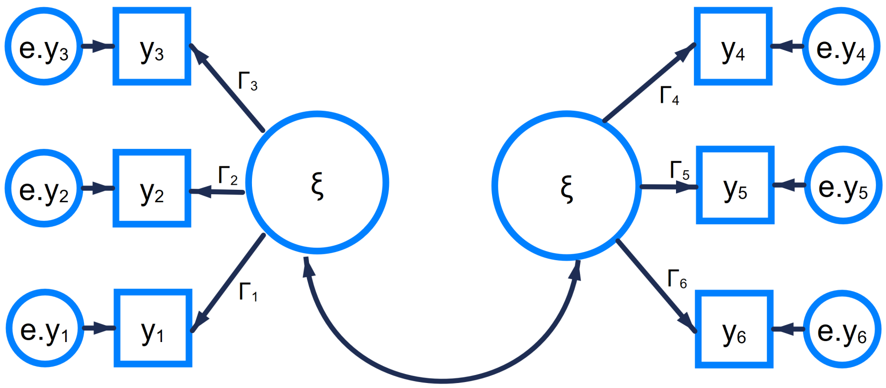
Introduction to structural equation modeling



Meghan K. Cain, Ph.D.

Assistant Director, Educational Services

StataCorp, LLC

Society for Prevention Research

July 18, 2024

If you have any questions about the course content, please email the instructor,
Meghan Cain: mcain@stata.com.

For questions about materials or licenses, please email Brooke Johnson: bjohnson@stata.com

Structural equation modeling using Stata

Contents

1 Introduction to the course

1.1 Course overview

1.1.1 Course goals

- What is structural equation modeling (SEM)?
- How to fit SEMs in Stata, using:
  - `sem`
  - SEM Builder
- How to evaluate model fit and interpret results

1.1.2 Course schedule

- We'll start with a conceptual introduction to SEM, and then demonstrate the following models:
  - Path analysis
  - Confirmatory factor analysis
  - Structural equation model
  - Multiple-group analysis
- We'll take short breaks roughly every hour.
- There will be independent exercises for you to work on throughout the course.

1.2 Working in Stata

1.2.1 Typography

- There are various fonts that will be used when describing commands.
  - `Items that must be typed as shown` will be in a `monospaced font`

```
. or on a separate line with a . in front
```

- When commands have <u>`under`</u>`lined` portions, that means you can type just that portion.
- *Items for which a substitution is needed* will be in *italics*.
- [Optional items] will be [in square brackets], though the brackets do not get typed.

1.2.2 Downloading the course files

- To download all the materials we will use for this workshop,visiting the following webpage: www.stata.com/news/conferences
- Click the link "Download here".

1.2.3 Handy navigation commands

- To print the current working directory, type `pwd`
- To change the working directory, type `cd` *folder*
- To go up one folder, type `cd ..`
- To list all files in a directory, type `ls`
- To list all Stata datasets in the working directory, type `ls *.dta`

2 Introduction to SEM

2.1 Definition

- Structural equation modeling (SEM) is a multivariate statistical analysis framework that allows simultaneous estimation of a system of equations.
  - Equations can include variables measured with error or be unobserved constructs.
  - Variables can be both predictors and outcomes at the same time.

2.2 Why do we need SEMs?

- SEM can be used to perform any of the following types of analyses:
  - Confirmatory factor analysis
  - Regression
  - ANOVA
  - Survival analysis
  - IRT analysis
  - Survey data analysis
  - Growth curve modeling
  - Multilevel modeling
  - Latent class analysis
  - Mixture modeling
  - And more!
- Furthermore, SEM can combine several models into one, i.e., a CFA as part of a regression or a growth curve model with IRT. As such, it is often regarded as the most flexible modeling framework.

2.3 Path diagrams

- Path diagrams are graphical representations of SEMs, in which different types of variables and model components are represented using different shapes and lines. See Figure 1.
- For this hypothetical study, we are interested in explaining differences in income in a group of 25-year-olds by their high school grade point average (GPA) and the quality of the college they attended. Furthermore, we think that the quality of the college they were able to attend was influenced by their GPA and scholastic aptitude in high school.

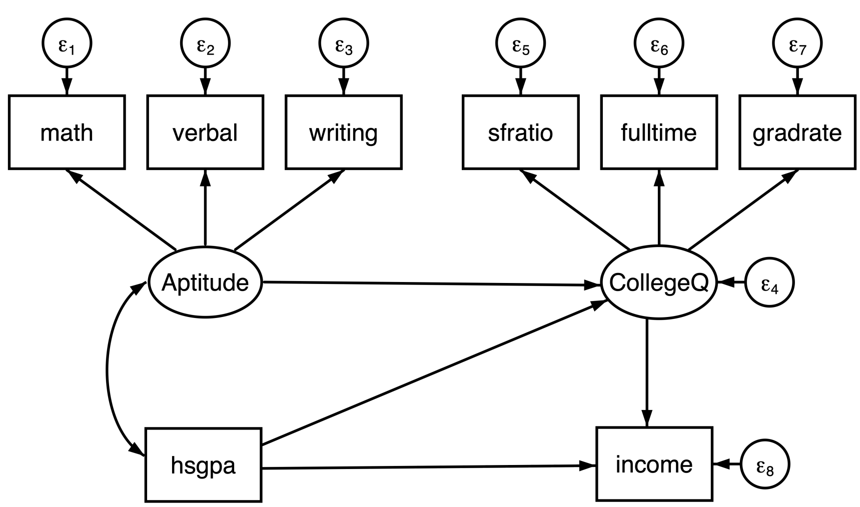


Figure 1: A path diagram.

- *Observed variables* are the measured variables in our dataset. They're represented by squares or rectangles, i.e., high school GPA (`hsgpa`), student-faculty ratio (`sfratio`), etc. Alternatively, we may have only summary statistics such as means, variances, and covariances for these variables.
- *Latent variables* are the unobserved, often more conceptual variables that are being measured by a set of observed variables. Latent variables are represented by circles or ovals, i.e., Aptitude and College Quality (`CollegeQ`). We do not have direct measurements of these variables in our dataset. Latent variables may represent (Bollen, 2002; Skrondal and Rabe-Hesketh, 2004):
  - a hypothetical construct
  - a variable that cannot be directly measured
  - a summarization of observed variables
  - the true value of a variable measured with error
  - unobserved heterogeneity
  - error or disturbance
- Because variables can be both predictors and outcomes in SEMs, the terms "independent variables" and "dependent variables" no longer apply. Rather, we use the terms "exogenous variables" and "endogenous variables".
  - *Endogenous variables* have at least one straight arrow pointing toward them. They may predict other endogenous variables. In our model, `CollegeQ` and `income` are examples of endogenous variables.
  - *Exogenous variables* do not have any straight arrows pointing toward them. They may covary with other variables. In our model, `hsgpa` and `Aptitude` are examples of exogenous variables.
- We estimate means and variances of exogenous variables and intercepts and residual variances of endogenous variables. Residual variances, ε_1-ε_8, can represent measurement error or unexplained variance. In Stata you don't need to add these yourself; they will automatically be added to your model for you.
- What sets SEM apart from other modeling frameworks is that SEM allows simultaneous estimation of a measurement model and a structural model.
  - The *measurement model* is the part or parts of the model where we are using observed variables to measure latent variables, i.e., `Aptitude` is being measured by `math`, `verbal`, and `writing`.
  - The *structural model* describes the relationships among our variables. It includes straight arrows and curved arrows.
    - Curved arrows represent covariances between variables, i.e., `Aptitude` and `hsgpa`.
    - Straight arrows can be either measurement paths, i.e., `math` on `Aptitude`, or regression paths where the arrow points to the outcome, i.e., `income` on `hsgpa`.
- Although SEM is sometimes referred to as causal modeling, note that simply specifying an arrow from X to Y in a model and finding that the model fits the data well is not enough to claim that X causes Y. At a minimum, in order to infer causality:
  - There must be an observed covariance between X and Y.
  - There are no other explanations for the covariance between X and Y such as a third variable that influences both X and Y.
  - The direction of the effect must be correct.

2.4 SEM analysis in Stata

- SEMs can be fit in Stata through the `sem` command, through the `gsem` command, or through the SEM Builder.
- `sem`
  - Standard linear SEMs
  - Has more estimation options and postestimation features than `gsem`

– Quicker and slightly more accurate than `gsem`

- `gsem`
  – Can accommodate generalized outcomes, i.e., binary, count, categorical, ordinal, survival
  – Can include random effects (multilevel components)
  – Can include latent categorical variables (such as in latent class analysis)

- Both `sem` and `gsem` models can be fit by drawing their path diagrams in the SEM Builder.

2.4.1 SEM Builder

- To open the SEM Builder, you can type `sembuilder` into the Command window. The tools along the left-hand side allow us to draw the path diagram (see Figure 2). Use the menus at the top to customize the appearance of the path diagram, fit the model, customize the results, and more.

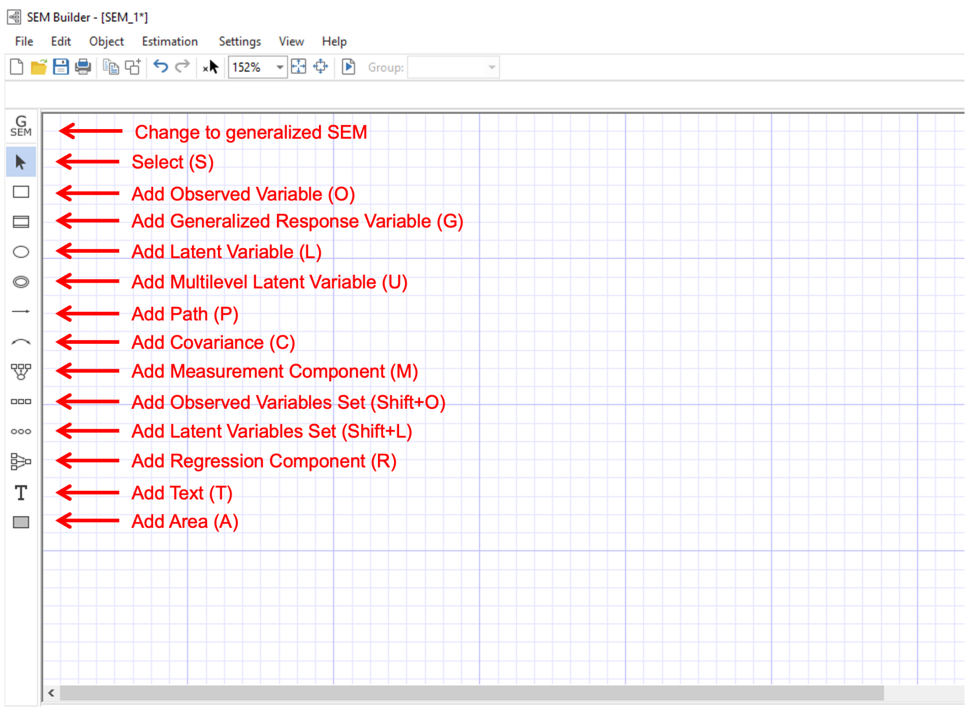


Figure 2: SEM Builder

- One setting you may consider changing that can make your path diagrams a bit prettier is to click on **Settings > Automation** then check **Attach based on position of variables**.

- You can click on the rectangles and arrows on the left side to build our model. You can also open an existing model in the builder by typing `sembuilder` followed by the file name (`.stsem` files).

- When we click the **Estimate** button at the top, an estimation window opens. We will discuss the other estimation options later. For now, we'll use the default, maximum likelihood. Once you click **Submit** at the bottom of the window, two things will happen:
  – The corresponding `sem` command and model results will appear in the Results window.
  – The estimated path coefficients will appear on your path diagram in the SEM Builder.

2.4.2 `sem`

- Both `sem` and `gsem` share the same syntax but differ in their available options. Their general syntax is

  `sem/gsem (`*path1*`) (`*path2*`) (`*...*`) (`*last path*`)` [`,` *options*]

- Paths are specified in parentheses, and the direction of relationships is specified using arrows, `(x->y)`. Arrows can point in either direction, `(x->y)` or `(y<-x)`. Paths can be specified individually, or multiple paths can be specified within a single set of parentheses, `(x1 x2 x3 -> y)`.

- We already saw the `sem` command output in the Results window from when we fit our model in the SEM Builder. This command usually has a lot of extra options we don't need. If we had wanted to fit our model through the Command window, we could have just typed

```
. sem (Aptitude -> math verbal writing)
```

- By default, Stata assumes that capitalized variables are latent and lowercase variables are observed. This can be changed using the `nocapslatent` option and specifying the latent variables with the `latent()` option.

- The `sem` command has more options that we will use later. We'll just stick with the basics for now.

- Although the `gsem` command will not be covered in this short course, it uses largely the same syntax and options as `sem`. Check out examples using both commands in the documentation: https://www.stata.com/manuals/sem.pdf.

3 Path analysis models

- Path analysis encompasses linear regression and multivariate regression models as well as mediation and cross-lagged panel models. It allows for including many observed variables with complex assumed relationships.

3.1 Path analysis example

- For this section, we'll be reproducing results from the paper "The conceptual roles of negative and positive affectivity in the stressor-strain relationship" (Rydstedt et al., 2013). This paper is included in your materials as `example.pdf`. We would like to fit the work situation model shown in Figure 3.

Table 1

Inter-Correlations (Pearson's r_{xy}) for the Variables in the Study

		M	*SD*	1	2	3	4
1	Positive Affect	3.43	0.52				
2	Negative Affect	1.73	0.55	-.12			
3	Role conflict	2.45	0.52	-.34	.43		
4	Role Ambiguity	2.04	0.64	-.32	.32	.53	
5	Stress	1.798	0.59	-.33	.54	.56	.39

Note. All inter-correlations $p < .01$. $N = 719$.

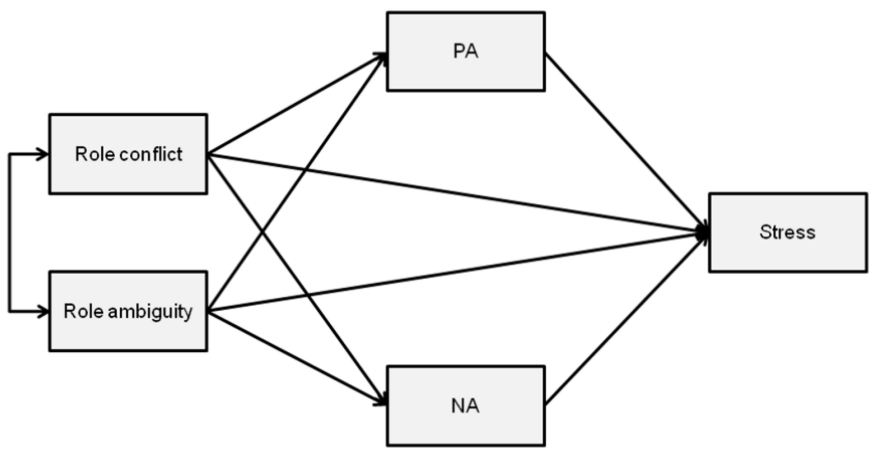


Figure 3: Summary statistics and work situation model from Rydstedt et al. (2013)

- In our example paper, the authors report the means, standard deviations, and correlation matrix of the observed variables, as shown in Figure 3.

- We start by inputting these summary statistics using the `ssd` commands. We initialize this process by supplying a list of variable names and setting the number of observations.

```
. clear
. ssd init pa na conflict ambiguity stress
Summary statistics data initialized.  Next use, in any order,
    ssd set observations (required)
        It is best to do this first.
    ssd set means (optional)
        Default setting is 0.
    ssd set variances or ssd set sd (optional)
        Use this only if you have set or will set correlations and, even then, this is optional
        but highly recommended.  Default setting is 1.
    ssd set covariances or ssd set correlations (required)
```

```
. ssd set observations 719
  (value set)
    Status:
                        observations:    set
                               means:  unset
                    variances or sd:  unset
        covariances or correlations:  unset (required to be set)
```

- Then we can list the means and standard deviations.

```
. ssd set means 3.43 1.73 2.45 2.04 1.798
  (values set)
    Status:
                        observations:    set
                               means:    set
                    variances or sd:  unset
        covariances or correlations:  unset (required to be set)
. ssd set sd 0.52 0.55 0.52 0.64 0.59
  (values set)
    Status:
                        observations:    set
                               means:    set
                    variances or sd:    set
        covariances or correlations:  unset (required to be set)
```

- It will be easiest to supply the correlation matrix by first creating a Stata matrix with the `matrix input` command. This can be done with the dialog box: Click **Data > Matrices, ado language > Input matrix by hand**. Check **Create symmatric matrix** to only put in the lower triangle, and remember to put 1s on the diagonal because this is a correlation matrix.

```
. matrix input cor = (1,-.12,-.34,-.32,-.33\-.12,1,.43,.32,.54\ ///
> -.34,.43,1,.53,.56\-.32,.32,.53,1,.39\-.33,.54,.56,.39,1)
. ssd set correlations (stata) cor
  (values set)
    Status:
                        observations:    set
                               means:    set
                    variances or sd:    set
        covariances or correlations:    set
```

- The `(stata)` part of the above code is to tell the `ssd` commands that we have a Stata matrix.
- We can review the summary statistics that we entered with `ssd list`.

```
. ssd list
  Observations = 719
  Means:
         pa         na   conflict  ambiguity     stress
       3.43       1.73       2.45       2.04      1.798
  Standard deviations:
         pa         na   conflict  ambiguity     stress
        .52        .55        .52        .64        .59
  Correlations:
         pa         na   conflict  ambiguity     stress
          1
       -.12          1
       -.34        .43          1
       -.32        .32        .53          1
       -.33        .54        .56        .39          1
```

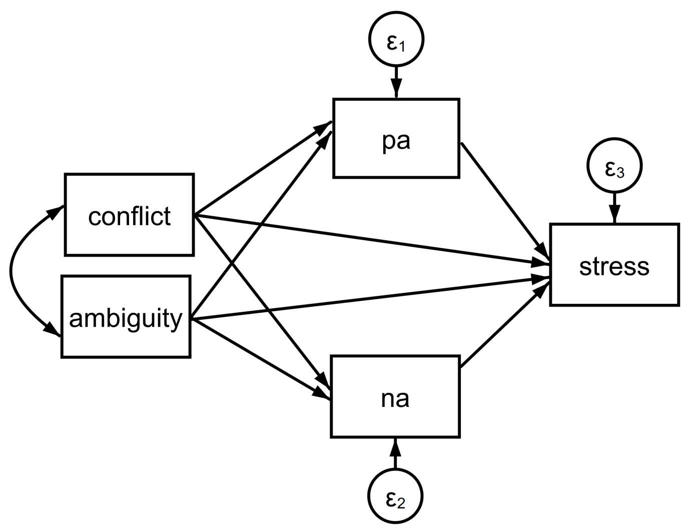
- Now, for this example and for the rest of this course, we'll follow the following four steps of SEM analysis in Stata:

1. Specify the model

2. Fit the model
3. Evaluate the model
4. Report and interpret the results

3.1.1 Specify the model

- Our goal in this example is to fit a path analysis model to explain stress. You can find the path diagram for this model in `stress.stsem`.



- In this model, we have two exogenous variables, `conflict` and `ambiguity`, and three endogenous variables, `pa`, `na`, and `stress`.
- All five variables are observed; there are no latent variables. Accordingly, there is no measurement model.
- We could specify this model in equation form as

  Measurement Model: (none)

  Structural Model:

$$\begin{aligned}
\texttt{pa} &= \Gamma_1\texttt{conflict} + \Gamma_2\texttt{ambiguity} + \texttt{e.pa} && (1)\\
\texttt{na} &= \Gamma_3\texttt{conflict} + \Gamma_4\texttt{ambiguity} + \texttt{e.na}\\
\texttt{stress} &= \Gamma_5\texttt{conflict} + \Gamma_6\texttt{ambiguity} + B_1\texttt{pa} + B_2\texttt{na} + \texttt{e.stress} && (2)
\end{aligned}$$

  Covariance Structures:

$$var\left(\begin{bmatrix}\texttt{conflict}\\ \texttt{ambiguity}\end{bmatrix}\right) = \begin{bmatrix}\Phi_{11} & \\ \Phi_{12} & \Phi_{22}\end{bmatrix}$$

$$var\left(\begin{bmatrix}\texttt{e.pa}\\ \texttt{e.na}\\ \texttt{e.stress}\end{bmatrix}\right) = \begin{bmatrix}\Psi_{11} & & \\ 0 & \Psi_{22} & \\ 0 & 0 & \Psi_{33}\end{bmatrix}$$

3.1.2 Fit the model

- We can fit this model either through the SEM Builder or the `sem` commands.

```
. sem (conflict ambiguity -> pa na stress) (pa na -> stress), noxconditional nomeans
Endogenous variables
  Observed: pa na stress
```

```
Exogenous variables
  Observed: conflict ambiguity

Fitting target model:
Iteration 0:  Log likelihood = -2560.2609
Iteration 1:  Log likelihood = -2560.2609

Structural equation model                               Number of obs = 719
Estimation method: ml

Log likelihood = -2560.2609

--------------------------------------------------------------------------------------
                       |                 OIM
                       | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-----------------------+--------------------------------------------------------------
Structural             |
  pa                   |
              conflict |  -.2369629   .0407181    -5.82   0.000    -.3167689   -.1571569
             ambiguity |  -.1579579   .0330835    -4.77   0.000    -.2228002   -.0931155
  ---------------------+--------------------------------------------------------------
  na                   |
              conflict |   .3830108   .0416908     9.19   0.000     .3012983    .4647233
             ambiguity |    .110066   .0338738     3.25   0.001     .0436746    .1764574
  ---------------------+--------------------------------------------------------------
  stress               |
                    pa |  -.1821824    .033928    -5.37   0.000      -.24868   -.1156847
                    na |   .3930666   .0331364    11.86   0.000     .3281204    .4580127
              conflict |   .3637271   .0400611     9.08   0.000     .2852088    .4422454
             ambiguity |   .0474406   .0307801     1.54   0.123    -.0128873    .1077684
-----------------------+--------------------------------------------------------------
              var(e.pa)|   .2314703    .012208                       .2087382    .2566781
              var(e.na)|   .2426615   .0127983                       .2188303     .269088
          var(e.stress)|   .1910345   .0100754                       .1722735    .2118387
          var(conflict)|   .2700239   .0142414                       .2435055    .2994303
         var(ambiguity)|   .4090303   .0215728                       .3688604    .4535748
-----------------------+--------------------------------------------------------------
cov(conflict,ambiguity)|   .1761387   .0140272    12.56   0.000     .1486458    .2036316
--------------------------------------------------------------------------------------
LR test of model vs. saturated: chi2(1) = 2.03                      Prob > chi2 = 0.1538
```

3.1.3 Evaluate the model

- Notice that this is an over-identified model. We have 5 variables in our model, so we have 15 unique covariance elements, $5(6)/2 = 15$, and we are estimating 14 parameters. Therefore, the degrees of freedom of our model is 1. We can evaluate our model with the `estat gof` command.

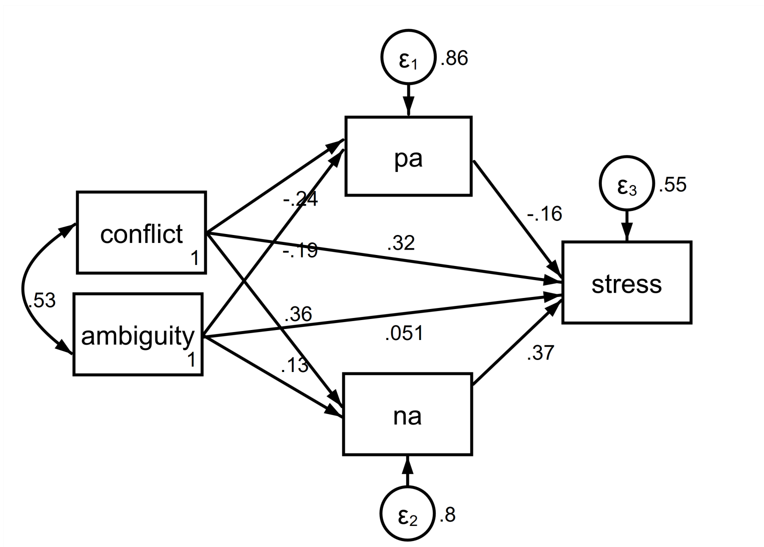
```
. estat gof, stats(all)

----------------------------------------------------------------------------
Fit statistic        |      Value   Description
---------------------+------------------------------------------------------
Likelihood ratio     |
          chi2_ms(1) |      2.034   model vs. saturated
            p > chi2 |      0.154
          chi2_bs(9) |    700.704   baseline vs. saturated
            p > chi2 |      0.000
---------------------+------------------------------------------------------
Population error     |
               RMSEA |      0.038   Root mean squared error of approximation
 90% CI, lower bound |      0.000
         upper bound |      0.115
              pclose |      0.468   Probability RMSEA <= 0.05
---------------------+------------------------------------------------------
Information criteria |
                 AIC |   5148.522   Akaike's information criterion
                 BIC |   5212.612   Bayesian information criterion
---------------------+------------------------------------------------------
Baseline comparison  |
                 CFI |      0.999   Comparative fit index
                 TLI |      0.987   Tucker-Lewis index
---------------------+------------------------------------------------------
Size of residuals    |
                SRMR |      0.012   Standardized root mean squared residual
                  CD |      0.391   Coefficient of determination
----------------------------------------------------------------------------
```

- Our model fits well, $\chi^2(1) = 2.034, p = 0.154$; RMSEA=0.038;CFI=0.999.

3.1.4 Report and interpret the results

- When reporting your structural equation modeling analysis, you should include two types of information: the covariance matrix of the observed variables and the model results. The covariance matrix allows other researchers to reproduce your results. The model results can be reported either on a path diagram or with a table, or both.
  - The paper already reported the descriptive statistics of the variables. We could also get these again with `ssd list`.
  - The paper reports standardized results, so let's do the same.
- To get standardized coefficients, we can replay our `sem` results and add the `standardized` option. Using `sem` in this way is not re-running the model, but displaying the results again for us. If you've already fit your model in the SEM Builder, you can get these by clicking **View > Standardized estimates**.



- Our results closely match those reported in the paper. They are not exactly the same simply because of rounding error.
- Increased conflict ($\hat{\beta} = 0.32\,[0.25, 0.39]$) and negative affect ($\hat{\beta} = 0.37\,[0.31, 0.42]$) increase stress, while positive affect reduces it ($\hat{\beta} = -0.16\,[-0.22, -0.10]$). Ambiguity does not have a statistically significant effect on stress ($\hat{\beta} = 0.05\,[-0.01, 0.12]$). Furthermore, as expected, conflict and ambiguity increase negative affect and decrease positive affect.

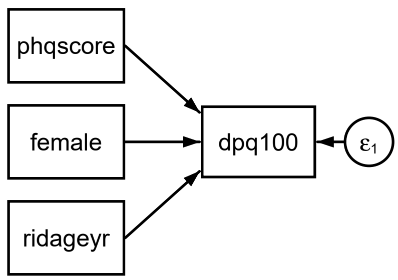
3.2 Exercise: Path analysis models

- Open the dataset `nhanes.dta`

  (Note: If you're interested in how this dataset was imported and cleaned, see `dataprep.do`.)

- These data contain demographics and results from a depression screener from the 2017-2018 National Health and Nutrition Examination Survey (NHANES).

  1. Fit the following model and interpret the results:



4 Measurement models

- In the last section, we saw a few examples of purely structural models. In this section, we will focus on measurement models. In later sections, we'll see models with both measurement and structural components.
- In the preceding section, we assumed that all the variables in our models were perfectly measured. We often make this assumption in empirical research, either implicitly or explicitly. This assumption is rarely true in practice.

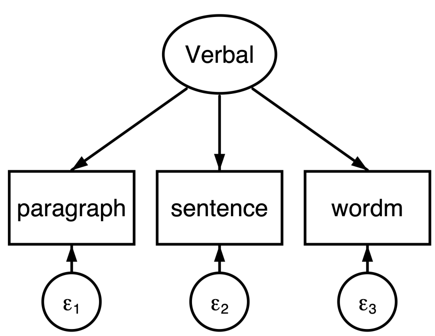
4.1 One-factor CFA

- Confirmatory factor analysis (CFA) is an approach to accounting for measurement error in a scale or other unobservable construct.
  - CFA involves collecting several items that are related to the concept of interest, each of which includes some error. For example, if we're interested in depression, we might ask about a series of symptoms (feeling unhappy, interest in hobbies, sleep problems).
  - The underlying concept is represented by a latent variable and is measured using a set of observed variables, often referred to as items or indicators. It's confirmatory in the sense that we already have an idea about which items measure a given concept.
- CFA models include one or more latent variables (factors). We will demonstrate a one-factor model, which aims to measure one underlying concept, verbal ability. Let's load the dataset now.

```
. use hsdata, clear
(A classic data set in psychometrics from Holzinger and Swineford (1939))
```

4.1.1 Specify the model

- To specify a measurement model, we put arrows leading from the latent variable to the observed variables. This is because the latent variable (verbal ability) is assumed to cause the observed variables (the test scores). When arrows go from the observed variables to the latent variables, they are often referred to as formative indicators; see Bollen and Bauldry (2011) for more details.
- You can find the path diagram for our model in `cfa.stsem`.



- This model has one latent exogenous variable and three observed endogenous variables.
- The corresponding equations are:

  Measurement Model:

$$\begin{aligned} \texttt{paragraph} &= \Gamma_1\texttt{Verbal} + \texttt{e.paragraph} \qquad (3)\\ \texttt{sentence} &= \Gamma_2\texttt{Verbal} + \texttt{e.sentence}\\ \texttt{wordm} &= \Gamma_3\texttt{Verbal} + \texttt{e.wordm} \end{aligned}$$

– Note: the coefficients from latent variables to their indicators are often called factor loadings. These are sometimes represented as λ or Λ rather than as Γ.

Structural Model: (none)

Covariance Structures:

$$var\,(\texttt{Verbal}) = \Phi$$

$$var\left(\begin{bmatrix} \texttt{e.paragraph} \\ \texttt{e.sentence} \\ \texttt{e.wordm} \end{bmatrix}\right) = \begin{bmatrix} \Psi_{11} & & \\ 0 & \Psi_{22} & \\ 0 & 0 & \Psi_{33} \end{bmatrix}$$

4.1.2 Fit the model

- We specify our model using a capitalized `Verbal` for the latent verbal ability with arrows pointing toward its indicators.

```
. sem (Verbal -> paragraph sentence wordm), noxconditional nomeans nolog
Endogenous variables
  Measurement: paragraph sentence wordm
Exogenous variables
  Latent: Verbal
Structural equation model                                Number of obs = 301
Estimation method: ml
Log likelihood = -2514.7407
 ( 1)  [paragraph]Verbal = 1
```

	Coefficient	OIM std. err.	z	P>|z|	[95% conf.	interval]
Measurement						
paragraph						
Verbal	1	(constrained)				
sentence						
Verbal	1.510525	.0893473	16.91	0.000	1.335407	1.685642
wordm						
Verbal	2.156426	.1315606	16.39	0.000	1.898572	2.41428
var(e.paragraph)	3.435149	.4401399			2.672283	4.415792
var(e.sentence)	6.658378	.9460044			5.040017	8.796398
var(e.wordm)	18.06818	2.159346			14.29507	22.83717
var(Verbal)	8.720832	1.009334			6.950904	10.94144

```
LR test of model vs. saturated: chi2(0) = 0.00                    Prob > chi2 = .
```

- Notice in our output that the path from `Verbal` to `paragraph` has been constrained to 1; this is for identification purposes. Because latent variables are unobservable constructs, they have no natural scale. Does a verbal ability of 1 inherently mean anything to you?

- In order to set the scale of our latent variable, we have two options:
  - Constrain the path from the latent variable to its first indicator to 1 (default). This will set the scale to be the same as that observed variable.
  - Explicitly specify the variance of the latent variable, usually to 1. By default, Stata also sets the mean of latent variables to 0, so when its variance is set to 1 it can be interpreted as a standard normal variable. This is helpful for interpretation in models with structural components.

```
. sem (Verbal -> paragraph sentence wordm), noxconditional nomeans var(Verbal@1) nolog
Endogenous variables
  Measurement: paragraph sentence wordm
Exogenous variables
  Latent: Verbal
```

```
Structural equation model                                Number of obs = 301
Estimation method: ml

Log likelihood = -2514.7407

 ( 1)  [/]var(Verbal) = 1

--------------------------------------------------------------------------------
                |                 OIM
                | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
----------------+---------------------------------------------------------------
Measurement     |
  paragraph     |
         Verbal |   2.953105   .1708937    17.28   0.000      2.61816    3.288051
  --------------+---------------------------------------------------------------
  sentence      |
         Verbal |   4.460739   .2504337    17.81   0.000     3.969898     4.95158
  --------------+---------------------------------------------------------------
  wordm         |
         Verbal |   6.368154   .3778412    16.85   0.000     5.627599    7.108709
----------------+---------------------------------------------------------------
var(e.paragraph)|   3.435149   .4401399                      2.672283    4.415792
 var(e.sentence)|   6.658378   .9460044                      5.040017    8.796398
    var(e.wordm)|   18.06818   2.159346                      14.29507    22.83717
     var(Verbal)|          1  (constrained)
--------------------------------------------------------------------------------
LR test of model vs. saturated: chi2(0) = 0.00                   Prob > chi2 = .
```

- Returning to our results, the factor loadings tell you how the latent variable is being measured. They represent the expected change in the item per unit change on the latent variable. Below that, we see the variances of the errors and the variance of the latent variable. The error variances represent the portion of the variances in the corresponding indicator that is not explained by the latent variable. This could also be called the unique variance or uniqueness.

4.1.3 Evaluate the model

- We have 3 variables in our model so we have 6 unique covariance elements, $3(4)/2 = 6$, and we are estimating 6 parameters (constrained parameters don't count). Therefore, the degrees of freedom of our model is 0. Because our model is just-identified, we have perfect fit. As a rule of thumb, a one-factor model with three indicators will be just-identified. Two indicators will not be sufficient unless there are other variables in the model.

4.1.4 Report and interpret the results

- Even in models with latent variables, we report descriptive statistics of the observed variables. I created a do file called `collect` that will do this for us if we supply the names of the observed variables.

```
. do collect "paragraph sentence wordm"

. dtable `1´                                              // table of means and SDs

-------------------------------------------
                                    Summary
-------------------------------------------
N                                       301
Paragraph Comprehension Test  9.183 (3.492)
Sentence Completion Test     17.362 (5.162)
Word Meaning Test            15.299 (7.669)
-------------------------------------------

. collect remap var=roweq                                 // remap to align with correlation
(8 items remapped in collection DTable)

.
. pwcorr `1´, sig                                         // correlations and significance

             | paragr~h sentence    wordm
-------------+---------------------------
   paragraph |   1.0000 
             |
             |
    sentence |   0.7332   1.0000 
             |   0.0000
             |
       wordm |   0.7045   0.7200   1.0000 
             |   0.0000   0.0000
             |
```

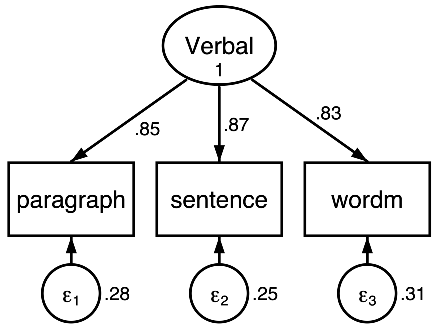
```
              |
. local N=r(N)
. collect get C=vech(r(C)) S=vech(r(sig))         // vech() to get lower triangle
. collect layout (result[mean sd] result[C]#rowname) (roweq)
Collection: DTable
      Rows: result[mean sd] result[C]#rowname
   Columns: roweq
   Table 1: 6 x 3
────────────────────────────────────────────────────────────────────────────────────────────────────
──
                               Paragraph Comprehension Test Sentence Completion Test Word Meaning Te
> st
────────────────────────────────────────────────────────────────────────────────────────────────────
──
Mean                                                  9.183                   17.362             15.2
> 99
Standard deviation                                  (3.492)                  (5.162)            (7.66
> 9)
C
>
  Paragraph Comprehension Test                        1.000
>
  Sentence Completion Test                            0.733                    1.000
>
  Word Meaning Test                                   0.704                    0.720              1.0
> 00
────────────────────────────────────────────────────────────────────────────────────────────────────
──
.
. ** make it pretty
. collect label levels result C "Correlation"
. collect stars S 0.01 "**" 0.05 "*", attach(C) shownote
. collect title "Descriptive statistics of the observed variables."
. collect note "N=`N´"
. collect preview
Descriptive statistics of the observed variables.
────────────────────────────────────────────────────────────────────────────────────────────────────
──
                               Paragraph Comprehension Test Sentence Completion Test Word Meaning Te
> st
────────────────────────────────────────────────────────────────────────────────────────────────────
──
Mean                                                  9.183                   17.362             15.2
> 99
Standard deviation                                  (3.492)                  (5.162)            (7.66
> 9)
Correlation
>
  Paragraph Comprehension Test                        1.000
>
  Sentence Completion Test                          0.733**                    1.000
>
  Word Meaning Test                                 0.704**                  0.720**              1.0
> 00
────────────────────────────────────────────────────────────────────────────────────────────────────
──
** p<.01, * p<.05
N=301
.
end of do-file
```

- To make this table fit on one page, we could display variable names rather than labels in the rows.

```
. collect style header rowname, level(value)
. collect preview
Descriptive statistics of the observed variables.
──────────────────────────────────────────────────────────────────────────────────────────
                   Paragraph Comprehension Test Sentence Completion Test Word Meaning Test
──────────────────────────────────────────────────────────────────────────────────────────
Mean                                      9.183                   17.362            15.299
```

```
Standard deviation                   (3.492)              (5.162)          (7.669)
Correlation
  paragraph                            1.000
  sentence                           0.733**                1.000
  wordm                              0.704**              0.720**            1.000
------------------------------------------------------------------------------------
** p<.01, * p<.05
N=301
```

- It will be easier to compare factor loadings and error variances across indicators with standardized results.



- When indicators load on a single latent variable, the standardized path coefficient is the correlation between the latent variable and the indicator. Squaring this value gives an estimate of the explained proportion of the item's variance (R^2). The standardized error variances are the proportion of the item's variance not explained by the latent variable.
- In our results, we see that our test scores each contribute to our measure of verbal ability pretty equally. The correlation between sentence completion and verbal ability is highest, followed by paragraph comprehension and word meaning. Likewise, we see that word meaning has the most measurement error. In other words, from among our three test scores, sentence completion score is the best indicator of verbal ability.

4.2 Exercise: Factor analysis models

- Returning to our `nhanes` dataset,
  1. Fit a CFA model for the Patient History Questionnaire, using items `dpq010-dpq090`.
  2. Interpret the results.

5 Structural equation models

- SEMs combine the concepts of path analysis and confirmatory factor analysis. One or more latent variables is included in the model with corresponding observed indicators. Unlike CFA, structural relationships may exist among latent and/or observed variables. This allows us to test our hypotheses about the structural relationships between our variables while accounting for their measurement errors.

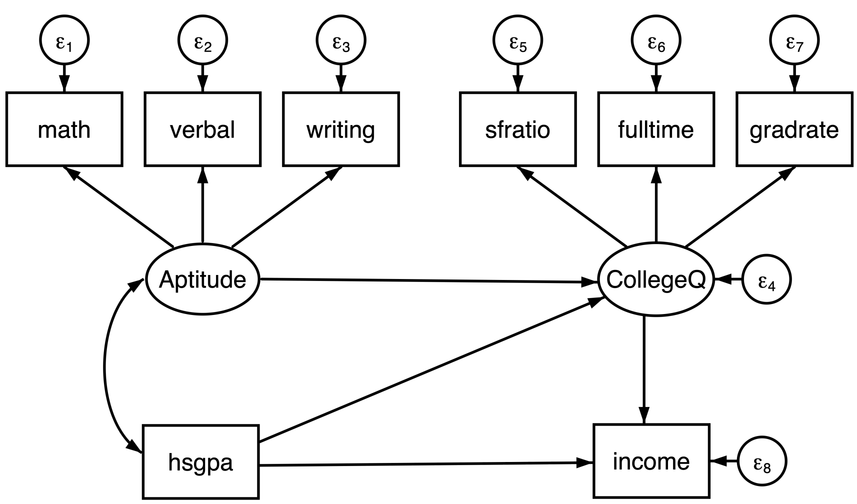
5.1 Models with measurement and structural paths

- For this section, we'll return to the very first example we talked about, where we would like to see how high school GPA and scholastic aptitude influence the quality of the college a group of 25-year-olds was able to attend and future income. The data for this example can be found in `sem.dta`. Let's take a look at the variables in this dataset.

```
. use sem, clear
. codebook, compact
Variable   Obs Unique      Mean   Min      Max  Label
-----------------------------------------------------------------------------------------------------
cohort     300      2  2010.933  2000     2020  Year of data collection
hsgpa      300    105     3.108  2.24        4  High School Grade Point Average (GPA)
math       300    179    522.29   281      754  Math Score (200-800)
verbal     300    189  577.5967   349      762  Verbal Score (200-800)
writing    300    178  519.5767   332      738  Writing Score (200-800)
grad       300      2        66     0      100  Attended College
sfratio    300     46  28.66667     3       54  Avg Student:Faculty Ratio
fulltime   300     43  51.75333    24       72  % Full time faculty
gradrate   300     31     65.58    48       82  6-year Graduation Rate
debt       300    300  35.45694     0  84.7608  Average debt (in thousands of dollars)
income     300     58     75.36    41      108  Income (in thousands of dollars)
enjoy      300      5         3     1        5  Enjoyed experience 1-5
inc_cat    300      5  2.056667     0        4  Income category 1-5
-----------------------------------------------------------------------------------------------------
```

5.1.1 Specify the model

- The path diagram for this model can be found in `sem.stsem`.



- This model has one observed exogenous variable, one latent exogenous variable, seven observed endogenous variables, and one latent endogenous variable.

- The corresponding equations are:

  Measurement Model:

$$
\begin{aligned}
\texttt{math} &= \Gamma_1\texttt{Aptitude} + \texttt{e.math} \\
\texttt{verbal} &= \Gamma_2\texttt{Aptitude} + \texttt{e.verbal} \\
\texttt{writing} &= \Gamma_3\texttt{Aptitude} + \texttt{e.writing} \\
\texttt{sfratio} &= B_1\texttt{CollegeQ} + \texttt{e.sfratio} \\
\texttt{fulltime} &= B_2\texttt{CollegeQ} + \texttt{e.fulltime} \\
\texttt{gradrate} &= B_3\texttt{CollegeQ} + \texttt{e.gradrate}
\end{aligned} \tag{4}
$$

Structural Model:

$$
\begin{aligned}
\texttt{CollegeQ} &= \Gamma_4\texttt{hsgpa} + \Gamma_5\texttt{Aptitude} + \texttt{e.CollegeQ} \\
\texttt{income} &= \Gamma_6\texttt{hsgpa} + B_4\texttt{CollegeQ} + \texttt{e.income}
\end{aligned}
$$

Covariance Structures:

$$
var\left(\begin{bmatrix} \texttt{hsgpa} \\ \texttt{Aptitude} \end{bmatrix}\right) = \begin{bmatrix} \Phi_{11} & \\ \Phi_{12} & \Phi_{22} \end{bmatrix}
$$

$$
var\left(\begin{bmatrix} \texttt{e.math} \\ \texttt{e.verbal} \\ \texttt{e.writing} \\ \texttt{e.sfratio} \\ \texttt{e.gradrate} \\ \texttt{e.avgSAT} \\ \texttt{e.CollegeQ} \\ \texttt{e.income} \end{bmatrix}\right) = \begin{bmatrix} \Psi_{11} & & & & & & & \\ 0 & \Psi_{22} & & & & & & \\ 0 & 0 & \Psi_{33} & & & & & \\ 0 & 0 & 0 & \Psi_{44} & & & & \\ 0 & 0 & 0 & 0 & \Psi_{55} & & & \\ 0 & 0 & 0 & 0 & 0 & \Psi_{66} & & \\ 0 & 0 & 0 & 0 & 0 & 0 & \Psi_{77} & \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \Psi_{88} \end{bmatrix}
$$

5.1.2 Fit the model

- In our `sem` command, we will specify the measurement paths first followed by the structural paths. This is not necessary, but it will make it easier for Stata to figure out which paths are measurement paths.

```
. sem (Aptitude -> math verbal writing) (CollegeQ -> sfratio fulltime gradrate) ///
>         (hsgpa Aptitude -> CollegeQ) (hsgpa CollegeQ -> income), ///
>         noxconditional nomeans nolog
Endogenous variables
  Observed:    income
  Measurement: math verbal writing sfratio fulltime gradrate
  Latent:      CollegeQ
Exogenous variables
  Observed: hsgpa
  Latent:   Aptitude
Structural equation model                              Number of obs = 300
Estimation method: ml
Log likelihood = -9144.7139
 ( 1)  [sfratio]CollegeQ = 1
 ( 2)  [math]Aptitude = 1
------------------------------------------------------------------------------
                   |                 OIM
                   | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------------+----------------------------------------------------------------
Structural         |
  income           |
          CollegeQ |   .6599806   .0961211     6.87   0.000     .4715868    .8483744
             hsgpa |   20.47139   2.057554     9.95   0.000     16.43865    24.50412
  -----------------+----------------------------------------------------------------
  CollegeQ         |
             hsgpa |   4.308577   1.836601     2.35   0.019     .7089063    7.908249
          Aptitude |   .1558428   .0313754     4.97   0.000     .0943481    .2173375
```

Measurement						
math						
Aptitude	1	(constrained)				
verbal						
Aptitude	1.240077	.2097064	5.91	0.000	.8290604	1.651094
writing						
Aptitude	1.227641	.2275876	5.39	0.000	.7815775	1.673704
sfratio						
CollegeQ	1	(constrained)				
fulltime						
CollegeQ	.8216257	.0759324	10.82	0.000	.672801	.9704504
gradrate						
CollegeQ	.6887297	.0608898	11.31	0.000	.5693879	.8080715
var(e.math)	4372.757	407.9178			3642.084	5250.016
var(e.verbal)	3387.774	377.0419			2723.83	4213.556
var(e.writing)	3576.065	368.9101			2921.425	4377.399
var(e.sfratio)	35.62925	4.241107			28.21528	44.99134
var(e.fulltime)	37.25718	3.771825			30.55177	45.43426
var(e.gradrate)	17.52231	2.059338			13.91722	22.06126
var(e.income)	66.77301	5.88145			56.18573	79.35528
var(e.CollegeQ)	15.37788	4.664471			8.486076	27.86671
var(hsgpa)	.0777953	.006352			.0662908	.0912964
var(Aptitude)	1132.689	336.9221			632.2927	2029.1
cov(hsgpa,Aptitude)	3.619961	.8875014	4.08	0.000	1.880491	5.359432

```
LR test of model vs. saturated: chi2(17) = 17.73            Prob > chi2 = 0.4058
```

5.1.3 Evaluate the model

- We have 8 variables in our model so we have 36 unique covariance elements, $8(9)/2 = 36$, and we are estimating 19 parameters. Therefore, the degrees of freedom of our model is 17. Because our model is over-identified, we can check the model fit.

```
. estat gof, stats(all)
```

Fit statistic	Value	Description
Likelihood ratio		
chi2_ms(17)	17.734	model vs. saturated
p > chi2	0.406	
chi2_bs(28)	706.639	baseline vs. saturated
p > chi2	0.000	
Population error		
RMSEA	0.012	Root mean squared error of approximation
90% CI, lower bound	0.000	
upper bound	0.054	
pclose	0.920	Probability RMSEA <= 0.05
Information criteria		
AIC	18327.428	Akaike´s information criterion
BIC	18397.800	Bayesian information criterion
Baseline comparison		
CFI	0.999	Comparative fit index
TLI	0.998	Tucker-Lewis index
Size of residuals		
SRMR	0.026	Standardized root mean squared residual
CD	0.841	Coefficient of determination

- We can see that our model has excellent fit, $\chi^2(17) = 17.734, p = 0.406$, RMSEA=0.012, CFI=0.999.

```
. estat eqgof
Equation-level goodness of fit
```

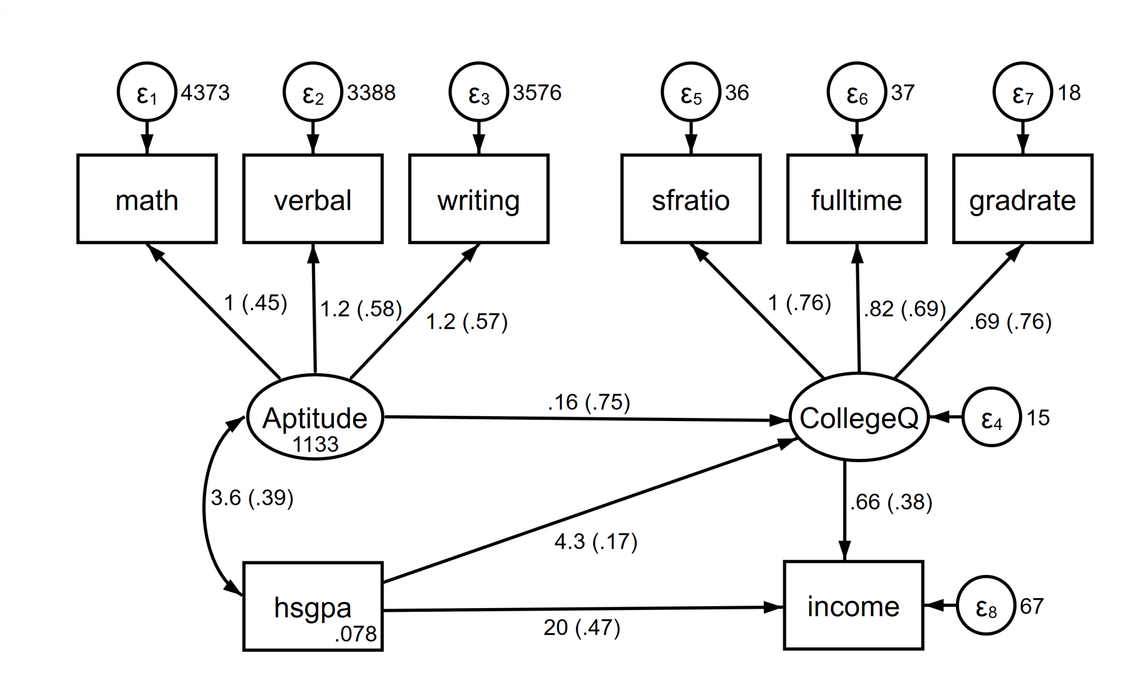
Dependent variables	Fitted	Variance Predicted	Residual	R-squared	mc	mc2
Observed						
math	5505.446	1132.689	4372.757	.2057398	.4535855	.2057398
verbal	5129.614	1741.84	3387.774	.3395655	.5827225	.3395655
writing	5283.144	1707.079	3576.065	.323118	.5684347	.323118
sfratio	84.82222	49.19298	35.62925	.5799539	.761547	.5799539
fulltime	70.46582	33.20865	37.25718	.4712731	.6864933	.4712731
gradrate	40.85693	23.33462	17.52231	.57113	.7557314	.57113
income	145.1037	78.33073	66.77301	.5398257	.7347283	.5398257
Latent						
CollegeQ	49.19298	33.8151	15.37788	.6873969	.829094	.6873969
Overall				.8412968		

```
mc  = Correlation between dependent variable and its prediction.
mc2 = mc^2 is the Bentler-Raykov squared multiple correlation coefficient.
```

- Our model explains 69% of the variance in college quality and 54% of the variance in income.
- At this point, if the model didn't fit well, you could fit each of the measurement models individually to see where the problem may lie.

5.1.4 Report and interpret the results

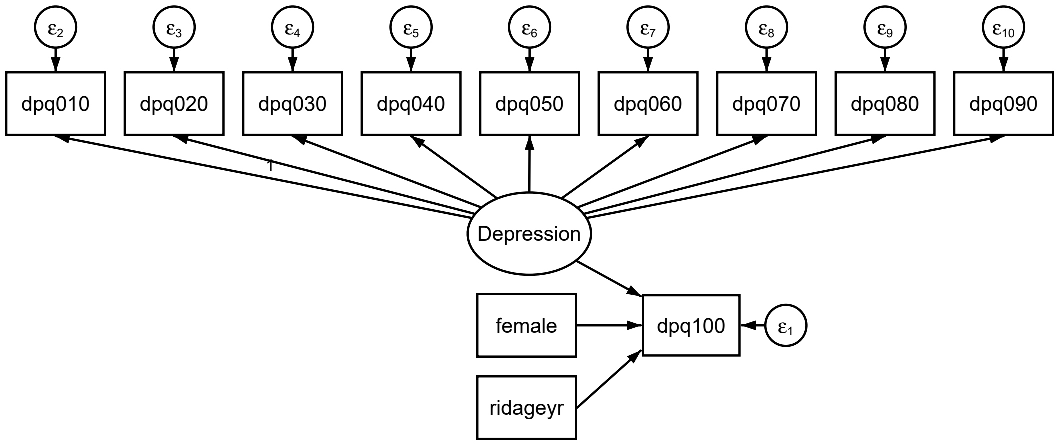
- Some of our path coefficients may be easier to interpret from the unstandardized coefficients, while others may be easier to interpret from the standardized coefficients. Let's report both on our path diagram.



- From the unstandardized results, we can see that every point increase in high school GPA corresponded to a \$20k (SE=\$2k) increase in expected future income, $z = 9.95, p < 0.001$.
- Standardized paths will be easier to interpret the measurement paths and paths to latent variables. Every standard deviation increase in aptitude corresponded to a 0.75 standard deviation increase (SE=0.075) in expected college quality. One standard deviation increase in college quality then led to a 0.38 standard deviation increase (SE=0.050) in expected income.

5.2 Exercise: Structural equation modeling

- Returning to our `nhanes` dataset,
  1. Fit our original model, this time replacing the PHQ sum score with the latent variable:



6 Multiple-group SEM

- Multiple-group SEM allows us to estimate model parameters separately across groups.
  - In measurement models, we can test whether the latent variables are measuring the same construct across groups. This is called measurement invariance.
  - In path models, we can compare the structural relationships between constructs across groups. This is a type of moderation analysis to evaluate whether the relationships among our variables change depending on group membership.
- It is highly recommended to check the measurement model for invariance before comparing the structural paths, so we'll start there in Section 6.1 and examine structural group differences in Section 6.2.
- For this section, we will use data from Marsh and Hocevar (1985) located in `sdq.dta`. This is summary data by group, meaning summary statistics for each group were separately provided and input.

```
. use sdq
(Two-factor CFA)
. ssd list
--------------------------------------------------------------------------------------------------
Group grade==1:
  Observations = 134
  Means:
    phyab1    phyab2    phyab3    phyab4   appear1   appear2   appear3   appear4  peerrel1
      8.34      8.34      8.37       8.4      7.51      7.22      7.03      7.13      8.44
  peerrel2  peerrel3  peerrel4   parrel1   parrel2   parrel3   parrel4
      7.62      7.06      7.89      9.32      9.39      8.69      9.13
  Standard deviations:
    phyab1    phyab2    phyab3    phyab4   appear1   appear2   appear3   appear4  peerrel1
       1.9      1.75      2.06      1.88       2.3      2.63      2.71      2.42      2.05
  peerrel2  peerrel3  peerrel4   parrel1   parrel2   parrel3   parrel4
      2.22      2.38      2.12      1.21      1.21      1.71      1.32
  Correlations:
    phyab1    phyab2    phyab3    phyab4   appear1   appear2   appear3   appear4  peerrel1
         1
        .5         1
       .59       .46         1
       .58       .43       .66         1
        .3       .27       .35       .46         1
       .32       .34       .38       .39       .71         1
       .38       .41       .43       .53       .68       .67         1
       .23       .29       .33       .43       .61       .63       .73         1
       .43       .32        .4       .42       .36       .34       .45       .42         1
       .38        .4       .38       .49       .53       .61       .69       .59       .59
       .27       .24       .41       .37       .43       .46       .57       .57       .61
       .43       .41       .37       .47       .51       .45       .63       .61       .59
        .2       .14       .15       .18       .22       .21       .13       .03       .15
       .29       .18       .26        .2       .25       .29       .17       .25       .35
       .37       .14       .34       .37       .34       .34       .35       .33       .42
       .13        .1       .16       .21       .33       .28       .23       .22       .23
  peerrel2  peerrel3  peerrel4   parrel1   parrel2   parrel3   parrel4
         1
       .59         1
       .58       .65         1
       .19       .12       .14         1
       .23       .23       .28       .25         1
       .36       .39       .39       .53        .5         1
       .25       .23       .28       .46       .43       .59         1
--------------------------------------------------------------------------------------------------
Group grade==2:
  Observations = 251
  Means:
    phyab1    phyab2    phyab3    phyab4   appear1   appear2   appear3   appear4  peerrel1
       8.2      8.23      8.17      8.56      7.41         7      7.17       7.4      8.81
  peerrel2  peerrel3  peerrel4   parrel1   parrel2   parrel3   parrel4
      7.94      7.52      8.29      9.35      9.13      8.67         9
  Standard deviations:
```

```
  phyab1    phyab2    phyab3    phyab4   appear1   appear2   appear3   appear4  peerrel1
    1.84      1.94      2.07      1.82      2.34      2.61      2.48      2.34      1.71

peerrel2  peerrel3  peerrel4   parrel1   parrel2   parrel3   parrel4
    1.93      2.18      1.94      1.31      1.57      1.77      1.47

Correlations:
  phyab1    phyab2    phyab3    phyab4   appear1   appear2   appear3   appear4  peerrel1
       1
     .31         1
     .52       .45         1
     .54       .46        .7         1
     .15       .33       .22       .21         1
     .14       .28       .21       .13       .72         1
     .16       .32       .35       .31       .59       .56         1
     .23       .29       .43       .36       .55       .51       .65         1
     .24       .13       .24       .23       .25       .24       .24        .3         1
     .19       .26       .22       .18       .34       .37       .36       .32       .38
     .16       .24       .36        .3       .33       .29       .44       .51       .47
     .16       .21       .35       .24       .31       .33       .41       .39       .47
     .08       .18       .09       .12       .19       .24       .08       .21       .21
     .01      -.01       .03       .02        .1       .13       .03       .05       .26
     .06       .19       .22       .22       .23       .24        .2       .26       .16
     .04       .17        .1       .07       .26       .24       .12       .26       .16

peerrel2  peerrel3  peerrel4   parrel1   parrel2   parrel3   parrel4
       1
      .5         1
     .47       .55         1
     .19       .19        .2         1
     .17       .23       .26       .33         1
     .23       .38       .24       .42        .4         1
     .22       .32       .17       .42       .42       .65         1
```

- Let's take a look at the notes.

```
. notes

_dta:
  1.  Source: Summary statistics data from Marsh, H. W., and D. Hocevar. 1985. Application of
      confirmatory factor analysis to the study of self-concept: First- and higher order factor
      models and their invariance across groups. Psychological Bulletin 97: 562-582.
      http://doi.org/10.1037/0033-2909.97.3.562.
  2.  Summary statistics based on 134 students in grade 4 and 251 students in grade 5 from Sydney,
      Australia.
  3.  Group 1 is grade 4, group 2 is grade 5.
  4.  Data collected using the Self-Description Questionnaire and includes sixteen subscales
      designed to measure nonacademic status: four intended to measure physical ability, four
      intended to measure physical appearance, four intended to measure relations with peers, and
      four intended to measure relations with parents.
```

- We would like to fit a two-factor model for appearance and peer relationships and compare these models between students in grade 4 and 5. Specifically, we have the following research questions.
  1. Are relationships with peers stronger in 5th grade than in 4th grade?
  2. Is physical appearance more strongly associated with peer relationships in 5th grade or 4th grade?
- Before we can answer these questions, we need to establish measurement invariance between the two groups. That is, is our questionnaire measuring the same constructs in the same way for both groups?

6.1 Measurement invariance

- When testing for measurement invariance, we ask (Vandenberg and Lance, 2000):
  - Do respondents from different cultures interpret a given measure in a conceptually similar manner?
  - Do rating sources define performance in similar ways when rating the same target on identical performance dimensions?
  - Are there gender, ethnic, or other individual differences that preclude responding to instruments in similar ways?

  - Does the very process of substantive interest (i.e., an intervention or experimental manipulation) alter the conceptual frame of reference against which a group responds to a measure over time?

- There are several procedures to test for measurement invariance in the literature, but they are all very similar. See Vandenberg and Lance (2000) or Leitgb et al. (2023) for a review. We will follow a four-step procedure in which we will test for four types of measurement invariance:

Table 1: Levels of measurement invariance (reproduced from Leitgb et al., 2023)

Table 1
Levels of measurement invariance.[4]

Invariance level	What it implies	Type of comparison across groups allowed	How the invariance level may be assessed
Configural invariance	The same items measuring the same constructs across groups	None	An MGCFA suggesting an acceptable fit to the data
(Full or partial) Metric invariance	The same items have the same factor loadings across groups (at least two equal factor loadings for partial metric invariance)	Unstandardized associations (covariances, unstandardized regression coefficients with other theoretical constructs of interest)	The model fit does not deteriorate considerably compared to the configural invariance model
(Full or partial) Scalar invariance	The same items have the same factor loadings *and* intercepts across groups (at least two items with equal factor loadings and intercepts for partial scalar invariance)	Unstandardized associations *and* latent means	The model fit does not deteriorate considerably compared to the (full or partial) metric invariance model
Strict invariance	The same items have the same factor loadings, intercepts, *and* error variances across groups	Unstandardized associations *and* latent means	The model fit does not deteriorate considerably compared to the (full or partial) scalar invariance model

- We will specify our model by group according to each of these types of invariance. The general procedure is to fit the model until you reach a model that doesn't fit well. Historically, this has been evaluated by comparing models in a likelihood ratio test. Some research has shown that comparing models' CFIs may have better performance when conducting measurement invariance (Cheung and Rensvold, 2002). If a model has a CFI of 0.01 less than previous model, we stop there. You can also compare models' AIC and BIC. We will show all three methods. Note that this is an ongoing research area.

- The groups will be defined by the grade variable. In Stata, we can specify group with the `group(grade)` option. Then, we can use `ginvariant()` to specify the types of parameters we would like to constrain across groups. All other variables will be estimated separately for each group. The `ginvariant()` option has the following suboptions:

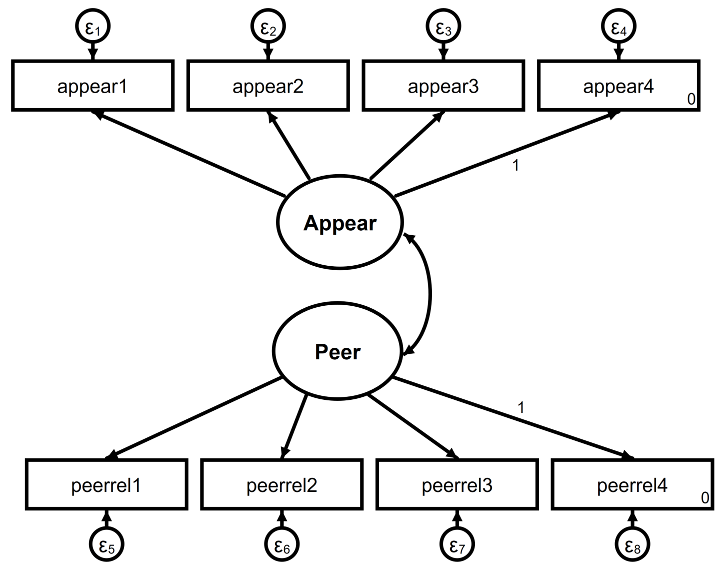
Table 2: `ginvariant()` suboptions

Option	Description
`mcoef`	measurement coefficients
`mcons`	measurement intercepts
`merrvar`	covariances of measurement errors
`scoef`	structural coefficients
`scons`	structural intercepts
`serrvar`	covariances of structural errors
`smerrcov`	covariances between structural and measurement errors
`meanex`	means of exogenous variables
`covex`	covariances of exogenous variables
`all`	all the above
`none`	none of the above

6.1.1 Specify the model

- We will be measuring physical appearance using four indicators, `appear1-appear4`, and peer relationships using four indicators, `peerrel1-peerrel4`.

- The path diagram for this model is in `sdq.stsem`.



- This is a two-factor measurement model with two latent exogenous variables and eight observed endogenous variables.
- Notice that the paths from `Appear` to `appear4` and from `Peer` to `peerrel4` have constraints on them; these are our anchor items — the items that we have reason to expect to be invariant across groups.
  - Anchor items should come from theory; how to statistically identify anchor items is an ongoing research area (for example, Chen et al., 2023).
  - The original paper on this assessment identified anchor items for us.
- The corresponding equations are:

  Measurement Model:

$$
\begin{aligned}
\texttt{appear}_\texttt{i} &= \alpha_i + \Gamma_i \texttt{Appear} + \texttt{e.appear}_\texttt{i} \\
\texttt{peerrel}_\texttt{j} &= \alpha_j + \Gamma_j \texttt{Peer} + \texttt{e.peerrel}_\texttt{j}
\end{aligned}
\tag{5}
$$

  Structural Model: (none)

  Mean Structure:

$$
mean \begin{pmatrix} \texttt{Appear} \\ \texttt{Peer} \end{pmatrix} = \begin{matrix} \kappa_1 \\ \kappa_2 \end{matrix}
$$

  Covariance Structures:

$$
var \begin{pmatrix} \texttt{Appear} \\ \texttt{Peer} \end{pmatrix} = \begin{bmatrix} \Phi_{11} & \\ \Phi_{12} & \Phi_{22} \end{bmatrix}
$$

$$
var \left(\begin{bmatrix} \texttt{e.appear}_1 \\ \texttt{e.appear}_2 \\ e.appear_3 \\ e.appear_4 \\ e.peerrel_1 \\ e.peerrel_2 \\ e.peerrel_3 \\ e.peerrel_4 \end{bmatrix} \right) = \begin{bmatrix} \Psi_{11} & & & & & & & \\ 0 & \Psi_{22} & & & & & & \\ 0 & 0 & \Psi_{33} & & & & & \\ 0 & 0 & 0 & \Psi_{44} & & & & \\ 0 & 0 & 0 & 0 & \Psi_{55} & & & \\ 0 & 0 & 0 & 0 & 0 & \Psi_{66} & & \\ 0 & 0 & 0 & 0 & 0 & 0 & \Psi_{77} & \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \Psi_{88} \end{bmatrix}
$$

  - Notice that we are now including intercepts, $\boldsymbol{\alpha}$, and means, $\boldsymbol{\kappa}$, in our model formulations.

6.1.2 Step 1: Configural invariance

- Configural invariance means that the factor structure is the same in each group: Are the same loadings positive/negative/zero? Does the factor structure fit well in both groups? This tests whether the latent variable has the same conceptual meaning in each group.
- We can fit this model in Stata using the `group(grade)` option with `ginvariant(none)` to specify that all parameters should be estimated separately for each grade. We set the factor loadings of the anchor items to 1 and their intercepts to 0. This will allow us to estimate the variances and means of the latent variables. To estimate means for both latent variables, we must add the `means()` option. Finally, we will add the `byparm` option to see the group estimates next to each other rather than in different tables and the `difficult` option to help with estimation.

```
. sem (Peer -> peerrel1 peerrel2 peerrel3 peerrel4@1) ///
>         (Appear -> appear1 appear2 appear3 appear4@1) (_cons@0 -> peerrel4 appear4), ///
>         means(Peer Appear) group(grade) ginvariant(none) byparm difficult nolog

Endogenous variables
  Measurement: peerrel1 peerrel2 peerrel3 peerrel4 appear1 appear2 appear3 appear4

Exogenous variables
  Latent: Peer Appear

Structural equation model                               Number of obs    = 385
Grouping variable: grade                                Number of groups =   2
Estimation method: ml

Log likelihood = -6125.2088

 ( 1)  [peerrel4]1bn.grade#c.Peer = 1
 ( 2)  [appear4]1bn.grade#c.Appear = 1
 ( 3)  [peerrel4]1bn.grade = 0
 ( 4)  [appear4]1bn.grade = 0
 ( 5)  [peerrel4]2.grade#c.Peer = 1
 ( 6)  [appear4]2.grade#c.Appear = 1
 ( 7)  [peerrel4]2.grade = 0
 ( 8)  [appear4]2.grade = 0
------------------------------------------------------------------------------
                |                 OIM
                | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
----------------+-------------------------------------------------------------
Measurement     |
  peerrel1      |
           Peer |
             1  |   .8608536   .1016299     8.47   0.000     .6616627    1.060044
             2  |   .7284655    .085748     8.50   0.000     .5604024    .8965285
          _cons |
             1  |   1.647865   .8170015     2.02   0.044     .0465719    3.249159
             2  |   2.771021   .7186088     3.86   0.000     1.362574    4.179469
  --------------+-------------------------------------------------------------
  peerrel2      |
           Peer |
             1  |   1.050271   .1123772     9.35   0.000     .8300159    1.270526
             2  |   .8919647   .0993817     8.98   0.000     .6971802    1.086749
          _cons |
             1  |  -.6666395   .9015634    -0.74   0.460    -2.433671    1.100392
             2  |   .5456129   .8323335     0.66   0.512    -1.085731    2.176957
  --------------+-------------------------------------------------------------
  peerrel3      |
           Peer |
             1  |   1.104391   .1147103     9.63   0.000     .8795634     1.32922
             2  |   1.180111   .1176168    10.03   0.000      .949586    1.410636
          _cons |
             1  |  -1.653649   .9220681    -1.79   0.073    -3.460869    .1535715
             2  |  -2.263118   .9839815    -2.30   0.021    -4.191687   -.3345498
  --------------+-------------------------------------------------------------
  peerrel4      |
           Peer |
           [*]  |          1  (constrained)
          _cons |
           [*]  |          0  (constrained)
  --------------+-------------------------------------------------------------
  appear1       |
         Appear |
             1  |   .9035148   .0904825     9.99   0.000     .7261723    1.080857
             2  |   1.086674   .1001424    10.85   0.000     .8903989     1.28295
```

```
           _cons |
              1  |    1.06794   .6657704     1.60   0.109    -.2369463    2.372826
              2  |  -.6313903    .753977    -0.84   0.402    -2.109158    .8463774
  ---------------+----------------------------------------------------------------
  appear2        |
          Appear |
              1  |   1.038415   .1029001    10.09   0.000     .8367341    1.240095
              2  |   1.167527   .1115815    10.46   0.000     .9488314    1.386223
           _cons |
              1  |  -.1838957   .7572859    -0.24   0.808    -1.668149    1.300357
              2  |    -1.6397   .8402607    -1.95   0.051    -3.286581    .0071807
  ---------------+----------------------------------------------------------------
  appear3        |
          Appear |
              1  |    1.20879   .1002443    12.06   0.000     1.012315    1.405266
              2  |    1.11336   .0932375    11.94   0.000     .9306178    1.296102
           _cons |
              1  |  -1.588676   .7372425    -2.15   0.031    -3.033645   -.1437069
              2  |  -1.068864   .7056349    -1.51   0.130    -2.451883    .3141551
  ---------------+----------------------------------------------------------------
  appear4        |
          Appear |
            [*]  |          1  (constrained)
           _cons |
            [*]  |          0  (constrained)
-----------------+----------------------------------------------------------------
      mean(Peer) |
              1  |       7.89   .1824555    43.24   0.000     7.532394    8.247606
              2  |       8.29   .1222075    67.84   0.000     8.050478    8.529522
    mean(Appear) |
              1  |       7.13   .2082746    34.23   0.000     6.721789    7.538211
              2  |        7.4    .147405    50.20   0.000     7.111092    7.688908
-----------------+----------------------------------------------------------------
 var(e.peerrel1) |
              1  |   2.075001   .2971312                       1.56722    2.747304
              2  |   1.853321   .1909065                      1.514504    2.267938
 var(e.peerrel2) |
              1  |   1.771553   .2933339                      1.280596    2.450734
              2  |   2.122149    .230136                      1.715803    2.624728
 var(e.peerrel3) |
              1  |   2.172223   .3420941                      1.595336    2.957716
              2  |   1.953904    .264817                       1.49809    2.548405
 var(e.peerrel4) |
              1  |   1.632327   .2637335                      1.189267    2.240449
              2  |   1.752742   .2158546                      1.376862    2.231237
  var(e.appear1) |
              1  |   2.064657   .3041667                      1.546851    2.755799
              2  |   1.931581   .2689719                      1.470224    2.537712
  var(e.appear2) |
              1  |   2.657062   .3951369                      1.985262    3.556194
              2  |   2.719127   .3481134                      2.115706     3.49465
  var(e.appear3) |
              1  |   1.586877   .3073131                      1.085677    2.319456
              2  |   2.428578   .3078832                      1.894266    3.113603
  var(e.appear4) |
              1  |    1.91007   .2946191                      1.411743    2.584301
              2  |   2.471043   .2926982                       1.95909    3.116781
       var(Peer) |
              1  |   2.828532   .5357446                       1.95136     4.10001
              2  |   1.995863   .3310647                      1.441907     2.76264
     var(Appear) |
              1  |   3.902626   .6943946                       2.75361    5.531097
              2  |   2.982742   .4747281                      2.183433    4.074661
-----------------+----------------------------------------------------------------
cov(Peer,Appear) |
              1  |   2.788274   .4724712     5.90   0.000     1.862248    3.714301
              2  |   1.549422   .2546297     6.09   0.000     1.050357    2.048487
----------------------------------------------------------------------------------
Note: [*] identifies parameter estimates constrained to be equal across groups.
LR test of model vs. saturated: chi2(38) = 122.80           Prob > chi2 = 0.0000
```

- We can store this model for comparison later. We'll also store the estimates in a matrix called `config_est` that we can use as starting values when we estimate our next model.

```
. estimates store config
```

```
. matrix config_est = e(b)
```

- Right now, we're just interested in the top part of the output to make sure that the factor loadings have similar configurations between our groups. It looks as though they do, so we have configural invariance!
- Let's start by looking at model fit by group using command `estat ggof`.

```
. estat ggof
Group-level fit statistics
```

	N	SRMR	CD	chi2	df	p>chi2
grade						
1	134	0.052	0.968	51.210	19	0.000
2	251	0.049	0.960	71.592	19	0.000

- The model fit looks comparable between the two groups. Let's look at more indices for overall model fit.

```
. estat gof, stats(all)
```

Fit statistic	Value	Description
Likelihood ratio		
chi2_ms(38)	122.802	model vs. saturated
p > chi2	0.000	
chi2_bs(56)	1537.655	baseline vs. saturated
p > chi2	0.000	
Population error		
RMSEA	0.108	Root mean squared error of approximation
90% CI, lower bound	0.087	
upper bound	0.129	
Information criteria		
AIC	12350.418	Akaike´s information criterion
BIC	12548.080	Bayesian information criterion
Baseline comparison		
CFI	0.943	Comparative fit index
TLI	0.916	Tucker-Lewis index
Size of residuals		
SRMR	0.051	Standardized root mean squared residual
CD	0.962	Coefficient of determination

```
Note: pclose is not reported because of multiple groups.
```

- Overall, the model fit looks adequate, $\chi^2(38) = 122.8, p < 0.001$; RMSEA=0.108,CFI=0.943.
- We would like to create a table containing model fit information for this model and the subsequent models we will fit. We do this using the `collect` system. We'll start by clearing any current collections, then use `collect get` to collect the results we want returned from the `estat gof` command. We tag all these results with `Model[‘‘Configural’’]`.

```
. collect clear
. collect get r(chi2_ms) r(df_ms) r(p_ms) r(rmsea) r(srmr) ///
>         r(cfi) r(tli) r(aic) r(bic), tags(Model["Configural"])
```

6.1.3 Step 2: Weak factor invariance

- Weak factor invariance means that the factor loadings (measurement coefficients) are the same in each group. This tests whether your construct is being measured in the same way across groups. If you would like to compare structural relationships with your latent variable across groups, we first need to demonstrate that we have weak factor invariance.

- We can fit this model in Stata using the `ginvariant(mcoef)` option. We use the `from()` option to use the last model's estimates as starting values.

```
. sem (Peer -> peerrel1 peerrel2 peerrel3 peerrel4@1) ///
>         (Appear -> appear1 appear2 appear3 appear4@1) (_cons@0 -> peerrel4 appear4), ///
>         means(Peer Appear) group(grade) ginvariant(mcoef) byparm from(config_est) nolog

Endogenous variables
  Measurement: peerrel1 peerrel2 peerrel3 peerrel4 appear1 appear2 appear3 appear4

Exogenous variables
  Latent: Peer Appear

Structural equation model                         Number of obs    = 385
Grouping variable: grade                          Number of groups =   2
Estimation method: ml

Log likelihood = -6129.4106

 ( 1)  [peerrel1]1bn.grade#c.Peer - [peerrel1]2.grade#c.Peer = 0
 ( 2)  [peerrel2]1bn.grade#c.Peer - [peerrel2]2.grade#c.Peer = 0
 ( 3)  [peerrel3]1bn.grade#c.Peer - [peerrel3]2.grade#c.Peer = 0
 ( 4)  [peerrel4]1bn.grade#c.Peer = 1
 ( 5)  [appear1]1bn.grade#c.Appear - [appear1]2.grade#c.Appear = 0
 ( 6)  [appear2]1bn.grade#c.Appear - [appear2]2.grade#c.Appear = 0
 ( 7)  [appear3]1bn.grade#c.Appear - [appear3]2.grade#c.Appear = 0
 ( 8)  [appear4]1bn.grade#c.Appear = 1
 ( 9)  [peerrel4]1bn.grade = 0
 (10)  [appear4]1bn.grade = 0
 (11)  [peerrel4]2.grade#c.Peer = 1
 (12)  [appear4]2.grade#c.Appear = 1
 (13)  [peerrel4]2.grade = 0
 (14)  [appear4]2.grade = 0
────────────────┬────────────────────────────────────────────────────────────────
                │                 OIM
                │ Coefficient  std. err.      z    P>|z|     [95% conf. interval]
────────────────┼────────────────────────────────────────────────────────────────
Measurement     │
  peerrel1      │
           Peer │
            [*] │   .7857895   .0659097    11.92   0.000     .6566088    .9149703
          _cons │
              1 │   2.240121   .5417164     4.14   0.000     1.178376    3.301865
              2 │   2.295805   .5569468     4.12   0.000     1.204209    3.387401
  ──────────────┼────────────────────────────────────────────────────────────────
  peerrel2      │
           Peer │
            [*] │    .962879   .0747474    12.88   0.000     .8163767    1.109381
          _cons │
              1 │    .022885   .6105801     0.04   0.970     -1.17383      1.2196
              2 │  -.0422666   .6314924    -0.07   0.947    -1.279969    1.195436
  ──────────────┼────────────────────────────────────────────────────────────────
  peerrel3      │
           Peer │
            [*] │   1.143154   .0813983    14.04   0.000     .9836162    1.302692
          _cons │
              1 │  -1.959485   .6659812    -2.94   0.003    -3.264784   -.6541859
              2 │  -1.956747   .6875035    -2.85   0.004    -3.304229   -.6092646
  ──────────────┼────────────────────────────────────────────────────────────────
  peerrel4      │
           Peer │
            [*] │          1  (constrained)
          _cons │
            [*] │          0  (constrained)
  ──────────────┼────────────────────────────────────────────────────────────────
  appear1       │
         Appear │
            [*] │   .9891779   .0664672    14.88   0.000     .8589047    1.119451
          _cons │
              1 │   .4571615   .5034424     0.91   0.364    -.5295675     1.44389
              2 │   .0900834   .5094728     0.18   0.860     -.908465    1.088632
  ──────────────┼────────────────────────────────────────────────────────────────
  appear2       │
         Appear │
            [*] │    1.09125   .0745531    14.64   0.000     .9451286    1.237371
          _cons │
              1 │  -.5606122   .5646186    -0.99   0.321    -1.667244    .5460199
              2 │   -1.07525   .5719551    -1.88   0.060    -2.196261    .0457618
```

```
-----------------+----------------------------------------------------------------
appear3          |
          Appear |
             [*] |   1.154801   .0678996    17.01   0.000     1.021721    1.287882
           _cons |
               1 |  -1.203734    .516358    -2.33   0.020    -2.215777   -.1916911
               2 |  -1.375531   .5234273    -2.63   0.009    -2.401429   -.3496319
-----------------+----------------------------------------------------------------
appear4          |
          Appear |
             [*] |          1  (constrained)
           _cons |
             [*] |          0  (constrained)
-----------------+----------------------------------------------------------------
      mean(Peer) |
               1 |       7.89   .1845991    42.74   0.000     7.528192    8.251808
               2 |       8.29   .1211811    68.41   0.000     8.052489    8.527511
    mean(Appear) |
               1 |       7.13    .206761    34.48   0.000     6.724756    7.535244
               2 |        7.4   .1481468    49.95   0.000     7.109638    7.690362
-----------------+----------------------------------------------------------------
 var(e.peerrel1) |
               1 |   2.106257   .2951676                      1.600388    2.772027
               2 |   1.832003   .1887595                      1.497004    2.241967
 var(e.peerrel2) |
               1 |   1.878607   .2928222                      1.384068    2.549848
               2 |   2.078753   .2256555                      1.680359    2.571601
 var(e.peerrel3) |
               1 |   2.090211   .3349893                      1.526763    2.861598
               2 |   2.037302   .2525425                      1.597868    2.597585
 var(e.peerrel4) |
               1 |   1.586238   .2553012                      1.157096    2.174539
               2 |   1.767036   .2075156                      1.403727    2.224375
  var(e.appear1) |
               1 |   1.981082   .3001704                      1.472074    2.666092
               2 |   2.117346   .2630426                       1.65976    2.701087
  var(e.appear2) |
               1 |   2.559455   .3845969                      1.906521    3.436003
               2 |   2.903224   .3444036                      2.300934    3.663168
  var(e.appear3) |
               1 |    1.75026   .3101521                      1.236707    2.477071
               2 |   2.251312   .2868143                      1.753855    2.889866
  var(e.appear4) |
               1 |   1.928033   .2944087                      1.429344    2.600711
               2 |   2.360937   .2710586                      1.885203    2.956723
       var(Peer) |
               1 |   2.980056   .4918101                       2.15649    4.118144
               2 |   1.918867   .2762699                       1.44708    2.544468
     var(Appear) |
               1 |   3.800482   .6058492                      2.780644    5.194359
               2 |    3.14788   .4146205                       2.43166    4.075055
-----------------+----------------------------------------------------------------
cov(Peer,Appear) |
               1 |   2.799433   .4445762     6.30   0.000     1.928079    3.670786
               2 |    1.58524   .2355576     6.73   0.000     1.123556    2.046924
----------------------------------------------------------------------------------
Note: [*] identifies parameter estimates constrained to be equal across groups.
LR test of model vs. saturated: chi2(44) = 131.21          Prob > chi2 = 0.0000
```

- Looking at the results table, we see the measurement coefficients are now the same across groups as represented by `[*]`.
- Again, we store this model's estimates.

```
. estimates store weak
. matrix weak_est = e(b)
```

- Let's see how the fit of this model compares to the previous.

```
. estat gof, stats(all)
----------------------------+----------------------------------------------------
```

```
Fit statistic            Value   Description
-------------------------------------------------------------------
Likelihood ratio
         chi2_ms(44)    131.205   model vs. saturated
            p > chi2      0.000
         chi2_bs(56)   1537.655   baseline vs. saturated
            p > chi2      0.000
-------------------------------------------------------------------
Population error
               RMSEA      0.102   Root mean squared error of approximation
 90% CI, lower bound      0.082
         upper bound      0.122
-------------------------------------------------------------------
Information criteria
                 AIC  12346.821   Akaike´s information criterion
                 BIC  12520.764   Bayesian information criterion
-------------------------------------------------------------------
Baseline comparison
                 CFI      0.941   Comparative fit index
                 TLI      0.925   Tucker-Lewis index
-------------------------------------------------------------------
Size of residuals
                SRMR      0.062   Standardized root mean squared residual
                  CD      0.961   Coefficient of determination
-------------------------------------------------------------------
Note: pclose is not reported because of multiple groups.
```

- Our model still fits well. Notice that the chi-squared test now has 44 degrees of freedom; the previous model had 38. That's because we are estimating six fewer parameters in this model. Previously, we estimated 12 factor loadings (6 for each group); now, we're estimating 6 factor loadings total. It's important to check that the difference in degrees of freedom is as you expect.

- Again, we collect model fit indices from this output, this time labeling them with tag `Model[‘‘Weak’’]`.

```
. collect get r(chi2_ms) r(df_ms) r(p_ms) r(rmsea) r(srmr) ///
>         r(cfi) r(tli) r(aic) r(bic), tags(Model["Weak"])
```

- We would also like to add the results from the likelihood ratio test comparing this model with the previous. We can compare our models in a likelihood ratio test with the `lrtest` command. The order that the models are specified does not matter.

```
. lrtest config weak
Likelihood-ratio test
Assumption: weak nested within config
 LR chi2(6) =   8.40
Prob > chi2 = 0.2100
. collect get r(chi2) r(df) r(p), tags(Model["Weak"])
```

- The LR statistic is exactly the difference in the two chi-squared values, so sometimes you will see the test referred to by this name ($\Delta\chi^2$). This is how we will label it in our table. The degrees of freedom is the difference in degrees of freedom.

- We create our table using the `collect layout` command. We put results in the rows and models in the columns. First, we use a style and label file previously created with the `collect` system.

```
. collect style use minv_sty
Collection: default
      Rows: result
   Columns: Model
   Table 1: 12 x 2
. collect label use minv_lab
. collect layout (result) (Model)
Collection: default
      Rows: result
   Columns: Model
   Table 1: 12 x 2
```

	Configural	Weak
χ2	122.8	131.2
df	38	44
p-val	<0.001	<0.001
RMSEA	0.108	0.102
SRMR	0.051	0.062
CFI	0.943	0.941
TLI	0.916	0.925
AIC	12350	12347
BIC	12548	12521
$\Delta\chi$2		8.4
Δdf		6
p-val		0.210

- The current model does not fit significantly worse than the previous model, the CFI has not substantially changed, and the AIC and BIC are smaller. Therefore, we have weak factor invariance!

6.1.4 Step 3: Strong factor invariance

- Strong factor invariance means that the factor loadings AND the measurement intercepts are the same in each group. This tests whether the expected scores of the measurement indicators are the same across groups when the latent variables are 0. If you would like to compare latent variable means across groups, we first need to demonstrate that we have strong factor invariance.
- We can fit this model in Stata using the `ginvariant(mcoef mcons)` option. We have added `mcons` to constrain the intercepts. This is the default constraint when the `group()` option is specified.

```
. sem (Peer -> peerrel1 peerrel2 peerrel3 peerrel4@1) ///
>         (Appear -> appear1 appear2 appear3 appear4@1) (_cons@0 -> peerrel4 appear4), ///
>         means(Peer Appear) group(grade) ginvariant(mcoef mcons) byparm from(weak_est) nolog

Endogenous variables
  Measurement: peerrel1 peerrel2 peerrel3 peerrel4 appear1 appear2 appear3 appear4

Exogenous variables
  Latent: Peer Appear

Structural equation model                              Number of obs    = 385
Grouping variable: grade                               Number of groups =   2
Estimation method: ml

Log likelihood = -6132.4177

 ( 1)  [peerrel1]1bn.grade#c.Peer - [peerrel1]2.grade#c.Peer = 0
 ( 2)  [peerrel2]1bn.grade#c.Peer - [peerrel2]2.grade#c.Peer = 0
 ( 3)  [peerrel3]1bn.grade#c.Peer - [peerrel3]2.grade#c.Peer = 0
 ( 4)  [peerrel4]1bn.grade#c.Peer = 1
 ( 5)  [appear1]1bn.grade#c.Appear - [appear1]2.grade#c.Appear = 0
 ( 6)  [appear2]1bn.grade#c.Appear - [appear2]2.grade#c.Appear = 0
 ( 7)  [appear3]1bn.grade#c.Appear - [appear3]2.grade#c.Appear = 0
 ( 8)  [appear4]1bn.grade#c.Appear = 1
 ( 9)  [peerrel1]1bn.grade - [peerrel1]2.grade = 0
 (10)  [peerrel2]1bn.grade - [peerrel2]2.grade = 0
 (11)  [peerrel3]1bn.grade - [peerrel3]2.grade = 0
 (12)  [peerrel4]1bn.grade = 0
 (13)  [appear1]1bn.grade - [appear1]2.grade = 0
 (14)  [appear2]1bn.grade - [appear2]2.grade = 0
 (15)  [appear3]1bn.grade - [appear3]2.grade = 0
 (16)  [appear4]1bn.grade = 0
 (17)  [peerrel4]2.grade#c.Peer = 1
 (18)  [appear4]2.grade#c.Appear = 1
 (19)  [peerrel4]2.grade = 0
 (20)  [appear4]2.grade = 0
-----------------------------------------------------------------------------
                 |                 OIM
                 | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-----------------------------------------------------------------------------
Measurement      |
  peerrel1       |
            Peer |
             [*] |   .7881482    .065323    12.07   0.000     .6601175    .9161789
           _cons |
```

[*]	2.259076	.5400828	4.18	0.000	1.200533	3.317619
peerrel2						
Peer						
[*]	.9591104	.0737563	13.00	0.000	.8145508	1.10367
_cons						
[*]	.0125904	.6078714	0.02	0.983	-1.178816	1.203996
peerrel3						
Peer						
[*]	1.1432	.0805006	14.20	0.000	.9854222	1.300979
_cons						
[*]	-1.957919	.6641913	-2.95	0.003	-3.25971	-.6561279
peerrel4						
Peer						
[*]	1	(constrained)				
_cons						
[*]	0	(constrained)				
appear1						
Appear						
[*]	.9881174	.066786	14.80	0.000	.8572193	1.119015
_cons						
[*]	.2379637	.4988898	0.48	0.633	-.7398424	1.21577
appear2						
Appear						
[*]	1.089232	.0750114	14.52	0.000	.9422124	1.236252
_cons						
[*]	-.8629376	.5606182	-1.54	0.124	-1.961729	.2358539
appear3						
Appear						
[*]	1.155363	.0680537	16.98	0.000	1.02198	1.288746
_cons						
[*]	-1.31349	.5103134	-2.57	0.010	-2.313686	-.3132941
appear4						
Appear						
[*]	1	(constrained)				
_cons						
[*]	0	(constrained)				
mean(Peer)						
1	7.892569	.1703408	46.33	0.000	7.558707	8.22643
2	8.288471	.113546	73.00	0.000	8.065925	8.511017
mean(Appear)						
1	7.271829	.1906136	38.15	0.000	6.898233	7.645424
2	7.308173	.1387199	52.68	0.000	7.036287	7.580059
var(e.peerrel1)						
1	2.105289	.2953939			1.599114	2.771687
2	1.830845	.1888212			1.495768	2.240985
var(e.peerrel2)						
1	1.885491	.292866			1.390627	2.556456
2	2.082436	.2255748			1.684098	2.574993
var(e.peerrel3)						
1	2.088163	.3347588			1.525127	2.859056
2	2.036096	.2522395			1.597158	2.595663
var(e.peerrel4)						
1	1.583933	.2549653			1.155365	2.171473
2	1.766171	.2073164			1.403192	2.223045
var(e.appear1)						
1	1.992392	.3027131			1.479272	2.6835
2	2.124025	.2643852			1.664207	2.71089
var(e.appear2)						
1	2.59801	.3916924			1.93334	3.491189
2	2.920461	.3472736			2.313314	3.686959
var(e.appear3)						
1	1.748895	.3116591			1.233323	2.479995
2	2.249298	.2875189			1.750819	2.8897
var(e.appear4)						
1	1.954001	.2996112			1.446799	2.639012
2	2.369747	.2729322			1.890889	2.969873
var(Peer)						

```
               1 |   2.983041   .4907614                          2.160837    4.118097
               2 |   1.920483   .2752821                          1.450104    2.543441
     var(Appear) |
               1 |    3.80179   .6074336                          2.779629    5.199833
               2 |   3.149011    .415582                          2.431304    4.078582
-----------------+----------------------------------------------------------------
cov(Peer,Appear) |
               1 |   2.801922   .4449025     6.30   0.000     1.929929    3.673915
               2 |   1.586567   .2356247     6.73   0.000     1.124751    2.048383
----------------------------------------------------------------------------------
Note: [*] identifies parameter estimates constrained to be equal across groups.
LR test of model vs. saturated: chi2(50) = 137.22          Prob > chi2 = 0.0000
```

- Now, the whole top half of our results table has been constrained across groups.
- Again, we store this model for comparison.

```
. estimates store strong
. matrix strong_est = e(b)
```

- Let's see how the fit of this model compares with the previous.

```
. estat gof, stats(all)
----------------------------------------------------------------------------
Fit statistic        |      Value   Description
---------------------+------------------------------------------------------
Likelihood ratio     |
         chi2_ms(50) |    137.220   model vs. saturated
            p > chi2 |      0.000
         chi2_bs(56) |   1537.655   baseline vs. saturated
            p > chi2 |      0.000
---------------------+------------------------------------------------------
Population error     |
               RMSEA |      0.095   Root mean squared error of approximation
 90% CI, lower bound |      0.076
         upper bound |      0.115
---------------------+------------------------------------------------------
Information criteria |
                 AIC |  12340.835   Akaike´s information criterion
                 BIC |  12491.059   Bayesian information criterion
---------------------+------------------------------------------------------
Baseline comparison  |
                 CFI |      0.941   Comparative fit index
                 TLI |      0.934   Tucker-Lewis index
---------------------+------------------------------------------------------
Size of residuals    |
                SRMR |      0.062   Standardized root mean squared residual
                  CD |      0.961   Coefficient of determination
----------------------------------------------------------------------------
Note: pclose is not reported because of multiple groups.
```

- Our model still fits well. Let's collect these results, then compare our models in a likelihood ratio test and collect those results. We label these results with tag `Model[‘‘Strong’’]`.

```
. collect get r(chi2_ms) r(df_ms) r(p_ms) r(rmsea) r(srmr) ///
>         r(cfi) r(tli) r(aic) r(bic), tags(Model["Strong"])
. lrtest weak strong
Likelihood-ratio test
Assumption: strong nested within weak
 LR chi2(6) =   6.01
Prob > chi2 = 0.4216
. collect get r(chi2) r(df) r(p), tags(Model["Strong"])
```

- We can see an updated version of our table using `collect preview`.

```
. collect preview
--------------------------------
       |
```

	Configural	Weak	Strong
$\chi 2$	122.8	131.2	137.2
df	38	44	50
p-val	<0.001	<0.001	<0.001
RMSEA	0.108	0.102	0.095
SRMR	0.051	0.062	0.062
CFI	0.943	0.941	0.941
TLI	0.916	0.925	0.934
AIC	12350	12347	12341
BIC	12548	12521	12491
$\Delta\chi 2$		8.4	6.0
Δdf		6	6
p-val		0.210	0.422

6.1.5 Step 4: Strict factor invariance

- Strict factor invariance means that the factor loadings, the measurement intercepts, AND the measurement error variances are the same in each group. This tests whether our constructs are being measured exactly the same across groups. If there is not strict factor variance, you can still compare groups on the latent variable, but you will need to model different amounts of error between groups. In other words, this is testing for homogeneity of variance.
- We can fit this model in Stata using the `ginvariant(mcoef mcons merrvar)` option. We have added `merrvar` to constrain the error variances.

```
. sem (Peer -> peerrel1 peerrel2 peerrel3 peerrel4@1) ///
>         (Appear -> appear1 appear2 appear3 appear4@1) (_cons@0 -> peerrel4 appear4), ///
>         means(Peer Appear) group(grade) ginvariant(mcoef mcons merrvar) byparm from(weak_est) nolo
> g
Endogenous variables
  Measurement: peerrel1 peerrel2 peerrel3 peerrel4 appear1 appear2 appear3 appear4
Exogenous variables
  Latent: Peer Appear
Structural equation model                              Number of obs    = 385
Grouping variable: grade                               Number of groups =   2
Estimation method: ml
Log likelihood = -6134.8064
 ( 1)  [peerrel1]1bn.grade#c.Peer - [peerrel1]2.grade#c.Peer = 0
 ( 2)  [peerrel2]1bn.grade#c.Peer - [peerrel2]2.grade#c.Peer = 0
 ( 3)  [peerrel3]1bn.grade#c.Peer - [peerrel3]2.grade#c.Peer = 0
 ( 4)  [peerrel4]1bn.grade#c.Peer = 1
 ( 5)  [appear1]1bn.grade#c.Appear - [appear1]2.grade#c.Appear = 0
 ( 6)  [appear2]1bn.grade#c.Appear - [appear2]2.grade#c.Appear = 0
 ( 7)  [appear3]1bn.grade#c.Appear - [appear3]2.grade#c.Appear = 0
 ( 8)  [appear4]1bn.grade#c.Appear = 1
 ( 9)  [/]var(e.peerrel1)#1bn.grade - [/]var(e.peerrel1)#2.grade = 0
 (10)  [/]var(e.peerrel2)#1bn.grade - [/]var(e.peerrel2)#2.grade = 0
 (11)  [/]var(e.peerrel3)#1bn.grade - [/]var(e.peerrel3)#2.grade = 0
 (12)  [/]var(e.peerrel4)#1bn.grade - [/]var(e.peerrel4)#2.grade = 0
 (13)  [/]var(e.appear1)#1bn.grade - [/]var(e.appear1)#2.grade = 0
 (14)  [/]var(e.appear2)#1bn.grade - [/]var(e.appear2)#2.grade = 0
 (15)  [/]var(e.appear3)#1bn.grade - [/]var(e.appear3)#2.grade = 0
 (16)  [/]var(e.appear4)#1bn.grade - [/]var(e.appear4)#2.grade = 0
 (17)  [peerrel1]1bn.grade - [peerrel1]2.grade = 0
 (18)  [peerrel2]1bn.grade - [peerrel2]2.grade = 0
 (19)  [peerrel3]1bn.grade - [peerrel3]2.grade = 0
 (20)  [peerrel4]1bn.grade = 0
 (21)  [appear1]1bn.grade - [appear1]2.grade = 0
 (22)  [appear2]1bn.grade - [appear2]2.grade = 0
 (23)  [appear3]1bn.grade - [appear3]2.grade = 0
 (24)  [appear4]1bn.grade = 0
 (25)  [peerrel4]2.grade#c.Peer = 1
 (26)  [appear4]2.grade#c.Appear = 1
 (27)  [peerrel4]2.grade = 0
 (28)  [appear4]2.grade = 0
------------------------------------------------------------------------------
                 |                 OIM
                 | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
```

Measurement						
peerrel1						
Peer						
[*]	.790374	.0648726	12.18	0.000	.6632261	.9175219
_cons						
[*]	2.239057	.5360408	4.18	0.000	1.188436	3.289678
peerrel2						
Peer						
[*]	.9550826	.0735518	12.99	0.000	.8109238	1.099242
_cons						
[*]	.0439556	.6071758	0.07	0.942	-1.146087	1.233998
peerrel3						
Peer						
[*]	1.142014	.0802919	14.22	0.000	.984645	1.299384
_cons						
[*]	-1.94841	.6628425	-2.94	0.003	-3.247558	-.6492626
peerrel4						
Peer						
[*]	1	(constrained)				
_cons						
[*]	0	(constrained)				
appear1						
Appear						
[*]	.9919702	.0670982	14.78	0.000	.8604602	1.12348
_cons						
[*]	.1974454	.5013822	0.39	0.694	-.7852456	1.180136
appear2						
Appear						
[*]	1.092433	.0758459	14.40	0.000	.943778	1.241088
_cons						
[*]	-.9047742	.5667847	-1.60	0.110	-2.015652	.2061034
appear3						
Appear						
[*]	1.149323	.0683129	16.82	0.000	1.015432	1.283214
_cons						
[*]	-1.275713	.5120066	-2.49	0.013	-2.279227	-.2721982
appear4						
Appear						
[*]	1	(constrained)				
_cons						
[*]	0	(constrained)				
mean(Peer)						
1	7.891786	.1711454	46.11	0.000	7.556347	8.227225
2	8.289046	.1132409	73.20	0.000	8.067098	8.510994
mean(Appear)						
1	7.284729	.1902374	38.29	0.000	6.911871	7.657588
2	7.317396	.138368	52.88	0.000	7.046199	7.588592
var(e.peerrel1)						
[*]	1.923141	.1618084			1.630771	2.267927
var(e.peerrel2)						
[*]	2.014873	.1824748			1.687172	2.406222
var(e.peerrel3)						
[*]	2.045882	.2045142			1.681865	2.488685
var(e.peerrel4)						
[*]	1.698119	.1648007			1.403977	2.053886
var(e.appear1)						
[*]	2.070261	.2048242			1.705337	2.513275
var(e.appear2)						
[*]	2.803378	.2685355			2.323513	3.382348
var(e.appear3)						
[*]	2.085094	.2231078			1.69062	2.571611
var(e.appear4)						
[*]	2.226312	.2092069			1.851819	2.67654
var(Peer)						
1	2.991798	.4945357			2.16387	4.136504
2	1.928961	.2748483			1.458948	2.550392


```
     var(Appear)
               1      3.735291   .6054324                             2.718679    5.132051
               2      3.172977   .4179768                              2.45097    4.107673
-------------------------------------------------------------------------------------------
cov(Peer,Appear)
               1      2.771789   .4447477     6.23   0.000              1.9001    3.643479
               2      1.592103   .2356791     6.76   0.000             1.13018    2.054025
-------------------------------------------------------------------------------------------
Note: [*] identifies parameter estimates constrained to be equal across groups.
LR test of model vs. saturated: chi2(58) = 142.00           Prob > chi2 = 0.0000
```

- Now, the entire measurement model has been constrained across groups.
- Let's see how the fit of this model compares with the previous and collect those results, labeling them with tag `Model[''Strict'']`.

```
. estat gof, stats(all)

----------------------------------------------------------------------------
Fit statistic               |      Value   Description
----------------------------+-----------------------------------------------
Likelihood ratio            |
         chi2_ms(58)        |    141.997   model vs. saturated
            p > chi2        |      0.000
         chi2_bs(56)        |   1537.655   baseline vs. saturated
            p > chi2        |      0.000
----------------------------+-----------------------------------------------
Population error            |
               RMSEA        |      0.087   Root mean squared error of approximation
 90% CI, lower bound        |      0.069
         upper bound        |      0.105
----------------------------+-----------------------------------------------
Information criteria        |
                 AIC        |  12329.613   Akaike´s information criterion
                 BIC        |  12448.210   Bayesian information criterion
----------------------------+-----------------------------------------------
Baseline comparison         |
                 CFI        |      0.943   Comparative fit index
                 TLI        |      0.945   Tucker-Lewis index
----------------------------+-----------------------------------------------
Size of residuals           |
                SRMR        |      0.064   Standardized root mean squared residual
                  CD        |      0.962   Coefficient of determination
----------------------------------------------------------------------------
Note: pclose is not reported because of multiple groups.
. collect get r(chi2_ms) r(df_ms) r(p_ms) r(rmsea) r(srmr) ///
>         r(cfi) r(tli) r(aic) r(bic), tags(Model["Strict"])
```

- RMSEA and CFI still indicate an adequate fit, while the chi-squared test does not.
- We can compare our models in a likelihood ratio test with the `lrtest` command. We use a period (.) to denote the most recent model in memory. We'll also collect these results.

```
. lrtest strong .
Likelihood-ratio test
Assumption: . nested within strong
 LR chi2(8) =   4.78
Prob > chi2 = 0.7811
. collect get r(chi2) r(df) r(p), tags(Model["Strict"])
```

- Again, we can see our updated table using `collect preview`.

```
. collect preview
```

	Configural	Weak	Strong	Strict
χ2	122.8	131.2	137.2	142.0
df	38	44	50	58
p-val	<0.001	<0.001	<0.001	<0.001

RMSEA	0.108	0.102	0.095	0.087
SRMR	0.051	0.062	0.062	0.064
CFI	0.943	0.941	0.941	0.943
TLI	0.916	0.925	0.934	0.945
AIC	12350	12347	12341	12330
BIC	12548	12521	12491	12448
$\Delta\chi2$		8.4	6.0	4.8
Δdf		6	6	8
p-val		0.210	0.422	0.781

- According to the LRT, CFI, AIC, and BIC, we have strict measurement invariance! This will allow us to make any structural comparisons we want between groups.

6.2 Structural invariance

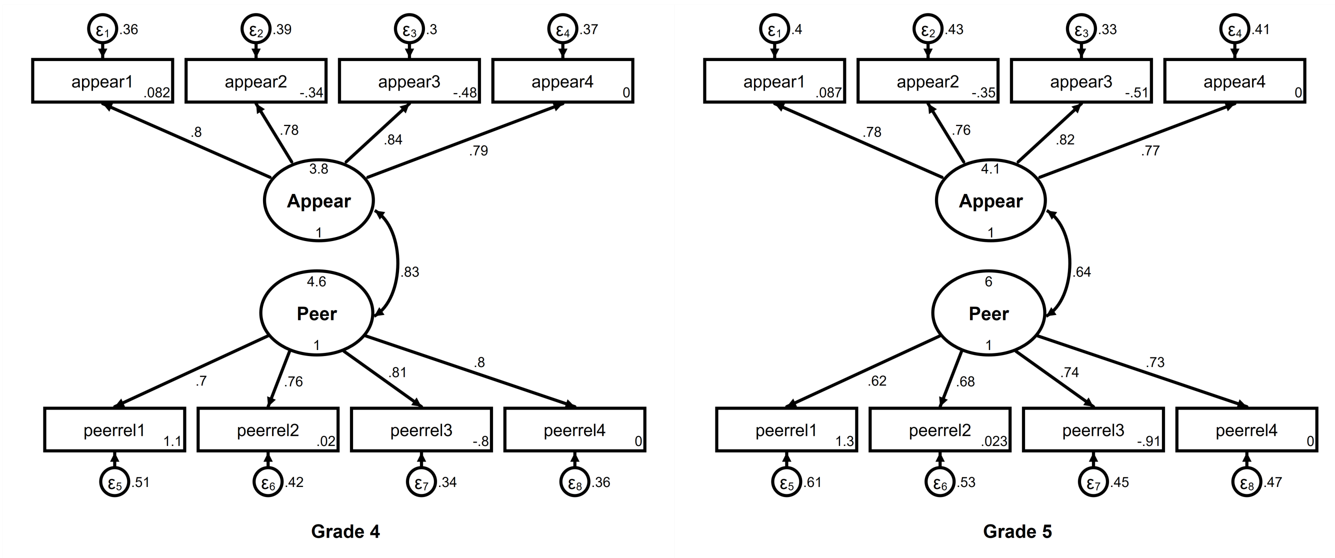
- Once measurement invariance has been established, we can compare group means and structural paths. We can test for invariance of all parameters across groups using `estat ginvariant`.

```
. estat ginvariant
Tests for group invariance of parameters
```

	Wald test chi2	df	P>chi2	Score test chi2	df	P>chi2
Measurement						
peerrel1						
Peer	.	.	.	0.033	1	0.8554
_cons	.	.	.	0.173	1	0.6772
peerrel2						
Peer	.	.	.	0.374	1	0.5410
_cons	.	.	.	0.202	1	0.6528
peerrel3						
Peer	.	.	.	0.070	1	0.7915
_cons	.	.	.	0.003	1	0.9597
peerrel4						
Peer	.	.	.	0.020	1	0.8865
_cons	.	.	.	0.001	1	0.9811
appear1						
Appear	.	.	.	0.422	1	0.5159
_cons	.	.	.	0.981	1	0.3221
appear2						
Appear	.	.	.	2.146	1	0.1429
_cons	.	.	.	2.633	1	0.1046
appear3						
Appear	.	.	.	0.205	1	0.6508
_cons	.	.	.	0.651	1	0.4198
appear4						
Appear	.	.	.	2.558	1	0.1097
_cons	.	.	.	2.903	1	0.0884
mean(Peer)	4.381	1	0.0363	.	.	.
mean(Appear)	0.023	1	0.8796	.	.	.
var(e.peerrel1)	.	.	.	0.624	1	0.4297
var(e.peerrel2)	.	.	.	0.315	1	0.5748
var(e.peerrel3)	.	.	.	0.001	1	0.9756
var(e.peerrel4)	.	.	.	0.366	1	0.5449
var(e.appear1)	.	.	.	0.260	1	0.6102
var(e.appear2)	.	.	.	0.661	1	0.4161
var(e.appear3)	.	.	.	1.643	1	0.2000
var(e.appear4)	.	.	.	1.236	1	0.2663
var(Peer)	4.781	1	0.0288	.	.	.
var(Appear)	0.832	1	0.3618	.	.	.


```
cov(Peer,Appear)|      6.343          1    0.0118           .           .         .
```

- We can see that mean peer relationship is higher in grade 5 than in grade 4, $\chi^2(1) = 4.38, p = 0.036$. We also see that the covariance between peer relationships and physical appearance is higher in grade 4 than in grade 5, $\chi^2(1) = 6.34, p = 0.012$.
- We can report estimates for each group by saving path diagrams for each, labeling them within the SEM Builder, and then combining them (or putting them side by side) in our report. Here, we report standardized coefficients to see the correlation between constructs.



- Physical appearance is higher in 5th grade, but, contrary to our hypothesis, we can actually see that physical appearance and peer relationships are more highly associated in 4th grade than in 5th.

6.3 Exercise: Multiple-group analysis

- Returning to our `nhanes` dataset,
  1. Does the PHQ inventory demonstrate measurement invariance between male and female respondents? Use the first item as the anchor.
  2. Is there a gender difference in the effect that depression has on the difficulty that it causes (`dpq100`)?

References

Bollen, K. A. (2002). “Latent Variables in Psychology and the Social Sciences”. In: *Annual Review of Psychology* 53.1, 605–634. ISSN: 0066-4308, 1545-2085. `https://doi.org/10.1146/annurev.psych.53.100901.135239`.

Bollen, Kenneth A and Shawn Bauldry (2011). “Three Cs in measurement models: Causal indicators, composite indicators, and covariates.” In: *Psychological methods* 16.3, 265.

Cheung, G. W. and R. B. Rensvold (2002). “Evaluating goodness-of-fit indexes for testing measurement invariance”. In: *Structural equation modeling* 9.2, 233–255.

Leitgb, Heinz et al. (2023). “Measurement invariance in the social sciences: Historical development, methodological challenges, state of the art, and future perspectives”. In: *Social Science Research* 110. Publisher: Elsevier1[]Publisher: Elsevier, 102805.

Marsh, Herbert W and Dennis Hocevar (1985). “Application of confirmatory factor analysis to the study of self-concept: First-and higher order factor models and their invariance across groups.” In: *Psychological bulletin* 97.3. Publisher: American Psychological Association1[]Publisher: American Psychological Association, 562.

Rydstedt, Leif W et al. (2013). “The conceptual roles of negative and positive affectivity in the stressor-strain relationship”. In: *Europe’s Journal of Psychology* 9.1, 93–103.

Skrondal, A and S Rabe-Hesketh (2004). *Generalized latent variable modeling: Multilevel, longitudinal, and structural equation models*. Crc Press, 2004.

Vandenberg, Robert J and Charles E Lance (2000). “A review and synthesis of the measurement invariance literature: Suggestions, practices, and recommendations for organizational research”. In: *Organizational research methods* 3.1, 4–70.

