Structural Equation Modeling for Ordinal Data

Chong Xing[1], Paul Johnson[1] & others[2]

[1]Center for Research Methods and Data Analysis
[2]Meaghan Sullivan, Ben Kite, and others

2019

CENTER FOR RESEARCH METHODS & DATA ANALYSIS
College of Liberal Arts & Sciences

Outline

Goals of this Session

During this session:

- Understand the non-normality issues in SEM
- SEM estimation
- Focus on ordinal data

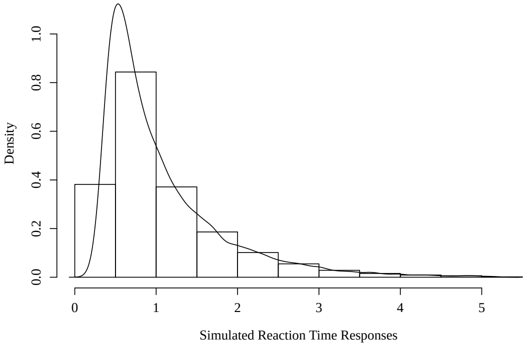
Outline

Outline

Meaning 1: Nonsymmetric, skewed distribution

"Non-normal data": sample distribution is not unimodal and symmetric.

- For instance, observations are often skewed in reaction time (RT) studies.



Meaning 1: Nonsymmetric, skewed distribution ...

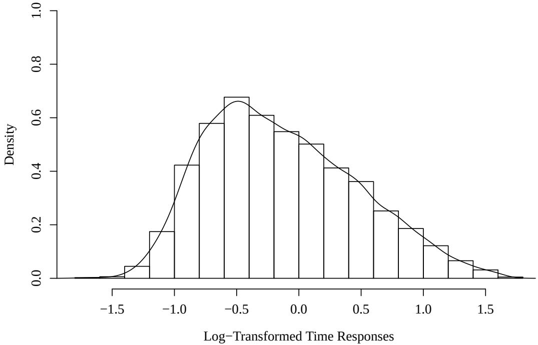
- E.g., Correll et al. (2002) conducted experimental studies on the reaction time for shooting decisions.
- Sometimes projects mathematically transform variables to make them "more normal"

Results and Discussion

To analyze the resulting reaction times, we excluded all trials on which the participant had either timed-out (i.e., failed to make a decision in the allotted 850-ms window) or made an incorrect response (e.g., shooting a target holding a non-gun). This resulted in the exclusion of data from 7% of the trials across participants, with a maximum of 20% of the trials for any one participant. Response latencies on the remaining trials were log-transformed and then averaged within subject across trials occurring in the same cell of the 2×2 within-subject research design. An analysis of variance (ANOVA) of the resulting mean latencies was then conducted, treating Target Ethnicity (White vs. African American) and Object Type (gun versus no gun) as within-subject factors.

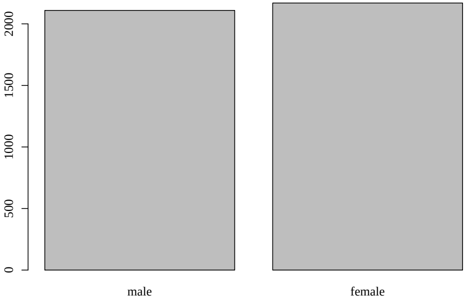
Meaning 1: Nonsymmetric, skewed distribution ...

- In our example data, the log-transformed data is slightly more like a Normal distribution.



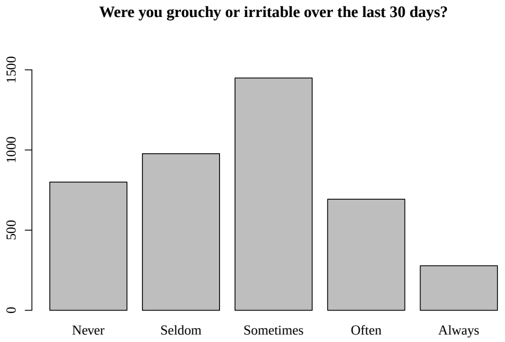
Meaning 2: Categorical Data

- **categorical variables** are discrete groupings of observations
- nominal: unordered categories, such as gender



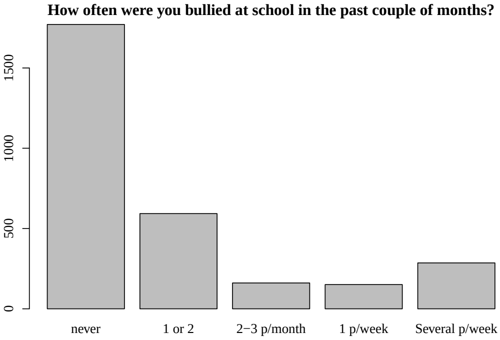
- ordinal variables: may appear unimodal, symmetric.

Meaning 2: Categorical Data ...



- But ordinal measures are inherently non-normal by nature (discrete, non-numeric; O'Brien, 1985).

Meaning 2: Categorical Data ...



- Models providing alternatives to the normal/numeric treatment
  - For **non-normal numeric data**, variants of the generalized linear model. These are not our primary focus today
  - For ordinal **categorical data**, new methods are more well-suited to the data. This is the focus of the remainder of this presentation.

Outline

Some Published Work on the Big Five Personality

Strongly Disagree	Disagree	Undecided	Agree	Strongly Agree
1	2	3	4	5

- A variable scored {1,2,3,4,5} cannot be "Normal". A normal variable is a floating point number, {1,2,3,4,5} are not.
- Nevertheless, it is common in many fields to treat them like numbers

Measures.

Big Five Inventory. As described above, the original Big Five Inventory (BFI; John, Donahue, & Kentle, 1991; see John et al., 2008) was developed to measure the prototypical features of each Big Five domain. Its 44 items are short, descriptive phrases that respondents rate on a 5-point scale ranging from *disagree strongly* to *agree strongly*. A total of 641 ESCS members completed the BFI; in this sample, the domain scales' alpha reliabilities were .86 for Extraversion, .82 for Agreeableness, .83 for Conscientiousness, .85 for Negative Emotionality, and .84 for Open-Mindedness.

(Soto & John, 2017a)

Treating a Likert Scale as Numbers

Strongly Disagree	Disagree	Undecided	Agree	Strongly Agree
1	2	3	4	5

- The original SEM was designed for numeric data.
- It was standard to treat "Likert-type" variables as if they were normally distributed.

> To examine the multidimensional structure of the BFI-2-S and BFI-2-XS, we submitted each short form's items to a random intercept EFA that extracted and varimax-rotated five factors, using maximum likelihood estimation. Random intercept EFA (Aichholzer, 2014) is a procedure for examining the multidimensional structure of an item set while using a method factor to model individual differences in acquiescent responding, thereby minimizing the negative structural effects of acquiescence variance (Rammstedt & Farmer, 2013; Soto et al., 2008). For both short

(Soto & John, 2017b)

Treating a Likert Scale as Numbers ...

- In the 1980s, alternative methods became available, including "robust" standard errors and re-scaled χ^2 statistics, and wholly new estimators.

> Prior to analysis, all items were checked for non-normality. Skewness and kurtosis of the data were −1.18 to 0.25 (kurtosis) and −0.87 to 0.95 (skeweness), showing that items were mainly close to normal. Because population weights were used to draw highly precise conclusions about the population, a restricted maximum likelihood (MLR) estimator was applied (Muthén & Muthén, 1998–2013). Factor models were separately fitted for each of the Big Five traits.

Another Big Five study (Göllner et al., 2017)

- Under what conditions is it reasonable to treat an ordinal variable as if it were a numeric?

The Long-Lasting "Interval-Ordinal Debate"

- Suppose data is a reflection of a "rounding" process, in which thresholds are imposed that separate responses into bins (as described in previous set of notes). The data may appear to be unimodal-symmetric.
- What should we do?

There are two sides in the debate

Claim 1: Ordered-categorical data should be analyzed only using methods developed for categorical data (e.g., WLSMV).
Claim 2: Ordered-categorical data can be analyzed using methods developed for continuous data (e.g., ML or MLR), **if...**

Support for Claim 1:

1. Categorical model has unique theoretical features (thresholds)
2. Less biased (more accurate) estimates
3. More valid statistical inferences

The Long-Lasting "Interval-Ordinal Debate" ...

Support for Claim 2:

Probably including

1. The numeric software is faster, simpler (estimates fewer coefficients)
2. Can view 1-2-3-4-5 data as "rounded" numbers.
3. **If** there are more than five response options, the data appear to be "normal enough"
4. Results are same either way, so we use the more familiar method
5. There is no suitable software for discrete data that tolerates
    - small sample size,
    - missing data

Why (or when) ordinal data should not be treated as if it were numeric

Support 1a. - Treating ordinal data as numeric can lead to model misspecifications

> *"With five-category Likert scales, the [indicator] variables are often scored by consecutive integers, say 0, 1, 2, 3, 4. The analysis then proceeds as if these [indicator] variables are interval scaled ....* **Strictly speaking, the model cannot be correct unless there are infinitely many categories for each** *[indicator variable] ..."* .
>
> *[For indicator variables based on Likert scales] with extreme skews it should further be noted that the ordinary measures of association, covariances and Pearson product moment correlations, my not be suitable. Not only the [indicator variables] discrete, but they are also limited in range with strong censoring.* **In such cases, a linear model for [ordinal data] may be unrealistic, and a non-linear model ... may be more appropriate**" *(B. Muthén & Kaplan, 1985, p. 172).*

Why (or when) ordinal data should not be treated as if it were numeric ...

Support 1b. - Applying modeling techniques developed for numeric data to ordinal data can lead to incorrect parameter and global model fit estimates (Beauducel & Herzberg, 2006; Flora & Curran, 2004a).

- Spuriously inflated model χ^2values (i.e., over-rejection of solutions)
- Modest underestimation of fit indices (e.g., TLI, CFI)
- moderate to severe underestimation of the standard errors of the parameter estimates (inflated Type I error)
- These issues are exacerbated when the sample size is small.

Why (or when) ordinal data should not be treated as if it were numeric ...

Support 1c. - Inferences are made from valid models that align with data

> Second, this study replicated previous results that factor loadings are typically underestimated by MLR but are essentially unbiased with WLSMV (Beauducel & Herzberg, 2006; DiStefano, 2002; Flora & Curran, 2004). Interestingly, a clear superiority of WLSMV over MLR in factor loading estimates was found in this study, irrespective of the number of categories. This study also revealed that the factor loadings obtained by WLSMV were more precise and accurate than those obtained by MLR when the latent normality assumption was moderately violated. Generally speaking, WLSMV was preferable to MLR across most of the conditions observed in this study, given its properties of being less biased and having small sampling variation in estimating factor loadings.

WLSMV vs. MLR (Li, 2016)

Why (or when) ordinal data should not be treated as if it were numeric ... ...

Support 1d. - Treating categorical as continuous is fundamentally wrong

> *"The usual numerical values assigned to these dichotomous variables are the ad hoc numbers '1' and '0', or '1' and '2'. Hence, the basic assumption that the data come from a continuous normal distribution* **is clearly violated***. In fact, even for polytomous variables that are closer to continuous variables than dichotomous variables, ignoring the discrete nature of the data* **leads to erroneous results***".*

BSEM with dichotomous variables (Lee & Song, 2003, p. 3074)

> *"In behavioral and social sciences reerach, many variables are only observed in dichotomous or polytomous form. Examples of such variables are attitude items, performance items, rating scales, etc. When analyzing this kind of data,* **a common approach used by nonrigorous statisticians** *is to assign integral values to the categories and proceed as if the data are measured on an interval scale".*

Why (or when) ordinal data should not be treated as if it were numeric ... ...

BSEM with polytomous variables (Song & Lee, 2002, p. 453)

Why (or when) ordinal data can be treated as if is numeric

If there are more categories (Bollen & Barb, 1981, p. 234) correlations are less incorrect

Why (or when) ordinal data can be treated as if is numeric ...

AMERICAN SOCIOLOGICAL REVIEW

Table 1. Comparison of Mean Correlation Coefficients for Original and Collapsed Variables

Number of Categories	Mean Collapsed Correlation				
2	0.117	0.264	0.414	0.579	0.719
3	0.149	0.302	0.445	0.617	0.727
4	0.165	0.336	0.504	0.669	0.774
5	0.182	0.359	0.533	0.712	0.806
6	0.180	0.371	0.554	0.733	0.833
7	0.192	0.380	0.563	0.750	0.850
8	0.193	0.384	0.571	0.760	0.860
9	0.194	0.389	0.577	0.771	0.870
10	0.197	0.391	0.581	0.772	0.875
Mean Original Correlation	0.203	0.404	0.599	0.796	0.901

Based on 50 samples of 500 observations each.

Why (or when) ordinal data can be treated as if is numeric ...

WLSM and WLSMV also reported sensitivity to extreme levels of nonnormality, but generally, with more categories (five, seven) and smaller sample sizes (200, 400). Under these conditions, WLSMV reported greater numbers of replications that did not converge than with the two- or three-category conditions. With many categories and item-level nonnormality, a contingency table between two observed variables could result in empty cells. The *Mplus* Web site noted the sensitivity of categorical estimators to empty cells and the software has implemented procedures to assist with categorical estimation beginning with *Mplus* version 4.0 (http://www.statmodel.com); however, some sensitivity to these conditions remains.

DiStefano and Morgan (2014, p. 430); also see Flora and Curran (2004a)

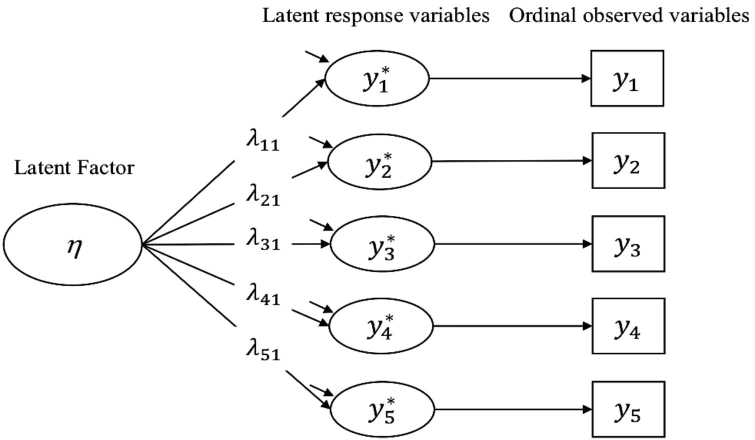
More serious issue: missing data treatment

- Categorical estimators (e.g., DWLS, WLSMV) require pairwise complete covariance matrix, which causes some distortion.
- Full information maximum likelihood (FIML; Arbuckle,1996) not universally available or practical for categorical variables.

Outline

Conceptual Framework: Thresholds

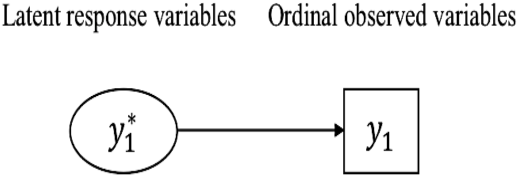
1. There is an "intermediate" latent variable y_1^* which is revealed to us as a categorical outcome y_1.



Conceptual Framework: Thresholds ...

Flora and Curran (2004b, p. 473)

2. Assume (y_j^*) is a normally distributed gives rise to an ordinal observation y_j that can fall into a discrete category (c; e.g., $c = 1 =$ "strongly disagree")

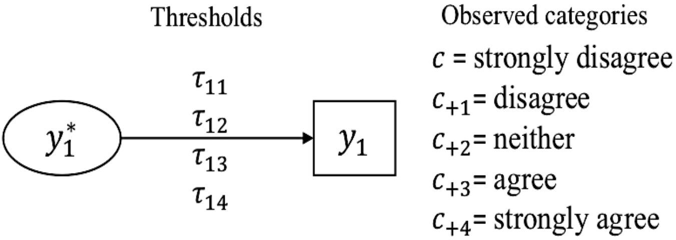


3. Assume there are threshold parameters (τ_j; tau). As y_j^* rises, it surpasses thresholds that separate the categories (e.g., from "strongly disagree" to "disagree")

$$y_j = c,\ if \tau_c < y_j^* < \tau_{c+1}$$

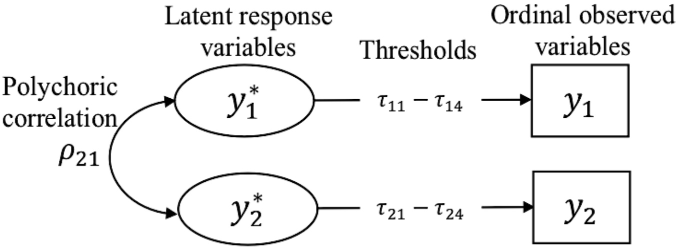
Conceptual Framework: Thresholds

Strongly Disagree	Disagree	Undecided	Agree	Strongly Agree
1	2	3	4	5



- A polychoric correlation (ρ) attempts to measure the correlation between 2 normally distributed variables that "underlie" 2 categorical variables.

Conceptual Framework: Thresholds ...



Polychoric correlations

- SEM software is designed to align covariance (or correlation) matrices
- The earliest strategies for categorical data used a polychoric correlation based on a threshold model to compare observed and theoretical covariance.
- polychoric correlations will replace the usual Pearson product-moment correlation matrix in the usual ML estimation for SEM

$$\Sigma(\theta) = \Lambda\Psi\Lambda' + \Theta_{\epsilon}$$

This substitution approach will generally yield consistent parameter estimates.

- Robust standard errors and mean/variance adjusted Chi-square statistics will be needed (Hence, the model is sometimes referred to as "WLSMV"", rather than just "WLS").

WLS estimators for categorical data

1. WLS - fully weighted least squares fitting function

$$F_{WLS} = [s - \sigma(\theta)]' W^{-1} [s - \sigma(\theta)]$$

   1. s is a vector of sample statistics (i.e., polychoric correlations and threshold estimates)
   2. $\sigma(\theta)$ is the model-implied vector of population elements in $\Sigma(\theta)$
   3. W is weight matrix.
2. Diagonally weighted least squares (developed to facilitate model estimation)
   1. DWLS - diagonal weighted least squares

$$F_{DWLS} = [s - \sigma(\theta)]' V^{-1} [s - \sigma(\theta)]$$

   2. V is a diagonal matrix (has 0's on all elements except main diagonal)
   3. Once parameter estimates are obtained using the DWLS fitting function, full W matrix used to calculate standard errors
3. WLSMV - weighted least squares-mean adjusted

WLS estimators for categorical data ...

1. In addition to using a diagonal weight matrix (V) and standard error corrections, Muthen(1993) developed a robust goodness-of-fit test via calculation of a mean- and variance-adjusted chi-square test statistics.
2. Calculation of WLSMV goodness-of-fit also involves the full W matrix, however it does not involve inversion.

Summary

1. Threshold estimates link observed ordinal responses to latent response variables
2. Polychoric correlations and full weighted least squares incorporate latent response variables into traditional SEM (common factor) framework
3. DWLS facilitates model estimations and provides corrected parameter standard errors
4. WLSMV provides more accurate goodness-of-fit statistics

Outline

The HBSC data

The demonstration data - a subset of HBSC Health Behaviour in School-Aged Children

```
str(hbsc)
```

```
'data.frame': 4284 obs. of  89 variables:
 $ stud_id : int  1 3 4 5 7 10 15 16 17 20 ...
 $ schl_id : int  101 66 170 169 58 109 180 66 61 24 ...
 $ Gender  : Factor w/ 2 levels "male","female": 1 1 1 2 2 1 2 2 1 1 ...
 $ Age     : int  13 13 12 12 13 11 12 12 11 12 ...
 $ Grade   : int  7 7 6 6 7 6 6 7 6 6 ...
 $ body1_o : Ord.factor w/ 5 levels "SA"<"A"<"U"<"D"<..: 4 4 NA NA 4 3
     5 4 NA NA ...
 $ body2_o : Ord.factor w/ 5 levels "SA"<"A"<"U"<"D"<..: 4 4 NA NA 4 2
     1 4 NA NA ...
 $ body3_o : Ord.factor w/ 5 levels "SA"<"A"<"U"<"D"<..: 5 5 NA NA 4 3
     5 3 NA NA ...
 $ body4_o : Ord.factor w/ 5 levels "SA"<"A"<"U"<"D"<..: 5 4 NA NA 4 3
     4 3 NA NA ...
 $ body5_o : Ord.factor w/ 5 levels "SA"<"A"<"U"<"D"<..: 5 5 NA NA 5 3
     5 3 NA NA ...
 $ phys1_o : Ord.factor w/ 5 levels "EveryDay"<"EveryWeek"<..: 4 1 5 5
     5 5 4 5 5 3 ...
```

The HBSC data ...

```
$ phys2_o : Ord.factor w/ 5 levels "EveryDay"<"EveryWeek"<..: 5 4 5 5
    5 2 4 5 5 4 ...
$ phys3_o : Ord.factor w/ 5 levels "EveryDay"<"EveryWeek"<..: 5 2 5 1
    5 2 2 3 5 5 ...
$ phys4_o : Ord.factor w/ 5 levels "EveryDay"<"EveryWeek"<..: 5 5 5 5
    5 1 5 5 5 5 ...
$ phys5_o : Ord.factor w/ 5 levels "EveryDay"<"EveryWeek"<..: 5 4 5 4
    4 1 4 5 5 4 ...
$ phys6_o : Ord.factor w/ 5 levels "EveryDay"<"EveryWeek"<..: 5 2 5 4
    4 1 3 4 5 5 ...
$ phys7_o : Ord.factor w/ 5 levels "EveryDay"<"EveryWeek"<..: 5 1 5 5
    3 1 3 1 5 5 ...
$ phys8_o : Ord.factor w/ 5 levels "EveryDay"<"EveryWeek"<..: 5 5 5 5
    5 1 4 5 5 4 ...
$ depre1_o: Ord.factor w/ 5 levels "Never"<"Seldom"<..: 2 2 3 3 2 3 1
    3 1 1 ...
$ depre2_o: Ord.factor w/ 5 levels "Never"<"Seldom"<..: 2 2 4 5 3 4 4
    3 1 2 ...
$ depre3_o: Ord.factor w/ 5 levels "Never"<"Seldom"<..: 1 5 2 3 2 4 1
    1 1 1 ...
$ depre4_o: Ord.factor w/ 5 levels "Never"<"Seldom"<..: 1 2 1 5 1 1 3
    1 3 2 ...
$ depre5_o: Ord.factor w/ 5 levels "Never"<"Seldom"<..: 2 2 1 3 3 1 3
    1 1 1 ...
$ depre6_o: Ord.factor w/ 5 levels "Never"<"Seldom"<..: 1 2 1 5 1 5 1
    1 1 2 ...
```

The HBSC data ...

```
$ gotBu1_o: Ord.factor w/ 5 levels "never"<"1 or 2"<..: 1 2 1 1 1 NA
    NA 1 1 1 ...
$ gotBu2_o: Ord.factor w/ 5 levels "never"<"1 or 2"<..: 1 1 1 1 1 NA
    NA 1 1 1 ...
$ gotBu3_o: Ord.factor w/ 5 levels "never"<"1 or 2"<..: 1 1 1 1 1 NA
    NA 1 1 1 ...
$ gotBu4_o: Ord.factor w/ 5 levels "never"<"1 or 2"<..: 1 1 1 3 1 NA
    NA 1 1 2 ...
$ gotBu5_o: Ord.factor w/ 5 levels "never"<"1 or 2"<..: 1 2 1 1 1 NA
    NA 1 1 1 ...
$ gotBu6_o: Ord.factor w/ 5 levels "never"<"1 or 2"<..: 1 1 1 4 1 NA
    NA 1 1 1 ...
$ gotBu7_o: Ord.factor w/ 5 levels "never"<"1 or 2"<..: 1 1 1 1 1 NA
    NA 1 1 1 ...
$ gotBu8_o: Ord.factor w/ 5 levels "never"<"1 or 2"<..: 1 1 1 NA 1 NA
    NA 1 1 2 ...
$ gotBu9_o: Ord.factor w/ 5 levels "never"<"1 or 2"<..: 1 1 1 5 1 NA
    NA 1 1 1 ...
$ buOth1_o: Ord.factor w/ 5 levels "never"<"1 or 2"<..: 2 1 1 4 1 NA
    NA 1 1 1 ...
$ buOth2_o: Ord.factor w/ 5 levels "never"<"1 or 2"<..: 1 1 1 1 1 NA
    NA 1 1 1 ...
$ buOth3_o: Ord.factor w/ 5 levels "never"<"1 or 2"<..: 1 1 1 4 1 NA
    NA 1 1 2 ...
$ buOth4_o: Ord.factor w/ 5 levels "never"<"1 or 2"<..: 1 1 1 1 1 NA
    NA 1 1 1 ...
```

The HBSC data ...

```
$ bu0th5_o: Ord.factor w/ 5 levels "never"<"1 or 2"<..: 1 1 1 1 1 NA
    NA 1 1 1 ...
$ bu0th6_o: Ord.factor w/ 5 levels "never"<"1 or 2"<..: 1 1 1 1 1 NA
    NA 1 1 1 ...
$ bu0th7_o: Ord.factor w/ 5 levels "never"<"1 or 2"<..: 1 1 1 1 1 NA
    NA 1 1 1 ...
$ bu0th8_o: Ord.factor w/ 5 levels "never"<"1 or 2"<..: 1 1 1 1 1 NA
    NA 1 1 1 ...
$ bu0th9_o: Ord.factor w/ 5 levels "never"<"1 or 2"<..: 1 1 1 1 1 NA
    NA 1 1 1 ...
$ alc1_o  : Ord.factor w/ 5 levels "Never"<"Rarely"<..: 1 1 1 1 1 2 1
    2 1 1 ...
$ alc2_o  : Ord.factor w/ 5 levels "Never"<"Rarely"<..: 1 1 1 1 1 1 1
    1 1 1 ...
$ alc3_o  : Ord.factor w/ 5 levels "Never"<"Rarely"<..: 1 1 1 1 1 1 1
    1 1 1 ...
$ alc4_o  : Ord.factor w/ 5 levels "Never"<"Rarely"<..: 1 1 1 1 1 1 1
    2 1 1 ...
$ alc5_o  : Ord.factor w/ 5 levels "Never"<"Rarely"<..: 1 1 1 1 1 1 1
    1 1 1 ...
$ body1_i : int  4 4 NA NA 4 3 5 4 NA NA ...
$ body2_i : int  4 4 NA NA 4 2 1 4 NA NA ...
$ body3_i : int  5 5 NA NA 4 3 5 3 NA NA ...
$ body4_i : int  5 4 NA NA 4 3 4 3 NA NA ...
$ body5_i : int  5 5 NA NA 5 3 5 3 NA NA ...
$ phys1_i : int  4 1 5 5 5 5 4 5 5 3 ...
```

The HBSC data ...

```
 $ phys2_i : int  5 4 5 5 5 2 4 5 5 4 ...
 $ phys3_i : int  5 2 5 1 5 2 2 3 5 5 ...
 $ phys4_i : int  5 5 5 5 5 1 5 5 5 5 ...
 $ phys5_i : int  5 4 5 4 4 1 4 5 5 4 ...
 $ phys6_i : int  5 2 5 4 4 1 3 4 5 5 ...
 $ phys7_i : int  5 1 5 5 3 1 3 1 5 5 ...
 $ phys8_i : int  5 5 5 5 5 1 4 5 5 4 ...
 $ depre1_i: int  2 2 3 3 2 3 1 3 1 1 ...
 $ depre2_i: int  2 2 4 5 3 4 4 3 1 2 ...
 $ depre3_i: int  1 5 2 3 2 4 1 1 1 1 ...
 $ depre4_i: int  1 2 1 5 1 1 3 1 3 2 ...
 $ depre5_i: int  2 2 1 3 3 1 3 1 1 1 ...
 $ depre6_i: int  1 2 1 5 1 5 1 1 1 2 ...
 $ gotBu1_i: int  1 2 1 1 1 NA NA 1 1 1 ...
 $ gotBu2_i: int  1 1 1 1 1 NA NA 1 1 1 ...
 $ gotBu3_i: int  1 1 1 1 1 NA NA 1 1 1 ...
 $ gotBu4_i: int  1 1 1 3 1 NA NA 1 1 2 ...
 $ gotBu5_i: int  1 2 1 1 1 NA NA 1 1 1 ...
 $ gotBu6_i: int  1 1 1 4 1 NA NA 1 1 1 ...
 $ gotBu7_i: int  1 1 1 1 1 NA NA 1 1 1 ...
 $ gotBu8_i: int  1 1 1 NA 1 NA NA 1 1 2 ...
 $ gotBu9_i: int  1 1 1 5 1 NA NA 1 1 1 ...
 $ buOth1_i: int  2 1 1 4 1 NA NA 1 1 1 ...
 $ buOth2_i: int  1 1 1 1 1 NA NA 1 1 1 ...
 $ buOth3_i: int  1 1 1 4 1 NA NA 1 1 2 ...
 $ buOth4_i: int  1 1 1 1 1 NA NA 1 1 1 ...
```

The HBSC data ...

```
$ bu0th5_i: int  1 1 1 1 1 NA NA 1 1 1 ...
$ bu0th6_i: int  1 1 1 1 1 NA NA 1 1 1 ...
$ bu0th7_i: int  1 1 1 1 1 NA NA 1 1 1 ...
$ bu0th8_i: int  1 1 1 1 1 NA NA 1 1 1 ...
$ bu0th9_i: int  1 1 1 1 1 NA NA 1 1 1 ...
$ alc1_i  : int  1 1 1 1 1 2 1 2 1 1 ...
$ alc2_i  : int  1 1 1 1 1 1 1 1 1 1 ...
$ alc3_i  : int  1 1 1 1 1 1 1 1 1 1 ...
$ alc4_i  : int  1 1 1 1 1 1 1 2 1 1 ...
$ alc5_i  : int  1 1 1 1 1 1 1 1 1 1 ...
```

The nature of your data

- Check that the levels are meaningfully ordered

```
## An ordered-factor variable in R should look
   like
levels(hbsc$depre1_o)
```

```
[1] "Never"     "Seldom"     "Sometimes" "Often"     "Always"
```

```
str(hbsc$depre1_o)
```

```
Ord.factor w/ 5 levels "Never"<"Seldom"<..: 2 2 3 3 2 3 1 3 1 1 ...
```

A one-factor categorical CFA model

```
## Specifying the model-structure object
cfa.01.ord <- '
 depress =~ NA*depre1_o + depre2_o + depre3_o +
                depre4_o + depre5_o + depre6_o
 depress ~~ 1*depress '
## Estimating the model
cfa.01.ord.fit <-
 cfa(model = cfa.01.ord, data = hbsc,
     mimic = "Mplus", estimator = "WLSMV",
     ordered = c("depre1_o", "depre2_o",
                 "depre3_o", "depre4_o",
                 "depre5_o", "depre6_o"))
## Requesting an estimation summary
summary(cfa.01.ord.fit, fit.measures = TRUE,
        standardized = TRUE)
```

A one-factor categorical CFA model ...

```
lavaan 0.6-3 ended normally after 12 iterations

  Optimization method                           NLMINB
  Number of free parameters                         30

                                                  Used       Total
  Number of observations                          4103        4284

  Estimator                                       DWLS      Robust
  Model Fit Test Statistic                     170.031     315.344
  Degrees of freedom                                 9           9
  P-value (Chi-square)                           0.000       0.000
  Scaling correction factor                                  0.540
  Shift parameter                                            0.231
    for simple second-order correction (WLSMV)

Model test baseline model:

  Minimum Function Test Statistic            16019.445   10984.581
  Degrees of freedom                                15          15
  P-value                                        0.000       0.000

User model versus baseline model:

  Comparative Fit Index (CFI)                    0.990       0.972
```

A one-factor categorical CFA model ...

```
  Tucker-Lewis Index (TLI)                      0.983        0.953

  Robust Comparative Fit Index (CFI)                            NA
  Robust Tucker-Lewis Index (TLI)                               NA

Root Mean Square Error of Approximation:

  RMSEA                                         0.066        0.091
  90 Percent Confidence Interval         0.058  0.075        0.083
      0.100
  P-value RMSEA <= 0.05                         0.001        0.000

  Robust RMSEA                                                  NA
  90 Percent Confidence Interval                                NA
      NA

Standardized Root Mean Square Residual:

  SRMR                                          0.034        0.034

Weighted Root Mean Square Residual:

  WRMR                                          2.088        2.088

Parameter Estimates:
```

A one-factor categorical CFA model ...

```
  Information                                   Expected
  Information saturated (h1) model          Unstructured
  Standard Errors                             Robust.sem

Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
  depress =~
    depre1_o          0.710    0.010   69.611    0.000    0.710    0.710
    depre2_o          0.655    0.011   59.557    0.000    0.655    0.655
    depre3_o          0.744    0.011   66.206    0.000    0.744    0.744
    depre4_o          0.715    0.011   66.123    0.000    0.715    0.715
    depre5_o          0.603    0.013   47.851    0.000    0.603    0.603
    depre6_o          0.589    0.013   45.786    0.000    0.589    0.589

Intercepts:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
   .depre1_o          0.000                               0.000    0.000
   .depre2_o          0.000                               0.000    0.000
   .depre3_o          0.000                               0.000    0.000
   .depre4_o          0.000                               0.000    0.000
   .depre5_o          0.000                               0.000    0.000
   .depre6_o          0.000                               0.000    0.000
    depress           0.000                               0.000    0.000

Thresholds:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
```

A one-factor categorical CFA model ...

```
    depre1_o|t1     -0.564    0.021  -27.187    0.000   -0.564   -0.564
    depre1_o|t2      0.133    0.020    6.756    0.000    0.133    0.133
    depre1_o|t3      1.095    0.024   44.708    0.000    1.095    1.095
    depre1_o|t4      1.786    0.036   49.021    0.000    1.786    1.786
    depre2_o|t1     -0.870    0.023  -38.650    0.000   -0.870   -0.870
    depre2_o|t2     -0.193    0.020   -9.812    0.000   -0.193   -0.193
    depre2_o|t3      0.737    0.022   34.072    0.000    0.737    0.737
    depre2_o|t4      1.510    0.030   49.857    0.000    1.510    1.510
    depre3_o|t1      0.289    0.020   14.542    0.000    0.289    0.289
    depre3_o|t2      0.668    0.021   31.451    0.000    0.668    0.668
    depre3_o|t3      1.177    0.025   46.350    0.000    1.177    1.177
    depre3_o|t4      1.624    0.033   49.899    0.000    1.624    1.624
    depre4_o|t1     -0.107    0.020   -5.446    0.000   -0.107   -0.107
    depre4_o|t2      0.255    0.020   12.894    0.000    0.255    0.255
    depre4_o|t3      0.888    0.023   39.214    0.000    0.888    0.888
    depre4_o|t4      1.436    0.029   49.520    0.000    1.436    1.436
    depre5_o|t1     -0.370    0.020  -18.421    0.000   -0.370   -0.370
    depre5_o|t2      0.072    0.020    3.668    0.000    0.072    0.072
    depre5_o|t3      0.669    0.021   31.481    0.000    0.669    0.669
    depre5_o|t4      1.232    0.026   47.292    0.000    1.232    1.232
    depre6_o|t1     -0.428    0.020  -21.173    0.000   -0.428   -0.428
    depre6_o|t2      0.085    0.020    4.323    0.000    0.085    0.085
    depre6_o|t3      0.741    0.022   34.219    0.000    0.741    0.741
    depre6_o|t4      1.255    0.026   47.633    0.000    1.255    1.255

Variances:
```

A one-factor categorical CFA model ...

```
                 Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
    depress         1.000                               1.000    1.000
   .depre1_o        0.496                               0.496    0.496
   .depre2_o        0.571                               0.571    0.571
   .depre3_o        0.446                               0.446    0.446
   .depre4_o        0.489                               0.489    0.489
   .depre5_o        0.637                               0.637    0.637
   .depre6_o        0.653                               0.653    0.653

Scales y*:
                 Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
    depre1_o        1.000                               1.000    1.000
    depre2_o        1.000                               1.000    1.000
    depre3_o        1.000                               1.000    1.000
    depre4_o        1.000                               1.000    1.000
    depre5_o        1.000                               1.000    1.000
    depre6_o        1.000                               1.000    1.000
```

A three-factor categorical CFA model

```
## Specifying the model-structure object
cfa.04.ord <- '
 gotBully =~ NA*gotBu1_o + gotBu2_o + gotBu3_o +
                gotBu4_o + gotBu5_o + gotBu6_o +
                gotBu7_o + gotBu8_o + gotBu9_o
 gotBully ~~ 1*gotBully

 depress =~ NA*depre1_o + depre2_o + depre3_o +
               depre4_o + depre5_o + depre6_o
 depress ~~ 1*depress

 alcohol =~ NA*alc1_o + alc2_o + alc3_o +
               alc4_o + alc5_o
 alcohol ~~ 1*alcohol '
## Estimating the model
cfa.04.ord.fit <-
 cfa(model = cfa.04.ord, data = hbsc,
```

A three-factor categorical CFA model ...

```
      mimic = "Mplus", estimator = "WLSMV")
## Requesting an estimation summery
summary(cfa.04.ord.fit, fit.measures = TRUE,
             standardized = TRUE)
```

```
lavaan 0.6-3 ended normally after 17 iterations

  Optimization method                         NLMINB
  Number of free parameters                      103

                                                Used       Total
  Number of observations                        2684        4284

  Estimator                                     DWLS      Robust
  Model Fit Test Statistic                   870.308    1046.718
  Degrees of freedom                             167         167
  P-value (Chi-square)                         0.000       0.000
  Scaling correction factor                                0.876
  Shift parameter                                         52.908
    for simple second-order correction (WLSMV)

Model test baseline model:
```

A three-factor categorical CFA model ...

```
  Minimum Function Test Statistic            75123.245   30297.601
  Degrees of freedom                               190         190
  P-value                                        0.000       0.000

User model versus baseline model:

  Comparative Fit Index (CFI)                    0.991       0.971
  Tucker-Lewis Index (TLI)                       0.989       0.967

  Robust Comparative Fit Index (CFI)                            NA
  Robust Tucker-Lewis Index (TLI)                               NA

Root Mean Square Error of Approximation:

  RMSEA                                          0.040       0.044
  90 Percent Confidence Interval          0.037  0.042       0.042
      0.047
  P-value RMSEA <= 0.05                          1.000       1.000

  Robust RMSEA                                                  NA
  90 Percent Confidence Interval                                NA
      NA

Standardized Root Mean Square Residual:

  SRMR                                           0.052       0.052
```

A three-factor categorical CFA model ...

```
Weighted Root Mean Square Residual:

  WRMR                                         1.795       1.795

Parameter Estimates:

  Information                                Expected
  Information saturated (h1) model       Unstructured
  Standard Errors                          Robust.sem

Latent Variables:
                  Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
  gotBully =~
    gotBu1_o         0.752    0.013   56.021    0.000    0.752    0.752
    gotBu2_o         0.780    0.013   59.261    0.000    0.780    0.780
    gotBu3_o         0.778    0.015   53.290    0.000    0.778    0.778
    gotBu4_o         0.807    0.011   70.920    0.000    0.807    0.807
    gotBu5_o         0.840    0.012   67.419    0.000    0.840    0.840
    gotBu6_o         0.866    0.013   65.080    0.000    0.866    0.866
    gotBu7_o         0.770    0.015   52.305    0.000    0.770    0.770
    gotBu8_o         0.861    0.014   61.460    0.000    0.861    0.861
    gotBu9_o         0.891    0.016   57.294    0.000    0.891    0.891
  depress =~
    depre1_o         0.692    0.014   49.499    0.000    0.692    0.692
    depre2_o         0.640    0.015   43.858    0.000    0.640    0.640
```

A three-factor categorical CFA model ...

```
    depre3_o         0.753    0.015   48.958    0.000    0.753    0.753
    depre4_o         0.710    0.015   47.969    0.000    0.710    0.710
    depre5_o         0.597    0.017   35.102    0.000    0.597    0.597
    depre6_o         0.628    0.016   38.109    0.000    0.628    0.628
  alcohol =~
    alc1_o           0.859    0.013   67.423    0.000    0.859    0.859
    alc2_o           0.774    0.016   47.124    0.000    0.774    0.774
    alc3_o           0.951    0.009  106.901    0.000    0.951    0.951
    alc4_o           0.882    0.010   84.795    0.000    0.882    0.882
    alc5_o           0.907    0.009  100.941    0.000    0.907    0.907

Covariances:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
  gotBully ~~
    depress          0.438    0.024   18.438    0.000    0.438    0.438
    alcohol          0.262    0.031    8.361    0.000    0.262    0.262
  depress ~~
    alcohol          0.354    0.028   12.467    0.000    0.354    0.354

Intercepts:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
   .gotBu1_o         0.000                               0.000    0.000
   .gotBu2_o         0.000                               0.000    0.000
   .gotBu3_o         0.000                               0.000    0.000
   .gotBu4_o         0.000                               0.000    0.000
   .gotBu5_o         0.000                               0.000    0.000
```

A three-factor categorical CFA model ...

```
    .gotBu6_o         0.000                                      0.000    0.000
    .gotBu7_o         0.000                                      0.000    0.000
    .gotBu8_o         0.000                                      0.000    0.000
    .gotBu9_o         0.000                                      0.000    0.000
    .depre1_o         0.000                                      0.000    0.000
    .depre2_o         0.000                                      0.000    0.000
    .depre3_o         0.000                                      0.000    0.000
    .depre4_o         0.000                                      0.000    0.000
    .depre5_o         0.000                                      0.000    0.000
    .depre6_o         0.000                                      0.000    0.000
    .alc1_o           0.000                                      0.000    0.000
    .alc2_o           0.000                                      0.000    0.000
    .alc3_o           0.000                                      0.000    0.000
    .alc4_o           0.000                                      0.000    0.000
    .alc5_o           0.000                                      0.000    0.000
     gotBully         0.000                                      0.000    0.000
     depress          0.000                                      0.000    0.000
     alcohol          0.000                                      0.000    0.000

Thresholds:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
     gotBu1_o|t1      0.266    0.025   10.870    0.000    0.266    0.266
     gotBu1_o|t2      0.852    0.028   30.781    0.000    0.852    0.852
     gotBu1_o|t3      1.070    0.030   35.696    0.000    1.070    1.070
     gotBu1_o|t4      1.319    0.034   39.208    0.000    1.319    1.319
     gotBu2_o|t1      0.470    0.025   18.640    0.000    0.470    0.470
```

A three-factor categorical CFA model ...

```
    gotBu2_o|t2      1.028    0.029   34.876    0.000    1.028    1.028
    gotBu2_o|t3      1.241    0.032   38.356    0.000    1.241    1.241
    gotBu2_o|t4      1.539    0.038   40.373    0.000    1.539    1.539
    gotBu3_o|t1      0.906    0.028   32.141    0.000    0.906    0.906
    gotBu3_o|t2      1.331    0.034   39.311    0.000    1.331    1.331
    gotBu3_o|t3      1.546    0.038   40.380    0.000    1.546    1.546
    gotBu3_o|t4      1.802    0.046   39.528    0.000    1.802    1.802
    gotBu4_o|t1      0.304    0.025   12.369    0.000    0.304    0.304
    gotBu4_o|t2      0.917    0.028   32.416    0.000    0.917    0.917
    gotBu4_o|t3      1.204    0.032   37.867    0.000    1.204    1.204
    gotBu4_o|t4      1.459    0.036   40.154    0.000    1.459    1.459
    gotBu5_o|t1      1.012    0.029   34.554    0.000    1.012    1.012
    gotBu5_o|t2      1.375    0.035   39.672    0.000    1.375    1.375
    gotBu5_o|t3      1.555    0.038   40.388    0.000    1.555    1.555
    gotBu5_o|t4      1.831    0.047   39.298    0.000    1.831    1.831
    gotBu6_o|t1      1.251    0.033   38.479    0.000    1.251    1.251
    gotBu6_o|t2      1.580    0.039   40.395    0.000    1.580    1.580
    gotBu6_o|t3      1.788    0.045   39.629    0.000    1.788    1.788
    gotBu6_o|t4      2.029    0.055   37.126    0.000    2.029    2.029
    gotBu7_o|t1      0.649    0.026   24.818    0.000    0.649    0.649
    gotBu7_o|t2      1.077    0.030   35.819    0.000    1.077    1.077
    gotBu7_o|t3      1.291    0.033   38.926    0.000    1.291    1.291
    gotBu7_o|t4      1.564    0.039   40.393    0.000    1.564    1.564
    gotBu8_o|t1      1.361    0.034   39.564    0.000    1.361    1.361
    gotBu8_o|t2      1.695    0.042   40.149    0.000    1.695    1.695
    gotBu8_o|t3      1.889    0.049   38.768    0.000    1.889    1.889
```

A three-factor categorical CFA model ...

```
    gotBu8_o|t4       2.083    0.057   36.366    0.000    2.083    2.083
    gotBu9_o|t1       1.492    0.037   40.273    0.000    1.492    1.492
    gotBu9_o|t2       1.744    0.044   39.911    0.000    1.744    1.744
    gotBu9_o|t3       1.889    0.049   38.768    0.000    1.889    1.889
    gotBu9_o|t4       2.075    0.057   36.484    0.000    2.075    2.075
    depre1_o|t1      -0.553    0.026  -21.595    0.000   -0.553   -0.553
    depre1_o|t2       0.163    0.024    6.712    0.000    0.163    0.163
    depre1_o|t3       1.095    0.030   36.155    0.000    1.095    1.095
    depre1_o|t4       1.812    0.046   39.455    0.000    1.812    1.812
    depre2_o|t1      -0.906    0.028  -32.141    0.000   -0.906   -0.906
    depre2_o|t2      -0.175    0.024   -7.174    0.000   -0.175   -0.175
    depre2_o|t3       0.749    0.027   27.912    0.000    0.749    0.749
    depre2_o|t4       1.533    0.038   40.365    0.000    1.533    1.533
    depre3_o|t1       0.293    0.025   11.908    0.000    0.293    0.293
    depre3_o|t2       0.687    0.026   26.041    0.000    0.687    0.687
    depre3_o|t3       1.168    0.031   37.350    0.000    1.168    1.168
    depre3_o|t4       1.642    0.041   40.317    0.000    1.642    1.642
    depre4_o|t1      -0.152    0.024   -6.249    0.000   -0.152   -0.152
    depre4_o|t2       0.247    0.024   10.101    0.000    0.247    0.247
    depre4_o|t3       0.897    0.028   31.934    0.000    0.897    0.897
    depre4_o|t4       1.478    0.037   40.228    0.000    1.478    1.478
    depre5_o|t1      -0.410    0.025  -16.429    0.000   -0.410   -0.410
    depre5_o|t2       0.053    0.024    2.200    0.028    0.053    0.053
    depre5_o|t3       0.656    0.026   25.041    0.000    0.656    0.656
    depre5_o|t4       1.237    0.032   38.306    0.000    1.237    1.237
    depre6_o|t1      -0.454    0.025  -18.069    0.000   -0.454   -0.454
```

A three-factor categorical CFA model ...

```
    depre6_o|t2        0.092    0.024    3.781    0.000    0.092    0.092
    depre6_o|t3        0.733    0.027   27.438    0.000    0.733    0.733
    depre6_o|t4        1.261    0.033   38.600    0.000    1.261    1.261
    alc1_o|t1          1.060    0.030   35.509    0.000    1.060    1.060
    alc1_o|t2          1.836    0.047   39.256    0.000    1.836    1.836
    alc1_o|t3          2.193    0.063   34.638    0.000    2.193    2.193
    alc1_o|t4          2.435    0.081   30.190    0.000    2.435    2.435
    alc2_o|t1          0.927    0.028   32.655    0.000    0.927    0.927
    alc2_o|t2          1.816    0.046   39.417    0.000    1.816    1.816
    alc2_o|t3          2.117    0.059   35.858    0.000    2.117    2.117
    alc2_o|t4          2.473    0.084   29.438    0.000    2.473    2.473
    alc3_o|t1          1.317    0.034   39.187    0.000    1.317    1.317
    alc3_o|t2          1.816    0.046   39.417    0.000    1.816    1.816
    alc3_o|t3          2.075    0.057   36.484    0.000    2.075    2.075
    alc3_o|t4          2.435    0.081   30.190    0.000    2.435    2.435
    alc4_o|t1          0.983    0.029   33.932    0.000    0.983    0.983
    alc4_o|t2          1.604    0.040   40.381    0.000    1.604    1.604
    alc4_o|t3          2.000    0.053   37.499    0.000    2.000    2.000
    alc4_o|t4          2.353    0.074   31.773    0.000    2.353    2.353
    alc5_o|t1          1.067    0.030   35.633    0.000    1.067    1.067
    alc5_o|t2          1.719    0.043   40.042    0.000    1.719    1.719
    alc5_o|t3          2.036    0.055   37.026    0.000    2.036    2.036
    alc5_o|t4          2.353    0.074   31.773    0.000    2.353    2.353

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
```

A three-factor categorical CFA model ...

```
    gotBully      1.000                                 1.000    1.000
    depress       1.000                                 1.000    1.000
    alcohol       1.000                                 1.000    1.000
   .gotBu1_o      0.435                                 0.435    0.435
   .gotBu2_o      0.391                                 0.391    0.391
   .gotBu3_o      0.395                                 0.395    0.395
   .gotBu4_o      0.348                                 0.348    0.348
   .gotBu5_o      0.295                                 0.295    0.295
   .gotBu6_o      0.249                                 0.249    0.249
   .gotBu7_o      0.407                                 0.407    0.407
   .gotBu8_o      0.259                                 0.259    0.259
   .gotBu9_o      0.206                                 0.206    0.206
   .depre1_o      0.521                                 0.521    0.521
   .depre2_o      0.591                                 0.591    0.591
   .depre3_o      0.432                                 0.432    0.432
   .depre4_o      0.495                                 0.495    0.495
   .depre5_o      0.643                                 0.643    0.643
   .depre6_o      0.606                                 0.606    0.606
   .alc1_o        0.262                                 0.262    0.262
   .alc2_o        0.401                                 0.401    0.401
   .alc3_o        0.095                                 0.095    0.095
   .alc4_o        0.221                                 0.221    0.221
   .alc5_o        0.178                                 0.178    0.178

Scales y*:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
```

A three-factor categorical CFA model ...

```
    gotBu1_o        1.000                                    1.000    1.000
    gotBu2_o        1.000                                    1.000    1.000
    gotBu3_o        1.000                                    1.000    1.000
    gotBu4_o        1.000                                    1.000    1.000
    gotBu5_o        1.000                                    1.000    1.000
    gotBu6_o        1.000                                    1.000    1.000
    gotBu7_o        1.000                                    1.000    1.000
    gotBu8_o        1.000                                    1.000    1.000
    gotBu9_o        1.000                                    1.000    1.000
    depre1_o        1.000                                    1.000    1.000
    depre2_o        1.000                                    1.000    1.000
    depre3_o        1.000                                    1.000    1.000
    depre4_o        1.000                                    1.000    1.000
    depre5_o        1.000                                    1.000    1.000
    depre6_o        1.000                                    1.000    1.000
    alc1_o          1.000                                    1.000    1.000
    alc2_o          1.000                                    1.000    1.000
    alc3_o          1.000                                    1.000    1.000
    alc4_o          1.000                                    1.000    1.000
    alc5_o          1.000                                    1.000    1.000
```

Outline

A structural model with ordinal indicators

```
## Specifying the model-structure object
sem.01.ord <- '
 gotBully =~ NA*gotBu1_o + gotBu2_o + gotBu3_o +
                gotBu4_o + gotBu5_o + gotBu6_o +
                gotBu7_o + gotBu8_o + gotBu9_o
 gotBully ~~ 1*gotBully

 alcohol =~ NA*alc1_o + alc2_o + alc3_o +
               alc4_o + alc5_o
 alcohol ~~ 1*alcohol

 alcohol ~ gotBully '
## Estimating the model
sem.01.ord.fit <-
 sem(model = sem.01.ord, data = hbsc,
     mimic = "Mplus", estimator = "WLSMV")
## Requesting an estimation summery
```

A structural model with ordinal indicators ...

```
summary(sem.01.ord.fit, fit.measures = TRUE,
        standardized = TRUE)
```

```
lavaan 0.6-3 ended normally after 22 iterations

  Optimization method                           NLMINB
  Number of free parameters                         71

                                                  Used       Total
  Number of observations                          2745        4284

  Estimator                                       DWLS      Robust
  Model Fit Test Statistic                     491.935     646.904
  Degrees of freedom                                76          76
  P-value (Chi-square)                           0.000       0.000
  Scaling correction factor                                  0.789
  Shift parameter                                           23.365
    for simple second-order correction (WLSMV)

Model test baseline model:

  Minimum Function Test Statistic            61631.106   23861.747
  Degrees of freedom                                91          91
  P-value                                        0.000       0.000
```

A structural model with ordinal indicators ...

```
User model versus baseline model:

  Comparative Fit Index (CFI)                    0.993       0.976
  Tucker-Lewis Index (TLI)                       0.992       0.971

  Robust Comparative Fit Index (CFI)                            NA
  Robust Tucker-Lewis Index (TLI)                               NA

Root Mean Square Error of Approximation:

  RMSEA                                          0.045       0.052
  90 Percent Confidence Interval          0.041  0.048       0.049
       0.056
  P-value RMSEA <= 0.05                          0.990       0.148

  Robust RMSEA                                                  NA
  90 Percent Confidence Interval                                NA
       NA

Standardized Root Mean Square Residual:

  SRMR                                           0.057       0.057

Weighted Root Mean Square Residual:

```

A structural model with ordinal indicators ...

```
  WRMR                                             1.829       1.829

Parameter Estimates:

  Information                                   Expected
  Information saturated (h1) model            Unstructured
  Standard Errors                               Robust.sem

Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
  gotBully =~
    gotBu1_o          0.745    0.013   55.379    0.000    0.745    0.745
    gotBu2_o          0.774    0.013   59.107    0.000    0.774    0.774
    gotBu3_o          0.790    0.014   56.927    0.000    0.790    0.790
    gotBu4_o          0.798    0.011   71.210    0.000    0.798    0.798
    gotBu5_o          0.842    0.012   70.209    0.000    0.842    0.842
    gotBu6_o          0.879    0.012   70.731    0.000    0.879    0.879
    gotBu7_o          0.762    0.014   53.164    0.000    0.762    0.762
    gotBu8_o          0.874    0.013   66.741    0.000    0.874    0.874
    gotBu9_o          0.905    0.014   62.407    0.000    0.905    0.905
  alcohol =~
    alc1_o            0.833    0.013   62.215    0.000    0.867    0.867
    alc2_o            0.757    0.016   47.938    0.000    0.788    0.788
    alc3_o            0.912    0.011   82.275    0.000    0.949    0.949
    alc4_o            0.846    0.012   70.018    0.000    0.881    0.881
    alc5_o            0.871    0.012   74.238    0.000    0.906    0.906
```

A structural model with ordinal indicators ...

```
Regressions:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
  alcohol ~
    gotBully          0.289    0.035    8.370    0.000    0.278    0.278

Intercepts:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
   .gotBu1_o          0.000                               0.000    0.000
   .gotBu2_o          0.000                               0.000    0.000
   .gotBu3_o          0.000                               0.000    0.000
   .gotBu4_o          0.000                               0.000    0.000
   .gotBu5_o          0.000                               0.000    0.000
   .gotBu6_o          0.000                               0.000    0.000
   .gotBu7_o          0.000                               0.000    0.000
   .gotBu8_o          0.000                               0.000    0.000
   .gotBu9_o          0.000                               0.000    0.000
   .alc1_o            0.000                               0.000    0.000
   .alc2_o            0.000                               0.000    0.000
   .alc3_o            0.000                               0.000    0.000
   .alc4_o            0.000                               0.000    0.000
   .alc5_o            0.000                               0.000    0.000
    gotBully          0.000                               0.000    0.000
   .alcohol           0.000                               0.000    0.000

Thresholds:
```

A structural model with ordinal indicators ...

	Estimate	Std.Err	z-value	P(>|z|)	Std.lv	Std.all
gotBu1_o|t1	0.272	0.024	11.224	0.000	0.272	0.272
gotBu1_o|t2	0.853	0.027	31.165	0.000	0.853	0.853
gotBu1_o|t3	1.066	0.030	36.028	0.000	1.066	1.066
gotBu1_o|t4	1.321	0.033	39.666	0.000	1.321	1.321
gotBu2_o|t1	0.473	0.025	18.982	0.000	0.473	0.473
gotBu2_o|t2	1.028	0.029	35.279	0.000	1.028	1.028
gotBu2_o|t3	1.242	0.032	38.803	0.000	1.242	1.242
gotBu2_o|t4	1.542	0.038	40.832	0.000	1.542	1.542
gotBu3_o|t1	0.902	0.028	32.407	0.000	0.902	0.902
gotBu3_o|t2	1.323	0.033	39.687	0.000	1.323	1.323
gotBu3_o|t3	1.539	0.038	40.829	0.000	1.539	1.539
gotBu3_o|t4	1.803	0.045	39.968	0.000	1.803	1.803
gotBu4_o|t1	0.305	0.024	12.516	0.000	0.305	0.305
gotBu4_o|t2	0.914	0.028	32.713	0.000	0.914	0.914
gotBu4_o|t3	1.200	0.031	38.242	0.000	1.200	1.200
gotBu4_o|t4	1.460	0.036	40.613	0.000	1.460	1.460
gotBu5_o|t1	1.007	0.029	34.832	0.000	1.007	1.007
gotBu5_o|t2	1.373	0.034	40.106	0.000	1.373	1.373
gotBu5_o|t3	1.551	0.038	40.842	0.000	1.551	1.551
gotBu5_o|t4	1.831	0.046	39.740	0.000	1.831	1.831
gotBu6_o|t1	1.250	0.032	38.900	0.000	1.250	1.250
gotBu6_o|t2	1.572	0.038	40.852	0.000	1.572	1.572
gotBu6_o|t3	1.785	0.045	40.101	0.000	1.785	1.785
gotBu6_o|t4	2.024	0.054	37.614	0.000	2.024	2.024
gotBu7_o|t1	0.651	0.026	25.163	0.000	0.651	0.651

A structural model with ordinal indicators ...

gotBu7_o|t2	1.079	0.030	36.272	0.000	1.079	1.079
gotBu7_o|t3	1.293	0.033	39.388	0.000	1.293	1.293
gotBu7_o|t4	1.572	0.038	40.852	0.000	1.572	1.572
gotBu8_o|t1	1.359	0.034	39.998	0.000	1.359	1.359
gotBu8_o|t2	1.682	0.041	40.652	0.000	1.682	1.682
gotBu8_o|t3	1.872	0.048	39.375	0.000	1.872	1.872
gotBu8_o|t4	2.061	0.056	37.103	0.000	2.061	2.061
gotBu9_o|t1	1.484	0.036	40.704	0.000	1.484	1.484
gotBu9_o|t2	1.737	0.043	40.397	0.000	1.737	1.737
gotBu9_o|t3	1.877	0.048	39.324	0.000	1.877	1.877
gotBu9_o|t4	2.076	0.056	36.878	0.000	2.076	2.076
alc1_o|t1	1.063	0.030	35.966	0.000	1.063	1.063
alc1_o|t2	1.822	0.046	39.820	0.000	1.822	1.822
alc1_o|t3	2.172	0.061	35.382	0.000	2.172	2.172
alc1_o|t4	2.392	0.076	31.376	0.000	2.392	2.392
alc2_o|t1	0.924	0.028	32.950	0.000	0.924	0.924
alc2_o|t2	1.808	0.045	39.933	0.000	1.808	1.808
alc2_o|t3	2.100	0.058	36.516	0.000	2.100	2.100
alc2_o|t4	2.443	0.080	30.370	0.000	2.443	2.443
alc3_o|t1	1.314	0.033	39.604	0.000	1.314	1.314
alc3_o|t2	1.812	0.045	39.896	0.000	1.812	1.812
alc3_o|t3	2.068	0.056	36.992	0.000	2.068	2.068
alc3_o|t4	2.425	0.079	30.721	0.000	2.425	2.425
alc4_o|t1	0.987	0.029	34.411	0.000	0.987	0.987
alc4_o|t2	1.601	0.039	40.840	0.000	1.601	1.601
alc4_o|t3	1.990	0.052	38.061	0.000	1.990	1.990

A structural model with ordinal indicators ...

```
    alc4_o|t4         2.319    0.071   32.776    0.000    2.319    2.319
    alc5_o|t1         1.068    0.030   36.059    0.000    1.068    1.068
    alc5_o|t2         1.717    0.042   40.503    0.000    1.717    1.717
    alc5_o|t3         2.031    0.054   37.517    0.000    2.031    2.031
    alc5_o|t4         2.347    0.073   32.252    0.000    2.347    2.347

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
    gotBully          1.000                               1.000    1.000
   .alcohol           1.000                               0.923    0.923
   .gotBu1_o          0.445                               0.445    0.445
   .gotBu2_o          0.401                               0.401    0.401
   .gotBu3_o          0.375                               0.375    0.375
   .gotBu4_o          0.364                               0.364    0.364
   .gotBu5_o          0.292                               0.292    0.292
   .gotBu6_o          0.227                               0.227    0.227
   .gotBu7_o          0.420                               0.420    0.420
   .gotBu8_o          0.236                               0.236    0.236
   .gotBu9_o          0.181                               0.181    0.181
   .alc1_o            0.248                               0.248    0.248
   .alc2_o            0.380                               0.380    0.380
   .alc3_o            0.100                               0.100    0.100
   .alc4_o            0.225                               0.225    0.225
   .alc5_o            0.178                               0.178    0.178

Scales y*:
```

A structural model with ordinal indicators ...

	Estimate	Std.Err	z-value	P(>|z|)	Std.lv	Std.all
gotBu1_o	1.000				1.000	1.000
gotBu2_o	1.000				1.000	1.000
gotBu3_o	1.000				1.000	1.000
gotBu4_o	1.000				1.000	1.000
gotBu5_o	1.000				1.000	1.000
gotBu6_o	1.000				1.000	1.000
gotBu7_o	1.000				1.000	1.000
gotBu8_o	1.000				1.000	1.000
gotBu9_o	1.000				1.000	1.000
alc1_o	1.000				1.000	1.000
alc2_o	1.000				1.000	1.000
alc3_o	1.000				1.000	1.000
alc4_o	1.000				1.000	1.000
alc5_o	1.000				1.000	1.000

An indirect effect model with ordinal data

- Nboot = 10 in order to save time; raise that to 1000 or 2000 in a real project.

```
## Specifying the model-structure object
sem.02.ord <- '
 ## the measurement model
 gotBully =~ NA*gotBu1_o + gotBu2_o + gotBu3_o +
               gotBu4_o + gotBu5_o + gotBu6_o +
              gotBu7_o + gotBu8_o + gotBu9_o
 gotBully ~~ 1*gotBully

 depress =~ NA*depre1_o + depre2_o + depre3_o +
              depre4_o + depre5_o + depre6_o
 depress ~~ 1*depress

 alcohol =~ NA*alc1_o + alc2_o + alc3_o +
              alc4_o + alc5_o
```

An indirect effect model with ordinal data ...

```
  alcohol ~~ 1*alcohol

  ## the structural model
  ## direct effect (the c path)
  alcohol ~ c*gotBully

  ## mediator paths (the a and b path)
  ## the a path - IV predicting ME
  depress ~ a*gotBully
  ## the b path - ME predicting DV
  alcohol ~ b*depress

  ## indirect effect (a*b)
  ## := operator defines new parameters
  ab := a*b

  ## total effect
```

An indirect effect model with ordinal data ...

```
 total := c + (a*b) '
## Estimating the model
Nboot <- 10
sem.02.ord.fit <-
 sem(model = sem.02.ord, data = hbsc,
     mimic = "Mplus", estimator = "DWLS",
     se = "bootstrap", verbose = TRUE,
     bootstrap = Nboot)
```

```
Estimating sample thresholds and correlations ... done
Quasi-Newton steps using NLMINB:
Objective function  = 1.6304782003700506
Objective function  = 4.3078918147412253
Objective function  = 0.5235747582570333
Objective function  = 0.4648468857015091
Objective function  = 0.2962602988611805
Objective function  = 0.5158296626795964
Objective function  = 0.1797810390781590
Objective function  = 0.1794633005086383
Objective function  = 0.1639437760851136
Objective function  = 0.1646792256203331
```

An indirect effect model with ordinal data ...

```
Objective function  = 0.1643922647221566
Objective function  = 0.1631750765883284
Objective function  = 0.1627453483523332
Objective function  = 0.1623158721478513
Objective function  = 0.1623687520295372
Objective function  = 0.1623792467402424
Objective function  = 0.1622108420797628
Objective function  = 0.1621514247023866
Objective function  = 0.1621324780354752
Objective function  = 0.1621388401677114
Objective function  = 0.1621310484748160
Objective function  = 0.1621297446938101
Objective function  = 0.1621298352766599
Objective function  = 0.1621290450479564
Objective function  = 0.1621291173392306
Objective function  = 0.1621289404154801
Objective function  = 0.1621289056118569
Objective function  = 0.1621289055046326
Objective function  = 0.1621288871349046
Objective function  = 0.1621288847249472
Objective function  = 0.1621288850714017
Objective function  = 0.1621288845983513
Objective function  = 0.1621288845924740
Objective function  = 0.1621288845727410
Objective function  = 0.1621288845793785
Objective function  = 0.1621288845727410
```

An indirect effect model with ordinal data ...

```
convergence status (0=ok):  0
nlminb message says:  relative convergence (4)
number of iterations:  24
number of function evaluations [objective, gradient]:  35 24
Computing VCOV for se = bootstrap ...
  ... bootstrap draw number:    1   OK -- niter =   31   fx =
      0.209263023
  ... bootstrap draw number:    2   OK -- niter =   31   fx =
      0.187160838
  ... bootstrap draw number:    3   OK -- niter =   29   fx =
      0.210081105
  ... bootstrap draw number:    4   OK -- niter =   36   fx =
      0.162921807
  ... bootstrap draw number:    5   OK -- niter =   30   fx =
      0.191199095
  ... bootstrap draw number:    6   OK -- niter =   27   fx =
      0.199532107
  ... bootstrap draw number:    7   OK -- niter =   26   fx =
      0.173135581
  ... bootstrap draw number:    8   OK -- niter =   28   fx =
      0.195254339
  ... bootstrap draw number:    9   OK -- niter =   29   fx =
      0.170001213
  ... bootstrap draw number:   10   OK -- niter =   28   fx =
      0.175450263
Number of successful bootstrap draws: 10
```

An indirect effect model with ordinal data ...

```
 done.
Computing TEST for test = standard ... done.
```

```
## Requesting an estimation summary
summary(sem.02.ord.fit, fit.measures = TRUE,
        standardized = TRUE)
```

```
lavaan 0.6-3 ended normally after 24 iterations

  Optimization method                           NLMINB
  Number of free parameters                        103

                                                  Used       Total
  Number of observations                          2684        4284

  Estimator                                       DWLS
  Model Fit Test Statistic                     870.308
  Degrees of freedom                               167
  P-value (Chi-square)                           0.000

Model test baseline model:

  Minimum Function Test Statistic            75123.245
```

An indirect effect model with ordinal data ...

```
  Degrees of freedom                                 190
  P-value                                          0.000

User model versus baseline model:

  Comparative Fit Index (CFI)                      0.991
  Tucker-Lewis Index (TLI)                         0.989

Root Mean Square Error of Approximation:

  RMSEA                                            0.040
  90 Percent Confidence Interval            0.037  0.042
  P-value RMSEA <= 0.05                            1.000

Standardized Root Mean Square Residual:

  SRMR                                             0.052

Weighted Root Mean Square Residual:

  WRMR                                             1.795

Parameter Estimates:

  Standard Errors                              Bootstrap
  Number of requested bootstrap draws                 10
```

An indirect effect model with ordinal data ...

```
  Number of successful bootstrap draws             10

Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
  gotBully =~
    gotBu1_o          0.752    0.016   45.589    0.000    0.752    0.752
    gotBu2_o          0.780    0.007  114.820    0.000    0.780    0.780
    gotBu3_o          0.778    0.018   43.671    0.000    0.778    0.778
    gotBu4_o          0.807    0.013   60.192    0.000    0.807    0.807
    gotBu5_o          0.840    0.012   69.129    0.000    0.840    0.840
    gotBu6_o          0.866    0.015   56.432    0.000    0.866    0.866
    gotBu7_o          0.770    0.014   55.174    0.000    0.770    0.770
    gotBu8_o          0.861    0.016   53.892    0.000    0.861    0.861
    gotBu9_o          0.891    0.014   64.221    0.000    0.891    0.891
  depress =~
    depre1_o          0.622    0.010   64.389    0.000    0.692    0.692
    depre2_o          0.575    0.009   62.699    0.000    0.640    0.640
    depre3_o          0.677    0.014   50.119    0.000    0.753    0.753
    depre4_o          0.639    0.021   30.132    0.000    0.710    0.710
    depre5_o          0.537    0.025   21.239    0.000    0.597    0.597
    depre6_o          0.564    0.013   42.330    0.000    0.628    0.628
  alcohol =~
    alc1_o            0.797    0.016   49.334    0.000    0.859    0.859
    alc2_o            0.718    0.013   56.980    0.000    0.774    0.774
    alc3_o            0.883    0.014   64.691    0.000    0.951    0.951
    alc4_o            0.819    0.016   51.826    0.000    0.882    0.882
```

An indirect effect model with ordinal data ...

```
    alc5_o             0.841    0.012   68.294    0.000    0.907    0.907

Regressions:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
  alcohol ~
    gotBully   (c)    0.143    0.047    3.036    0.002    0.133    0.133
  depress ~
    gotBully   (a)    0.487    0.040   12.314    0.000    0.438    0.438
  alcohol ~
    depress    (b)    0.286    0.053    5.367    0.000    0.295    0.295

Intercepts:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
   .gotBu1_o          0.000                               0.000    0.000
   .gotBu2_o          0.000                               0.000    0.000
   .gotBu3_o          0.000                               0.000    0.000
   .gotBu4_o          0.000                               0.000    0.000
   .gotBu5_o          0.000                               0.000    0.000
   .gotBu6_o          0.000                               0.000    0.000
   .gotBu7_o          0.000                               0.000    0.000
   .gotBu8_o          0.000                               0.000    0.000
   .gotBu9_o          0.000                               0.000    0.000
   .depre1_o          0.000                               0.000    0.000
   .depre2_o          0.000                               0.000    0.000
   .depre3_o          0.000                               0.000    0.000
   .depre4_o          0.000                               0.000    0.000
```

An indirect effect model with ordinal data ...

```
    .depre5_o          0.000                                      0.000    0.000
    .depre6_o          0.000                                      0.000    0.000
    .alc1_o            0.000                                      0.000    0.000
    .alc2_o            0.000                                      0.000    0.000
    .alc3_o            0.000                                      0.000    0.000
    .alc4_o            0.000                                      0.000    0.000
    .alc5_o            0.000                                      0.000    0.000
     gotBully          0.000                                      0.000    0.000
    .depress           0.000                                      0.000    0.000
    .alcohol           0.000                                      0.000    0.000

Thresholds:
                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
     gotBu1_o|t1       0.266    0.024   11.129    0.000    0.266    0.266
     gotBu1_o|t2       0.852    0.025   33.712    0.000    0.852    0.852
     gotBu1_o|t3       1.070    0.040   26.959    0.000    1.070    1.070
     gotBu1_o|t4       1.319    0.047   27.952    0.000    1.319    1.319
     gotBu2_o|t1       0.470    0.021   21.969    0.000    0.470    0.470
     gotBu2_o|t2       1.028    0.018   57.123    0.000    1.028    1.028
     gotBu2_o|t3       1.241    0.020   63.134    0.000    1.241    1.241
     gotBu2_o|t4       1.539    0.034   45.853    0.000    1.539    1.539
     gotBu3_o|t1       0.906    0.031   29.072    0.000    0.906    0.906
     gotBu3_o|t2       1.331    0.036   37.065    0.000    1.331    1.331
     gotBu3_o|t3       1.546    0.041   37.896    0.000    1.546    1.546
     gotBu3_o|t4       1.802    0.046   38.771    0.000    1.802    1.802
     gotBu4_o|t1       0.304    0.015   20.290    0.000    0.304    0.304
```

An indirect effect model with ordinal data ...

```
    gotBu4_o|t2       0.917    0.021   42.775    0.000    0.917    0.917
    gotBu4_o|t3       1.204    0.040   30.099    0.000    1.204    1.204
    gotBu4_o|t4       1.459    0.031   47.584    0.000    1.459    1.459
    gotBu5_o|t1       1.012    0.044   22.826    0.000    1.012    1.012
    gotBu5_o|t2       1.375    0.041   33.501    0.000    1.375    1.375
    gotBu5_o|t3       1.555    0.053   29.501    0.000    1.555    1.555
    gotBu5_o|t4       1.831    0.050   36.635    0.000    1.831    1.831
    gotBu6_o|t1       1.251    0.045   28.081    0.000    1.251    1.251
    gotBu6_o|t2       1.580    0.053   29.793    0.000    1.580    1.580
    gotBu6_o|t3       1.788    0.058   30.879    0.000    1.788    1.788
    gotBu6_o|t4       2.029    0.051   39.506    0.000    2.029    2.029
    gotBu7_o|t1       0.649    0.037   17.614    0.000    0.649    0.649
    gotBu7_o|t2       1.077    0.025   43.744    0.000    1.077    1.077
    gotBu7_o|t3       1.291    0.022   57.465    0.000    1.291    1.291
    gotBu7_o|t4       1.564    0.026   59.079    0.000    1.564    1.564
    gotBu8_o|t1       1.361    0.023   59.044    0.000    1.361    1.361
    gotBu8_o|t2       1.695    0.034   49.493    0.000    1.695    1.695
    gotBu8_o|t3       1.889    0.045   42.038    0.000    1.889    1.889
    gotBu8_o|t4       2.083    0.056   37.469    0.000    2.083    2.083
    gotBu9_o|t1       1.492    0.032   46.509    0.000    1.492    1.492
    gotBu9_o|t2       1.744    0.041   42.727    0.000    1.744    1.744
    gotBu9_o|t3       1.889    0.044   43.041    0.000    1.889    1.889
    gotBu9_o|t4       2.075    0.047   44.068    0.000    2.075    2.075
    depre1_o|t1      -0.553    0.024  -22.682    0.000   -0.553   -0.553
    depre1_o|t2       0.163    0.023    7.209    0.000    0.163    0.163
    depre1_o|t3       1.095    0.030   36.002    0.000    1.095    1.095
```

An indirect effect model with ordinal data ...

```
    depre1_o|t4        1.812    0.045   40.615    0.000    1.812    1.812
    depre2_o|t1       -0.906    0.016  -56.190    0.000   -0.906   -0.906
    depre2_o|t2       -0.175    0.021   -8.350    0.000   -0.175   -0.175
    depre2_o|t3        0.749    0.022   34.483    0.000    0.749    0.749
    depre2_o|t4        1.533    0.043   35.647    0.000    1.533    1.533
    depre3_o|t1        0.293    0.027   10.846    0.000    0.293    0.293
    depre3_o|t2        0.687    0.028   24.871    0.000    0.687    0.687
    depre3_o|t3        1.168    0.033   35.902    0.000    1.168    1.168
    depre3_o|t4        1.642    0.045   36.773    0.000    1.642    1.642
    depre4_o|t1       -0.152    0.019   -8.014    0.000   -0.152   -0.152
    depre4_o|t2        0.247    0.022   11.418    0.000    0.247    0.247
    depre4_o|t3        0.897    0.023   38.568    0.000    0.897    0.897
    depre4_o|t4        1.478    0.034   44.095    0.000    1.478    1.478
    depre5_o|t1       -0.410    0.027  -15.402    0.000   -0.410   -0.410
    depre5_o|t2        0.053    0.023    2.309    0.021    0.053    0.053
    depre5_o|t3        0.656    0.029   22.299    0.000    0.656    0.656
    depre5_o|t4        1.237    0.034   36.805    0.000    1.237    1.237
    depre6_o|t1       -0.454    0.026  -17.201    0.000   -0.454   -0.454
    depre6_o|t2        0.092    0.021    4.326    0.000    0.092    0.092
    depre6_o|t3        0.733    0.013   55.029    0.000    0.733    0.733
    depre6_o|t4        1.261    0.025   49.514    0.000    1.261    1.261
    alc1_o|t1          1.060    0.040   26.290    0.000    1.060    1.060
    alc1_o|t2          1.836    0.044   41.357    0.000    1.836    1.836
    alc1_o|t3          2.193    0.053   41.417    0.000    2.193    2.193
    alc1_o|t4          2.435    0.090   27.033    0.000    2.435    2.435
    alc2_o|t1          0.927    0.030   30.765    0.000    0.927    0.927
```

An indirect effect model with ordinal data ...

```
    alc2_o|t2         1.816    0.051   35.610    0.000    1.816    1.816
    alc2_o|t3         2.117    0.053   40.178    0.000    2.117    2.117
    alc2_o|t4         2.473    0.079   31.118    0.000    2.473    2.473
    alc3_o|t1         1.317    0.029   45.335    0.000    1.317    1.317
    alc3_o|t2         1.816    0.036   49.846    0.000    1.816    1.816
    alc3_o|t3         2.075    0.050   41.333    0.000    2.075    2.075
    alc3_o|t4         2.435    0.103   23.677    0.000    2.435    2.435
    alc4_o|t1         0.983    0.024   40.889    0.000    0.983    0.983
    alc4_o|t2         1.604    0.038   42.494    0.000    1.604    1.604
    alc4_o|t3         2.000    0.029   68.147    0.000    2.000    2.000
    alc4_o|t4         2.353    0.060   39.476    0.000    2.353    2.353
    alc5_o|t1         1.067    0.027   39.306    0.000    1.067    1.067
    alc5_o|t2         1.719    0.030   56.984    0.000    1.719    1.719
    alc5_o|t3         2.036    0.041   49.952    0.000    2.036    2.036
    alc5_o|t4         2.353    0.079   29.659    0.000    2.353    2.353

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
    gotBully          1.000                               1.000    1.000
   .depress           1.000                               0.808    0.808
   .alcohol           1.000                               0.861    0.861
   .gotBu1_o          0.435                               0.435    0.435
   .gotBu2_o          0.391                               0.391    0.391
   .gotBu3_o          0.395                               0.395    0.395
   .gotBu4_o          0.348                               0.348    0.348
   .gotBu5_o          0.295                               0.295    0.295
```

An indirect effect model with ordinal data ...

```
    .gotBu6_o        0.249                                              0.249      0.249
    .gotBu7_o        0.407                                              0.407      0.407
    .gotBu8_o        0.259                                              0.259      0.259
    .gotBu9_o        0.206                                              0.206      0.206
    .depre1_o        0.521                                              0.521      0.521
    .depre2_o        0.591                                              0.591      0.591
    .depre3_o        0.432                                              0.432      0.432
    .depre4_o        0.495                                              0.495      0.495
    .depre5_o        0.643                                              0.643      0.643
    .depre6_o        0.606                                              0.606      0.606
    .alc1_o          0.262                                              0.262      0.262
    .alc2_o          0.401                                              0.401      0.401
    .alc3_o          0.095                                              0.095      0.095
    .alc4_o          0.221                                              0.221      0.221
    .alc5_o          0.178                                              0.178      0.178

Scales y*:
                   Estimate   Std.Err   z-value   P(>|z|)    Std.lv   Std.all
     gotBu1_o        1.000                                              1.000      1.000
     gotBu2_o        1.000                                              1.000      1.000
     gotBu3_o        1.000                                              1.000      1.000
     gotBu4_o        1.000                                              1.000      1.000
     gotBu5_o        1.000                                              1.000      1.000
     gotBu6_o        1.000                                              1.000      1.000
     gotBu7_o        1.000                                              1.000      1.000
     gotBu8_o        1.000                                              1.000      1.000
```

An indirect effect model with ordinal data ...

```
    gotBu9_o          1.000                                        1.000    1.000
    depre1_o          1.000                                        1.000    1.000
    depre2_o          1.000                                        1.000    1.000
    depre3_o          1.000                                        1.000    1.000
    depre4_o          1.000                                        1.000    1.000
    depre5_o          1.000                                        1.000    1.000
    depre6_o          1.000                                        1.000    1.000
    alc1_o            1.000                                        1.000    1.000
    alc2_o            1.000                                        1.000    1.000
    alc3_o            1.000                                        1.000    1.000
    alc4_o            1.000                                        1.000    1.000
    alc5_o            1.000                                        1.000    1.000

Defined Parameters:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
    ab                0.140    0.026    5.295    0.000    0.129    0.129
    total             0.283    0.037    7.652    0.000    0.262    0.262
```

Outline

Measurement invariance testing - configural invariance

```
## Specifying the model-structure object
model.config.WLSMV <- '
 depress =~ depre1_o + depre2_o + depre3_o +
            depre4_o + depre5_o + depre6_o '
## Estimating the model
fit.config.WLSMV <-
 cfa(model = model.config.WLSMV, data = hbsc,
     mimic = "Mplus", estimator = "WLSMV",
     std.lv = TRUE, group = "Grade",
     ordered = c("depre1_o", "depre2_o",
        "depre3_o",
                 "depre4_o", "depre5_o",
                    "depre6_o"))
## Requesting an estimation summary
summary(fit.config.WLSMV, fit.measures = TRUE,
         standardized = TRUE)
```

Measurement invariance testing - configural invariance ...

```
lavaan 0.6-3 ended normally after 10 iterations

  Optimization method                             NLMINB
  Number of free parameters                           60

                                                    Used       Total
  Number of observations per group
  7                                                 1811        1880
  6                                                 2292        2404

  Estimator                                         DWLS      Robust
  Model Fit Test Statistic                       180.000     333.412
  Degrees of freedom                                  18          18
  P-value (Chi-square)                             0.000       0.000
  Scaling correction factor                                    0.541
  Shift parameter for each group:
    7                                                          0.240
    6                                                          0.304
    for simple second-order correction (WLSMV)

Chi-square for each group:

  7                                               68.577     127.057
  6                                              111.423     206.355
```

Measurement invariance testing - configural invariance ...

```
Model test baseline model:

  Minimum Function Test Statistic            16164.928   11074.106
  Degrees of freedom                                30          30
  P-value                                        0.000       0.000

User model versus baseline model:

  Comparative Fit Index (CFI)                    0.990       0.971
  Tucker-Lewis Index (TLI)                       0.983       0.952

  Robust Comparative Fit Index (CFI)                            NA
  Robust Tucker-Lewis Index (TLI)                               NA

Root Mean Square Error of Approximation:

  RMSEA                                          0.066       0.092
  90 Percent Confidence Interval          0.058  0.075       0.084
      0.101
  P-value RMSEA <= 0.05                          0.001       0.000

  Robust RMSEA                                                  NA
  90 Percent Confidence Interval                                NA
      NA

Standardized Root Mean Square Residual:
```

Measurement invariance testing - configural invariance ...

```
  SRMR                                        0.035       0.035

Weighted Root Mean Square Residual:

  WRMR                                        2.148       2.148

Parameter Estimates:

  Information                                 Expected
  Information saturated (h1) model          Unstructured
  Standard Errors                           Robust.sem


Group 1 [7]:

Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
  depress =~
    depre1_o          0.704    0.016   45.352    0.000    0.704    0.704
    depre2_o          0.649    0.016   39.702    0.000    0.649    0.649
    depre3_o          0.749    0.017   44.639    0.000    0.749    0.749
    depre4_o          0.733    0.016   46.213    0.000    0.733    0.733
    depre5_o          0.600    0.019   31.717    0.000    0.600    0.600
    depre6_o          0.619    0.018   34.057    0.000    0.619    0.619
```

Measurement invariance testing - configural invariance ...

```
Intercepts:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
   .depre1_o          0.000                                0.000    0.000
   .depre2_o          0.000                                0.000    0.000
   .depre3_o          0.000                                0.000    0.000
   .depre4_o          0.000                                0.000    0.000
   .depre5_o          0.000                                0.000    0.000
   .depre6_o          0.000                                0.000    0.000
    depress           0.000                                0.000    0.000

Thresholds:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
    depre1_o|t1      -0.591    0.031  -18.818    0.000   -0.591   -0.591
    depre1_o|t2       0.142    0.030    4.813    0.000    0.142    0.142
    depre1_o|t3       1.089    0.037   29.607    0.000    1.089    1.089
    depre1_o|t4       1.780    0.055   32.594    0.000    1.780    1.780
    depre2_o|t1      -0.945    0.035  -27.171    0.000   -0.945   -0.945
    depre2_o|t2      -0.214    0.030   -7.205    0.000   -0.214   -0.214
    depre2_o|t3       0.747    0.033   22.876    0.000    0.747    0.747
    depre2_o|t4       1.535    0.046   33.153    0.000    1.535    1.535
    depre3_o|t1       0.294    0.030    9.827    0.000    0.294    0.294
    depre3_o|t2       0.702    0.032   21.762    0.000    0.702    0.702
    depre3_o|t3       1.210    0.039   31.169    0.000    1.210    1.210
    depre3_o|t4       1.722    0.052   32.872    0.000    1.722    1.722
    depre4_o|t1      -0.094    0.030   -3.170    0.002   -0.094   -0.094
    depre4_o|t2       0.267    0.030    8.938    0.000    0.267    0.267
```

Measurement invariance testing - configural invariance ...

```
    depre4_o|t3       0.907    0.034    26.424    0.000    0.907    0.907
    depre4_o|t4       1.504    0.045    33.104    0.000    1.504    1.504
    depre5_o|t1      -0.394    0.030   -13.000    0.000   -0.394   -0.394
    depre5_o|t2       0.070    0.029     2.372    0.018    0.070    0.070
    depre5_o|t3       0.697    0.032    21.627    0.000    0.697    0.697
    depre5_o|t4       1.279    0.040    31.862    0.000    1.279    1.279
    depre6_o|t1      -0.481    0.031   -15.645    0.000   -0.481   -0.481
    depre6_o|t2       0.069    0.029     2.325    0.020    0.069    0.069
    depre6_o|t3       0.727    0.032    22.387    0.000    0.727    0.727
    depre6_o|t4       1.230    0.039    31.389    0.000    1.230    1.230

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
   .depre1_o          0.504                                0.504    0.504
   .depre2_o          0.578                                0.578    0.578
   .depre3_o          0.438                                0.438    0.438
   .depre4_o          0.463                                0.463    0.463
   .depre5_o          0.640                                0.640    0.640
   .depre6_o          0.617                                0.617    0.617
    depress           1.000                                1.000    1.000

Scales y*:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
    depre1_o          1.000                                1.000    1.000
    depre2_o          1.000                                1.000    1.000
    depre3_o          1.000                                1.000    1.000
```

Measurement invariance testing - configural invariance ...

```
    depre4_o          1.000                                       1.000    1.000
    depre5_o          1.000                                       1.000    1.000
    depre6_o          1.000                                       1.000    1.000


Group 2 [6]:

Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
  depress =~
    depre1_o          0.716    0.013   53.121    0.000    0.716    0.716
    depre2_o          0.659    0.015   44.443    0.000    0.659    0.659
    depre3_o          0.745    0.015   49.442    0.000    0.745    0.745
    depre4_o          0.701    0.015   47.571    0.000    0.701    0.701
    depre5_o          0.604    0.017   35.803    0.000    0.604    0.604
    depre6_o          0.566    0.018   31.571    0.000    0.566    0.566

Intercepts:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
   .depre1_o          0.000                                0.000    0.000
   .depre2_o          0.000                                0.000    0.000
   .depre3_o          0.000                                0.000    0.000
   .depre4_o          0.000                                0.000    0.000
   .depre5_o          0.000                                0.000    0.000
   .depre6_o          0.000                                0.000    0.000
    depress           0.000                                0.000    0.000
```

Measurement invariance testing - configural invariance ...

```
Thresholds:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
    depre1_o|t1      -0.543    0.028  -19.637    0.000   -0.543   -0.543
    depre1_o|t2       0.125    0.026    4.759    0.000    0.125    0.125
    depre1_o|t3       1.100    0.033   33.486    0.000    1.100    1.100
    depre1_o|t4       1.791    0.049   36.599    0.000    1.791    1.791
    depre2_o|t1      -0.815    0.030  -27.511    0.000   -0.815   -0.815
    depre2_o|t2      -0.177    0.026   -6.719    0.000   -0.177   -0.177
    depre2_o|t3       0.729    0.029   25.240    0.000    0.729    0.729
    depre2_o|t4       1.490    0.040   37.208    0.000    1.490    1.490
    depre3_o|t1       0.285    0.027   10.717    0.000    0.285    0.285
    depre3_o|t2       0.642    0.028   22.722    0.000    0.642    0.642
    depre3_o|t3       1.151    0.034   34.279    0.000    1.151    1.151
    depre3_o|t4       1.557    0.042   37.320    0.000    1.557    1.557
    depre4_o|t1      -0.117    0.026   -4.467    0.000   -0.117   -0.117
    depre4_o|t2       0.246    0.026    9.303    0.000    0.246    0.246
    depre4_o|t3       0.874    0.030   28.964    0.000    0.874    0.874
    depre4_o|t4       1.386    0.038   36.729    0.000    1.386    1.386
    depre5_o|t1      -0.350    0.027  -13.084    0.000   -0.350   -0.350
    depre5_o|t2       0.073    0.026    2.797    0.005    0.073    0.073
    depre5_o|t3       0.647    0.028   22.883    0.000    0.647    0.647
    depre5_o|t4       1.197    0.034   34.907    0.000    1.197    1.197
    depre6_o|t1      -0.388    0.027  -14.410    0.000   -0.388   -0.388
    depre6_o|t2       0.097    0.026    3.716    0.000    0.097    0.097
    depre6_o|t3       0.752    0.029   25.872    0.000    0.752    0.752
```

Measurement invariance testing - configural invariance ...

```
    depre6_o|t4       1.275    0.036   35.805    0.000    1.275    1.275

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
   .depre1_o          0.487                               0.487    0.487
   .depre2_o          0.565                               0.565    0.565
   .depre3_o          0.446                               0.446    0.446
   .depre4_o          0.509                               0.509    0.509
   .depre5_o          0.635                               0.635    0.635
   .depre6_o          0.680                               0.680    0.680
    depress           1.000                               1.000    1.000

Scales y*:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
    depre1_o          1.000                               1.000    1.000
    depre2_o          1.000                               1.000    1.000
    depre3_o          1.000                               1.000    1.000
    depre4_o          1.000                               1.000    1.000
    depre5_o          1.000                               1.000    1.000
    depre6_o          1.000                               1.000    1.000
```

Measurement invariance testing - metric/weak invariance

```
## Specifying the model-structure object
model.metric.WLSMV <- '
 depress =~ depre1_o + depre2_o + depre3_o +
            depre4_o + depre5_o + depre6_o
 depress ~~ c(1, NA)*depress '
## Estimating the model
fit.metric.WLSMV <-
 cfa(model = model.metric.WLSMV, data = hbsc,
     mimic = "Mplus", estimator = "WLSMV",
     std.lv = TRUE, group = "Grade",
     ordered = c("depre1_o", "depre2_o",
        "depre3_o", "depre4_o", "depre5_o",
        "depre6_o"),
     group.equal = "loadings")
## Requesting an estimation summary
summary(fit.metric.WLSMV, fit.measures = TRUE,
        standardized = TRUE)
```

Measurement invariance testing - metric/weak invariance ...

```
lavaan 0.6-3 ended normally after 11 iterations

  Optimization method                           NLMINB
  Number of free parameters                         61
  Number of equality constraints                     6

                                                  Used       Total
  Number of observations per group
  7                                               1811        1880
  6                                               2292        2404

  Estimator                                       DWLS      Robust
  Model Fit Test Statistic                     188.866     306.223
  Degrees of freedom                                23          23
  P-value (Chi-square)                           0.000       0.000
  Scaling correction factor                                  0.614
  Shift parameter for each group:
    7                                                       -0.654
    6                                                       -0.827
    for simple second-order correction (WLSMV)

Chi-square for each group:
```

Measurement invariance testing - metric/weak invariance ...

```
  7                                                73.253     118.692
  6                                               115.613     187.532

Model test baseline model:

  Minimum Function Test Statistic               16164.928   11074.106
  Degrees of freedom                                   30          30
  P-value                                           0.000       0.000

User model versus baseline model:

  Comparative Fit Index (CFI)                       0.990       0.974
  Tucker-Lewis Index (TLI)                          0.987       0.967

  Robust Comparative Fit Index (CFI)                               NA
  Robust Tucker-Lewis Index (TLI)                                  NA

Root Mean Square Error of Approximation:

  RMSEA                                             0.059       0.077
  90 Percent Confidence Interval          0.052     0.067       0.070
      0.085
  P-value RMSEA <= 0.05                             0.024       0.000
```

Measurement invariance testing - metric/weak invariance ...

```
  Robust RMSEA                                                  NA
  90 Percent Confidence Interval                                NA
      NA

Standardized Root Mean Square Residual:

  SRMR                                          0.036        0.036

Weighted Root Mean Square Residual:

  WRMR                                          2.201        2.201

Parameter Estimates:

  Information                                Expected
  Information saturated (h1) model       Unstructured
  Standard Errors                          Robust.sem


Group 1 [7]:

Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
  depress =~
```

Measurement invariance testing - metric/weak invariance ...

```
    depre1_ (.p1.)    0.716    0.012   57.884    0.000    0.716    0.716
    depre2_ (.p2.)    0.660    0.013   52.740    0.000    0.660    0.660
    depre3_ (.p3.)    0.753    0.013   56.179    0.000    0.753    0.753
    depre4_ (.p4.)    0.721    0.013   57.062    0.000    0.721    0.721
    depre5_ (.p5.)    0.607    0.014   43.681    0.000    0.607    0.607
    depre6_ (.p6.)    0.595    0.014   42.722    0.000    0.595    0.595

Intercepts:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
   .depre1_o          0.000                               0.000    0.000
   .depre2_o          0.000                               0.000    0.000
   .depre3_o          0.000                               0.000    0.000
   .depre4_o          0.000                               0.000    0.000
   .depre5_o          0.000                               0.000    0.000
   .depre6_o          0.000                               0.000    0.000
    depress           0.000                               0.000    0.000

Thresholds:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
    depre1_o|t1      -0.591    0.031  -18.818    0.000   -0.591   -0.591
    depre1_o|t2       0.142    0.030    4.813    0.000    0.142    0.142
    depre1_o|t3       1.089    0.037   29.607    0.000    1.089    1.089
    depre1_o|t4       1.780    0.055   32.594    0.000    1.780    1.780
    depre2_o|t1      -0.945    0.035  -27.171    0.000   -0.945   -0.945
```

Measurement invariance testing - metric/weak invariance ...

```
    depre2_o|t2      -0.214    0.030   -7.205    0.000   -0.214   -0.214
    depre2_o|t3       0.747    0.033   22.876    0.000    0.747    0.747
    depre2_o|t4       1.535    0.046   33.153    0.000    1.535    1.535
    depre3_o|t1       0.294    0.030    9.827    0.000    0.294    0.294
    depre3_o|t2       0.702    0.032   21.762    0.000    0.702    0.702
    depre3_o|t3       1.210    0.039   31.169    0.000    1.210    1.210
    depre3_o|t4       1.722    0.052   32.872    0.000    1.722    1.722
    depre4_o|t1      -0.094    0.030   -3.170    0.002   -0.094   -0.094
    depre4_o|t2       0.267    0.030    8.938    0.000    0.267    0.267
    depre4_o|t3       0.907    0.034   26.424    0.000    0.907    0.907
    depre4_o|t4       1.504    0.045   33.104    0.000    1.504    1.504
    depre5_o|t1      -0.394    0.030  -13.000    0.000   -0.394   -0.394
    depre5_o|t2       0.070    0.029    2.372    0.018    0.070    0.070
    depre5_o|t3       0.697    0.032   21.627    0.000    0.697    0.697
    depre5_o|t4       1.279    0.040   31.862    0.000    1.279    1.279
    depre6_o|t1      -0.481    0.031  -15.645    0.000   -0.481   -0.481
    depre6_o|t2       0.069    0.029    2.325    0.020    0.069    0.069
    depre6_o|t3       0.727    0.032   22.387    0.000    0.727    0.727
    depre6_o|t4       1.230    0.039   31.389    0.000    1.230    1.230

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
    depress           1.000                               1.000    1.000
   .depre1_o          0.487                               0.487    0.487
```

Measurement invariance testing - metric/weak invariance ...

```
    .depre2_o         0.565                                        0.565    0.565
    .depre3_o         0.434                                        0.434    0.434
    .depre4_o         0.480                                        0.480    0.480
    .depre5_o         0.632                                        0.632    0.632
    .depre6_o         0.646                                        0.646    0.646

Scales y*:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
     depre1_o         1.000                                        1.000    1.000
     depre2_o         1.000                                        1.000    1.000
     depre3_o         1.000                                        1.000    1.000
     depre4_o         1.000                                        1.000    1.000
     depre5_o         1.000                                        1.000    1.000
     depre6_o         1.000                                        1.000    1.000


Group 2 [6]:

Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
   depress =~
     depre1_ (.p1.)    0.716    0.012   57.884    0.000    0.706    0.706
     depre2_ (.p2.)    0.660    0.013   52.740    0.000    0.650    0.650
     depre3_ (.p3.)    0.753    0.013   56.179    0.000    0.742    0.742
```

Measurement invariance testing - metric/weak invariance ...

```
    depre4_  (.p4.)    0.721    0.013   57.062    0.000    0.711    0.711
    depre5_  (.p5.)    0.607    0.014   43.681    0.000    0.598    0.598
    depre6_  (.p6.)    0.595    0.014   42.722    0.000    0.587    0.587

Intercepts:
                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
   .depre1_o           0.000                               0.000    0.000
   .depre2_o           0.000                               0.000    0.000
   .depre3_o           0.000                               0.000    0.000
   .depre4_o           0.000                               0.000    0.000
   .depre5_o           0.000                               0.000    0.000
   .depre6_o           0.000                               0.000    0.000
    depress            0.000                               0.000    0.000

Thresholds:
                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
    depre1_o|t1       -0.543    0.028  -19.637    0.000   -0.543   -0.543
    depre1_o|t2        0.125    0.026    4.759    0.000    0.125    0.125
    depre1_o|t3        1.100    0.033   33.486    0.000    1.100    1.100
    depre1_o|t4        1.791    0.049   36.599    0.000    1.791    1.791
    depre2_o|t1       -0.815    0.030  -27.511    0.000   -0.815   -0.815
    depre2_o|t2       -0.177    0.026   -6.719    0.000   -0.177   -0.177
    depre2_o|t3        0.729    0.029   25.240    0.000    0.729    0.729
    depre2_o|t4        1.490    0.040   37.208    0.000    1.490    1.490
```

Measurement invariance testing - metric/weak invariance ...

```
    depre3_o|t1       0.285    0.027   10.717    0.000    0.285    0.285
    depre3_o|t2       0.642    0.028   22.722    0.000    0.642    0.642
    depre3_o|t3       1.151    0.034   34.279    0.000    1.151    1.151
    depre3_o|t4       1.557    0.042   37.320    0.000    1.557    1.557
    depre4_o|t1      -0.117    0.026   -4.467    0.000   -0.117   -0.117
    depre4_o|t2       0.246    0.026    9.303    0.000    0.246    0.246
    depre4_o|t3       0.874    0.030   28.964    0.000    0.874    0.874
    depre4_o|t4       1.386    0.038   36.729    0.000    1.386    1.386
    depre5_o|t1      -0.350    0.027  -13.084    0.000   -0.350   -0.350
    depre5_o|t2       0.073    0.026    2.797    0.005    0.073    0.073
    depre5_o|t3       0.647    0.028   22.883    0.000    0.647    0.647
    depre5_o|t4       1.197    0.034   34.907    0.000    1.197    1.197
    depre6_o|t1      -0.388    0.027  -14.410    0.000   -0.388   -0.388
    depre6_o|t2       0.097    0.026    3.716    0.000    0.097    0.097
    depre6_o|t3       0.752    0.029   25.872    0.000    0.752    0.752
    depre6_o|t4       1.275    0.036   35.805    0.000    1.275    1.275

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
    depress           0.973    0.033   29.079    0.000    1.000    1.000
   .depre1_o          0.501                               0.501    0.501
   .depre2_o          0.577                               0.577    0.577
   .depre3_o          0.449                               0.449    0.449
   .depre4_o          0.494                               0.494    0.494
```

Measurement invariance testing - metric/weak invariance ...

```
   .depre5_o        0.642                                        0.642    0.642
   .depre6_o        0.655                                        0.655    0.655

Scales y*:
                 Estimate   Std.Err   z-value   P(>|z|)   Std.lv   Std.all
    depre1_o        1.000                                  1.000    1.000
    depre2_o        1.000                                  1.000    1.000
    depre3_o        1.000                                  1.000    1.000
    depre4_o        1.000                                  1.000    1.000
    depre5_o        1.000                                  1.000    1.000
    depre6_o        1.000                                  1.000    1.000
```

```
## Conducting a corrected chi-square test
## Comparing the configural and the metric model
lavTestLRT(fit.config.WLSMV, fit.metric.WLSMV,
            method = "satorra.bentler.2010",
            A.method = "delta")
```

Measurement invariance testing - metric/weak invariance ...

```
Scaled Chi Square Difference Test (method = "satorra.bentler.2010")

                  Df AIC BIC  Chisq Chisq diff Df diff Pr(>Chisq)
fit.config.WLSMV 18          180.00
fit.metric.WLSMV 23          188.87     10.124       5     0.0718 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Measurement invariance testing - scalar/storng invariance

```
## Specifying the model-structure object
model.scalar.WLSMV <- '
 depress =~ depre1_o + depre2_o + depre3_o +
            depre4_o + depre5_o + depre6_o
 depress ~~ c(1, NA)*depress
 depress ~ c(0, NA)*1'
## Estimating the model
fit.scalar.WLSMV <-
 cfa(model = model.scalar.WLSMV, data = hbsc,
     mimic = "Mplus", estimator = "WLSMV",
     std.lv = TRUE, group = "Grade",
     ordered = c("depre1_o", "depre2_o",
        "depre3_o", "depre4_o", "depre5_o",
        "depre6_o"),
     group.equal = c("loadings", "thresholds"))
## Requesting an estimation summary
summary(fit.scalar.WLSMV, fit.measures = TRUE,
```

KU

Measurement invariance testing - scalar/storng invariance ...

```
            standardized = TRUE)
```

```
lavaan 0.6-3 ended normally after 27 iterations

  Optimization method                           NLMINB
  Number of free parameters                         68
  Number of equality constraints                    30

                                                  Used       Total
  Number of observations per group
  7                                               1811        1880
  6                                               2292        2404

  Estimator                                       DWLS      Robust
  Model Fit Test Statistic                     201.376     330.890
  Degrees of freedom                                40          40
  P-value (Chi-square)                           0.000       0.000
  Scaling correction factor                                  0.603
  Shift parameter for each group:
    7                                                       -1.415
    6                                                       -1.791
    for simple second-order correction (WLSMV)
```

Measurement invariance testing - scalar/storng invariance ...

```
Chi-square for each group:

  7                                              79.770      130.929
  6                                             121.605      199.960

Model test baseline model:

  Minimum Function Test Statistic             16164.928    11074.106
  Degrees of freedom                                 30           30
  P-value                                         0.000        0.000

User model versus baseline model:

  Comparative Fit Index (CFI)                     0.990        0.974
  Tucker-Lewis Index (TLI)                        0.992        0.980

  Robust Comparative Fit Index (CFI)                              NA
  Robust Tucker-Lewis Index (TLI)                                 NA

Root Mean Square Error of Approximation:

  RMSEA                                           0.044        0.060
  90 Percent Confidence Interval          0.038   0.051        0.054
      0.066
```

Measurement invariance testing - scalar/storng invariance ...

```
  P-value RMSEA <= 0.05                          0.933       0.004

  Robust RMSEA                                               NA
  90 Percent Confidence Interval                             NA
       NA

Standardized Root Mean Square Residual:

  SRMR                                           0.037       0.037

Weighted Root Mean Square Residual:

  WRMR                                           2.272       2.272

Parameter Estimates:

  Information                                 Expected
  Information saturated (h1) model        Unstructured
  Standard Errors                           Robust.sem


Group 1 [7]:

Latent Variables:
```

Measurement invariance testing - scalar/storng invariance ...

```
                      Estimate   Std.Err   z-value   P(>|z|)    Std.lv   Std.all
  depress =~
    depre1_  (.p1.)      0.701     0.013    52.266     0.000     0.701     0.701
    depre2_  (.p2.)      0.656     0.014    47.410     0.000     0.656     0.656
    depre3_  (.p3.)      0.760     0.015    51.150     0.000     0.760     0.760
    depre4_  (.p4.)      0.732     0.014    52.459     0.000     0.732     0.732
    depre5_  (.p5.)      0.607     0.016    38.229     0.000     0.607     0.607
    depre6_  (.p6.)      0.598     0.015    38.668     0.000     0.598     0.598

Intercepts:
                      Estimate   Std.Err   z-value   P(>|z|)    Std.lv   Std.all
    depress              0.000                                   0.000     0.000
   .depre1_o             0.000                                   0.000     0.000
   .depre2_o             0.000                                   0.000     0.000
   .depre3_o             0.000                                   0.000     0.000
   .depre4_o             0.000                                   0.000     0.000
   .depre5_o             0.000                                   0.000     0.000
   .depre6_o             0.000                                   0.000     0.000

Thresholds:
                      Estimate   Std.Err   z-value   P(>|z|)    Std.lv   Std.all
    dpr1_|1  (.p9.)     -0.585     0.027   -21.689     0.000    -0.585    -0.585
    dpr1_|2  (.10.)      0.121     0.024     4.979     0.000     0.121     0.121
    dpr1_|3  (.11.)      1.096     0.032    34.556     0.000     1.096     1.096
```

Measurement invariance testing - scalar/storng invariance ...

```
    dpr1_|4  (.12.)    1.795    0.046   39.323    0.000    1.795    1.795
    dpr2_|1  (.13.)   -0.910    0.029  -30.906    0.000   -0.910   -0.910
    dpr2_|2  (.14.)   -0.212    0.024   -8.750    0.000   -0.212   -0.212
    dpr2_|3  (.15.)    0.746    0.027   27.431    0.000    0.746    0.746
    dpr2_|4  (.16.)    1.542    0.039   39.662    0.000    1.542    1.542
    dpr3_|1  (.17.)    0.288    0.025   11.339    0.000    0.288    0.288
    dpr3_|2  (.18.)    0.685    0.028   24.794    0.000    0.685    0.685
    dpr3_|3  (.19.)    1.216    0.034   35.475    0.000    1.216    1.216
    dpr3_|4  (.20.)    1.687    0.044   38.010    0.000    1.687    1.687
    dpr4_|1  (.21.)   -0.125    0.025   -4.968    0.000   -0.125   -0.125
    dpr4_|2  (.22.)    0.255    0.025   10.072    0.000    0.255    0.255
    dpr4_|3  (.23.)    0.919    0.030   30.898    0.000    0.919    0.919
    dpr4_|4  (.24.)    1.496    0.039   38.146    0.000    1.496    1.496
    dpr5_|1  (.25.)   -0.395    0.025  -15.905    0.000   -0.395   -0.395
    dpr5_|2  (.26.)    0.063    0.023    2.678    0.007    0.063    0.063
    dpr5_|3  (.27.)    0.682    0.027   25.599    0.000    0.682    0.682
    dpr5_|4  (.28.)    1.265    0.035   36.570    0.000    1.265    1.265
    dpr6_|1  (.29.)   -0.458    0.025  -18.069    0.000   -0.458   -0.458
    dpr6_|2  (.30.)    0.076    0.024    3.238    0.001    0.076    0.076
    dpr6_|3  (.31.)    0.757    0.027   27.625    0.000    0.757    0.757
    dpr6_|4  (.32.)    1.289    0.034   37.377    0.000    1.289    1.289

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
```

Measurement invariance testing - scalar/storng invariance ...

```
    depress          1.000                                      1.000    1.000
   .depre1_o         0.509                                      0.509    0.509
   .depre2_o         0.569                                      0.569    0.569
   .depre3_o         0.423                                      0.423    0.423
   .depre4_o         0.465                                      0.465    0.465
   .depre5_o         0.632                                      0.632    0.632
   .depre6_o         0.643                                      0.643    0.643

Scales y*:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
    depre1_o         1.000                                      1.000    1.000
    depre2_o         1.000                                      1.000    1.000
    depre3_o         1.000                                      1.000    1.000
    depre4_o         1.000                                      1.000    1.000
    depre5_o         1.000                                      1.000    1.000
    depre6_o         1.000                                      1.000    1.000


Group 2 [6]:

Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
  depress =~
    depre1_ (.p1.)   0.701    0.013   52.266    0.000    0.736    0.719
```

Measurement invariance testing - scalar/storng invariance ...

```
    depre2_ (.p2.)    0.656    0.014   47.410    0.000    0.689    0.653
    depre3_ (.p3.)    0.760    0.015   51.150    0.000    0.798    0.736
    depre4_ (.p4.)    0.732    0.014   52.459    0.000    0.768    0.702
    depre5_ (.p5.)    0.607    0.016   38.229    0.000    0.637    0.598
    depre6_ (.p6.)    0.598    0.015   38.668    0.000    0.627    0.585

Intercepts:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
    depress          -0.036    0.037   -0.973    0.331   -0.034   -0.034
   .depre1_o          0.000                                0.000    0.000
   .depre2_o          0.000                                0.000    0.000
   .depre3_o          0.000                                0.000    0.000
   .depre4_o          0.000                                0.000    0.000
   .depre5_o          0.000                                0.000    0.000
   .depre6_o          0.000                                0.000    0.000

Thresholds:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
    dpr1_|1 (.p9.)   -0.585    0.027  -21.689    0.000   -0.585   -0.572
    dpr1_|2 (.10.)    0.121    0.024    4.979    0.000    0.121    0.118
    dpr1_|3 (.11.)    1.096    0.032   34.556    0.000    1.096    1.070
    dpr1_|4 (.12.)    1.795    0.046   39.323    0.000    1.795    1.754
    dpr2_|1 (.13.)   -0.910    0.029  -30.906    0.000   -0.910   -0.863
    dpr2_|2 (.14.)   -0.212    0.024   -8.750    0.000   -0.212   -0.201
```

Measurement invariance testing - scalar/storng invariance ...

```
    dpr2_|3 (.15.)   0.746   0.027   27.431   0.000   0.746   0.708
    dpr2_|4 (.16.)   1.542   0.039   39.662   0.000   1.542   1.462
    dpr3_|1 (.17.)   0.288   0.025   11.339   0.000   0.288   0.265
    dpr3_|2 (.18.)   0.685   0.028   24.794   0.000   0.685   0.631
    dpr3_|3 (.19.)   1.216   0.034   35.475   0.000   1.216   1.121
    dpr3_|4 (.20.)   1.687   0.044   38.010   0.000   1.687   1.556
    dpr4_|1 (.21.)  -0.125   0.025   -4.968   0.000  -0.125  -0.114
    dpr4_|2 (.22.)   0.255   0.025   10.072   0.000   0.255   0.233
    dpr4_|3 (.23.)   0.919   0.030   30.898   0.000   0.919   0.840
    dpr4_|4 (.24.)   1.496   0.039   38.146   0.000   1.496   1.368
    dpr5_|1 (.25.)  -0.395   0.025  -15.905   0.000  -0.395  -0.370
    dpr5_|2 (.26.)   0.063   0.023    2.678   0.007   0.063   0.059
    dpr5_|3 (.27.)   0.682   0.027   25.599   0.000   0.682   0.639
    dpr5_|4 (.28.)   1.265   0.035   36.570   0.000   1.265   1.187
    dpr6_|1 (.29.)  -0.458   0.025  -18.069   0.000  -0.458  -0.427
    dpr6_|2 (.30.)   0.076   0.024    3.238   0.001   0.076   0.071
    dpr6_|3 (.31.)   0.757   0.027   27.625   0.000   0.757   0.706
    dpr6_|4 (.32.)   1.289   0.034   37.377   0.000   1.289   1.203

Variances:
                  Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
    depress          1.102    0.073   15.039    0.000    1.000    1.000
   .depre1_o         0.506                               0.506    0.483
   .depre2_o         0.637                               0.637    0.573
```

Measurement invariance testing - scalar/storng invariance ...

```
   .depre3_o        0.540                                        0.540    0.459
   .depre4_o        0.607                                        0.607    0.507
   .depre5_o        0.730                                        0.730    0.643
   .depre6_o        0.755                                        0.755    0.657

Scales y*:
                 Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
    depre1_o        0.977    0.025   39.050    0.000    0.977    1.000
    depre2_o        0.948    0.023   41.409    0.000    0.948    1.000
    depre3_o        0.922    0.026   35.106    0.000    0.922    1.000
    depre4_o        0.914    0.025   36.803    0.000    0.914    1.000
    depre5_o        0.938    0.026   35.431    0.000    0.938    1.000
    depre6_o        0.933    0.026   35.854    0.000    0.933    1.000
```

```
## Conducting a corrected chi-square test
## Comparing the metric and the scalar model
lavTestLRT(fit.metric.WLSMV, fit.scalar.WLSMV,
            method = "satorra.bentler.2010",
            A.method = "delta")
```

Measurement invariance testing - scalar/storng invariance ...

```
Scaled Chi Square Difference Test (method = "satorra.bentler.2010")

                 Df AIC BIC  Chisq Chisq diff Df diff Pr(>Chisq)
fit.metric.WLSMV 23          188.87
fit.scalar.WLSMV 40          201.38     21.469      17      0.206
```

References

Arbuckle, J. L. (1996). Advanced structural equation modeling: Issues and techniques. In G. A. Marcoulides & R. E. Schumacker (Eds.), (p. 243-277). Mahwah, NJ: Lawrence Erlbaum. doi:10.4324/9781315827414

Beauducel, A., & Herzberg, P. Y. (2006). On the performance of maximum likelihood versus means and variance adjusted weighted least squares estimation in CFA. *Structural Equation Modeling*, *13*, 186-203. doi:10.1207/s15328007sem1302_2

Bollen, K. A., & Barb, K. H. (1981). Pearson's R and coarsely categorized measures. *American Sociological Review*, *46*, 232-239. doi:10.2307/2094981

References ...

Correll, J., Park, B., Judd, C. M., & Wittenbrink, B. (2002). The police officer's dilemma: Using ethnicity to disambiguate potentially threatening individuals. *Journal of Personality and Social Psychology: Attitudes and Social Cognition*, *83*, 1314-1329. doi:10.1037//0022-3514.83.6.1314

DiStefano, C., & Morgan, G. B. (2014). A comparison of diagonal weighted least squares robust estimation techniques for ordinal data. *Structural Equation Modeling*, *21*, 425-438. doi:10.1080/10705511.2014.915373

Flora, D. B., & Curran, P. J. (2004a). An empirical evaluation of alternative methods of estimation for confirmatory factor analysis with ordinal data. *Psychological Methods*, *9*, 466-491. doi:10.1037/1082-989x.9.4.466

References ...

Flora, D. B., & Curran, P. J. (2004b). An empirical evaluation of alternative methods of estimation for confirmatory factor analysis with ordinal data. *Psychological Methods*, *9*, 466-491. doi:10.1037/1082-989x.9.4.466

Göllner, R., Roberts, B. W., Damian, R. I., Lüdtke, O., Jonkmann, K., & Trautwein, U. (2017). Whose "strom and stress" is it? Parent and child reports of personality development in the transition to early adolescence. *Journal of Personalily*, *85*, 376-387. doi:10.1111/jopy.12246

Lee, S.-Y., & Song, X.-Y. (2003). Bayesian analysis of structural equation models with dichotomous variables. *Statistics in Medicine*, *22*, 3073-3088. doi:10.1002/sim.1544

References ...

Li, C. -H. (2016). Confirmatory factor analysis with ordinal data: Comparing robust maximum likelihood and diagonally weighted least squares. *Behavior Research Methods*, *48*, 936-949. doi:10.3758/s13428-015-0619-7

Muthén, B., & Kaplan, D. (1985). A comparison of some methodologies for the factor analysis of non-normal likert variables. *British Journal of Mathematical and Statistical Psycholgoy*, *38*, 171-189. doi:10.1111/j.2044-8317.1985.tb00832.x

Muthén, B. O. (1993). Testing structural equation models. In K. A. Bollen & S. J. Long (Eds.), (p. 205-243). CA: Newbury Park: Sage.

O'Brien, R. M. (1985). The relationship between ordinal measures and their underlying values: Why all the disagreement? *Quality and Quantity*, *19*, 265-277. doi:10.1007/bf00170998

References ...

Song, X.-Y., & Lee, S.-Y. (2002). Bayesian estimation and model selection of multivariate linear model with polytomous variables. *Multivariate Behavioral Research*, *37*, 453-477. doi:10.1207/s15327906mbr3704_02

Soto, C. J., & John, O. P. (2017a). The next Big Five Inventory (BFI-2): Developing and assessing a hierarchical model with 15 facets to enhence bandwidth, fidelity, and predictive power. *Journal of Personality and Social Psychology*, *113*, 117-143. doi:10.1037/pspp0000096

Soto, C. J., & John, O. P. (2017b). Short and extra-short forms of the Big Five Inventory-2: The BFI-2-S and BFI-2-XS. *Journal of Research in Personality*, *68*, 69-81. doi:10.1016/j.jrp.2017.02.004

Session

```
sessionInfo()
```

```
R version 3.6.0 (2019-04-26)
Platform: x86_64-pc-linux-gnu (64-bit)
Running under: Ubuntu 19.04

Matrix products: default
BLAS:   /usr/lib/x86_64-linux-gnu/atlas/libblas.so.3.10.3
LAPACK: /usr/lib/x86_64-linux-gnu/atlas/liblapack.so.3.10.3

locale:
 [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
      LC_TIME=en_US.UTF-8
 [4] LC_COLLATE=en_US.UTF-8     LC_MONETARY=en_US.UTF-8
      LC_MESSAGES=en_US.UTF-8
 [7] LC_PAPER=en_US.UTF-8       LC_NAME=C                  LC_ADDRESS=C
[10] LC_TELEPHONE=C             LC_MEASUREMENT=en_US.UTF-8
     LC_IDENTIFICATION=C

attached base packages:
[1] stats     graphics  grDevices utils     datasets  methods   base

other attached packages:
[1] plyr_1.8.4   lavaan_0.6-3 kutils_1.69
```

Session ...

20

```
loaded via a namespace (and not attached):
 [1] MASS_7.3-51.4  compiler_3.6.0 tools_3.6.0    foreign_0.8-71
     Rcpp_1.0.1     mnormt_1.5-5
 [7] pbivnorm_0.6.0 xtable_1.8-4   zip_2.0.2      openxlsx_4.1.0
     stats4_3.6.0
```