The ‘Structural After Measurement’ (SAM) approach to SEM

Yves Rosseel
Department of Data Analysis
Ghent University – Belgium

CFE-CMStatistics – King’s College London
14 December, 2024

acknowledgements

- Wen Wei Loh (Maastricht University)
- Sara Dhaene (Delaware BeLux), Julie De Jonckere (UGent)
- Jasper Bogaert (UGent), Seda Can (University of Izmir)
- Andres Perez Alonso (Tilburg), Jennifer Dang Guay (KUL), Jeroen Vermunt (Tilburg) & Kim De Roover (KUL)

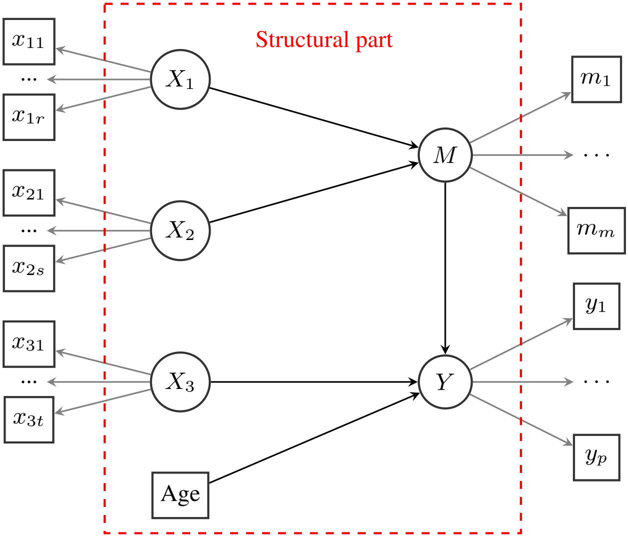
paper

Rosseel, Y., & Loh, W.W. (2024). A structural after measurement approach to structural equation modeling. *Psychological Methods, 29*(3), 561–588.
https://doi.org/10.1037/met0000503

slides

https://lavaan.org

the setting: structural equation models



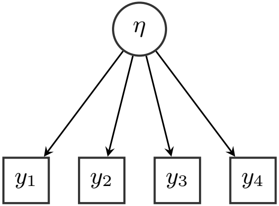
the setting (2)

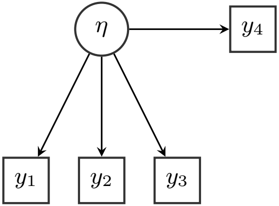
- structural equation models
- we focus on 'large' models with many (say, > 100) parameters:
  - many constructs (motivation, ability, personality traits, . . .)
  - each construct is measured by a set of (observed) indicators
  - many 'background' variables (age, gender, . . .)
  - multilevel data, missing data, . . .
- we are mostly interested in the structural part of the model:
  - regression model: variables are either dependent or independent
  - path analysis model: includes mediating effects, perhaps non-recursive
- assumption: the measurement instruments for the latent variables are well established, and usually fit (reasonably) well
- BUT: the sample size is not large (say, $N = 150$)

the standard estimation approach in SEM: 'system-wide' estimation

- all parameters (measurement and structural) are estimated jointly
- frequentist: typically using an iterative optimization approach (e.g., ML); Bayesian: typically using MCMC
- advantages:
  - one-step, and therefore efficient (in terms of sampling variability)
  - inference is straightforward (standard errors, hypothesis testing)
  - (relatively) easy to handle constraints, missing data, ...
- works very well if the following conditions are met:
  - correctly specified model, large sample size
  - (normally distributed data)
- but under less ideal circumstances, system-wide estimation does not (always) work well (bias, instability, nonconvergence, improper solutions, ...)
- in addition: combining 'structural' and 'measurement' may lead to interpretational confounding (the meaning of the latent variables changes, if the structural model changes)

example (stolen from Roy Levy, 2023)





- left panel: only measurement
- right panel: measurement + structural
- conceptually very different
- mathematically identical (in system-wide SEM)

generate some data

```
> library(lavaan)
> Sigma <- matrix(c(2.0, 1.0, 1.0, 1.0, 1.5,
                    1.0, 2.0, 1.0, 1.0, 0.1,
                    1.0, 1.0, 2.0, 1.0, 0.1,
                    1.0, 1.0, 1.0, 2.0, 0.1,
                    1.5, 0.1, 0.1, 0.1, 2.0), nrow = 5, ncol = 5)
> rownames(Sigma) <- colnames(Sigma) <- c("y1", "y2", "y3", "y4", "z")
> Sigma

    y1  y2  y3  y4   z
y1 2.0 1.0 1.0 1.0 1.5
y2 1.0 2.0 1.0 1.0 0.1
y3 1.0 1.0 2.0 1.0 0.1
y4 1.0 1.0 1.0 2.0 0.1
z  1.5 0.1 0.1 0.1 2.0

> set.seed(3)
> Data <- MASS::mvrnorm(n = 200L, mu = rep(0, 5), Sigma = Sigma)
```

R code left panel (model1)

```
> model1 <- '
      f =~ y1 + y2 + y3 + y4
 '
> fit1 <- sem(model1, data = Data)
> summary(fit1)
```

```
lavaan 0.6-20.2250 ended normally after 25 iterations

  Estimator                                         ML
  Optimization method                           NLMINB
  Number of model parameters                         8

  Number of observations                           200

Model Test User Model:

  Test statistic                                 0.140
  Degrees of freedom                                 2
  P-value (Chi-square)                           0.932

Parameter Estimates:

  Standard errors                             Standard
  Information                                 Expected
  Information saturated (h1) model          Structured

Latent Variables:
```

```
                     Estimate  Std.Err  z-value  P(>|z|)
  f =~
    y1                  1.000
    y2                  0.996    0.123    8.122    0.000
    y3                  0.894    0.116    7.689    0.000
    y4                  0.911    0.120    7.608    0.000

Variances:
                     Estimate  Std.Err  z-value  P(>|z|)
   .y1                  1.096    0.148    7.388    0.000
   .y2                  0.837    0.127    6.601    0.000
   .y3                  0.987    0.128    7.694    0.000
   .y4                  1.081    0.138    7.820    0.000
    f                   1.031    0.208    4.959    0.000
```

R code right panel (model2)

```
> model2 <- '
     f =~ y1 + y2 + y3
     y4 ~ f
  '
> fit2 <- sem(model2, data = Data)
> summary(fit2)
```

```
lavaan 0.6-20.2250 ended normally after 26 iterations

  Estimator                                         ML
  Optimization method                           NLMINB
  Number of model parameters                         8

  Number of observations                           200

Model Test User Model:

  Test statistic                                 0.140
  Degrees of freedom                                 2
  P-value (Chi-square)                           0.932

Parameter Estimates:

  Standard errors                             Standard
  Information                                 Expected
  Information saturated (h1) model          Structured
```

```
Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)
  f =~
    y1                1.000
    y2                0.996    0.123    8.122    0.000
    y3                0.894    0.116    7.689    0.000

Regressions:
                   Estimate  Std.Err  z-value  P(>|z|)
  y4 ~
    f                 0.911    0.120    7.608    0.000

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)
   .y1                1.096    0.148    7.388    0.000
   .y2                0.837    0.127    6.601    0.000
   .y3                0.987    0.128    7.694    0.000
   .y4                1.081    0.138    7.820    0.000
    f                 1.031    0.208    4.959    0.000
```

change outcome variable (y4 becomes z) (model3)

```
> model3 <- '
     f =~ y1 + y2 + y3
     z ~ f
 '
> fit3 <- sem(model3, data = Data)
> summary(fit3)
lavaan 0.6-20.2250 ended normally after 58 iterations

  Estimator                                         ML
  Optimization method                           NLMINB
  Number of model parameters                         8

  Number of observations                           200

Model Test User Model:

  Test statistic                                50.659
  Degrees of freedom                                 2
  P-value (Chi-square)                           0.000

Parameter Estimates:

  Standard errors                             Standard
  Information                                 Expected
  Information saturated (h1) model          Structured
```

```
Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)
  f =~
    y1                1.000
    y2                0.138    0.053    2.601    0.009
    y3                0.133    0.051    2.591    0.010

Regressions:
                   Estimate  Std.Err  z-value  P(>|z|)
  z ~
    f                 0.276    0.087    3.156    0.002

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)
   .y1               -4.169    1.801   -2.315    0.021
   .y2                1.740    0.173   10.071    0.000
   .y3                1.699    0.168   10.086    0.000
   .z                 1.677    0.211    7.953    0.000
    f                 6.296    1.715    3.671    0.000
```

interpretational confounding

- replacing y_4 by z (= changing the structural part) also changes the parameters of the measurement model
- if the resulting parameters of the measurement model imply a different 'meaning' of the latent variable than was intended by the researcher, we have a problem
- this problem was coined "interpretational confounding" by Burt (1976)

  Burt, R.S. (1976). Interpretational confounding of unobserved variables in structural equation models. *Sociological Methods & Research, 5*(1), 3–52.

- Burt (1976) already suggested the solution: first fit the measurement part of the model, and then fit the structural part of the model

solution: replace sem() by sam()

```
> fit3.sam <- sam(model3, data = Data)
> parameterEstimates(fit3.sam, remove.step1 = FALSE, ci = FALSE,
                     output = "text")

Latent Variables:
                   Step Estimate  Std.Err  z-value  P(>|z|)
  f =~
    y1                1    1.000
    y2                1    0.969    0.139    6.977    0.000
    y3                1    0.884    0.127    6.979    0.000

Regressions:
                   Step Estimate  Std.Err  z-value  P(>|z|)
  z ~
    f                 2    0.540    0.121    4.471    0.000

Variances:
                   Step Estimate  Std.Err  z-value  P(>|z|)
   .y1                1    1.067    0.170    6.287    0.000
   .y2                1    0.865    0.151    5.723    0.000
   .y3                1    0.983    0.142    6.900    0.000
   .z                 2    1.846    0.195    9.462    0.000
    f                 2    1.059    0.226    4.678    0.000
```

the structural-after-measurement (SAM) approach

- SAM is an umbrella term to describe many different (estimation) approaches that have the following in common:
  - first step: we estimate the parameters related to the measurement part
  - second step: we estimate the parameters related to the structural part
- the term SAM was used by Rosseel & Loh (2024), to avoid the overloaded terms ‘two-step’, ‘two-stage’, …
- various SAM approaches have been suggested in the literature:
  - early references: Burt (1976), Hunter & Gerbing (1982), Lance, Cornwell & Mulaik (1988)
  - (uncorrected and bias-corrected) factor score regression (FSR)
  - SAM is the default approach in many other fields
- …but they never received much attention in the SEM literature/community
- Rosseel & Loh (2024) proposed a special case: ‘local SAM’ (LSAM)

local SAM: rationale

- the measurement model:

$$\boldsymbol{y} = \boldsymbol{\nu} + \boldsymbol{\Lambda}\boldsymbol{\eta} + \boldsymbol{\epsilon}$$

- to solve this for $\boldsymbol{\eta}$, we proceed as follows:

$$\begin{aligned} \boldsymbol{\nu} + \boldsymbol{\Lambda}\,\boldsymbol{\eta} + \boldsymbol{\epsilon} &= \boldsymbol{y} \\ \boldsymbol{\Lambda}\,\boldsymbol{\eta} &= \boldsymbol{y} - \boldsymbol{\nu} - \boldsymbol{\epsilon} \\ \mathbf{M}\boldsymbol{\Lambda}\,\boldsymbol{\eta} &= \mathbf{M}\left[\boldsymbol{y} - \boldsymbol{\nu} - \boldsymbol{\epsilon}\right] \\ \boldsymbol{\eta} &= \mathbf{M}\left[\boldsymbol{y} - \boldsymbol{\nu} - \boldsymbol{\epsilon}\right] \end{aligned}$$

  where $\mathbf{M}$ is $M \times P$ mapping matrix such that $\mathbf{M}\boldsymbol{\Lambda} = \mathbf{I}_M$

- we assume $\mathrm{E}(\boldsymbol{\epsilon}) = \mathbf{0}$ and write $\mathrm{Var}(\boldsymbol{\epsilon}) = \boldsymbol{\Theta}$; it follows that

$$\begin{aligned} \mathrm{E}(\boldsymbol{\eta}) &= \mathbf{M}\left[\mathrm{E}(\boldsymbol{y}) - \boldsymbol{\nu}\right] \\ \mathrm{Var}(\boldsymbol{\eta}) &= \mathbf{M}\left[\mathrm{Var}(\boldsymbol{y}) - \boldsymbol{\Theta}\right]\mathbf{M}^T \end{aligned}$$

local SAM: estimation

- first stage: estimation of the measurement part of the model (only)
- this results in estimates of:
  - $\mathrm{E}(\boldsymbol{\eta})$: the mean vector of the latent variables
  - $\mathrm{Var}(\boldsymbol{\eta})$: the variance-covariance matrix of the latent variables
  - $\Gamma(\boldsymbol{\eta})$ ('Gamma'): capturing the sampling variability of these sample statistics
- second stage: a regression or path analysis is performed, using the sample statistics of the latent variables as input
  - twostep-corrected standard errors and fit measures
  - we can use ML, GLS, ... or we can use noniterative estimators (OLS, TSLS)
- typical choice for the mapping matrix: the 'ML/Bartlett' matrix

$$\mathbf{M} = (\boldsymbol{\Lambda}^T \boldsymbol{\Theta}^{-1} \boldsymbol{\Lambda})^{-1} \boldsymbol{\Lambda}^T \boldsymbol{\Theta}^{-1}$$

application 1: adding latent quadratic and interaction terms (2)

- in the joint setting, adding latent quadratic/interaction terms is not trivial
- two popular methods are the product-indicator (PI) approach, and the so-called 'Latent Moderated Structural Equations' (LMS) approach
- none of these scale well: they cannot handle many quadratic and latent interaction terms simultaneously
- but if you can decouple the measurement and structural part, this becomes feasible
- a very general SAM solution (allowing for polynomial relations between latent variables) was already described in Wall & Amemiya (2000)
- the local SAM approach: find an explicit expression for

$$\mathrm{E}(\boldsymbol{\eta} \otimes \boldsymbol{\eta}) \qquad \text{and} \qquad \mathrm{Var}(\boldsymbol{\eta} \otimes \boldsymbol{\eta})$$

where $\otimes$ denotes the tensor (or Kronecker) product

the augmented (latent) sample statistics

- augmented mean vector:

$$\mathrm{E}(\boldsymbol{\eta} \otimes \boldsymbol{\eta}) = \mathrm{vec}[\mathrm{Var}(\boldsymbol{\eta})] + \mathrm{E}(\boldsymbol{\eta}) \otimes \mathrm{E}(\boldsymbol{\eta})$$

- augmented variance-covariance matrix (simple version, assuming normality for the measurement error):

$$\mathrm{Var}(\boldsymbol{\eta} \otimes \boldsymbol{\eta}) \approx \; \mathrm{Var}(\mathbf{f} \otimes \mathbf{f}) - \left[\mathbf{Q} + \mathbf{K}_m\mathbf{Q} + \mathbf{Q}\mathbf{K}_m^T + \mathbf{K}_m\mathbf{Q}\mathbf{K}_m^T + \boldsymbol{\Gamma}_{22}^{\star(NT)}(\mathbf{r})\right]$$

where

$$\mathbf{Q} = \mathrm{Var}(\boldsymbol{\eta}) \otimes \mathrm{Var}(\mathbf{r}) + \mathrm{E}(\boldsymbol{\eta})\mathrm{E}(\boldsymbol{\eta})^T \otimes \mathrm{Var}(\mathbf{r})$$

and

$$\boldsymbol{\Gamma}_{22}^{\star(NT)}(\mathbf{r}) = (\mathbf{I}_{m^2} + \mathbf{K}_m)\,(\mathrm{Var}(\mathbf{r}) \otimes \mathrm{Var}(\mathbf{r}))$$

- $\mathbf{K}_m$ is the commutation matrix
- $\boldsymbol{\Gamma}_{22}^{\star(NT)}(\mathbf{r})$ is the ‘Gamma’ matrix of the measurement error $(\mathbf{r})$

implementation in lavaan

```
> model <- '
     # measurement part
     f1 =~ y1 + y2 + y3
     f2 =~ y4 + y5 + y6
     f3 =~ y7 + y8 + y9

     # structural part
     f3 ~ f1 + f2 + f1:f1 + f2:f2 + f1:f2
 '
> fit <- sam(model, data = Data, se = "none") # or se = "bootstrap"
```

- no two-step analytic standard errors yet; but bootstrapping is possible
- forthcoming paper:

Rosseel, Y., Burghgraeve, E., Loh, W.W., Schermelleh-Engel, K. (in press). Structural after Measurement (SAM) approaches for accommodating latent quadratic and interaction effects. *Behavior Research Methods*.

application 2: noniterative SEM

- for CFA, many noniterative estimators are available; some (i.e., the multiple group method) perform better than ML in terms of mean squared error

  Dhaene, S. & Rosseel, Y. (2023). An Evaluation of Non-Iterative Estimators in Confirmatory Factor Analysis. *Structural Equation Modeling: A Multidisciplinary Journal*.

- we can use these noniterative estimators for the measurement part in SAM

  Dhaene, S., & Rosseel, Y. (2023). An Evaluation of Non-Iterative Estimators in the Structural after Measurement (SAM) Approach to Structural Equation Modeling (SEM). *Structural Equation Modeling: A Multidisciplinary Journal*, 30(6), 926–940

- “[the] local SAM approach outperforms traditional SEM in small to moderate samples (both in terms of convergence and MSE values), especially when reliability drops. ”

application 3: comparing structural relations across many groups

- reference (open access):

  Perez Alonso, A.F., Rosseel, Y., Vermunt, J.K., & De Roover, K. (in press). Mixture Multigroup Structural Equation Modeling: A Novel Method for Comparing Structural Relations Across Many Groups. *Psychological Methods*. https://doi.org/10.1037/met0000667

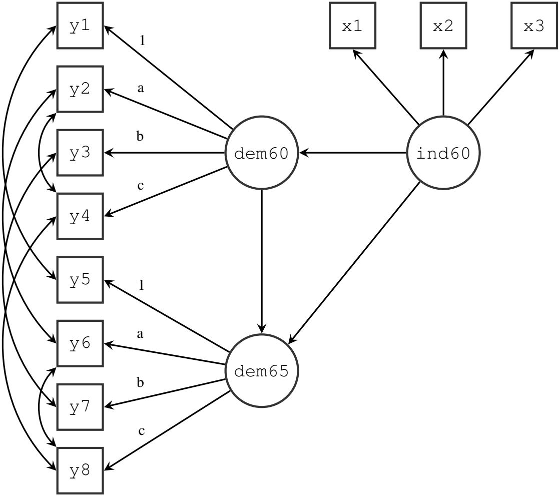
- relationships between latent variables are often different across groups (e.g., countries); but some groups may be similar in the sense that they have similar values for the regression coefficients
- we like to 'discover' these hidden clusters of similar groups
- in a first step, we estimated the measurement part across all groups (fixing the factor loadings to be the same across groups); this resulted in (model-implied) latent (co)variance matrices for all the groups
- in a second step, a mixture modeling approach is used to find homogeneous clusters that share similar regression coefficients

SAM: software implementation

- the SAM approach has been implemented in the sam() function in the R package lavaan
- available methods:
  - `sam.method = "local"` (default)
  - `sam.method = "global"`
  - `sam.method = "fsr"` (using Bartlett factor scores)
- typical call:

```
> fit.sam <- sam(model, data = PoliticalDemocracy,
                 sam.method = "local",
                 # link measurement blocks
                 mm.list = list(ind = "ind60", dem = c("dem60", "dem65")),
                 # measurement options
                 mm.args = list(estimator = "ML"),
                 # structural options
                 struc.args = list(estimator = "GLS"),
                 # global options
                 meanstructure = FALSE)
```

diagram of the Political Democracy model



sam() output

```
> # just for illustration, we also show the estimated parameters
> # of the measurement blocks
> #
> summary(fit.sam, remove.step1 = FALSE)
This is lavaan 0.6-20.2250 -- using the SAM approach to SEM

  SAM method                                          LOCAL
  Mapping matrix M method                                ML
  Number of measurement blocks                            2
  Estimator measurement part                             ML
  Estimator  structural part                            GLS

  Number of observations                                 75

Summary Information Measurement + Structural:

  Block      Latent Nind Chisq Df
      1       ind60    3  0.00  0
      2 dem60,dem65    8 15.32 16

  Model-based reliability latent variables:

  ind60 dem60 dem65
  0.966 0.868  0.87

  Summary Information Structural part:
```

```
  chisq df cfi rmsea srmr
      0  0   1     0    0

Parameter Estimates:

  Standard errors                                Twostep
  Information                                   Expected
  Information saturated (h1) model            Structured

Latent Variables:
                      Step Estimate  Std.Err  z-value  P(>|z|)
  ind60 =~
    x1                   1    1.000
    x2                   1    2.193    0.142   15.403    0.000
    x3                   1    1.824    0.153   11.883    0.000
  dem60 =~
    y1                   1    1.000
    y2          (a)      1    1.213    0.143    8.483    0.000
    y3          (b)      1    1.210    0.125    9.690    0.000
    y4          (c)      1    1.273    0.122   10.453    0.000
  dem65 =~
    y5                   1    1.000
    y6          (a)      1    1.213    0.143    8.483    0.000
    y7          (b)      1    1.210    0.125    9.690    0.000
    y8          (c)      1    1.273    0.122   10.453    0.000

Regressions:
```

```
                      Step Estimate  Std.Err  z-value  P(>|z|)
  dem60 ~
    ind60                2    1.454    0.389    3.741    0.000
  dem65 ~
    ind60                2    0.558    0.225    2.480    0.013
    dem60                2    0.871    0.076   11.497    0.000

Covariances:
                      Step Estimate  Std.Err  z-value  P(>|z|)
 .y1 ~~
   .y5                   1    0.577    0.364    1.585    0.113
 .y2 ~~
   .y4                   1    1.390    0.685    2.030    0.042
   .y6                   1    2.068    0.733    2.822    0.005
 .y3 ~~
   .y7                   1    0.727    0.611    1.190    0.234
 .y4 ~~
   .y8                   1    0.476    0.453    1.049    0.294
 .y6 ~~
   .y8                   1    1.257    0.583    2.156    0.031

Variances:
                      Step Estimate  Std.Err  z-value  P(>|z|)
   .x1                   1    0.084    0.020    4.140    0.000
   .x2                   1    0.108    0.074    1.455    0.146
   .x3                   1    0.468    0.091    5.124    0.000
   .y1                   1    1.879    0.431    4.355    0.000
   .y2                   1    7.530    1.363    5.523    0.000
```

```
 .y3              1   4.966   0.966   5.141   0.000
 .y4              1   3.214   0.722   4.449   0.000
 .y5              1   2.499   0.518   4.824   0.000
 .y6              1   4.809   0.924   5.202   0.000
 .y7              1   3.302   0.699   4.722   0.000
 .y8              1   3.227   0.720   4.482   0.000
  ind60           2   0.446   0.087   5.135   0.000
 .dem60           2   3.766   0.848   4.439   0.000
 .dem65           2   0.189   0.224   0.843   0.399
```

last slide

- why should we decouple the measurement and structural part?
  - because we should (avoid interpretational confounding)
  - because we can (we can still do SEM)
  - this is what is done in most fields outside SEM
  - good performance in simulation studies (small/moderate sample sizes)
  - opens up modeling possibilities that were (computationally) difficult (if not impossible) in a joint estimation approach
- but still some obstacles:
  - analytic standard errors not always available (yet)
  - limitations (e.g., no higher-order measurement models)
  - more studies are needed to discover potential weaknesses
- the SAM approach deserves the (renewed) interest of the SEM community

Thank you!

(questions?)

https://lavaan.org

https://lavaan.ugent.be/about/donate.html

references

Bakk, Z., & Kuha, J. (2021). Relating latent class membership to external variables: An overview. *British Journal of Mathematical and Statistical Psychology*, 74(2), 340–362.
https://doi.org/10.1111/bmsp.12227 (Open Access)

Burt, R.S. (1976). Interpretational confounding of unobserved variables in structural equation models. *Sociological Methods & Research*, 5(1), 3–52.

Dhaene, S., & Rosseel, Y. (2023). An Evaluation of Non-Iterative Estimators in the Structural after Measurement (SAM) Approach to Structural Equation Modeling (SEM). *Structural Equation Modeling: A Multidisciplinary Journal*, 30(6), 926–940
https://doi.org/10.1080/10705511.2023.2220135 (Open Access)

Dhaene, S. & Rosseel, Y. (2023). An Evaluation of Non-Iterative Estimators in Confirmatory Factor Analysis. *Structural Equation Modeling: A Multidisciplinary Journal*.
https://doi.org/10.1080/10705511.2023.2187285. (Open Access)

Fornell, C., & Yi, Y. (1992). Assumptions of the two-step approach to latent variable modeling. *Sociological Methods & Research*, 20(3), 291–320.

Fuller, W A. (1987). *Measurement error models*. John Wiley & Sons.

Hunter, J.E., & Gerbing, D. W. (1982). Unidimensional measurement, second-order factor analysis, and causal models. *Research in Organizational Behavior*, 4, 267–320.

Lance, C.E., Cornwell, J.M., & Mulaik, S.A. (1988). Limited information parameter estimates for latent or mixed manifest and latent variable models. *Multivariate Behavioral Research*, 23 (2), 171–187.

Levy, R. (2023). Precluding interpretational confounding in factor analysis with a covariate or outcome via measurement and uncertainty preserving parametric modeling. *Structural Equation Modeling: A Multidisciplinary Journal.*
https://doi.org/10.1080/10705511.2022.2154214

Perez Alonso, A.F., Rosseel, Y., Vermunt, J.K., & De Roover, K. (in press). Mixture Multigroup Structural Equation Modeling: A Novel Method for Comparing Structural Relations Across Many Groups. *Psychological Methods*.

Rosseel, Y., & Loh, W.W. (2024). A structural after measurement approach to structural equation modeling. *Psychological Methods, 29*(3), 561–588.
https://doi.org/10.1037/met0000503
https://osf.io/pekbm/ (includes original version)

Rosseel, Y., Burghgraeve, E., Loh, W.W., Schermelleh-Engel, K. (accepted). Structural after Measurement (SAM) approaches for accommodating latent quadratic and interaction effects. *Behavior Research Methods*.

Wall, M.M., & Amemiya, Y. (2000). Estimation for polynomial structural equation models. *Journal of the American Statistical Association*, 95(451), 929–940.