Case based vs. Covariance based SEM

Reinhard Oldenburg

Received: date / Accepted: date

Abstract The traditional approach to structural equation modeling (SEM) estimation is based on fitting the model-implied covariance matrix to the empirical covariance matrix. This approach will be contrasted to a direct least square fitting of the model equations. Principle strengths and weaknesses of this approach are discussed and simulation studies are performed to reveal problems and potentials of the two approaches and how they compare to other alternative estimation strategies.

Keywords Equation modeling · Simulation · Least squares · Latent variables

1 Introduction

It is needless to say anything about the importance of structural equational modelling. A good overview of its development and application is given in Hoyle (2012). However, despite the many virtues of SEM, there are a number of reasons why one may want to look for ways to extend the framework. This paper will introduce a very easy least squares estimation procedure which is however computationally demanding. More than 40 years ago, when SEM took shape, computers were simply not fast enough to perform these calculations. Hence, it was a clever step not to work directly with the model equations but to derive a theoretical and parameter-dependent covariance matrix from the equations and fit this to the covariance matrix of the data. With modern computers, however, fitting the model equations directly, is no longer a big problem, and hence it is worth investigating the benefits (and the drawbacks) of this approach.
The most common approach to SEM is based on fitting the empirical and the model covariance matrix in various ways. This family of methods (including various estimation techniques such as unweighted or weighted least squares, maximum likelihood and variations) will be called tradition SEM (TSEM). Despite its many strengths, there are several reasons why TSEM may be regarded to be not optimal in certain situations:

1. TSEM requires the nontrivial step from model equations to the model implied covariance matrix. Although software packages for SEM perform this for the user, it is still a step away from the data and makes results more detached from the original data. More precisely, this step can only be taken under strong assumptions of the independence of error variables. Therefore TSEM forbids certain linear models because they are not identified although the model itself is sensible and well defined.

2. TSEM is well suited only for linear models. There are some extensions to non-linear models (see e.g. Schumacker and Marcoulides 1998, Umbach et al. 2017) but they lack the ease of use that linear models have. Some approaches are limited to very special generalizations, e.g. quadratic terms. While this has the advantage that of lot of theory can be carried over (e.g. Kelava et al. 2011, Dijkstra and Schermelleh-Engel 2014), it may mean that the modelling flexibility is too small to match a certain situation, e.g. piecewise linear models cannot be treated that way. However, it seems that especially the ability to use piece-wise functions like

\[
y = \begin{cases} 
  a \cdot x + b & x < x_0 \\
  a \cdot x_0 + b + c \cdot (x - x_0) & x \geq x_0
\end{cases}
\]

would enhance the benefit of SEM by allowing to identify points ( \( x_0 \) ) where the slope of the linear relation changes, e.g. beyond a certain extent an influence may no longer enhance some result. Moreover, relations between two (normalized) variables \( x, y \) of the form \( \Theta(x) \cdot (y - k \cdot x) = 0 \) (where \( \Theta \) is the Heaviside function) model implicative relations as will be shown in a follow-up paper.

3. TSEM does not give estimates of the values of latent variables for each case. However, Bayesian variants can do this, e.g. the blavaan software package (Merkle and Rosseel 2015), but this usually takes even more computer time than the proposed version. Moreover, the PLS approach to SEM (Esposito Vinci et al. 2017) first estimates latent variables by regression on the measurement model and then estimates the structural model based on this values. This, however, restricts the class of models to such models that have an appropriate block structure.

4. Maximum likelihood estimation in TSEM requires the covariance matrix of observed data to be non-singular. However, in a reflexive measurement model with two observed variables measuring the same latent variable, this has the following drawback: If the two measurements are so perfect that they produce (almost) the same results, the covariance matrix will be (almost) singular. Thus, improving the measurement procedure can render a perfect model to be out of reach of the estimation procedure.
5. The result of TSEM depends much on the estimation method used. While maximum likelihood based estimation often turns out to give good results in simulation studies and while there are some results about equivalence (e.g. Lei and Wu 2012) this remains an issue, especially if assumptions of the estimation procedures cannot be verified easily.

6. When the assumptions of TSEM are violated, results can be misleading. The hard assumptions made in TSEM have been one reason for Wold to develop the PLS approach, according to (Esposito Vinci et al. 2017, pp. 24).

7. TSEM requires to distinguish exogenous and endogenous latent variables. While in some models this is quite natural, in others it may not be adequate. One way to characterize this is to say that TSEM is less structural equational modelling and more structural functional modelling.

8. TSEM needs a rather high number of observations to be reliable.

A lot of these points have been dealt with in the literature and there are various other approaches that address some of them. In general, one may distinguish methods that eliminate latent variables (e.g. TSEM, Generalized component analysis (Hwang and Takane 2015)) and methods that estimate latent variables (e.g. Factor score path analysis, (Devlieger and Rosseel 2017)).

The method presented in this paper addresses all of these points. Of course, as is to be expected, the method also has some issues and drawbacks, which will be discussed later.

The new method will be compared not only to TSEM and its Bayesian variant but also to the following methods (all available within R (R Core Team 2019)):

- MIIVsem. This method (Fisher et al. 2019) transforms latent to observed variables by substituting indicator and error variables. This yields a two-step approach that relaxes the assumptions made by TSEM and is rather stable as we shall see in the simulation studies below.

- Nlsem (Umbach et al. 2017) is a method that approximates nonlinear SEM by linear ones.

- semPLS (Monecke and Leisch 2012) is an implementation of the PLS (partial least squares) approach that originated from the work of Wold (1982) needs less assumptions about the data but has its own rigid structure.

- gesca (Kim et al. 2017) and cSEM (Rademaker and Schuberth 2019) are implementations of the method of Hwang and Takana (2015). This has some similarity to the present paper because it combines measurement and structural model into one set of equations that are estimated. A difference is, however, that it requires a special structure of the model that allows to eliminate the latent variables (Hwang and Takanae 2015, p. 19).

Some more methods that exist have not been taken part in the comparison studies. One is the method of Devlieger and Rosseel (2017) that applies a two-step approach to estimate values for the latent variables. This method was not included in the simulation study below because I couldn’t find an easy-to-use...
implementation. However, one important issue that the authors stress is that in TSEM a misspecification in one part of the model renders the whole model to fit badly and they consider it an advantage to proceed in parts. While this view has its legitimation, I prefer to have methods that test and fit whole theories because this is in line with epistemic holism (Quine 1951).

One quick objection to the approach taken here that calculates ordinary least squares of some equations might be that this general approach is plagued by some well-known issues. However, simulation studies indicate that these issues are not as severe as commonly thought. One trivial reason: The calculations given below show that in some situations a wrong result may simply be due to inadequate calculation precision (standard 64bit floating-point numbers are often not sufficient). Another problem usually attributed to ordinary least square methods (OLS) is that it underestimates regression coefficients when the independent variable is measured with error (e.g. Fuller 1987, formula (1.1.6)). However, this is in a sense just the result of inadequate model specification. The minimization of $\sum_i (y_i - k \cdot x_i)^2$ minimizes the error in $y_i$, (graphically speaking, the vertical error). If the $x_i$ are measured with error as well, a better estimate is obtained by minimizing $\sum_i (y_i - k \cdot x_i)^2 + (x_i - k^{-1} \cdot y_i)^2$.

The approach suggested in this paper does not resolve this issue in general but it gives the flexibility of modeling such situations.

2 Frameworks

This section describes traditional SEM shortly in the first subsection in order to give the basis for the comparison with the approach proposed in this paper that is described in the second subsection.

2.1 Traditional SEM: TSEM

This paragraph quickly recalls the main definitions from Bollen (1989, pp. 319) to fix notations: A TSEM model consists of measurement models for exogenous ($\xi$, this is a vector of random variables) and endogenous ($\eta$, this is a vector of random variables) latent variables in terms of observed variables x,y:

$$ x = A_\xi \xi + \delta, y = A_\eta \eta + \epsilon. $$

The structural model is given by the linear relation between the latent variables and it involves another error term $\zeta$:

$$ \eta = B \eta + \Gamma \xi + \zeta. $$

The latent variables are to be eliminated, only their covariances remain as variables to be estimated. To achieve this, one needs the equation $\eta = (1 - B)^{-1} (\Gamma \xi + \zeta)$ and hence one must have that $1 - B$ is non-singular.

The model is specified by fixing certain entries in the structural matrices $A_\xi, A_\eta, B, \Gamma$ to be zero (or maybe some other fixed number, mostly 1) while the other entries are parameters that are determined when estimating the model.
Further independency assumptions are needed (some of them can be relaxed somewhat): Usually, it is assumed that all components of error vectors are independent of each other and moreover
\[
\text{cov}(\delta, \epsilon) = \text{cov}(\xi, \delta) = \text{cov}(\xi, \epsilon) = \text{cov}(\zeta, \delta) = \text{cov}(\zeta, \epsilon) = 0.
\]
Under these assumptions the parameter implied covariance matrix can be calculated:
\[
\Sigma = \begin{pmatrix}
\text{cov}(x, x') & \text{cov}(x, y') \\
\text{cov}(x, y') & \text{cov}(y, y') 
\end{pmatrix}
\]
(1)

As details are given in (Bollen 1989, pp. 323), I will just repeat the calculation for the first entry to recall the role of the independency assumptions stated above. The calculation of \( \text{cov}(x, x') = E(xx') \) is
\[
E((A_x \xi + \delta) (\xi' A_x' + \delta')) = A_x E(\xi \xi') A_x' + A_x E(\delta \delta') A_x' + E(\delta \xi') A_x' + E(\delta' \xi). 
\]
(2)

Here, the second and third term are zero and the last one is diagonal, both because of the independency assumptions made above. Parameters that must be estimated are the non-fixed entries of \( A_x \) and the covariance matrix of the exogenous latent variables. Similar arguments are to be carried out for the rest of \( \Sigma \).

Finally, the estimation procedure fits the parameters of \( \Sigma \) in such a way that this matrix resembles the observed covariance matrix \( S \) of the manifest data \( x, y \) as closely as possible, where closeness is measured by an objective function, namely for unweighted least square estimation \( F_{ULS} = \frac{1}{2} \text{tr}((S - \Sigma)^2) \) and for maximum likelihood estimation \( F_{ML} = \text{tr}(S \Sigma^{-1}) + \log |\Sigma| - \log |S| - p \), where \( p \) is the number of observed variables.

The independency assumptions above can be relaxed somewhat by allowing some covariance to be nonzero. These covariances are then additional parameters and yield additional terms in the calculation of \( \text{cov}(x, x'), \text{cov}(x, y'), \text{cov}(y, y') \). However, to keep the model identified, a necessary condition is that the number of parameters does not exceed the number of independent entries in the \( p \times p \) covariance matrix which is \( \frac{p(p+1)}{2} \). Allowing all manifest variables to have correlating error variables would give \( \frac{p(p-1)}{2} \) parameters so that only the difference \( p \) parameters are left. However, each manifest variable usually need a weight parameter to be estimated so that the system would not be identified.

2.2 The theory of case based least square SEM (CLSSEM)

The data that is analysed by the methods describes in the paper consist of a numeric matrix \( (A_{i,j}) \), \( i \in \{1, \ldots, n\} \), \( j \in \{1, \ldots, k\} \), \( A_{i,j} \in \mathbb{R} \) of \( n \) cases for which \( k \) measurements have been performed. These measurements are denoted by manifest variables \( x_1, \ldots, x_k \). Hence, the column vector \( A_{.,j} \in \mathbb{R}^n \) contains the \( n \)
measurements of $x_j$. The equational model that is to be fitted to this data consists of a set of $m$ equations $g_l(\{x_j\}, \{z_q\}, \{p_s\}) = 0, l \in \{1, \ldots, m\}$ that relate the measured variables with latent variables $z_q, q \in \{1, \ldots, Q\}$ and parameters $p_s \in \mathbb{R}, s \in \{1, \ldots, S\}$. Latent variables are thought to depend on the case just like the observed variables, i.e. for each case there is a hypothetical value $Z_{i,q} \in \mathbb{R}$ while the parameters are case independent properties of the model. The equations may be linear or nonlinear and may also include inequalities (note that the inequality $h \geq 0$ is equivalent to $f(h) = 0$ for the function $f(x) = x - |x|$).

The instantiation of an equation $g_l(\{x_j\}, \{z_q\}, \{p_s\}) = 0$ for case $i \in \{1, \ldots, n\}$ is the equation $g_l(\{A_{i,j}\}, \{Z_{i,q}\}, \{p_s\}) = 0$ and due to measurement errors and due to possible differences between the model and the “true” structure, these equations are expected to hold only with an error, i.e.

$$g_l(\{A_{i,j}\}, \{Z_{i,q}\}, \{p_s\}) = \epsilon_{i,l}$$

where the error vector variables $\epsilon_l$ are required to have mean 0 and as small variance as possible. The aim of estimating the model is to determine the $n \cdot Q$ numbers $Z_{i,q}$ and the $S$ parameters $p_s$ such that the errors are as small as possible. Hence, following the least square approximation idea, the goal is to minimize the objective function

$$F_1(\{Z_{i,q}\}, \{p_s\}) := \sum_{i=1}^{n} \sum_{l=1}^{m} \epsilon_{i,l}^2 = \sum_{i=1}^{n} \sum_{l=1}^{m} (g_l(\{A_{i,j}\}, \{Z_{i,q}\}, \{p_s\}))^2$$

subject to the constraints $\forall l \in \{1, \ldots, m\} : \sum_{i=1}^{n} \epsilon_{i,k} = 0$.

Additional constraints may come from models that prescribe the variance of latent variables to, e.g., 1. If one has reason to assume the $\epsilon_l$ to be independent and normally distributed, then, of course, one can perform a $\chi^2$ hypothesis test of model fit, but this is not necessary to carry out the fitting procedure.

If one assumes the $\epsilon_{i,k}$ to be distributed according to $N(0, \sigma^2)$ then the minimizer of $F_1$, i.e. a least square estimate is also a maximum likelihood estimation by standard arguments (e.g. almost literally the same calculation as in Seber and Wild, p. 32).

Note that general calculus principles guarantee the existence of a minimizer $\{\hat{Z}_{i,q}\}, \{\hat{p}_s\}$, i.e. values such that the function value of the objective function equals the infimum over the total parameter space $F_1(\{\hat{Z}_{i,q}\}, \{\hat{p}_s\}) = \inf F_1(\{Z_{i,q}\}, \{p_s\})$ if the parameter space is compact and all the $g_l$ are continuous. If there is a reasonable argument that the absolute values of latent variables and parameters are bounded by some (maybe very large) number, compactness is given. Hence, existence of a minimizer is guaranteed in most situations. Uniqueness, however, is not guaranteed. If the functions $g_l$ are twice differentiable, it can be checked (Hesse matrix) if the minimizer is locally unique. However, it is a question of the optimization method if other minimizers may be found. When applying the method, one thus should consider global methods as well. In the Mathematica implementation one can
freely choose between the most common algorithms, e.g. simulated annealing or random search.

Yet another point is that of consistency. It seems that there are no results on this for problems of this form. Regarding maximum likelihood estimation of nonlinear models, Philipps (1982) has shown that the wide-spread belief that non-normality of errors leads to non-consistency is false. In this paper, the use of least square is not derived from a maximum likelihood approach but set in axiomatic fashion: The parameters identified are optimal in the sense that they minimize this objective function. But if one makes some distribution assumptions, in general, consistency cannot be assumed. Ivanov (1997, chapter 1) gives a number of examples where nonlinear least square estimators are not consistent. However, if one assumes the $g_l$ to be differentiable and that the measurement error of the manifest variables $x_j$ are deviations from the true values $\tilde{x}_j$ by some error $\delta_j$ with expectation value 0, then

$$F = \sum g(x_j, Z, p)^2 = \sum g(\tilde{x}_j + \delta_j, Z, p)^2 \approx \sum g(\tilde{x}_j, Z, p)^2 + \sum \left( g(\tilde{x}_j, Z, p) + \frac{\partial g(\tilde{x}_j, Z, p)}{\partial x_j} \delta_j \right)^2 = \sum g(\tilde{x}_j, Z, p)^2 + R$$

(4)

where the rest $R$ has expectation 0 so that one can assume that with growing sample size one should find minimizers that hold for the true model (as long as the linear approximation made is adequate).

The objective function $F_1$ realizes a uniform least square approach. Experiments suggested to consider weighted versions as well as:

$$F_2 (\{Z_{i,q}\}, \{p_s\}) := n \sum_{i=1}^{n} \sum_{l=1}^{m} w_l \cdot \epsilon_{i,l}^2 = n \sum_{i=1}^{n} \sum_{l=1}^{m} w_l \cdot (g_l (\{A_{i,j}\}, \{Z_{i,q}\}, \{p_s\}))^2$$

(5)

There is a somewhat natural choice for the weight factor: Assuming that the measured data is of good quality, one may assume that equations that relate just one latent variable to the manifest data are important while equations that relate many latent variables are of more hypothetical nature and thus should not be so important. This motivates the choice $w_l = \frac{1}{n L_l}$ where $L_l$ is the number of latent variables in equation $g_l$. I call this latent weighted CLSSEM.

In cases where the model equations itself don’t identify all parameters, it may be adequate to extend the objective functions by further requirements, e.g. it may be sensible to demand that the sum of squares of parameters to be minimized as well, because among different models that reproduce the measured values with similar precision one may favor the model with smaller absolute values of parameters. A situation that asks for this is given in the examples below.
Issues of the practical implementation of this approach in the Mathematica computer algebra system are given in (Oldenburg 2019).

The structural equations that TSEM starts with are a special case of CLSSEM. The model equations $x = A_x\xi + \delta, y = A_y\eta + \epsilon, \eta = B\eta + \Gamma\xi + \zeta$ yield the following objective function in vector notation (i.e. $x$ is a vector of vectors etc):

$$F_1(\eta, \xi, A_x, A_y, B, \Gamma) = (x - A_x\xi)^2 + (y - A_y\eta)^2 + (B - 1 + \Gamma\xi)^2 =$$

$$xx' - x\xi' A_x' - A_x\xi'\xi + A_x\xi' A_x' + yy' - y\eta' A_y' - A_y\eta'\eta + A_y\eta' A_y' +$$

$$BB' - B + B\xi' \Gamma' - B' + 1 - \xi' \Gamma' + \Gamma\xi B' - \Gamma\xi + \Gamma\xi' \Gamma'$$

(6)

It is rather obvious that it is not possible to generate closed form solutions for the system that states that the gradient of this function is zero. Thus, numerical methods must be applied. However, it can be seen from this expression, that there is no need for $B - 1$ to be non-singular as it is in TSEM.

2.3 Fitting measures for CLSSEM

An obvious measure for model fit is the mean of residuals, i.e. $T := \sqrt{\frac{F_{\text{min}}}{n \cdot m}}$. This allows to compare different models if their equations have been defined in a comparable manner. However, it is not possible to perform a hypothesis test because the distribution of $F_{\text{min}}$ is not known.

What can be done, however, is to compare the fit to the fit of the null data, i.e. one randomly permutes the $n$ entries of each of the $k$ data vectors. The resulting minimum of the objective function can then be compared to the one from the model-data fit.

2.4 A comment on Identification

TSEM requires a large number of cases to be estimated reliably. For the CLSSEM optimization problem to be sensible, it is sufficient that partial derivatives vanish only on a discrete set. Consider the rather trivial measurement model for one latent variable $L$ measured by two observed variables $x_1 = 1 \cdot L, x_2 = c \cdot L$ with $n$ observations made. Then: $F_1(c, Z_1, \ldots, Z_n) := \sum_{i=1}^{n} (x_{1i} - L_i)^2 + (x_{2i} - c \cdot L_i)^2$ and the partial derivatives are:

$$\frac{\partial F}{\partial c} = -2 \sum_{i=1}^{n} (x_{2i} - c \cdot L_i) \cdot L_i, \quad \frac{\partial F}{\partial L_j} = -2 (x_{1j} - L_j) - 2c(x_{2j} - c \cdot L_j)$$

(7)

Equating these derivatives to zero gives a system that can be solved exactly for any $n$ and the solution shows that there are exactly two critical points, a minimum and a maximum. Thus, this model is identified in CLSSEM, but not
in TSEM. In general, as the simulations below will show, this method works well with medium sized data sets.

3 Case studies

This paragraph compares the performance of both frameworks on real and simulated data.

3.1 Bollen’s democracy data set

An by now classic example of a non-trivial SEM is Bollen’s model of democracy and industrialization (Bollen 1989, pp. 332). This example is also included in the documentation of lavaan (Rosseel 2019) and the data were exported from there and imported into Mathematica to perform CLSSEM.

The model has three latent Variables $ind_{60}, dem_{60}, dem_{65}$ and eleven manifest variables $x_1, \ldots, x_3, y_1, \ldots, y_8$. The model equations are (omitting error variables but including intercept variables):

\[
x_1 = 1 \cdot ind_{60} + t_1, x_2 = c_2 \cdot ind_{60} + t_2, x_3 = c_3 \cdot ind_{60} + t_3, y_1 = 1 \cdot dem_{60} + s_1, y_2 = d_2 \cdot dem_{60} + s_2, y_3 = d_3 \cdot dem_{60} + s_3, y_4 = d_4 \cdot dem_{60} + s_4, y_5 = 1 \cdot dem_{65} + s_5, y_6 = d_6 \cdot dem_{65} + s_6, y_7 = d_7 \cdot dem_{65} + s_7, y_8 = d_8 \cdot dem_{65} + s_8
\]

\[
dem_{60} = b_1 \cdot ind_{60}, dem_{65} = b_2 \cdot ind_{60} + b_3 \cdot dem_{60}
\]

Furthermore, Bollen assumed that (only) the following residual covariances may differ from 0:

\[
\epsilon(y_1) \sim \epsilon(y_5), \epsilon(y_2) \sim \epsilon(y_4), \epsilon(y_3) \sim \epsilon(y_7), \epsilon(y_4) \sim \epsilon(y_8), \epsilon(y_6) \sim \epsilon(y_8)
\]

The following table shows an overview of estimating this model with various methods. Besides TSEM with estimators ULS and ML, I also include the Bayesian estimation calculated with the R-packages blavaan and the results delivered by nlsem, MIIVsem, semPLS and gesca.

These results can be summarized by saying that the results differ all over, but that CLSSEM weighted gives results that differ no more from TSEM as different methods within TSEM differ and less than other alternative methods differ. CLSSEM unweighted, however, differs a bit more for TSEM. However, in this case, there is no clear answer about what the “true” values are. This motivates the simulation studies given below.

An important feature of TSEM is that it allows to judge model fit by a number of fit indices and thus enable model comparison. For CLSSEM, there is up to now only one candidate, the quantity $T$ defined above, that is the square root of the mean residuum of all equations. Consider an alternative model of Bollen’s data set that results from omitting the observed variable $x_3$. This alternative model does not fit the data as well as the original one. In TSEM it brings up RMSEA from 0.035 to 0.060 and in CLSSEM it increases
Table 1 Comparison of TSEM and CLSSEM (CS) on Bollen’s model

<table>
<thead>
<tr>
<th>Var.</th>
<th>lavaan</th>
<th>lavaan</th>
<th>blavaan</th>
<th>MIIV</th>
<th>nlsem</th>
<th>semPLS</th>
<th>gesca</th>
<th>CS</th>
<th>CS weighted</th>
</tr>
</thead>
<tbody>
<tr>
<td>b1</td>
<td>1.48</td>
<td>1.35</td>
<td>1.46</td>
<td>1.26</td>
<td>0.37</td>
<td>0.38</td>
<td>1.00</td>
<td>1.22</td>
<td></td>
</tr>
<tr>
<td>b2</td>
<td>0.57</td>
<td>0.43</td>
<td>0.52</td>
<td>1.12</td>
<td>0.82</td>
<td>0.18</td>
<td>0.17</td>
<td>0.58</td>
<td></td>
</tr>
<tr>
<td>b3</td>
<td>0.84</td>
<td>0.84</td>
<td>0.88</td>
<td>0.72</td>
<td>-0.26</td>
<td>0.74</td>
<td>0.75</td>
<td>1.05</td>
<td></td>
</tr>
<tr>
<td>c2</td>
<td>2.18</td>
<td>2.06</td>
<td>2.28</td>
<td>2.08</td>
<td>0.97</td>
<td>1.02</td>
<td>1.02</td>
<td>2.19</td>
<td></td>
</tr>
<tr>
<td>c3</td>
<td>1.82</td>
<td>1.64</td>
<td>1.86</td>
<td>1.75</td>
<td>0.94</td>
<td>0.97</td>
<td>0.97</td>
<td>1.96</td>
<td></td>
</tr>
<tr>
<td>d2</td>
<td>1.26</td>
<td>1.24</td>
<td>1.32</td>
<td>1.14</td>
<td>1.01</td>
<td>0.92</td>
<td>0.91</td>
<td>2.82</td>
<td></td>
</tr>
<tr>
<td>d3</td>
<td>1.06</td>
<td>0.99</td>
<td>1.10</td>
<td>0.97</td>
<td>1.09</td>
<td>0.91</td>
<td>0.89</td>
<td>2.00</td>
<td></td>
</tr>
<tr>
<td>d4</td>
<td>1.27</td>
<td>1.30</td>
<td>1.33</td>
<td>1.21</td>
<td>1.03</td>
<td>1.02</td>
<td>1.02</td>
<td>2.45</td>
<td></td>
</tr>
<tr>
<td>d6</td>
<td>1.19</td>
<td>1.19</td>
<td>1.23</td>
<td>1.05</td>
<td>0.96</td>
<td>1.00</td>
<td>1.00</td>
<td>1.72</td>
<td></td>
</tr>
<tr>
<td>d7</td>
<td>1.28</td>
<td>1.31</td>
<td>1.32</td>
<td>1.18</td>
<td>1.04</td>
<td>1.02</td>
<td>1.03</td>
<td>1.67</td>
<td></td>
</tr>
<tr>
<td>d8</td>
<td>1.27</td>
<td>1.30</td>
<td>1.31</td>
<td>1.20</td>
<td>1.10</td>
<td>1.06</td>
<td>1.06</td>
<td>1.71</td>
<td></td>
</tr>
</tbody>
</table>

T slightly from 1.38 to 1.42. This shows that CLSSEM also provides a tool to assess model fit but this seems to be less sensitive. Moreover, there is no established theory to give guidelines of acceptable model fit. In interpreting the absolute value of T of approx. 1.4 one should bear in mind that we work with non-normalized data and that the range of the manifest variables span the interval [0,10]. Thus, an absolute mean error of 1.4 is far below standard deviations of the variables.

3.2 Simulated data sets

The main drawback of this well-known model is that it is not known what should be considered to be the “true” values of the parameters that a perfect SEM estimation procedure should yield. To remedy this situation, I used simulated data.

The simulated data were intended for Bollen’s model as well. In the first set of simulations, the aim was to produce data that fit the model very well and should be well suited for TSEM in the sense that it fulfills all requirements of TSEM, especially independence of error variables. In the description of the algorithm, the notion $N(m, \sigma, n)$ is used to denote a vector in $\mathbb{R}^n$ with normally distributed, independent random numbers with given mean $m$ and standard deviation $\sigma$. The algorithm produces

$$X_1, X_2, X_3, Y_1, Y_2, \ldots, Y_8, ind60, dem60, dem65 \in \mathbb{R}^n$$

from the following input:

$$n \in \mathbb{N}, b_1, b_2, b_3, c_2, d_2, \ldots, d_4, d_6, \ldots, d_8, \sigma \in \mathbb{R}$$

Algorithm 1

1. $\text{ind60} := N(0, 1, n)$
Table 2 Comparison of TSEM and CLSSEM on simulated data for Bollen’s model

<table>
<thead>
<tr>
<th>Var</th>
<th>true value</th>
<th>Lavaan ML</th>
<th>Lavaan ULS</th>
<th>CLSSEM</th>
<th>CLSSEM weighted</th>
</tr>
</thead>
<tbody>
<tr>
<td>b1</td>
<td>0.8</td>
<td>0.66</td>
<td>0.67</td>
<td>0.60</td>
<td>0.66</td>
</tr>
<tr>
<td>b2</td>
<td>0.5</td>
<td>0.51</td>
<td>0.51</td>
<td>0.50</td>
<td>0.51</td>
</tr>
<tr>
<td>b3</td>
<td>1.2</td>
<td>1.19</td>
<td>1.18</td>
<td>1.68</td>
<td>1.19</td>
</tr>
<tr>
<td>c2</td>
<td>0.5</td>
<td>0.49</td>
<td>0.50</td>
<td>0.49</td>
<td>0.49</td>
</tr>
<tr>
<td>c3</td>
<td>0.8</td>
<td>0.79</td>
<td>0.80</td>
<td>0.79</td>
<td>0.79</td>
</tr>
<tr>
<td>d2</td>
<td>0.3</td>
<td>0.29</td>
<td>0.30</td>
<td>0.44</td>
<td>0.30</td>
</tr>
<tr>
<td>d3</td>
<td>0.9</td>
<td>0.90</td>
<td>0.91</td>
<td>1.33</td>
<td>0.91</td>
</tr>
<tr>
<td>d4</td>
<td>1.7</td>
<td>1.70</td>
<td>1.70</td>
<td>2.51</td>
<td>1.71</td>
</tr>
<tr>
<td>d6</td>
<td>0.6</td>
<td>0.60</td>
<td>0.60</td>
<td>0.59</td>
<td>0.60</td>
</tr>
<tr>
<td>d7</td>
<td>0.4</td>
<td>0.40</td>
<td>0.40</td>
<td>0.40</td>
<td>0.40</td>
</tr>
<tr>
<td>d8</td>
<td>1.3</td>
<td>1.29</td>
<td>1.30</td>
<td>1.30</td>
<td>1.29</td>
</tr>
</tbody>
</table>

2. \( X_1 := 1.0 \cdot \text{ind60} + N(0, \sigma, n) ; X_2 := c_2 \cdot \text{ind60} + N(0, \sigma, n) ; \)
   \( X_3 := c_3 \cdot \text{ind60} + N(0, \sigma, n) ; \)

3. \( \text{dem60} := b_1 \cdot \text{ind60} + N(0, \sigma, n) \)
4. \( Y_1 := 1.0 \cdot \text{dem60} + N(0, \sigma, n) ; Y_2 := d_2 \cdot \text{dem60} + N(0, \sigma, n) ; \)
   \( Y_3 := d_3 \cdot \text{dem60} + N(0, \sigma, n) ; Y_4 := d_4 \cdot \text{dem60} + N(0, \sigma, n) ; \)
5. \( \text{dem65} := b_1 \cdot \text{ind60} + b_3 \cdot \text{dem60} + N(0, \sigma, n) \)
6. \( Y_5 := 1.0 \cdot \text{dem60} + N(0, \sigma, n) ; Y_6 := d_6 \cdot \text{dem60} + N(0, \sigma, n) ; \)
   \( Y_7 := d_7 \cdot \text{dem60} + N(0, \sigma, n) ; Y_8 := d_8 \cdot \text{dem60} + N(0, \sigma, n) ; \)

The parameters where chosen as in the column “true value” indicated below and furthermore \( n = 100, \sigma = 0.1 \). Then a run of all programs gives the results in the following table. All TSEM results are calculated with R 3.6.1 and the development version of lavaan (October 2019). R has been called from within Mathematica in order to avoid data mismatch.

However, repeating the simulation revealed that the results depend on the simulated data to an extent that cannot be neglected. Thus, for each value of \( n, \sigma \) ten data sets have been simulated. In each case the Euclidean distance between the vector \((b_1, b_2, b_3, c_2, c_3, d_2, \ldots, d_4, d_6, \ldots, d_8) \in \mathbb{R}^{11}\) given by the true and by the estimated value has been calculated. Mean (in the cells in the table the upper entry) and standard deviations (the lower entry in the cells) of the 10 runs are recorded in the following table with the best value on each simulation typeset in bold (table 3).

Investigating these numbers one first notices that CLSSEM unweighted may deliver very bad estimations especially when the standard deviation of random noise is very small. Inspection shows that this is due to misestimations of \( b_2, b_3 \) while all other estimates are pretty good. This fact is rather easy to explain: When \( \sigma \) is small, then \( \text{dem60, ind60} \) correlate highly with each other, even more, they are almost identical (as they have almost the same mean and variance, too). Hence, the transformation \((b_2, b_3) \mapsto (b_2 + z, b_3 - z)\) for any \( z \in \mathbb{R} \) will leave the value of the objective function almost unchanged. Hence, the minimization algorithm is likely to hunt along this path leaving finally with very large absolute values of these two parameters unchanged (that this
Table 3 Comparison of TSEM and CLSSEM (CS) on repeated simulations (mean stacked over sd), best mean values in bold

<table>
<thead>
<tr>
<th>n</th>
<th>σ</th>
<th>TSEM ML</th>
<th>TSEM ULS</th>
<th>MIIV cSEM</th>
<th>cSEM cPLS</th>
<th>cSEM GSCA</th>
<th>CS unw. weighted</th>
<th>CS para min</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>0.02</td>
<td>0.141</td>
<td>0.516</td>
<td>0.233</td>
<td>0.229</td>
<td>0.232</td>
<td>0.867</td>
<td><strong>0.107</strong></td>
</tr>
<tr>
<td></td>
<td>0.1</td>
<td>0.164</td>
<td>0.235</td>
<td>0.194</td>
<td>0.206</td>
<td>0.252</td>
<td>0.875</td>
<td>0.089</td>
</tr>
<tr>
<td></td>
<td>0.5</td>
<td>0.195</td>
<td>0.419</td>
<td>0.172</td>
<td>0.221</td>
<td>0.216</td>
<td>1.066</td>
<td><strong>0.047</strong></td>
</tr>
<tr>
<td>200</td>
<td>0.02</td>
<td>0.096</td>
<td>0.239</td>
<td>0.089</td>
<td>0.028</td>
<td>0.044</td>
<td>0.542</td>
<td>0.031</td>
</tr>
<tr>
<td></td>
<td>0.1</td>
<td>0.133</td>
<td>0.218</td>
<td>0.362</td>
<td>0.346</td>
<td>0.316</td>
<td>3.803</td>
<td><strong>0.107</strong></td>
</tr>
<tr>
<td></td>
<td>0.5</td>
<td>0.153</td>
<td>0.150</td>
<td>0.267</td>
<td>0.243</td>
<td>0.216</td>
<td>4.300</td>
<td>0.050</td>
</tr>
<tr>
<td>500</td>
<td>0.02</td>
<td>0.112</td>
<td>0.117</td>
<td>0.133</td>
<td>0.051</td>
<td>0.055</td>
<td>1.141</td>
<td>0.066</td>
</tr>
<tr>
<td></td>
<td>0.1</td>
<td>0.108</td>
<td>0.104</td>
<td>0.231</td>
<td>0.229</td>
<td>0.226</td>
<td>1.064</td>
<td><strong>0.099</strong></td>
</tr>
<tr>
<td></td>
<td>0.5</td>
<td>0.308</td>
<td>0.086</td>
<td>0.179</td>
<td>0.067</td>
<td>0.062</td>
<td>0.876</td>
<td>0.063</td>
</tr>
<tr>
<td>1000</td>
<td>0.5</td>
<td>0.144</td>
<td>0.135</td>
<td>0.498</td>
<td>0.086</td>
<td>0.070</td>
<td>4.196</td>
<td>0.094</td>
</tr>
<tr>
<td>2000</td>
<td>0.5</td>
<td>0.181</td>
<td>0.188</td>
<td>0.282</td>
<td>0.286</td>
<td>0.281</td>
<td>9.400</td>
<td><strong>0.177</strong></td>
</tr>
<tr>
<td>5000</td>
<td>0.5</td>
<td>0.069</td>
<td>0.068</td>
<td>0.232</td>
<td>0.039</td>
<td>0.028</td>
<td>0.788</td>
<td>0.060</td>
</tr>
<tr>
<td>100</td>
<td>1.5</td>
<td>0.193</td>
<td>0.196</td>
<td>0.180</td>
<td>0.258</td>
<td>0.258</td>
<td>0.826</td>
<td>0.213</td>
</tr>
<tr>
<td>200</td>
<td>1.5</td>
<td>0.198</td>
<td>0.195</td>
<td>0.160</td>
<td>0.119</td>
<td>0.237</td>
<td>8.530</td>
<td>0.209</td>
</tr>
<tr>
<td>500</td>
<td>1.5</td>
<td>0.199</td>
<td>0.176</td>
<td>0.111</td>
<td>0.186</td>
<td>0.135</td>
<td>0.310</td>
<td>0.083</td>
</tr>
<tr>
<td>1000</td>
<td>1.5</td>
<td>0.393</td>
<td>0.394</td>
<td>0.482</td>
<td>0.376</td>
<td>0.358</td>
<td>1.755</td>
<td>0.651</td>
</tr>
<tr>
<td>2000</td>
<td>1.5</td>
<td>0.145</td>
<td>0.145</td>
<td>0.173</td>
<td>0.039</td>
<td>0.045</td>
<td>0.327</td>
<td>0.086</td>
</tr>
</tbody>
</table>

Explanation is correct can be seen e.g. by replacing σ by 3σ in step 3 of the above simulation algorithm and observing that then unweighted CLSSEM is approximately as good as MIIVsem and especially it shown no extreme misestimations). The weighted version avoids this problem and delivers very good estimates, except in cases where the variance of the noise is very large. Besides using CLSSEM weighted there is another trick to cope with this problem: Just add \( b_1^2 + b_2^2 + b_3^2 + \ldots \) to the objective function (an option in the Mathematica procedure allows this to be done automatically). The rationale behind this trick is the argument that among almost equivalently fitting parameter sets the best one is the one that can explain the data with the smallest parameters possible. The results are contained in the last column of the table above.

Despite recovering path weights from simulated data, another issue is how accurate the methods reproduce the latent variables that were used in the simulation. To measure this, the average absolute difference between estimated and “real” (i.e. simulated) latent variable values have been recorded. The results are contained in table 4. It shows that CLSSEM performs quite well when compared to other methods.

Now, the situation may look quite different when there are substantial correlations of error variables. It has been discussed in (Henseler et al. 2014) that TSEM applications are typically based on a common factor structure.
Table 4 Mean absolute errors of latent variable estimation (estimated over 10 runs, for blavaan only one run) (best values in bold)

<table>
<thead>
<tr>
<th>n</th>
<th>σ</th>
<th>Blavaan</th>
<th>cSEM PLS</th>
<th>cSEM GSCA</th>
<th>CLSSEM unw.</th>
<th>CLSSEM weighted</th>
<th>CLSSEM parmin</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>0.1</td>
<td>1.294</td>
<td>0.261</td>
<td>0.260</td>
<td><strong>0.150</strong></td>
<td>0.151</td>
<td><strong>0.150</strong></td>
</tr>
<tr>
<td>200</td>
<td>0.1</td>
<td>1.392</td>
<td>0.215</td>
<td>0.213</td>
<td><strong>0.083</strong></td>
<td>0.086</td>
<td><strong>0.083</strong></td>
</tr>
<tr>
<td>500</td>
<td>0.1</td>
<td>1.364</td>
<td>0.220</td>
<td>0.218</td>
<td>0.079</td>
<td>0.076</td>
<td><strong>0.072</strong></td>
</tr>
</tbody>
</table>

with its assumptions of independence of error variables of manifest variables and that this assumption is inadequate in many situations, including many real studies. My own limited experiences underpin this judgement. Moreover, I’m concerned with the impact of correlations between error variables of latent variables. To test the reliability of algorithms in a situation where all these correlations occur, I invented the following simulation. To describe one of the algorithms used, the following additional notation is needed: \( \text{runif}(a, b) \) is an equally distributed random number from the real interval \([a, b]\), \( \text{rnorm}(\sigma) \) is a random number, normally distributed with mean 0 and standard deviation \( \sigma \), \( \text{vector}(f(i), i, 1, n) \) is the vector of \( n \) components that has \( f(i) \) as its \( i \)-th component.

Algorithm 2

1. \( Z_1 := N(0, \sigma, n) \); \( Z_0 := \text{vector}\left(\frac{3i \cdot \text{rnorm}(\sigma)}{n}, i, 1, n\right) \);
2. \( \text{ind60} := \text{vector}(\text{runif}(\frac{b_1}{n}, \frac{b_2}{n}), i, 1, n) \)
3. \( X_1 := 1.0 \cdot \text{ind60} + N(0, \sigma, n) \); \( X_2 := c_2 \cdot \text{ind60} + N(0, \sigma, n) + Z_1 \);
   \( X_3 := \text{vector}(c_3 \cdot \text{ind60}, + \frac{3i}{n} \cdot \text{rnorm}(\sigma), i, 1, n) \);
4. \( \text{dem60} := \text{vector}(b_1 \cdot \text{ind60} - \frac{Z_1}{2}, i, 1, n) \)
5. \( Y_1 := 1.0 \cdot \text{dem60} + N(0, \sigma, n) \); \( Y_2 := d_2 \cdot \text{dem60} + N(0, \sigma, n) + Z_1 \);
6. \( Y_3 := d_3 \cdot \text{dem60} + N(0, \sigma, n) + \frac{Z_1}{2}; \)
7. \( Y_4 := d_4 \cdot \text{dem60} + N(0, \sigma, n) + \text{rnorm}(\sigma) + Z_1 \);
8. \( Y_5 := 1.0 \cdot \text{dem60} + N(0, \sigma, n) \); \( Y_6 := d_6 \cdot \text{dem60} + N(0, \sigma, n) + Z_0; \)
9. \( Y_7 := d_7 \cdot \text{dem60} + N(0, \sigma, n) \); \( Y_8 := d_8 \cdot \text{dem60} + N(0, \sigma, n) \);

The data sets generated that way still have the same underlying linear model as in the above simulations but independency assumptions of TSEM are violated in multiple ways. One should thus assume, that TSEM will not perform well. The table of results (table 5) shows this. (Note: For small variations CLSSEM weighted may suffer from numerical problems. Thus, the cases with \( \sigma = 0.02 \) were run with higher precision (30 decimal places) to avoid instability).

Obviously, with such massive violations of independency assumptions, TSEM estimations are useless while all three versions of CLSSEM deliver useful results in most cases. An exception is the weighted version which gives most of
Table 5 Comparison of TSEM and CLSSEM on repeated simulations with violation of independency assumptions according to algorithm 2 (best mean values in bold)

<table>
<thead>
<tr>
<th>n</th>
<th>σ</th>
<th>lavaan ML</th>
<th>lavaan ULS</th>
<th>cSEM cPLS</th>
<th>cSEM GSCA</th>
<th>CLSSEM unweighted</th>
<th>CLSSEM weighted</th>
<th>CLSSEM paramin</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>0.02</td>
<td>57.85</td>
<td>0.755</td>
<td>1.078</td>
<td>2.122</td>
<td>0.376</td>
<td><strong>0.121</strong></td>
<td>0.132</td>
</tr>
<tr>
<td>200</td>
<td>0.02</td>
<td>112.0</td>
<td>0.799</td>
<td>0.901</td>
<td>3.475</td>
<td>0.276</td>
<td>0.096</td>
<td>0.005</td>
</tr>
<tr>
<td>500</td>
<td>0.02</td>
<td>27.25</td>
<td>0.434</td>
<td>0.487</td>
<td>0.623</td>
<td>0.397</td>
<td>0.050</td>
<td>0.002</td>
</tr>
<tr>
<td>100</td>
<td>0.1</td>
<td>4.292</td>
<td>60.99</td>
<td>1.488</td>
<td>11.36</td>
<td>0.747</td>
<td>0.165</td>
<td><strong>0.139</strong></td>
</tr>
<tr>
<td>200</td>
<td>0.1</td>
<td>5.224</td>
<td>132.4</td>
<td>2.020</td>
<td>31.74</td>
<td>0.662</td>
<td>0.129</td>
<td>0.032</td>
</tr>
<tr>
<td>500</td>
<td>0.1</td>
<td>35.85</td>
<td>131.7</td>
<td>1.434</td>
<td>1.092</td>
<td>0.299</td>
<td><strong>0.084</strong></td>
<td>0.140</td>
</tr>
<tr>
<td>100</td>
<td>0.5</td>
<td>97.80</td>
<td>227.4</td>
<td>0.277</td>
<td>1.715</td>
<td>0.271</td>
<td>0.054</td>
<td>0.022</td>
</tr>
<tr>
<td>200</td>
<td>0.5</td>
<td>3.254</td>
<td>1.471</td>
<td>1.057</td>
<td>0.893</td>
<td>0.294</td>
<td><strong>0.081</strong></td>
<td>0.152</td>
</tr>
<tr>
<td>500</td>
<td>0.5</td>
<td>2.063</td>
<td>0.989</td>
<td>1.278</td>
<td>1.015</td>
<td>0.211</td>
<td>0.045</td>
<td>0.045</td>
</tr>
<tr>
<td>100</td>
<td>1.5</td>
<td>50.51</td>
<td>122.2</td>
<td>0.961</td>
<td>2.563</td>
<td>0.477</td>
<td><strong>0.173</strong></td>
<td>0.287</td>
</tr>
<tr>
<td>200</td>
<td>1.5</td>
<td>104.4</td>
<td>299.0</td>
<td>0.756</td>
<td>3.423</td>
<td>0.324</td>
<td>0.069</td>
<td>0.142</td>
</tr>
<tr>
<td>500</td>
<td>1.5</td>
<td>2.520</td>
<td>209.5</td>
<td>2.221</td>
<td>4.813</td>
<td>0.189</td>
<td>0.065</td>
<td>0.116</td>
</tr>
<tr>
<td>100</td>
<td>1.5</td>
<td>2.064</td>
<td>579.0</td>
<td>0.779</td>
<td>0.908</td>
<td>0.261</td>
<td><strong>0.154</strong></td>
<td>0.229</td>
</tr>
<tr>
<td>200</td>
<td>1.5</td>
<td>1.638</td>
<td>985.3</td>
<td>0.410</td>
<td>0.757</td>
<td>0.161</td>
<td>0.058</td>
<td>0.137</td>
</tr>
<tr>
<td>500</td>
<td>1.5</td>
<td>3.881</td>
<td>4.401</td>
<td>2.080</td>
<td>1.164</td>
<td>0.683</td>
<td>0.079</td>
<td>1.171</td>
</tr>
<tr>
<td>100</td>
<td>1.5</td>
<td>5.011</td>
<td>9.738</td>
<td>2.859</td>
<td>1.057</td>
<td>0.683</td>
<td>0.068</td>
<td>0.544</td>
</tr>
<tr>
<td>200</td>
<td>1.5</td>
<td>148.1</td>
<td>1.881</td>
<td>1.682</td>
<td>0.658</td>
<td>0.244</td>
<td>1.209</td>
<td>1.374</td>
</tr>
<tr>
<td>500</td>
<td>1.5</td>
<td>460.7</td>
<td>2.762</td>
<td>2.109</td>
<td>0.465</td>
<td>0.408</td>
<td>0.356</td>
<td>0.378</td>
</tr>
<tr>
<td>100</td>
<td>1.5</td>
<td>2.241</td>
<td>1.295</td>
<td>19.29</td>
<td><strong>0.667</strong></td>
<td>1.418</td>
<td>1.134</td>
<td>1.390</td>
</tr>
<tr>
<td>200</td>
<td>1.5</td>
<td>1.272</td>
<td>1.163</td>
<td>55.49</td>
<td>0.368</td>
<td>0.230</td>
<td>0.210</td>
<td>0.222</td>
</tr>
</tbody>
</table>

the time the best estimates but may fall into a numerical trap when variances are very small.

Summing up the results from the simulation studies, one is faced with these key findings:

- When the error variables are simulated to be uncorrelated, both TSEM and CLSSEM give good results with TSEM leading for large data sets and larger standard deviations, while CLSSEM is rather good with small data sets.
- When errors variables violate the independence assumptions TSEM often gives very bad estimations while CLSSEM remains rather stable.
- Weighted CLSSEM overall shows a very good performance.
- All versions of CLSSEM show good performance in the recovery of latent variable values in the simulation studies.

4 Conclusion

This paper has presented a very simple least square approach to SEM. The results show that at least the weighted version of CLSSEM is a good option to estimate SEM. Both versions of CLSSEM, weighted and unweighted, have
the advantage to address all issues mentioned in the introduction. However, a lot of research questions are still open. The difference between weighted and unweighted CLSSEM is bigger than expected and it has to be clarified under what conditions which method is preferable. Moreover, one needs to develop reliable fit indices and techniques to allow hypothesis testing (extending the ideas sketched above). Moreover, the possibilities arising from using non-linear fit functions have to be evaluated: Especially piece-wise linear functions and implicative linkings as mentioned in the introduction will be analyzed in follow-up research. Thus, summing up, this paper provides some new insights and opens up new areas for research.

Acknowledgements Some names will be mentioned after the review process

References