

NOTES TO SEM LECTURES (in yellow: not explicitly discussed)

Harry BG Ganzeboom

Melbourne, October 25 2017

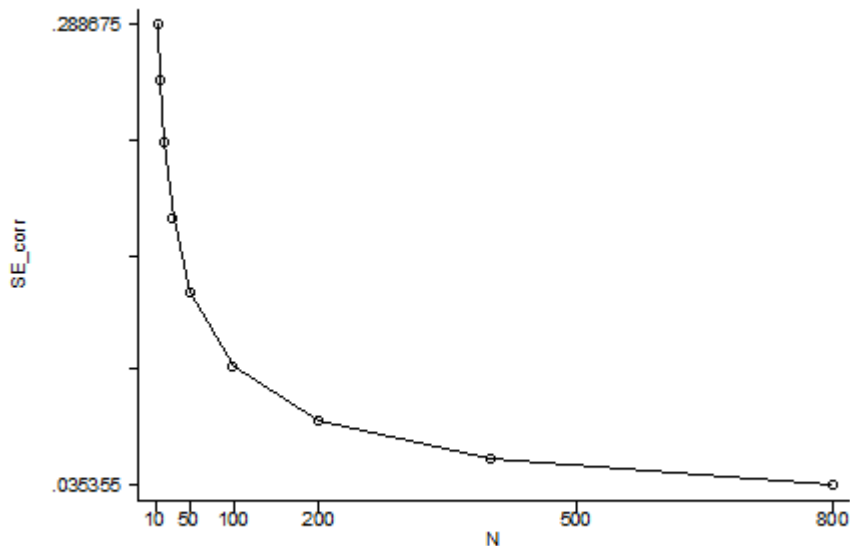
Lecture 2

Arguments why you should be doing SEM for the rest of your life (additions in red)

1. SEM makes you think about the world in terms of causality, which happens to be what science is all about.
2. SEM makes you aware that all we are observing is covariation (correlations); it is the task of a scientist to invent a causal theory / model that explains the covariation. The world that we see is a correlation matrix.
3. SEM allows the estimation of a causal process by simultaneous (or: a system of) equations; this has certain statistical advantages (to come), but it is also conceptually relevant.
4. SEM allows for an easy method to include all available data in your estimation, when there are missing values. The method is called MLMV (Maximum Likelihood with Missing Values) or FIML (Full Information Maximum Likelihood).
5. SEM makes you distinguish between the facts and your data (observations) and can estimate the relationship between the two: measurement error.
6. SEM allows you not only to diagnose measurement error, but also so to correct for it.
7. SEM makes you work on measurement models, using information on the relationships between indicators and structural variables that are not directly influencing these indicators (full information).
8. SEM makes it possible to distinguish between random and systematic measurement error, and can indeed quantify both: reliability and validity coefficients.
9. SEM makes it possible to estimate reciprocal causal effects.

A special SE

A formula dearly missing from most introductory statistics books (and many advanced ones too) is the $SE(r)$, the standard error of the Pearson correlation. It is approximately: $1/\sqrt{N}$. (In fact it is: $1/\sqrt{N-2}$ or $1/\sqrt{N-3}$ – this does not make much of a difference.) Say, if you have a sample of 100, the sampling fluctuation (the standard deviation of the sampling distribution) is $1/10=0.10$. If we have a sample of $N=25$, it is 0.20 . So in a sample of 100, a correlation of $r=0$ would with 95% confidence vary between -0.20 and $+0.20$, and sample values outside this range would be statistically significant. In a sample of 25, this range would be between -0.40 and $+0.40$. **Remember this formula.**



Caveat: the formula and CI calculations apply under the H0 – in the case the population $r=0$. It cannot be used to build confidence intervals around much stronger correlations. In those cases we need a much more complicated procedure, called Fisher Z transformation.

Notice that our sample in class is around $N=10000$. $SE(r)$ should then be $1/\sqrt{10000} = 0.01$. The SE's of our standardized effects are indeed around this value

The same formula is a rather good approximation of the $SE(\beta)$ in a simple regression, which is a quantity that arises in Stata (and Mplus?) but not in other programs. However, in multiple regression models the $SE(\beta)$ is also informed by explained variance, the spread of the X-variables and explained variance (just like $SE(B)$).

There is something surprising about these $SE(\beta)$ in Stata SEM, and for that matter for statistical testing in standardized models in Stata SEM: the results differ from the unstandardized model (compare t-values, so β/SE , with B/SE). I have not found an explanation in the Stata SEM manuals, but I think the reason must be that β -values (like r) are bounded in the $-1..1$ range, and so must have asymmetric and truncated confidence intervals. Traditionally, this is solved by a formula called Fisher's Z, which transforms correlations to a quantity $0.5 \cdot \ln((1+r)/(1-r)) = \arctan(r)$, which is more or less normally distributed and has symmetric CI, even at values of r close to -1 or $+1$.

Missing values

Missing values are one of the major practical problems in data-analysis and coping with them high on the list ten reasons why you would want to be doing SEM for the rest of your life. The remedy in SEM is extremely simple: `sem (model) , method(mlmv)`. `mlmv` stands for Maximum Likelihood with Missing Values. When you use this, the following will happen:

- The estimation of the model becomes more complicated: more iterations, more chances that the model does not converge.

- Observations with only missing data still will be excluded from the estimation. All the other observations will be used, even if they have information on only one variable and do not inform about correlations. Stata SEM will print the number of observations, but is otherwise not very informative about missing values patterns.
- The coefficients will come up in the usual way. What you would hope for is that they do not differ very much from the ones you obtain from the 'listwise' ('complete case') analysis, but that the SE's become smaller (and the associated coefficients 'more significant'), in particular for the variables that are most harmed by the listwise deletion. **These are the effects NOT for the variables with the most missing cases, but the variables with the *least* missing cases.**

It is extremely easy to do this, and also fairly easy to interpret the results, but it takes quite a bit of understanding what you are doing, why you would do this and what could go wrong.

Missing values can be treated in two principally correct ways:

- **Multiple imputation (MI).** In (single) imputation, you try to guess what the correct value would have been, given what you know about a case. One way to do this is to use the mean as imputation, which is the same as saying that you have no other information about the case that informs you about the true value. Other methods are to look for similar cases in the data (either by sorting of by using a predicting equation) and borrow a value from a similar case (nearest neighbor imputation). Modern techniques of **multiple** imputation add three steps to this procedure:
 - Do not only impute the cases as precise as you can, but also add a random component to the imputation that is similar to the random variation among the similar cases for which you have a valid observation.
 - Do this multiple (10-20) times. This leads to 10-20 different datasets that you are now going to analyze jointly.
 - Use some voodoo formulas, invented by Donald Rubin, to calculate unbiased standard errors, from the combination of these imputed files. The computer programs do this for you.
- MI Procedures are available in SPSS, Stata and many other programs. The principal choice you have to make is what variables you use to generate the best imputations. It is easy, but (computer) time consuming to use a great many, relevant and irrelevant ones. Analysis of the multiply imputed files is also somewhat time-consuming, but again: computer time consuming.
- **Maximum likelihood (ML)** estimation, like in Stata's SEM models. This procedure is based on the model that you specify (your structural and measurement model) and then tries to find best fitting parameters given all (partial) evidence that the data provide. In case this data is incomplete (missing values), the iterative procedures are fairly complex and time-consuming.
- In **m1mv** (elsewhere called FIML – Full Information Maximum Likelihood) the aim of the procedure clearly is to use the part of the data that is observed. In multiple imputation, the misunderstanding that often arises, is that you want to use data that are NOT observed – which

is not true, but the imputation procedure makes you think so. It is all about bringing back in the data that you HAVE observed.

- Although the MI and ML approaches appear to be radically different, they are asymptotically (in the long run) equivalent!! There are some practical differences:
 - In MI it is easier to take a whole lot of variables into account, including irrelevant ones. This is also possible in ML, but less practical.
 - In MI the missing values imputation is a separate step from the model building step – while the essence of ML is that this happens simultaneously. In practical work MI is often more convenient to do.
 - MI takes more time, but goes quicker.
- The background needed to operate either procedure is that you try to understand how your missing values arise in the first place. There are three possibilities:
 - MCAR (Missing Completely at Random). A simple lottery has determined the MV pattern. In this case the true correlation matrix (not observed), the pairwise correlation matrix and the listwise correlation matrix tend to be the same (although based on different N).
 - MAR (Missing at Random). A more complicated lottery has determined the MV patterns, namely a lottery that knows your true (population) model and then started to generate MV's in a random way on the predicted values from that model. In this case the true correlation matrix, the pairwise matrix and the listwise matrix are different (and the listwise matrix is biased), but the model estimated on the pairwise matrix can be correct (=equal to the population model), but we cannot know this, only hope for it.
 - MNAR (Missing Not at Random). The MVs are not even determined by lottery, but by evil golems: the true values have disappeared in 'systematic' ways, but you have no information on how this happened. This case cannot be repaired by MI or ML – you need 'selection [heckman] models' – and they do not work well if you have no information on the MV mechanism.

So, in MCAR everything that you do will work, in MAR you have to come up with the right model underlying the pattern of associations in the population.

It is always useful to consider your analysis to be an instance of MCAR. Whether data more or less fit this condition can be tested by comparing the pairwise and listwise correlation matrix. If these are the same, the parameter estimates deriving from it will be the same. But if so, you could ask yourself: "Why bother? Whatever I do, ML, MI or nothing at all, my estimated model (coefficients) will tend to be the same..." True, and that shows you why you should bother to do ML or MI: the intended effect is NOT on the coefficients (point estimates), but ON THE STANDARD ERRORS! You do all of this **to increase the power of your model**. If the coefficients change after missing values treatment, you should start worrying.

What if the listwise and pairwise correlations are different? This would show that your data are not MCAR, but they could still be either MAR or MNAR. The first thing to notice in this situation now is to understand that using the listwise correlations is by no means better or more representative than using the pairwise correlations. On the contrary, the listwise correlations will be biased. So, if you are not going to be able to apply selection models, there is little alternative than assume MAR and invent the best model you can in a MLMV mode.

There are two practical disadvantages to MLMV in Stata:

- Models take longer to converge or may not converge at all. If so, you must usually rethink your design and the missing values pattern and one good place to start would be the pairwise correlation matrix: `pwcorr _all, obs`.
- MLMV estimation in Stata has the particular inconvenience that `estat, residuals` does not work, while it is quite useful to look at these residuals, in particular when you have standardized the observed variables and you are looking at a correlation (not covariance) matrix. (In LISREL you can obtain these residuals.)

Elements of a Stata SEM model

The general formula for a Stata SEM model is:

```
sem (effect) (effect) (effect) (etc) if(condition) [weight] , var()  
covar() standardized method(mlmv) iterate(#)
```

(effect) can be one of the following:

(var) This would imply that the model fits the covariances between **var** and all other exogenous variables

(x → y) Causal effect of **x** on **y**

(y ← x) Causal effect of **x** on **y**. So the direction of the arrow does not matter.

(X → x) Measurement effect of latent variable **X** on observed variable **x**. We call these effects 'measurement', and the output groups them accordingly, but there is no principled distinction between causal and measurement effects.

(x → x1 x2 x3) Does three measurement effects at once. This is shorter, but you cannot have constraints in this form.

(effect@#) constrains the effect to be equal to **#**; very often **#=1** to fix a reference effect, or a latent variance in exogenous variables

(effect@name) constrains the effect to be equal to **name**, which becomes relevant when **name** also appears in another **(effect@name)**, in which case we obtain an equality constraint. It is common practice to use single or double characters (**a b c aa bb cc**) for this, but you can invent your own name.

var(x@1) constrains the variance of the latent variable **x** at 1 (so standardized at the latent level). This can only be applied to exogenous latent variables.

var(e.x@0) Residuals in variables can be addressed using **e.var**. This particular specification would say that there is no residual variance in **x**, which would imply that **x** is identical to its underlying latent variable **X**. This is sometime useful when you want to include observed variables in your model.

covar(e.x1*e.x2) Specifies that a covariance of residuals is to be estimated between **x1** and **x2**. These are also known as **correlated residuals**.

NB: For some mysterious reason the **var** and **covar** specification should be after the comma, as if they are a sort of options. They are not, they are parameters of the model.

if(condition) specifies a selection of the sample, e.g. **if(age > 30)** would exclude the younger cases.

[weight] applies the weight if present in the data.

Useful post-estimation commands

sem, standardized redisplay the model in standardized format

estat mindices displays the modification indices, which indicate L2 changes of you would introduce an additional parameter in the model. Unlike with residuals, the logic of modification indices is that of an influence statistic: how would the model fit change?

estat residuals displays the difference (residuals) between the fitted and the observed covariances. You obtain these in unstandardized form, even if you have standardized your model. But residual correlations are in fact easier to interpret than residual covariances, so this is one reason why I recommend to run your models also with standardized data.

estat residuals, normalized and **estat residuals, standardized** display the residuals relative to the sampling distribution – so as t-test. This is not the same as residuals on standardized data.

estat teffects shows total, direct and indirect effects, with associated SE and t-tests

estat teffect, standardized shows total, direct and indirect effects in standardized form, with associated SE and t-tests

estat gof, stat(all) will show some of the fit statistics that many SEM-people are so hung-up about. Important ones are:

- LR-chi2 test will test the significance of the difference between the observed correlation matrix and the correlation matrix. Can be used to argue that your model fits well, but also used in nested model comparisons. LR-chi2 is heavily dependent upon sample size when there is any misfit in the data.
- CFI, TTI are statistics that are not sensitive to N. Often reported, should be over 0.95

- RMSEA Root Mean Squared Error of Approximation. This is basically a function of the residuals adjusted for sample size. It comes with a CI and a test whether it is significantly different from .05. After LR-chi2 this is the most often reported fit measure.

It is interesting that Stata SEM is not overly concerned with fit statistics, unlike LISREL and other SEM programs. I think this is a good thing: when comparing models, we should first concentrate on how parameters change and are sensitive to model specification, before looking at the fit statistics.

Standardization and the standardized solution

Standardization means that variables are expressed as Z values: variance = SD = 1, and mean = 0. By standardizing variables their values become comparable, and so become effects in regression models. In factor analysis we are so used to standardization of both the latent and observed variables, that we would think it cannot be done in a different way; in fact we can, even in SPSS.

In SEM you can distinguish between three sorts of standardization:

- Standardization of only the observed variables
- Standardization of only the latent variables
- Standardization of the latent and the observed variables (**complete** standardization).

If you ask Stata for a standardized solution, you get the complete standardization. **However, I have not been able to interpret the intercepts in this solution. If you want the intercepts to be zero, you have to standardize the variables beforehand yourself.**

Standardized solutions can also be hard to interpret in multiple groups or with missing values treated with mlmv.

Generally, I like to start my analysis with standardized observed variables (note that the test file contains these). In LISREL, estimation goes a lot quicker when you start from standardized data – this I have not observed in Stata SEM.

However, what constitutes properly standardized variables in a file with missing values is not trivial – and may change with different selections and treatments of variables with missing values. Fortunately, the variations in solutions are often not large, but it can be a lot of work to get it straight.

Developing the intergenerational occupational reproduction model

The ISSP project and the double indicator design

The data we have been working with are from the 2009 Social Inequality IV module of the International Social Survey Programme. See www.issp.org for more information and access to the data (which are free for anyone). The ISSP is a social attitudes survey that runs around the world in about 45 countries. The 2009 data have now been collected in 43 countries (some as late as 2014). The module deals with attitudes about social inequality, but it also includes measures needed for a **complete intergenerational status attainment model**, which includes father's, mother's, respondent's first and current/last occupations (and respondent's education). These occupations in

ISSP are to be measured by an open question that is post-coded into the 4-digit International Standard Classification of Occupation 1988 [ISCO in your data). ISKO is a detailed nominal classification. To make the data fit for metric analysis, these codes are converted into the International Socio-Economic Index of occupational status [ISEI], a number between 10 and 90 that can be treated as interval.

Introducing the double indicator design for measuring occupations

The unique feature of the ISSP-2009 data then is that these four occupations were *also* measured with a second indicator, which is a *crude question* on the occupations. These four questions were optional in the ISSP design, and some countries decided to ask them, others did not. 16 countries participated in this research design one way or another, but not all did it in the same way. 10 countries implemented the complete research design (these countries are indicated by the variable “complete” in the data-file), 6 others did something of relevance, but somewhat differently. The crude occupation questions are also scaled in the ISEI metric.

Before embarking on modelling the full data, we try our hands on the relationship between father’s occupation and respondent’s (male and female offspring) occupations, which is a relationship of core interest to stratification researchers as the primary indicator of the openness / closedness of societies. Both occupation are measured using detailed indicators (fisei, isei) and a crude showcard indicator (fosei, osei). There are several research questions that can be answered by these data. One is about the correlation between father’s and offspring’s occupation, corrected for measurement error. Another important question is about the measurement quality of detailed and crude occupation measures. This is a question with great relevance as many researchers hate to do the costly occupation coding. However, conventional wisdom is that occupations cannot be validly / reliably measured with a showcard – the measurement quality of the crude questions muse somehow be inferior.

Let’s start by look at a few descriptives of the data at hand:

Variable	Obs	Mean	Std. Dev.	Min	Max
fisei	8142	36.14742	20.79816	9.29	89.35
fosei	8356	35.29344	14.85599	16	70
isei	7146	44.33541	21.36431	9.29	89.35
osei	7120	42.24059	15.44771	16	70

```
. sum fisei fosei isei osei if fisei < 100 & osei < 100 & fosei < 100 & isei < 100
```

Variable	Obs	Mean	Std. Dev.	Min	Max
fisei	6121	37.79409	20.74594	9.29	89.35
fosei	6121	36.04313	15.14541	16	70
isei	6121	44.75557	21.35317	9.29	89.35
osei	6121	42.58487	15.43417	16	70

We have a total of 9144 cases in 10 countries.

- All variables have missing values, but to a different degree.

- Fathers have on average lower status occupations than their offspring, according to both measurement methods.
- Despite being both expressed in the ISEI scale, the detailed measures have more variation than the crude measures.
- The descriptive are very much alike between pairwise and listwise, in particular with the crucial element, the SD.

Let have a look at the correlations between these four variables:

```
. corr fisei fosei isei osei
(obs=6121)

-----+-----
          |      fisei      fosei      isei      osei
-----+-----
fisei |      1.0000
fosei |      0.7651      1.0000
isei  |      0.3714      0.3656      1.0000
osei  |      0.3358      0.3689      0.7347      1.0000

. pwcorr fisei fosei isei osei, obs

-----+-----
          |      fisei      fosei      isei      osei
-----+-----
fisei |      1.0000
      |      8142
fosei |      0.7620      1.0000
      |      8062      8356
isei  |      0.3663      0.3582      1.0000
      |      6389      6520      7146
osei  |      0.3309      0.3669      0.7356      1.0000
      |      6331      6500      6881      7120
```

It is important to see that the pairwise and listwise correlations are not terribly different. This is good argument to assume that the missing values occur MCAR.

The basic model is:

```
. sem (FOCC -> fisei) (FOCC -> fosei) (OCC -> isei) (OCC -> osei)
(3023 observations with missing values excluded)

-----+-----
          |              OIM
          |      Coef.      Std. Err.      z      P>|z|      [95% Conf. Interval]
-----+-----
Measurement
fisei <-
  FOCC |              1 (constrained)
  _cons |      37.79409      .2651468      142.54      0.000      37.27441      38.31376
-----+-----
fosei <-
  FOCC |      .7529738      .0173031      43.52      0.000      .7190603      .7868872
  _cons |      36.04313      .1935684      186.20      0.000      35.66374      36.42252
-----+-----
isei <-
  OCC |              1 (constrained)
  _cons |      44.75557      .2729077      164.00      0.000      44.22068      45.29046
-----+-----
osei <-
  OCC |      .695508      .0171468      40.56      0.000      .6619009      .729115
  _cons |      42.58487      .197259      215.88      0.000      42.19825      42.97149
-----+-----
var(e.fisei) |      111.1247      7.010232              98.20033      125.75
```

```

var(e.fosei) | 48.36983 3.907018 41.28761 56.66689
var(e.isei) | 107.8232 8.1016 93.05823 124.9308
var(e.osei) | 69.80716 4.007731 62.37797 78.12116
var(FOCC) | 319.199 10.07867 300.044 339.5769
var(OCC) | 348.0603 11.22255 326.7451 370.766
-----
cov(FOCC,OCC) | 160.2708 6.092257 26.31 0.000 148.3302 172.2114
-----
LR test of model vs. saturated: chi2(1) = 38.27, Prob > chi2 = 0.0000

```

Notes:

- I run the model as a factor-analytic model, i.e. with a covariance between FOCC and OCC. This could be replaced by a causal effect (FOCC → OCC).
- The model LR (38.3, df=1) is statistically significant, which implies that the fitted covariance matrix does not fit the observed matrix. However, RMSEA = 0.054, pclose < 0.12, which means that we could get away with it.

Given that the detailed and crude measures have different variances, it is hard to compare their loadings. For this we could model the standardized indicators.

```
. sem (FOCC -> zfisei) (FOCC -> zfosei) (OCC -> zisei) (OCC -> zosei)
(3023 observations with missing values excluded)
```

```

-----
|                               OIM
|                               Coef. Std. Err. z P>|z| [95% Conf. Interval]
-----+-----
Measurement
zfisei <-
FOCC | 1 (constrained)
_cons | .0080079 .0127268 0.63 0.529 -.0169361 .0329519
-----+-----
zfosei <-
FOCC | 1.079337 .0272689 39.58 0.000 1.025891 1.132783
_cons | .0060604 .0127093 0.48 0.633 -.0188493 .0309701
-----+-----
zisei <-
OCC | 1 (constrained)
_cons | .0242117 .012779 1.89 0.058 -.0008346 .049258
-----+-----
zosei <-
OCC | 1.016364 .0274298 37.05 0.000 .962603 1.070126
_cons | .0203657 .0127892 1.59 0.111 -.0047005 .045432
-----+-----
var(e.zfisei) | .2949427 .0171611 .2631545 .3305707
var(e.zfosei) | .1773222 .0192713 .1433028 .2194176
var(e.zisei) | .2867443 .0185937 .252522 .3256044
var(e.zosei) | .2648152 .0190569 .229979 .3049284
var(FOCC) | .6964783 .0236393 .6516535 .7443863
var(OCC) | .712829 .0248689 .6657161 .7632762
-----+-----
cov(FOCC,OCC) | .3150088 .013445 23.43 0.000 .2886572 .3413604
-----
LR test of model vs. saturated: chi2(1) = 22.85, Prob > chi2 = 0.0000

```

- The fit changes (less misfit) – it should not. I will later explain why.
- We now can compare the loading of crude to detailed, and between the crudes for fathers and offspring. The conclusion would be that crude is at least as good as detailed, and that the measurement models are not too different between fathers and offspring. We can test this as follows:

```
. sem (FOCC -> zfisei) (FOCC -> zfosei@a) (OCC -> zisei) (OCC -> zosei@a)
(3023 observations with missing values excluded)
```

	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	

Measurement						
zfisei <-						
FOCC	1	(constrained)				
_cons	.0080079	.0127566	0.63	0.530	-.0169945	.0330103

zfosei <-						
FOCC	1.050094	.0206057	50.96	0.000	1.009707	1.09048
_cons	.0060604	.0126884	0.48	0.633	-.0188084	.0309292

zisei <-						
OCC	1	(constrained)				
_cons	.0242117	.0127465	1.90	0.058	-.0007709	.0491943

zosei <-						
OCC	1.050094	.0206057	50.96	0.000	1.009707	1.09048
_cons	.0203657	.0128152	1.59	0.112	-.0047516	.045483

var(e.zfisei)	.2793439	.0148253			.2517471	.3099659
var(e.zfosei)	.1951186	.0159068			.166305	.2289243
var(e.zisei)	.30525	.0150268			.2771742	.3361696
var(e.zosei)	.24522	.0158858			.2159801	.2784186
var(FOCC)	.7167259	.0207987			.6770989	.7586721
var(OCC)	.6892432	.020601			.6500259	.7308264

cov(FOCC,OCC)	.3143737	.0134319	23.41	0.000	.2880477	.3406997

LR test of model vs. saturated: chi2(2)	=	25.91,	Prob > chi2 = 0.0000			

- The LR difference is 25.9-22.9 = 3.0 (df=1). This is not a significant difference, as the magic number is 3.84 in a 1 df chi2-test.
- Crude is a marginally better measure than detailed. We could constrain the crude and detailed loadings to be equal, but the difference turns out to be significant.

Notice that the model is still unstandardized at the latent level. An interesting way to do standardization, is:

```
. sem (FOCC -> zfisei@a) (FOCC -> zfosei@a) (OCC -> zisei@a) (OCC -> zosei@a), var(FOCC@1)
var(OCC@1)
(3023 observations with missing values excluded)
```

	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	

Measurement						
zfisei <-						
FOCC	.8594736	.0069665	123.37	0.000	.8458195	.8731278
_cons	.0080079	.0127053	0.63	0.529	-.016894	.0329098

zfosei <-						
FOCC	.8594736	.0069665	123.37	0.000	.8458195	.8731278
_cons	.0060604	.0125701	0.48	0.630	-.0185765	.0306973

zisei <-						
OCC	.8594736	.0069665	123.37	0.000	.8458195	.8731278
_cons	.0242117	.0128684	1.88	0.060	-.0010098	.0494332

zosei <-						
OCC	.8594736	.0069665	123.37	0.000	.8458195	.8731278
_cons	.0203657	.0128705	1.58	0.114	-.0048601	.0455915

var(e.zfisei)	.2493806	.0087714			.2327682	.2671785

```

var(e.zfousei) | .2284629 .0085638 .21228 .2458795
var(e.zisei) | .274909 .0096123 .2567003 .2944092
var(e.zosei) | .2752546 .0096155 .2570392 .2947608
var(FOCC) | 1 (constrained)
var(OCC) | 1 (constrained)
-----+-----
cov(FOCC,OCC) | .4480227 .012523 35.78 0.000 .423478 .4725674
-----+-----
LR test of model vs. saturated: chi2(4) = 33.92, Prob > chi2 = 0.0000

```

- The LR difference with the previous model $33.9 - 25.9 = 8.0$ is marginally significant with $DF=2$.
- We obtain in this model an error corrected correlation between FOCC and OCC of 0.45, which is substantially stronger than the four observed correlations (which are around 0.36).
- Otherwise, we conclude that the crude measure has better measurement quality by a small, but statistically significant margin.

However, lets inspect the residual correlations:
`. estat residuals`

Residuals of observed variables

Mean residuals

	zfisei	zfosei	zisei	zosei
raw	0.000	0.000	0.000	0.000

Covariance residuals

	zfisei	zfosei	zisei	zosei
zfisei	-0.004			
zfosei	-0.004	-0.002		
zisei	0.010	-0.004	0.004	
zosei	-0.012	0.005	0.003	0.002

As I model z-standardized observed variables, these are actually residual correlation, not covariances. So it is easy to see that we are looking at very small numbers. However, there is still something of a pattern, which indicates a method effect: same-method correlations have a positive residual, different-method correlations have a negative residual. The presence of such method effects is of considerable interest when we try to answer the research question of measurement quality of crude and detailed measurement.

An interesting model to look at is:

```

. sem (FOCC -> zfisei@a) (FOCC -> zfosei) (OCC -> zisei@a) (OCC -> zosei), var(FOCC@1)
var(OCC@1) covar(e.zfosei*e.zosei) covar(e.zfisei*e.zisei)
(3023 observations with missing values excluded)

```

		Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
Measurement zfisei <-	FOCC	.8423736	14.47622	0.06	0.954	-27.53049 29.21523
	_cons	.0080079	.0127642	0.63	0.530	-.0170095 .0330252
zfosei <-	FOCC	.898343	15.43805	0.06	0.954	-29.35968 31.15637
	_cons	.0060604	.0127324	0.48	0.634	-.0188947 .0310155
zisei <-						

```

      OCC | .8423736 14.47622 0.06 0.954 -27.53049 29.21523
    _cons | .0242117 .0127418 1.90 0.057 -.0007618 .0491852
-----+-----
zosei <-
      OCC | .8543277 14.68165 0.06 0.954 -27.92118 29.62983
    _cons | .0203657 .0127683 1.60 0.111 -.0046597 .0453912
-----+-----
      var(e.zfisei) | .2876689 24.38876 1.97e-73 4.21e+71
      var(e.zfosei) | .1852828 27.73733 6.9e-129 5.0e+126
      var(e.zisei) | .2841755 24.38876 2.52e-74 3.21e+72
      var(e.zosei) | .2680284 25.08587 5.76e-81 1.25e+79
      var(FOCC) | 1 (constrained)
      var(OCC) | 1 (constrained)
-----+-----
cov(e.zfisei,e.zisei) | .0153171 10.66132 0.00 0.999 -20.88049 20.91113
cov(e.zfosei,e.zosei) | .0158854 11.53103 0.00 0.999 -22.58453 22.6163
cov(FOCC,OCC) | .4371408 .0127293 34.34 0.000 .4121918 .4620898
-----+-----
LR test of model vs. saturated: chi2(0) = 0.58, Prob > chi2 = .

```

- The model seems to converge without much problem, but the results are very strange. We obtain the two requested effects, but they are not significant. As a matter of fact, none of measurement effects is significant anymore. This is a sign that the model is not identified, despite having 0 DF. It is also strange that the fit is not perfect, with DF=0.
- Note that the point estimates of the loadings are still believable – they suggest a minor edge for crude measurement. Also note that the FOCC, OCC correlation is well estimated and did not change.

Tim suggested at this point to equalize the two residual correlations. This would help to identify the model, but is substantively unattractive: we want to know about the difference between crude and detailed. A better idea is to “buy your DF’s somewhere else”, by bringing in **auxiliary variables**. These should be other variables that are correlated with these occupations, but can be assumed to operate primarily via the latent occupations. There are several possible choices here: education, income, other occupations (such as mother’s). Best is the combination of education and income, but I will show this with only education:

```

. sem (FOCC -> zfisei@a) (FOCC -> zfosei) (OCC -> zisei@a) (OCC -> zosei) (zdegree) ,
var(FOCC@1) var(OCC@1) covar(e.zfosei*e.zosei) covar(e.zfisei*e.zisei)
(3031 observations with missing values excluded)
-----+-----
Measurement
zfisei <-
      FOCC | .8604452 .0085297 100.88 0.000 .8437273 .8771632
    _cons | .0085135 .0120081 0.71 0.478 -.015022 .0320489
-----+-----
zfosei <-
      FOCC | .8928553 .0138504 64.46 0.000 .865709 .9200015
    _cons | .0066523 .0118622 0.56 0.575 -.0165973 .0299019
-----+-----
zisei <-
      OCC | .8604452 .0085297 100.88 0.000 .8437273 .8771632
    _cons | .0245131 .0097533 2.51 0.012 .0053971 .0436291
-----+-----
zosei <-
      OCC | .8278169 .0103843 79.72 0.000 .8074639 .8481698
    _cons | .0207398 .0100736 2.06 0.040 .000996 .0404837
-----+-----
      var(e.zfisei) | .2713105 .0144562 .2444062 .3011765
      var(e.zfosei) | .2031964 .0155632 .1748723 .2361082
      var(e.zisei) | .2442218 .0092756 .2267021 .2630955
      var(e.zosei) | .3081388 .00946 .2901444 .3272493
      var(FOCC) | 1 (constrained)
      var(OCC) | 1 (constrained)

```

```

-----+-----
cov(e.zfisei,e.zisei) | .0081444 .0059989 1.36 0.175 -.0036133 .019902
cov(e.zfosei,e.zosei) | .02444 .0061145 4.00 0.000 .0124557 .0364242
cov(zdegree,F OCC) | .4150359 .0109969 37.74 0.000 .3934824 .4365894
cov(zdegree,OCC) | .7302173 .0064408 113.37 0.000 .7175936 .7428409
cov(F OCC,OCC) | .4369346 .0123854 35.28 0.000 .4126596 .4612096
-----+-----
LR test of model vs. saturated: chi2(2) = 6.29, Prob > chi2 = 0.0431

```

- Observe that adding the auxiliary variable solves the identification problem. All the odd results have disappeared.
- The two method effects are different: negligible and non-significant for detailed, and statistically significant, but also not terribly strong for crude.
- Crude seems to be working better for father than for offspring, detailed for offspring. A pooled estimation would make them almost equal.

Identification of SEM models

As we are getting to more complicated models, an important question becomes whether these models are identified. This is not a trivial matter, even for those who study this topic mathematically. The Stata SEM manual says about this:

... books have been written on this subject, and we will refer you to them. A few are [Bollen \(1989\)](#), [Brown \(2006\)](#), [Kline \(2011\)](#), and [Kenny \(1979\)](#). We will refer you to them, but do not be surprised if they refer you back to us. [Brown \(2006, 202\)](#) writes, "Because latent variable software programs are capable of evaluating whether a given model is identified, it is often most practical to simply try to estimate the solution and let the computer determine the model's identification status." That is not bad advice.

To which I must add, that the non-identification of a model may not be immediately apparent from the output. E.g. in our case we still obtain convergence, good L2 and good point estimates. If you do not study your results closely, in particular by comparing different models, you can easily get it wrong. The most interesting part to study is the SE column. The other part of interest is to check how your DF develops, and in particular whether you understand the development in conformity with the rule: **DF = number of correlations – number of estimated parameters**.

Some general guidelines on how to develop a SEM model

- Prepare your data in both the natural metric and in z-standardized metric. If your variables do not have an interesting natural metric, it is probably a good idea to standardize them to begin with.
- Study the univariate descriptives in a complete case and in an available case scenario to treat missing values.
- Also compare the correlations between variables of interest in a listwise and pairwise matrix. If the two different treatments of missing values give more or less the same numbers, be happy, the data are MCAR. Otherwise try to understand what kind of process is going on.
- Develop first your measurement model in a factor-analytic design, i.e. with saturated covariances / correlations between latent variables / constructs. The measurement model may contain such features as correlated residuals (method effects), simple structure, cross-loadings,

equality (or even more complicated) constraints. This stage is very similar to Exploratory Factor Analysis with oblique rotation – and it may even be a good idea to start there.

- I prefer to work on standardized data when developing the measurement model. This makes it possible to look at residual correlations, which are easier to interpret than residual covariances. I generally find little use of modification indices, but feel free to look at them.
- Once you have decided how your measurement model should look like and you are happy with the fit, you can start developing the structural part of the model, by imposing your assumed causal order and possibly trimming the effects that you deem to be unnecessary. While doing this, keep comparing fit statistics and coefficients of interest between models. In developing models, I generally prefer arguments about changes in coefficients of interest over arguments about fit statistics.
- If you are sort of happy with the model, re-estimate it with `method=mlmv` and see whether you are still happy (and whether the SE have indeed become smaller).
- Never publish only the fit statistics – the coefficients are the numbers of substantive interest.