpose

This seminar introduces basic concepts of structural equation modeling (SEM) using lavaan in the R programming language. The emphasis is on **identifying various manifestations of SEM models** and **interpreting the output.**.

Fundamental topics covered include:

- Matrix notation
- Identification
- Model fit
- Various kind of models...

Assumption: All variables are continuous and normally distributed.

Introduction

Introduction

What's SEM?

Structural equation modeling (SEM) is a linear model framework that models both **simultaneous regression equations** with **latent variables**.

Special cases of SEM:

- linear regression
- multivariate regression
- path analysis
- confirmatory factor analysis
- structural regression

What can we do in SEM?

Using SEM, one can model the following relationships

- observed to observed variables (e.g., regression)
- latent to observed variables (e.g., confirmatory factor analysis)
- latent to latent variables (e.g., structural regressoins)

We can fit **measurement** (relating observed to latent variables) models and **structural** (relating latent to latent variables) models

Let's get started!

Regression and Path Analysis

Simple Regression

You will most likely know this equation:

$y_1 = b_0 + b_1 x_1 + \epsilon_1$

Here, b_0 is the intercept, b_1 is the coefficient and x_1 is the observed predictor while ϵ_1 is the residual. However, in SEM one often finds the LISREL notation, which reads as follows:

$y_1 = \alpha + \gamma x_1 + \zeta_1$

Simple regression

$y_1 = \alpha + \gamma x_1 + \zeta_1$

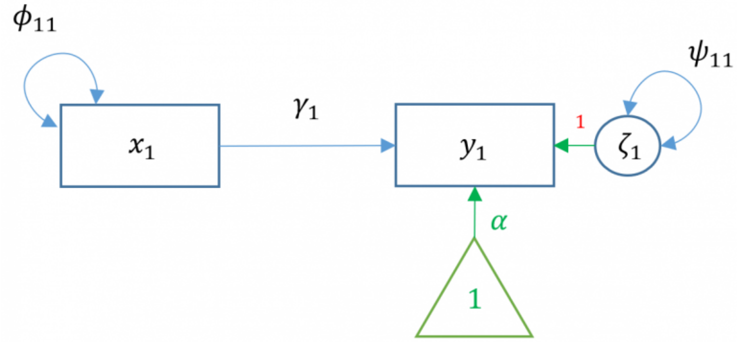

Figure 4: Visual representation of the matrix

Simple regression

The standard way to run a linear regression in R is the lm() function from base R.

```
m1a <- lm(Bout_t1 ~ Psysa_t1,data=df)
fit1a <- summary(m1a)
fit1a
```

```
##
## Call:
## lm(formula = Bout_t1 ~ Psysa_t1, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.95311 -0.58007 -0.08007  0.48537  2.29297
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.69136    0.18900  19.531  < 2e-16 ***
## Psysa_t1    -0.24608    0.04903  -5.019 6.34e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7791 on 852 degrees of freedom
## Multiple R-squared:  0.02871,    Adjusted R-squared:  0.02757
## F-statistic: 25.19 on 1 and 852 DF,  p-value: 6.336e-07
```

Simple regression in lavaan

Now we estimate the same regression in lavaan(). Here, the intercept is not included by default so we have to add it.

```
m1b <- '
  # regression
  Bout_t1 ~ 1 + Psysa_t1
  # Variance
  Psysa_t1 ~~ Psysa_t1 # this is estimated by default in lavaan
'
fit1b <- sem(m1b, data=df)
summary(fit1b)
```

```
## lavaan 0.6-18 ended normally after 10 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                         5
##
##   Number of observations                           854
##
## Model Test User Model:
##
##   Test statistic                                 0.000
##   Degrees of freedom                                 0
##
## Parameter Estimates:
##
##   Standard errors                             Standard
##   Information                                 Expected
##   Information saturated (h1) model          Structured
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   Bout_t1 ~
```

Maximum Likelihood vs. least squares

Multiple regression

Model Fit Statistics

Measurement Model

Structural Model