**Introduction to Latent Transition Analysis**

**Solution to Exercises**

1. **Test latent class measurement models for the indicators of behavioural and emotional difficulties collected in 1986 and 1988, i.e. ranti86 ranx86 rhypr86 rdep86 rpeer86 and ranti88 ranx88 rhypr88 rdep88 rpeer88, respectively. In other words, test latent class models of participants’ difficulties in 1986, and latent class models of participants’ difficulties in 1988. Test models that include *n* classes between 1 and 8.**

The Input, Output and .gh5 files relative to this exercise are in folder *Solutions🡪 1.Measurement models*. For example, a latent class model with three categories explaining variability in the observed behaviour difficulties in 1986 is defined by this input:

USEVAR=ranti86 ranx86 rhypr86 rdep86 rpeer86;

CATEGORICAL=ranti86 ranx86 rhypr86 rdep86 rpeer86;

Classes=x(3);

IDVAR=cpubid\_xrnd;

Analysis:

Type = MIXTURE ;

STARTS=2000 200;

PROCESS=4;

MODEL:

OUTPUT:

Tech11 tech14 svalues standardized cinterval;

PLOT:

type=PLOT3;

series= ranx86(1) rdep86(2) ranti86(3) rhypr86(4) rpeer86(5);

Note that I have called the latent variable explaining 1986 behaviours “**x**”. The files in the folder represent inputs for models where the number of classes of the latent class variable at each time point increases from 1 to 8.

1. **Inspect the outputs from the previous task to select the optimal measurement models of individuals’ behaviour difficulties in 1986 and the optimal measurement models of individuals’ behaviour difficulties in 1988.**

For this purpose, one has to inspect the outputs of the files in folder *Solutions🡪 1.Measurement models🡪Outputs and graphs*.

To compare the models, one can extract the statistics of interest from each output file. However, this tedious task can be facilitated by using the *Mplus Automation* library in R, see **Hallquist, M. N. & Wiley, J. F. (2018). MplusAutomation: An R Package for Facilitating Large-Scale Latent Variable Analyses in Mplus. Structural Equation Modeling, 25, 621-638. doi: 10.1080/10705511.2017.1402334.**

Using Mplus Automation in R, I have produced a similar table that reports the key statistics of interest in comparing the latent class models in 1986:

A picture containing text, screenshot, number, parallel

Description automatically generated

The results indicate that there is not a single optimal solution. For example, the BIC would favour a 3-class solution (lower information criteria indicate more adequate models), while the aBIC would indicate a 4-class solution as preferable. Other statistics indicate other models as the optimal ones.

A similar problem is also evident when we consider the latent class solutions in 1988:

A picture containing text, screenshot, number, font

Description automatically generated

Once again, the BIC favours a 3-class solution, while the aBIC favours a 4-class solution, and other statistics indicate other models.

Finding the more adequate solutions should also include considerations regarding the plausibility of the models, their interpretability, as well as other considerations about their fit (e.g. inspection of model residuals). Furthermore, scholars suggest retaining a pool of models for further analysis in successive LTA stages.

1. **Consider a model with 3 latent classes in 1986 and 3 latent classes in 1988, and test full measurement invariance across time.**

The Input and Output files for this problem are in folder 2*.Measurement models across time* of the solutions. The file *NLS\_86\_88\_c3\_no\_measinv* specifies a model *without* measurement invariance, i.e. where measurement parameters of the two latent variables in 1986 and 1988 are freely estimated. The file *NLS\_86\_88\_c3\_measinv* specifies a model with *full measurement invariance* across time.

In particular, the latter input specifies two categorical latent variables, both with 3 classes:

[…]

USEVAR=ranti86 ranx86 rhypr86 rdep86 rpeer86

ranti88 ranx88 rhypr88 rdep88 rpeer88;

CATEGORICAL=ranti86 ranx86 rhypr86 rdep86 rpeer86

ranti88 ranx88 rhypr88 rdep88 rpeer88;

**Classes=x(3) y(3);**

[…]

I called the first categorical latent variable **x**, and the second **y** (you could give them any other name, say dis86, and dis88).

To ensure that latent variable **x** relates only to behaviours in 1986 and latent variable **y** relates only to behaviours in 1988, the associations between latent variables and indicators are specified in the **MODEL:** command as follows:

%OVERALL%

MODEL x:

**%x#1%**

[ ranti86$1\*-2 ] (x1a1);

[ ranti86$2\*-1 ] (x1a2);

[ ranx86$1\*-2 ] (x1b1);

[ ranx86$2\*-1 ] (x1b2);

[ rhypr86$1\*-2 ] (x1c1);

[ rhypr86$2\*-1 ] (x1c2);

[ rdep86$1\*-2 ] (x1d1);

[ rdep86$2\*-1 ] (x1d2);

[ rpeer86$1\*-2 ] (x1e1);

[ rpeer86$2\*-1 ] (x1e2);

**%x#2%**

[ ranti86$1\*0 ] (x2a1);

[ ranti86$2\*1 ] (x2a2);

[ ranx86$1\*0 ] (x2b1);

[ ranx86$2\*1 ] (x2b2);

[ rhypr86$1\*0 ] (x2c1);

[ rhypr86$2\*1] (x2c2);

[ rdep86$1\*0 ] (x2d1);

[ rdep86$2\*1 ] (x2d2);

[ rpeer86$1\*0 ] (x2e1);

[ rpeer86$2\*1 ] (x2e2);

**%x#3%**

[ ranti86$1\*1] (x3a1);

[ ranti86$2\*2 ] (x3a2);

[ ranx86$1\*1 ] (x3b1);

[ ranx86$2\*2 ] (x3b2);

[ rhypr86$1\*1 ] (x3c1);

[ rhypr86$2\*2 ] (x3c2);

[ rdep86$1\*1 ] (x3d1);

[ rdep86$2\*2 ] (x3d2);

[ rpeer86$1\*1 ] (x3e1);

[ rpeer86$2\*2 ] (x3e2);

MODEL y:

**%y#1%**

[ ranti88$1\*-2 ] (x1a1);

[ ranti88$2\*-1 ] (x1a2);

[ ranx88$1\*-2 ] (x1b1);

[ ranx88$2\*-1 ] (x1b2);

[ rhypr88$1\*-2 ] (x1c1);

[ rhypr88$2\*-1 ] (x1c2);

[ rdep88$1\*-2 ] (x1d1);

[ rdep88$2\*-1 ] (x1d2);

[ rpeer88$1\*-2 ] (x1e1);

[ rpeer88$2\*-1 ] (x1e2);

**%y#2%**

[ ranti88$1\*0 ] (x2a1);

[ ranti88$2\*1 ] (x2a2);

[ ranx88$1\*0 ] (x2b1);

[ ranx88$2\*1 ] (x2b2);

[ rhypr88$1\*0 ] (x2c1);

[ rhypr88$2\*1] (x2c2);

[ rdep88$1\*0 ] (x2d1);

[ rdep88$2\*1 ] (x2d2);

[ rpeer88$1\*0 ] (x2e1);

[ rpeer88$2\*1 ] (x2e2);

**%y#3%**

[ ranti88$1\*1] (x3a1);

[ ranti88$2\*2 ] (x3a2);

[ ranx88$1\*1 ] (x3b1);

[ ranx88$2\*2 ] (x3b2);

[ rhypr88$1\*1 ] (x3c1);

[ rhypr88$2\*2 ] (x3c2);

[ rdep88$1\*1 ] (x3d1);

[ rdep88$2\*2 ] (x3d2);

[ rpeer88$1\*1 ] (x3e1);

[ rpeer88$2\*2 ] (x3e2);

Note that I have not imposed any structural relationship between **x** and **y** in this MODEL: (there is no model specification after %OVERALL% which pertains to all latent variables).

The constraints that ensure the two classes have the same measurement parameters are provided by the strings between parentheses following the variables’ thresholds. For example, the same strings are used for the thresholds of the antisocial behaviour variable **ranti86** and **ranti88** in latent class 1 of **x** and in latent class 1 of **y**:

[…]

MODEL x:

**%x#1%**

[ ranti86$1\*-2 ] **(x1a1)**;

[ ranti86$2\*-1 ] **(x1a2)**;

[…]

MODEL y:

**%y#1%**

[ ranti88$1\*-2 ] **(x1a1)**;

[ ranti88$2\*-1 ] **(x1a2)**;

[…]

This ensures that the probability of answering the item “antisocial behaviour” are the same for respondents in 1986 latent class 1 as those for respondents in 1988 latent class 1, see this extract from the output:

A screenshot of a calculator

Description automatically generated with low confidence

A picture containing text, font, screenshot, white

Description automatically generated

Note that Mplus provides the item response conditional probabilities for latent class patterns. For example, **Latent Class Pattern 1 1** indicates **x**=1 and **y**=1, **Latent Class Pattern 1 2** indicates **x**=1 and **y**=2, and so on.

To compare the two models using a Likelihood Ratio Test (LRT), we will have to consider the Log-Likelihood of the two models, their number of parameters, and their scaling factors, since the LRT must take into account these scaling factors when we use the MLR estimator (as by default in Mplus):

The formulas for the LRT when using a MLR estimator are:

LR test = ; ; Degrees of freedom = ;

Where:

L0  = Log-Likelihood of the null model (that with equality constraints);

L1  = Log-Likelihood of the model without the added constraints;

c0  = Scaling correction factor of the null model (that with equality constraints);

c1  = Scaling correction factor of the model without the added constraints;

p0  = Free parameters in the null model (that with equality constraints);

p1  = Free parameters in the model without the added constraints;

The relevant statistics of the model with no measurement invariance are:

A white background with black text

Description automatically generated with low confidence

While those of the model with measurement invariance are:

A white background with black text

Description automatically generated with low confidence

Using these, we can calculate that the LRT *χ2*(30) = 60.26, which corresponds to *p* = 0.001. We should thus reject the null hypothesis that two models provide a similar fit, and we therefore have to reject the more restrictive model with measurement invariance.

However, inspection of the two models indicate that the conditional probabilities (measurement parameters) of the latent variables in 1986 and 1988 are similar. Furthermore, information criteria indicate that the model with measurement invariance provides a more adequate fit. The approximation of the LRT to a *χ2* distribution has been questioned in the case of models with latent variables, and this may be particularly relevant to LTA whereby categorical variables may create cross-tabs with sparse data.

1. **Assume a model with full measurement invariance with 3 classes in 1986 and 1988. Use the Three-Step Approach to investigate the associations between latent classes in 1986 and in 1988.**

The relative input and output files are in folder *Solutions*🡪*3.LTA with 3-step approach*.

Since I am imposing full measurement invariance to the latent variables in 1986 and 1988, the first step of the Three-Step approach is preceded by estimation of the invariant measurement parameters (see Section *The Three-Step Approach applied to LTA: Models with and without measurement invariance*  of document *1.1 Three-Step Approach\_OliverPerra\_IntroLTA*).

The measurement parameters I need are extracted from the starting values of the file with measurement invariance produced for the previous exercise, *nls\_86\_88\_c3\_measinv.out*.

In particular:

A screenshot of a computer code

Description automatically generated with medium confidence

Taking these values, I can estimate latent class affiliation of individuals into latent class **x** in 1986. The relative input file is: *Step1\_c86\_NLS\_c3x3\_measinv.inp*.

In this file, I estimated a latent class **x** with 3 categories, using the indicators collected in 1986:

VARIABLES:

[…]

USEVAR=ranti86 ranx86 rhypr86 rdep86 rpeer86;

CATEGORICAL=ranti86 ranx86 rhypr86 rdep86 rpeer86;

Classes=x(3) ;

IDVAR=cpubid\_xrnd;

Analysis:

Type = MIXTURE ;

STARTS=0;

Note that I have instructed **STARTS=0** insofar the measurement parameters of the models are fixed to the values provided by the full measurement invariance model. In fact, the **MODEL:** command specifies these fixed values for the measurement parameters:

MODEL:

%OVERALL%

[ x#1@0.38849 ];

[ x#2@-0.85835 ];

%x#1%

[ ranti86$1@-0.30531 ] (x1a1);

[ ranti86$2@2.26673 ] (x1a2);

[ ranx86$1@-0.69807 ] (x1b1);

[ ranx86$2@1.87243 ] (x1b2);

[ rhypr86$1@-1.45152 ] (x1c1);

[ rhypr86$2@0.89495 ] (x1c2);

[ rdep86$1@-0.27052 ] (x1d1);

[ rdep86$2@2.44723 ] (x1d2);

[ rpeer86$1@0.38220 ] (x1e1);

[ rpeer86$2@2.42341 ] (x1e2);

%x#2%

[…]

Finally, I instructed Mplus to save a data file where the latent class affiliation of each participant is saved (together with the relative probability of being in each latent class):

**SAVEDATA:**

File=step1\_nlsy\_86\_3x3.txt;

save=Cprob;

Since I also included respondents’ IDs (variable **cpubid\_xrnd**) in the **VARIABLE:** command, see:

IDVAR=cpubid\_xrnd;

the data file that Mplus will create and that will include latent class affiliation, will also include the participants’ ID. In this way, I can use other software to match-merge respondents’ the latent class affiliation to other variables.

The output created by *Step1\_c86\_NLS\_c3x3\_measinv.inp* also provides the information about latent class allocation uncertainty that I will feed to the model in Step 3 (see Section *The Three-Step Approach in Mplus* of document *1.1 Three-Step Approach\_OliverPerra\_IntroLTA*).

In particular, the output returns the logits for the classification probabilities:

*A picture containing text, font, line, screenshot

Description automatically generated*

which represent the information concerning classification error, i.e. Step 2 in the Three-Step Approach.

After applying the same procedure to the latent class variable and indicators in 1988 (see *Step1\_c88\_NLS\_c3x3\_measiv.inp*, its output and the data file created), I match-merge the files created by Mplus which contain participants’ most-likely class affiliation with the rest of the data, creating a new data file called:

**NLS\_extract\_2.dta.dat.**

This data file contains the same variables as the previous NLS extract file, with the addition of two variables **x** and **y**, which represent participants’ most likely class affiliation in 1986 and 1988 respectively.

These two variables are subsequently considered as nominal indicators. In input file *NLS\_LTA\_3x3.inp*, I used these variables as indicators of latent class:

VARIABLES :

[…]

**USEVAR=x y;**

**NOMINAL=x y;**

**CLASSES= ex(3) ey(3);**

IDVAR=cpubid\_xrnd;

Analysis:

Type = MIXTURE ;

STARTS=0;

Note that I called the latent classes in this input file **ex** and **ey**, intended to relate to indicators **x** and **y** respectively, and they both have 3 classes. Also note that, once again, the number of starts in command **ANALYSIS:** is constrained to be zero, since the measurement model specified by this input is fixed.

In fact, the rest of the model instructs a regression of **ey** on **ex**, while the associations between latent classes and their respective nominal indicators (measurement parameters) are fixed to the values estimated in Step 2, i.e., the logits of classification probabilities reported above:

MODEL:

%OVERALL%

ey ON ex;

MODEL ex:

%ex#1%

[x#1 **@ 2.101**];

[x#2 **@ -0.906**];

%ex#2%

[x#1 **@ 4.397**];

[x#2 **@ 5.719**];

%ex#3%

[x#1 **@ -1.447**];

[x#2 **@ -10.521**];

The output of this model reports the prevalence of the latent classes:

A picture containing text, screenshot, font, number

Description automatically generated

As well as the latent transition probabilities:

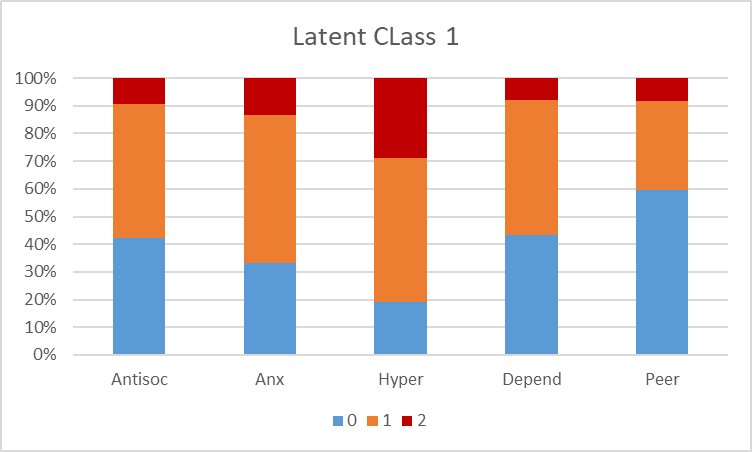
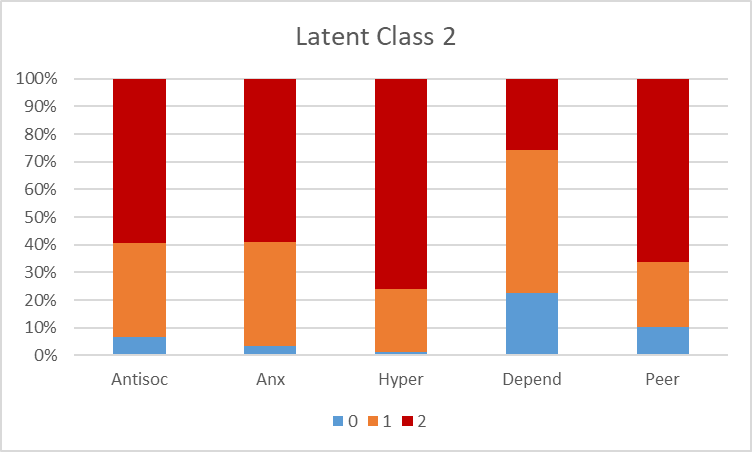
A picture containing text, screenshot, font, number

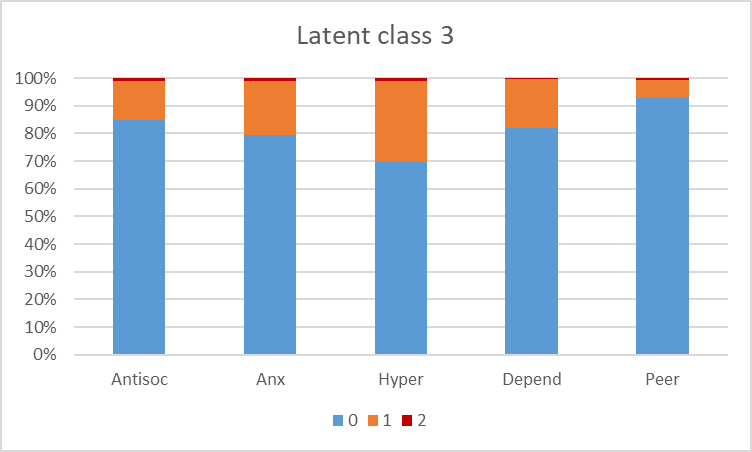
Description automatically generated

1. **Considering that the model with three classes at each age indicates these classes as representing respondents with “Low” , “Moderate”, and “High” levels of difficulties, define a model where the transition probabilities from Low to High and from High to Low are constrained to be zero. Compare this model with the model where transition probabilities are freely estimated using the Likelihood Ratio test.**

The meaning of the classes estimated cannot be interpreted from the output of Step 3 of the Three-Step Approach. To understand the meaning of the classes we need to go back to Step 1 and inspect the associations between the behaviour indicators and the latent classes (see also *nls\_86\_88\_c3\_measinv.out*, which preceded Step 1 providing the starting values of the measurement invariance model). Inspection of these files indicate that in the files produced, latent class 1 corresponds to a class of individuals with “moderate” levels of problem behaviour, latent class 2 corresponds to individuals with “high” levels of problem behaviour, and latent class 3 corresponds to individuals with “low” levels of problem behaviour. Because I imposed full measurement invariance, the meaning of these classes was the same in 1986 and 1988.

The item responses stacked conditional probabilities by classes are presented here:





The transition probabilities across time have been reported above as the output of *NLS\_LTA\_3x3.inp*:

A picture containing text, screenshot, font, number

Description automatically generated

Which indicate that the probability of transitions from High@86 to Low@88 (categories 2 and 3, respectively) is .080, while the probability of transitions from Low@86 to High@88 (categories 3 and 2, respectively) is .019. Since these probabilities are relatively small, we may hypothesise we observe these just as an effect of measurement error, and thus constrain these to be equal to zero.

To test this hypothesis, we need to specify a model with these constraints on the transition probabilities, and then compare the constrained and the unconstrained model using the Likelihood Ratio Test (LRT). To impose these constraints, we can use the probability parameterization option in the **ANALYSIS:** command (see *Stage 4* in document *1.2 Stages of LTA\_IntroLTA\_OPerra*). The relative files for these comparisons are in folder *4.LTA with constrained trans.prob* of the Solutions.

The file *NLS\_LTA\_3x3\_para\_prob.inp* specifies a model where the transition probabilities are freely estimated using the probability parameterization. The file *NLS\_LTA\_3x3\_const* runs the model with constraints on the transition probabilities.

To understand how to specify these constraints, consider that the probabilities parameters that Mplus consider are these:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **ey#1** | **ey#2** | **ey#3** |
| **ex#1** | *p11* | *p12* | *0* |
| **ex#2** | *p21* | *p22* | *0* |
| **ex#3** | *p31* | *p32* | *0* |

Namely, since the probabilities sum up to 1 in the rows, the probabilities in the last column are not estimated (these probabilities are provided by subtracting from 1 the sum of the other two probabilities in the rows).

In Mplus syntax these parameters take the following names:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **ey#1** | **ey#2** | **ey#3** |
| **ex#1** | ey#1 ON ex#1 | ey#2 ON ex#1 | 0 |
| **ex#2** | ey#1 ON ex#2 | ey#2 ON ex#2 | 0 |
| **ex#3** | ey#1 ON ex#3 | ey#2 ON ex#3 | 0 |

Since we want to fix the probability of transitions from Low@86 to High@88 (categories 3 and 2, respectively) the parameter we must fix to value 0 is **ey#2 ON ex#3@0**.

The other probability we want to fix to zero is that of moving from High@86 to Low@88 (categories 2 and 3, respectively) to zero. However, based on the table above, we do not have an **ey#3 ON ex#2** parameter to fix. Therefore, to ensure that the probability of the cell that corresponds to **ey#3 ON ex#2** is zero, we need to ensure that the sum of the other two parameters in the row, **ey#1 ON ex2** and **ey#2 ON ex#2**, is equal to 1.

To do so, we need to name these parameters in the **MODEL:** command, and then impose some **MODEL CONSTRAINT:** that ensures the sum of these two parameters is 1:

Analysis:

Type = MIXTURE ;

STARTS=0;

**PARAMETERIZATION=PROBABILITY;**

MODEL:

%OVERALL%

ey ON ex;

**ey#2 ON ex#3@0;**

**ey#1 ON ex#2 (p1);**

**ey#2 ON ex#2 (p2);**

[…]

MODEL CONSTRAINT:

**p2=1-p1;**

The constrains of the two parameters I called **p1** and **p2** , that is, **p2=1-p1**, may appear unsual (a statement like p2+p1=1 would have been more straightforward). However, moving the terms of the equation p2=1-p1, it will result that p2+p1=1. This indirect way to express the model constraint is dictated by the fact that model constraints in Mplus should contain either a named parameter (e.g. **p1** or **p2**) or zero as the first term on the left side of the equal sign.

The input above produces a model with constrained transition probabilities as desired:

A picture containing text, screenshot, font, number

Description automatically generated

The log-likelihood, scale factors and number of parameters of the model whereby the transition probabilities were freely estimated are:

A white background with black text

Description automatically generated with low confidence

while those of the model with constrained transition probabilities are:

A white background with black text

Description automatically generated with low confidence

The LRT test with scaling correction factors returns *χ2* (2) =12.93, with *p* = .002. Therefore, we would reject the null hypothesis of no difference, and therefore retain the less constrained model whereby transition probabilities are freely estimated.

1. **Use covariates male, bla\_his (ethnicity) and csage86 (age), and test their associations with the latent classes in 1986 and 1988.**

The relative input and output files are in folder *5.LTA with covariates and distal outcomes* of the Solutions. In particular, the input *NLS\_LTA\_3x3\_cov.inp* specifies a model where latent classes in 1986 (latent variable **ex**) and in 1988 (latent variable **ey**) are regressed on the covariates, while the measurement model parameters have been fixed (according to the Three-Step Approach).

The regression of the latent classes on the covariates are specified in the **MODEL:** command:

MODEL:

%OVERALL%

ey ON ex;

**ex ON csage86 male bla\_his;**

**ey ON csage86 male bla\_his;**

**[…]**

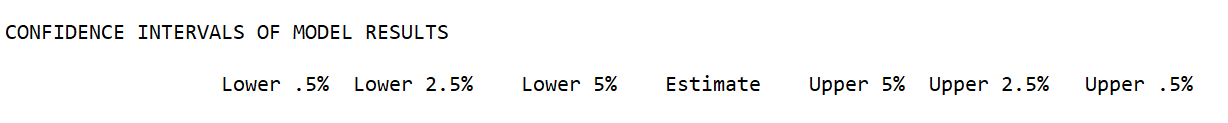
Mplus reports the regression parameters in the logit scale as well as Odds Ratios:

A screenshot of a computer

Description automatically generated with medium confidence

By specifying **CINTERVAL** in command **OUTPUT:** we can also obtain the confidence intervals of these Odds Ratios:

A screenshot of numbers

Description automatically generated with low confidence

1. Add to the previous model a moderation effect of **male** and interpret the output.
2. Investigate the associations between latent classes in 1988 and participants’ **ppvtz90**. What do the results indicate?

Solutions to these tasks are provided in a different file.

For example, we can see that, compared to females, males have *OR*=2.32 of being in latent class 2 @ 1988 (i.e. High level of problem behaviour) rather being in latent class 3 @1988 (the reference category, and the Low level category), and the 95% *CI* of the *OR* are 1.44 to 3.74. Therefore, compared to females, males display over a 2-fold increase in the odds of high levels rather than low levels of problem behaviour, once we control for their previous level of problem behaviour.

By instructing Mplus to include **TECH15** in the **OUTPUT:** command, it is possible to obtain calculations of the transition probabilities fixing the covariates at different values. Once the output file is generated, this option will appear in the menu:

A screenshot of a computer

Description automatically generated with medium confidence

Which will provide a drop-down menu that includes **LTA Calculator**. Clicking on **LTA Calculator** will open a dialog window like this one:

A screenshot of a computer

Description automatically generated

This allows to give a name to the “scenario” of covariates set and their values. Once you give a name to this set of covariates values, you can specify the desired values of the covariates:

A screenshot of a computer

Description automatically generated

Mplus will consider the sample mean of the covariates by default, but you can specify other values and set them.

The output produced in the example I am using will look like this:

A screenshot of a computer

Description automatically generated with medium confidence

The example above reports transition probabilities for males who are not Black or Hispanic, and have age fixed at the sample mean. The same probabilities for females who have the same values in the other covariates are as follow:

A screenshot of a computer

Description automatically generated with medium confidence

1. **Add to the previous model a moderation effect of male and interpret the output.**

The relevant input file in folder *5.LTA with covariates and distal outcomes*  is *NLS\_LTA\_3x3\_cov\_moderation.inp*. Some hints on how this model can be specified were provided in Section *Stage 5* of the document *1.2 Stages of LTA\_IntroLTA\_OPerra*.

The moderation effect is specified by instructing Mplus to estimate regressions of **ey** categories on gender within each category of **ex**:

MODEL:

%OVERALL%

ey ON ex;

ex ON csage86 male bla\_his;

ey ON csage86 bla\_his;

**MODEL ex:**

**%ex#1%**

[x#1 @ 2.101];

[x#2 @ -0.906];

**ey#1 ON male ;**

**ey#2 ON male** ;

**%ex#2%**

[x#1 @ 4.397];

[x#2 @ 5.719];

**ey#1 ON male ;**

**ey#2 ON male ;**

**%ex#3%**

[x#1 @ -1.447];

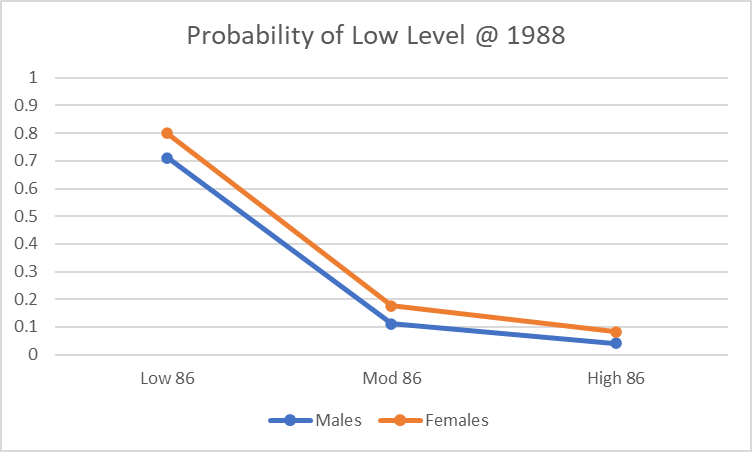
[x#2 @ -10.521];

**ey#1 ON male ;**

**ey#2 ON male ;**

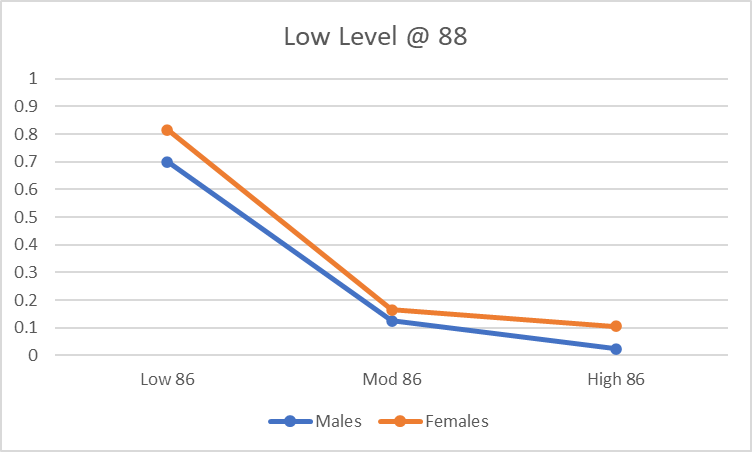
[…]

To understand the effect of this moderation on the transition probabilities, we can consider firstly the results of the model without a moderation effect. I focus here the probabilities of being in the latent class characterised by low level of problem behaviours in 1988 for males and females. In the graph below, I report these probabilities according to the latent class status in 1986. The different lines represent the probabilities of males and females:



The two lines appear roughly parallel, indicating that the associations between latent variables across time are similar in strength and sign for male and females.

However, the graph looks different when gender is considered a moderating variable of these relationships. The next graph reports the probability of being in the class characterised by low level of problem behaviours in 1988, depending on latent class affiliation two years before, 1986. Different lines represent the probabilities of males and females:



Overall, it appears that females may be more likely to show more radical transitions than males (see also the LTA Calculator created by the moderation model).

1. **Investigate the associations between latent classes in 1988 and participants’ ppvtz90. What do the results indicate?**

The relevant input file in folder *5.LTA with covariates and distal outcomes*  is *NLS\_LTA\_3x3\_cov\_distal\_outcome.inp*. Some hints on how this model can be specified were provided in Section *Stage 6* of the document *1.2 Stages of LTA\_IntroLTA\_OPerra*.

The variable **ppvtz90**  is continuous (and standardised), so we can ask Mplus to estimate the average values and variances of this outcome across the three classes of **ey**. However, since there are other covariates in the model, we can also instruct Mplus to control for the effect of these covariates on the distal outcome:

MODEL:

%OVERALL%

ey ON ex;

ex ON csage86 male bla\_his;

ey ON csage86 male bla\_his;

**ppvtz1990 ON csage86 male bla\_his;**

**[…]**

MODEL ey:

**%ey#1%**

[y#1 @ 2.036];

[y#2 @ -1.023];

**[ppvtz1990];**

**ppvtz1990;**

**%ey#2%**

[y#1 @ 4.627];

[y#2 @ 5.975];

**[ppvtz1990] (p1);**

**ppvtz1990;**

**%ey#3%**

[y#1 @ -1.536];

[y#2 @ -9.644];

**[ppvtz1990] (p2);**

**ppvtz1990;**

[…]

The statements **[ppvtz19980]** within **%ey#1%** etc. is asking Mplus to estimate the adjusted average of the outcome within latent class 1 of **ey**, and so on. This is the adjusted average since, at the same time, we are also controlling for the regression of **ppvtz90**  on other covariates in the model (gender, ethnicity, age). The **ppvtz1990** within **%ey#1%** etc. is asking Mplus to estimate the residual variance of the outcome within latent class 1 of **ey**, and so on.

The adjusted averages of **ppvtz90**  in **%ey#2%** and**%ey#3%**  are given names (**p1** and **p2** respectively) because this allows to ask for a Wald test that compares whether these parameters are similar:

MODEL TEST:

**0=p1-p2;**

The output reports the results of the Wald test as follows:

A white background with black text

Description automatically generated with low confidence

which suggests we can reject the null hypothesis that the two parameters are identical.

The output also provides the estimated values of these adjusted averages:

A picture containing text, screenshot, font, number

Description automatically generated

A picture containing text, screenshot, font, number

Description automatically generated

Remember that **Latent Class Pattern 1 2** indicates pattern **ex**=1 and **ey**=2, and so on. Therefore, the estimated adjusted average PPVT scores of those in the Low behaviour problems level (latent class 3) was 101.43, whereas the adjusted average PPVT scores of those in the High behaviour problems level (latent class 2) was 95.07.