N-Level Structural Equations Modeling

xxM User's Guide

Version 1.0

Paras Mehta

2013

1 ABOUT XXM

xxM is an R package for Structural Equations Modeling with complex dependent data structures. **xxM** implements a modeling framework called n-Level Structural Equation Modeling (NL-SEM). In other words, xxM allows models with any number of levels. Observed and latent variables are allowed at all levels. A conventional SEM model may be specified for each level and across-any two levels. Random-effects of observed variables are allowed both within and across levels.

1.1 WEBSITE

- http://xxm.times.uh.edu

1.2 MODEL SPECIFICATION

- Uses a simple graphical representation for fairly complex NL-SEM models. There is one-to-one correspondence between the graphical and mathematical representation of the model. ‘If you can draw the model, you can estimate the model’.
- Uses a LEGO like ‘building-block’ approach for constructing models. Once the basic building blocks are understood, arbitrarily complex models are constructed by repeating the same key steps.
- Uses standard SEM matrices for specifying the model. If you already know LISREL, transitioning to **xxM** would be easy.

1.3 DEPENDENT DATA-STRUCTURES

- Hierarchically nested data (e.g., students, classrooms, schools, districts).
- Cross-classified data (e.g., students nested within primary and secondary schools).
- Partial nesting (e.g., at-risk students in a classroom receive additional instruction by a tutor).
- Longitudinal data (long or wide).
- Longitudinal data with switching classification (e.g., students changing classrooms/teachers over time).
- Round-robin design (e.g., each person rates every other person in a small group).
- 360 performance evaluation data.

1.4 MODEL TYPES

- Multilevel models with random-effects of observed variables.
- Structural Equation Models with observed and latent variables at all levels.
- Multivariate Linear Mixed-Effects Models (LME) with constraints on both **G** and **R** side of the model.
- Linear Growth Curve Models.
- Kenny’s Social Relations Model (SRM) for reciprocal dyadic ratings.

1.5 ESTIMATION FEATURES

- Maximum Likelihood (ML) estimation.
- Missing data.
- Profile-likelihood confidence intervals (CI).
- Equality constraints on model parameters.

1.6 DOCUMENTATION

- Extensive user's guide provided many examples with real and simulated data. The process of model specification is presented using equations, diagrams and the corresponding xxM script.
- Datasets used in the documentation are distributed as part of the package.
- Built-in R help-file documents all commands and packaged example datasets. A vignette illustrates the process of model specification.

1.7 SUPPORT

- http://xxm.times.uh.edu/support/
- Discussion forum. http://xxm.times.uh.edu/support/forums/

1.8 PRICE

- Priceless!

1.9 ACKNOWLEDGEMENT

Development of **xxM** was supported by the Institute of Education Sciences, U.S. Department of Education, through grant R305D090024 awarded to Paras D. Mehta.

CONTENTS

1 Introduction 1-7

1.1 Motivating example 1-7

1.2 Bivariate random-intercepts model 1-7

1.2.1 Equations 1-7

1.2.2 Path-diagram 1-8

1.3 Fitting bivariate random inteRcepts model in xxM 1-8

1.3.1 The Main Model 1-8

1.3.2 Levels 1-9

1.3.3 Submodels 1-10

1.3.4 Datasets 1-11

1.3.5 Model specification 1-12

1.3.6 Bivariate random intercepts model: xxM matrices 1-12

1.3.7 Do Compute 1-18

1.4 Bivariate Random Intercepts Model: Code Listing 1-18

1.4.1 SAS: Proc Mixed 1-18

1.4.2 Mplus 1-19

1.4.3 XXM 1-19

1.5 Complete xxM model 1-20

1.5.1 Observed dependent variables vs. exogenous independent variables 1-21

1.5.2 What is a latent variable in xxM? 1-21

1.5.3 Within-matrices 1-21

1.5.4 Across-matrices: Teacher to Student 1-22

2 Random slopes model 2-23

2.1 Random slopes model 2-23

2.1.1 Two-level equations 2-23

2.1.2 Path Diagram 2-24

2.1.3 XXM Model Matrices 2-24

2.2 Code Listing 2-26

2.2.1 SAS Proc mixed 2-26

2.2.2 MPLUS 2-27

2.2.3 XXM 2-27

3 Univariate Latent Growth curve model 3-30

Univariate LGC model – long format 3-30

3.1.1 Two-level equations 3-30

3.1.2 Path Diagram 3-31

3.1.3 XXM Model Matrices 3-31

3.2 Code Listing 3-32

3.2.1 SAS Proc mixed 3-32

3.2.2 MPLUS 3-33

3.2.3 XXM 3-33

3.3 Results 3-35

4 Multivariate Latent Growth Curve Model 4-37

4.1 Univariate LGC Model –Wide Format 4-37

4.1.1 Equations 4-37

4.1.2 Path Diagram 4-37

4.1.3 xxM Model Matrices 4-38

4.1.4 Code Listing 4-40

5 BIVARIATE CROSS-CLASSIFIED MODEL 5-45

5.1 Motivating Example 5-45

5.2 Bivariate Cross-Classified Random Intercepts Model 5-45
5.2.1 MLM notation 5-45
5.2.2 General notation .... **Error! Bookmark not defined.**
5.2.3 Path Diagram 5-47
5.2.4 XXM Model Matrices 5-47
5.3 Code Listing 5-51
5.3.1 SAS proc mixed 5-51
5.3.2 XXM 5-51
5.4 Results 5-53
6 TWO LEVEL CONFIRMATORY FACTOR ANALYSIS 6-54
6.1 Motivating Example 6-54
6.2 Two-Level CFA Model 6-54
6.2.1 Scalar Representation 6-54
6.2.2 XXM Model Matrices 6-55
6.2.3 Model Matrices Summary 6-57
6.2.4 Path Diagram 6-57
6.3 Code Listing 6-58
6.3.1 MPLUS 6-58
6.3.2 XXM 6-59
6.4 Results 6-60
7 Two level confirmatory factor analysis with a random slope 7-62
7.1 Motivating Example 7-62
7.2 Conditional Two-Level CFA with a random slope 7-62
7.2.1 Scalar Representation 7-62
7.2.2 XXM Model Matrices 7-63

7.2.3 Model Matrices Summary 7-65

7.2.4 Path Diagram 7-66

7.3 Code Listing 7-66

7.3.1 xxM 7-66

7.4 Results 7-69

8 Three level hierarchical model with observed and latent variables at multiple levels 8-70

8.1 Motivating Example 8-70

8.2 Three Level Random Intercepts Model with Latent Regression 8-70

8.2.1 Scalar Representation 8-70

8.2.2 XXM Model Matrices 8-72

8.2.3 Model Matrices Summary 8-76

8.2.4 Path Diagram 8-76

8.3 Code Listing 8-77

8.3.1 XXM 8-77

8.4 Results 8-80

1 INTRODUCTION

This chapter introduces core concepts of xxM from a practical standpoint. Key elements of model specification in xxM are introduced in the context of fitting a bivariate random-intercepts model.

1.1 MOTIVATING EXAMPLE

We use the bivariate random intercepts model example from Mehta, Neale, and Flay (2005).

1.2 BIVARIATE RANDOM-INTERCEPTS MODEL

1.2.1 EQUATIONS

LEVEL 1

$$y_{pij} = 1 \times \eta_{pj} + e_{pij}$$

$$e \sim N(0, R)$$

LEVEL 2:

$$\eta_{pj} = \alpha_{p0} + u_{pj}$$

$$u \sim N(0, G)$$

Mixed-effects model matrices R and G correspond to Θ and Ψ matrices in SEM. Henceforth, we will use Θ and Ψ for the observed and latent residual covariance matrices, respectively. For two dependent variables, we can write the equation as:

$$y_{1ij} = 1 \times \eta_{1j} + 0 \times \eta_{2j} + e_{1ij}$$

$$y_{2ij} = 0 \times \eta_{1j} + 1 \times \eta_{2j} + e_{2ij}$$

$$e \sim N\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \theta_{11} & \\ \theta_{21} & \theta_{22} \end{bmatrix}\right),$$

$$\eta \sim N\left(\begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix}, \begin{bmatrix} \psi_{11} & \\ \psi_{21} & \psi_{22} \end{bmatrix}\right).$$

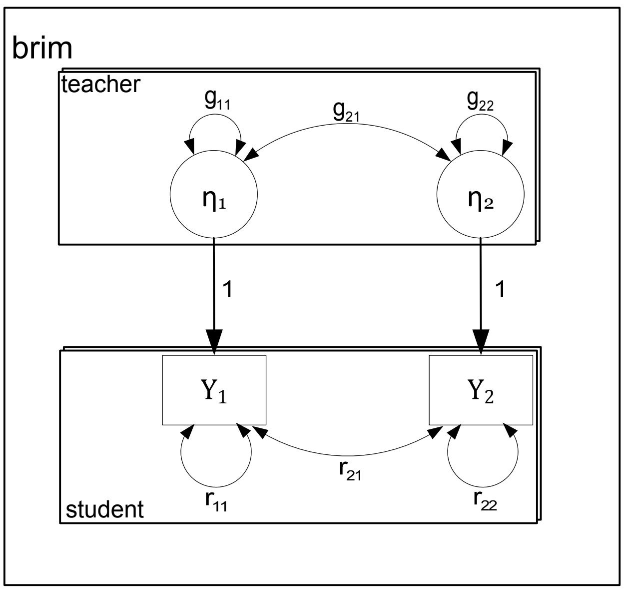
The above bivariate random-intercepts model has 8 parameters:

(a) Covariance among level-1 residuals (e_{pij}), denoted as θ_{21} in the $\mathbf{\Theta}$ matrix.
(b) Covariance among level-2 random-intercepts (u_{pj}), denoted as ψ_{21}in the $\mathbf{\Psi}$ matrix.
(c) Grand-means of y_1 and y_2, denoted as α_1 and α_2 in the $\boldsymbol{\alpha}$ vector.
(d) Variances of level-1 residuals, denoted as θ_{11} and θ_{22} in the $\mathbf{\Theta}$ matrix.
(e) Variances of the level-2 random-intercepts, denoted as ψ_{11} and ψ_{22} in the $\mathbf{\Psi}$ matrix.

1.2.2 PATH-DIAGRAM

The following two-level path diagram accurately represents all parameters except the grand means.

(a) Level-1 residual variances and covariance is represented by curved arrows labeled with the letter *r*.
(b) Level-2 variances and covariance among intercepts is represented by curved arrows labeled with the letter *g*.
(c) Each level-1 dependent variable (y_{pij}) is influenced by the corresponding level-2 "intercept" (η_{pj}).



By definition the effect is 1.0.

1.3 FITTING BIVARIATE RANDOM INTERCEPTS MODEL IN XXM

XXM is an R extension package. In order to use XXM, it must first be loaded. The R function library() does the job.

```
library(xxM)
```

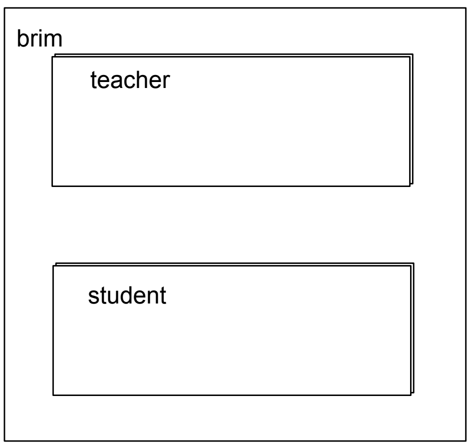
1.3.1 THE MAIN MODEL

An *n* level XXM model is composed of *n* sub models. The very first step in specifying an XXM model is to create a model object by invoking *xxmModel()*. The idea is to declare names of all levels.

```
brim <- xxmModel( levels = c( "student", "teacher") )
```

Internally, xxM assigns level numbers ($lnum = \{1,2, \dots, P\}$) to each level declared in *xxMModel().* The order of levels is also important. In this case, students are influenced by teachers. Hence, the student level must be declared before the teacher level.

The intent behind this function is obvious. The function creates an object called *brim* (i.e., bivariate random intercepts model) from a list of level names. The left hand side is the name of the XXM model. The choice of the name is arbitrary and other names such as *ladyGaga* or *cow* would work. However, it is better to use a short, but descriptive name, as this name will be used in all subsequent commands. Internally, the above invocation literally creates a model with placeholders for the student and teacher



submodels, as depicted below.

More formally, the function takes a single parameter aptly called *levels* and expects to receive a list of level names. The function creates an XXM object and returns a handle or a pointer to the object in memory. The following command presents a generic invocation of *xxmModel().*

```
xxmObject <- xxmModel(levels = list)
```

1.3.2 LEVELS

At this stage it appears that we have data for students and teachers. Presumably students are nested within teachers. In multilevel modeling jargon, we have two levels with students hierarchically nested within teachers. While the term *level* is obvious in this simple case, xxM can be used with fairly complex data structures in which the notion of levels may not be so easy to identify.

Levels represent any concrete or abstract set of entities across which some attribute is expected to vary. Very simply, a level involves multiple entities of some kind (e.g., students, situations, responses, occasions etc.) for whom there is an attribute or a variable (e.g., IQ or achievement) of interest. Each level may have its own set of observed and/or latent variables. The definition of levels also implies a flow of influence. Levels have a one-directional relationship. A higher order level is said to influence a lower order level. In this case, we know that teachers impact student outcomes. The directional relationships are implicitly conveyed to xxM in how the levels are ordered in the list. Our goal is to be able to specify a latent variable model within and across levels to capture the hypothesized flow of influence. The actual model will be presented later; for now we focus on the mechanics of specifying a model in xxM. We begin with the sub-models for students and teachers.

1.3.3 SUBMODELS

Each level may have its very own complete SEM model with observed dependent and exogenous independent variables, latent variables, measurement model, and structural model involving all possible regressions (observed on observed, observed on latent, latent on observed, and latent on latent). Before we can begin to specify the actual model, we need to provide our model object, *brim*, with basic information about each level. This is accomplished by the *xxmSubmodel()* function.

```
brim <- xxmSubmodel( model = “brim”,
      level = “student”,
      parents = c(“teacher”),
      ys = c(“y1”, “y2”),
      xs = ,
      etas = ,
      data = student )
```

These assignment statements build the sub-model with the constituent parts. The *xxmSubmodel()* function adds basic information to our XXM model object, *brim*. The first parameter *model*, asks for the name of the xxM object to which this information is being added. The second parameter identifies the level for the sub-model. In this case, we are adding information about the student level. The next parameter, *parents*, defines the nesting relationship involving students. Students are nested within teachers and the nesting is captured by the notion of parent and child levels in xxM. In this case, the teacher level is a parent of the student level. If there were additional levels of nesting, these would be added to the list of parents as well. The following code provides an example with four levels:

```
parents = c( “family”, “teacher”, “school”, “neighborhood” )
```

The next three arguments are for names of (a) observed dependent variables, (b) observed independent variables, and (c) latent variables. In this case, we have two dependent variables for student (y1 and y2).

There are no exogenous predictors or latent variables at the student level. The final parameter, *data,* is for an R dataset with student data. The corresponding model for the teacher level is:

```
brim <- xxmSubmodel( model = "brim",
      level = "teacher",
      parents = ,
      ys = ,
      xs = ,
      etas = c( "eta1", "eta2" ),
      data = teacher )
```

The teacher level does not have a parent level, nor does it have observed dependent or independent variables. The teacher level does have two latent variables. The latent variables represent random intercepts of student level dependent variables (y1 and y2). If teachers were nested within a higher level such as school, the *parents* argument would be:

```
parents = c( "school" )
```

1.3.4 DATASETS

For two-level data structures, a single dataset is adequate. However, with complex dependent data structures, it is most convenient to provide data for each level separately. Each dataset must include information about how each observation at a lower level is linked to a higher level. In general, datasets must have three types of variables: (a) one or more columns of IDs or variables with linking information, (b) zero or more columns of dependent variables corresponding to the list of *ys*, and (c) zero or more columns of independent variables corresponding to the list of *xs*. While the last two variable types are straight forward, the first one requires additional explanation. The student data (*studentData*) must have the following structure:

student	teacher	Y1	Y2
0	0	.314	-2.115
1	0	-1.205	-1.972
2	1	.055	0.917
3	1	1.03	1.486

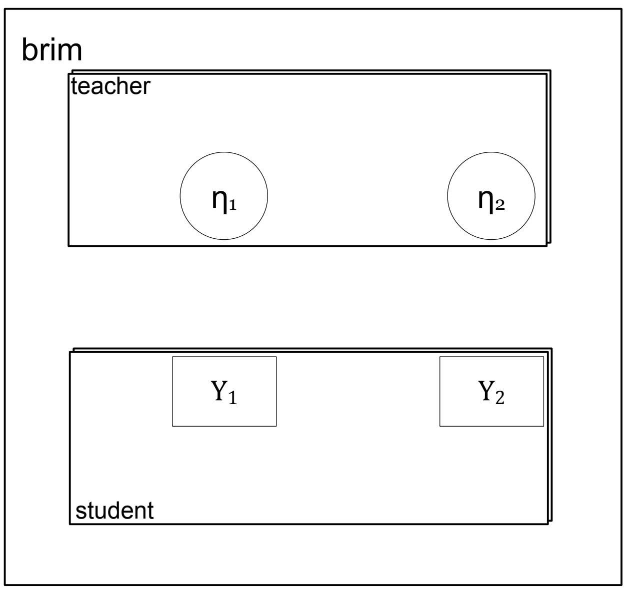
The first column is for the ID variable for the current level, in this case student. The next set of zero or more columns is for the ID variables of each parent of the current level. In this case, student has a single parent, teacher. For now, the ID columns must have the same name as the name of the corresponding level. Practically, it means that if in your dataset the ID variables are named *studentID* and *teacherID*, these must be renamed to *student* and *teacher*.

The teacher dataset (*teacherData*) has no observed variables, nor does it have parents. Yet, it is necessary to have a dataset listing teacher IDs.

teacher
0
1
2
3

1.3.5 MODEL SPECIFICATION

So far we created a model object called *brim* by invoking *xxmModel()*and declared sub-models for student and teacher by invoking *xxmSubmodel()*. At this point, our model object called *brim*, looks as follows and is just a shell of the final desired model.



The model object also has actual data for students and teachers along with the linking information. The next logical step would be to specify the actual model, i.e., how observed and latent variables relate to each other.

1.3.6 BIVARIATE RANDOM INTERCEPTS MODEL: XXM MATRICES

From an XXM perspective, the above model is specified in terms of parameters and matrices associated with each level and links among variables across levels. This sounds complicated, but in reality we will simply repeat what we have already stated in previous sections:

1.3.6.1 LEVEL 1: WITHIN-STUDENT MODEL MATRICES

At level-1, we only have variances and a covariance for the residuals. The residual covariance matrix is called the *theta* matrix (Θ). The matrix is symmetric with three free parameters: Two variances and a covariance ($\theta_{12} = \theta_{21}$).

$$\Theta = \begin{bmatrix} \theta_{11} & \\ \theta_{21} & \theta_{22} \end{bmatrix} \quad (1)$$

1.3.6.2 WITHIN-TEACHER MODEL MATRICES

At level-2, we have a covariance and variances among the latent variables, along with their means. Latent covariance and mean matrices are called Ψ (*psi*) and α (*alpha*), respectively:

$$\Psi = \begin{bmatrix} \psi_{11} & \\ \psi_{21} & \psi_{22} \end{bmatrix} \quad (2)$$

$$\alpha = \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} \quad (3)$$

In this example, we have observed variables only at level-1 and latent variables only at level-2. For more complex models, we may have observed and latent variables at multiple levels. For such models, we may have Ψ and Θ matrices at each such level. Complexities of representing such matrices will be introduced in subsequent chapters.

1.3.6.3 ACROSS-LEVEL MODEL MATRICES: FROM TEACHERS TO STUDENTS

The above three matrices (Θ, Ψ, & α) capture information about variables within each level. The missing piece is the link among variables across levels. In this case, the teacher-level latent variables influence student-level observed variables. Very concretely, the effects are:

$$y_{1ij} = 1 \times \eta_{1j} + 0 \times \eta_{2j} + e_{1ij}$$

$$y_{2ij} = 0 \times \eta_{1j} + 1 \times \eta_{2j} + e_{2ij}$$

We can represent the effects in a table, with columns representing the independent variables and rows representing the dependent variables. In xxM matrices, columns will always represent independent variables and rows will always represent dependent variables, when connecting independent and dependent variables.

	η_{1j}	η_{2j}
y_{1ij}	1	0
y_{2ij}	0	1

With this convention we can now represent the coefficients in a single matrix Λ (*lambda*):

$$\Lambda = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (4)$$

In LISREL and xxM, Λ matrix is used to capture measurement relationship. In this case, level-2 latent variables are said to be measured by level-1 observed variables.

1.3.6.4 XXM SPCIFICATION OF MODEL MATRICES: OVERVIEW

We have now defined four matrices that completely specify the underlying bivariate random-intercepts model. Once the model itself is clearly defined, the actual specification is really very trivial. There are just two commands for specifying model matrices:

```
brim <- xxmWithinMatrix(…)

brim <- xxmBetweenMatrix(…)
```

Essentially, we want to add the above four matrices to complete the model. Of the four matrices, the first matrix is a within-student matrix (Θ), the next two matrices are within-teacher matrices (Ψ and α) and the last matrix (Λ) connects the two levels and is therefore an across-level or between matrix. At this point several things must be obvious: (a) we will need to call *xxmWithinMatrix()* three times, first for the student level and then twice for the teacher level; and (b) *xxmBetweenMatrix()* will be called once connecting the teacher level to the student level.

```
brim <- xxmWithinMatrix( model = "brim",
      level = "student",
      type = "theta"… )
brim <- xxmWithinMatrix( model = "brim",
      level = "teacher",
      type = "psi", … )
brim <- xxmBetweenMatrix( model = "brim",
      parent = "teacher",
      child = "student",
      type = "lambda", …)
```

Now we know the general procedure for adding a matrix to the model. Let us now examine how a matrix to be added is actually specified.

1.3.6.5 XXM MATRICES: FREE VS. FIXED

Note that the first three parameter matrices (theta, psi, and alpha) are somewhat different from the last matrix (lambda). The first three matrices include model parameters that are to be freely estimated. In contrast, all four elements of the last matrix are fixed. We already know their values. This idea of free vs. fixed parameters is central in SEM. In essence, for each parameter we need to tell xxM if the parameter is to be estimated or if the parameter is to be fixed to some known value. For each parameter matrix, we need to define *two* separate matrices: (a) a pattern matrix indicating the pattern of free (=1) or fixed (= 0) parameters and (b) a value matrix providing numeric values for fixed-parameters or start-values for free parameters. It is easier done than said. We use a two part name including: (a) the type of the matrix (theta, psi, alpha, or lambda) and (b) the role of the matrix (pattern matrix or value matrix).

```
lambda_pattern <- matrix( c(0,0,0,0), 2,2 )
lambda_value   <- matrix( c(1,0,0,1), 2,2 )
```

$\Lambda_{pat} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}, \Lambda_{val} = \begin{bmatrix} 1.0 & 0.0 \\ 0.0 & 1.0 \end{bmatrix}$

All elements of pattern matrix for Λ are zero indicating that none of the parameters are free to be estimated. Instead, all parameters are to be fixed to some known values. The value matrix provides the corresponding values. The diagonal elements are to be fixed to 1.0, whereas the off-diagonal elements are to be fixed to 0.0. Compare the specification of Λ with that of the Θ matrix:

```
theta_pattern <- matrix( c(1,1,1,1), 2,2 )
theta_value   <- matrix( c(1.1,.2,.2,2.3), 2,2 )
```

$\Theta_{pat} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}, \Theta_{val} = \begin{bmatrix} 1.1 & 0.2 \\ 0.2 & 2.3 \end{bmatrix}$

All four elements of the Θ matrix are to be freely estimated. Hence all four elements in the pattern matrix are 1s. The value matrix provides start values. At this point, you may complain that you do not actually have any idea as to what these values may be. In general, almost any reasonable set of values will work. More specifically, for a residual covariance matrix, the following rules work very well in practice:

(1) Start values for the residual variances or the diagonal elements may be close to the observed variances of the respective variables.
(2) Start values for the residual covariances or the off-diagonal elements may be close to zero.

So for the following observed covariances matrix, $y = \begin{bmatrix} 6.2 & 3.1 \\ 3.1 & 5.7 \end{bmatrix}$, a reasonable set of starting values may be:

$\Theta_{val} = \begin{bmatrix} 5.1 & 1.1 \\ 1.1 & 4.9 \end{bmatrix}$.

Again, actual values do not matter much.

1.3.6.6 BIVARIATE RANDOM INTERCEPTS MODEL: COMPLETE XXM SPCIFICATION

First, we create pattern and value matrices corresponding to each of the four model matrices.

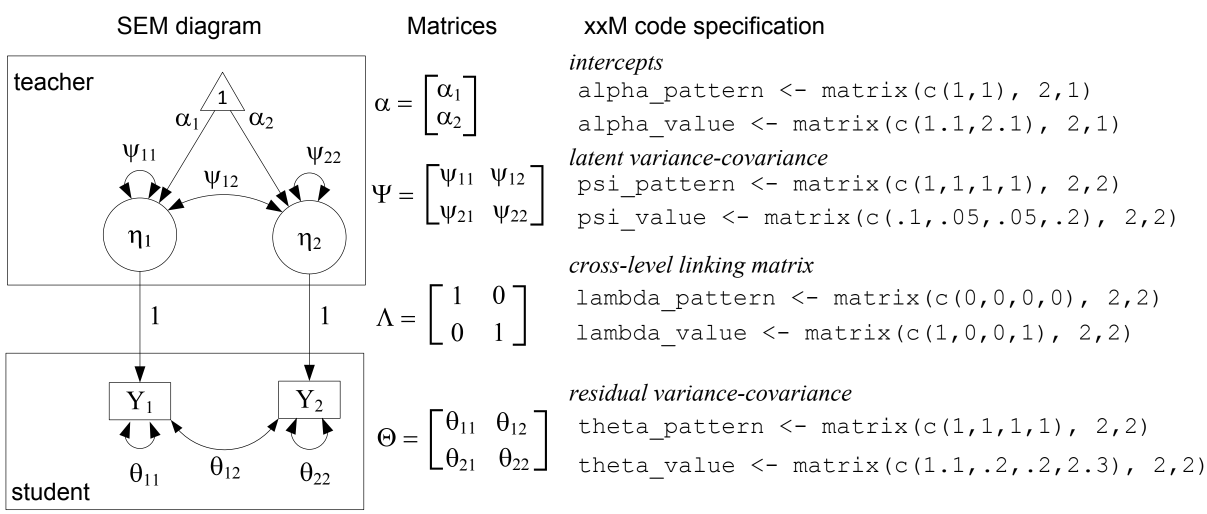
```
#Within-Student Model Matrices
theta_pattern <- matrix( c(1,1,1,1), 2,2 )
theta_value   <- matrix( c(1.1,.2,.2,2.3), 2,2 )
#Within-Teacher Model Matrices
psi_pattern <- matrix( c(1,1,1,1), 2,2 )
psi_value   <- matrix( c(.1,.05,.05,.2), 2,2 )
alpha_pattern <- matrix( c(1,1) , 2,1)
alpha_value <- matrix( c(1.1,2.1) , 2,1)
# Teacher->Student Across Matrices
lambda_pattern <- matrix( c(0,0,0,0), 2,2 )
lambda_value   <- matrix( c(1,0,0,1), 2,2 )
```

Once model matrices are created we add these matrices to our model as described earlier:

```
#Within-Student Model Matrices
brim <- xxmWithinMatrix( model = "brim",
      level = "student",
      type = "theta",
      pattern = theta_pattern,
      value = theta_value)
#Within-Teacher Model Matrices
brim <- xxmWithinMatrix( model = "brim",
      level = "teacher",
      type = "psi",
      pattern = psi_pattern,
      value = psi_value)
brim <- xxmWithinMatrix( model = "brim",
```

```
      level = "teacher",
      type = "alpha",
      pattern = alpha_pattern,
      value = alpha_value)
#Teacher->Student Across Model Matrices
brim <- xxmBetweenMatrix( model = "brim",
      parent = "teacher",
      child = "student",
      type = "lambda",
      pattern = lambda_pattern,
      value = lambda_value)
```

The following diagram illustrates a one-to-one correspondence between the path-diagram, corresponding model matrices, and xxM specification.



Note:

(1) All elements of the factor covariance matrix (psi:Ψ), residual covariance matrix (theta:Θ), and the factor-mean matrix (alpha:α) are freely estimated. All values in respective pattern matrices are 1.

(2) Factor covariance matrix and residual covariance matrix are symmetric. Hence, there are only three free-parameters. The xxM package constrains off-diagonal elements to be equal ($\theta_{12} = \theta_{21}$).

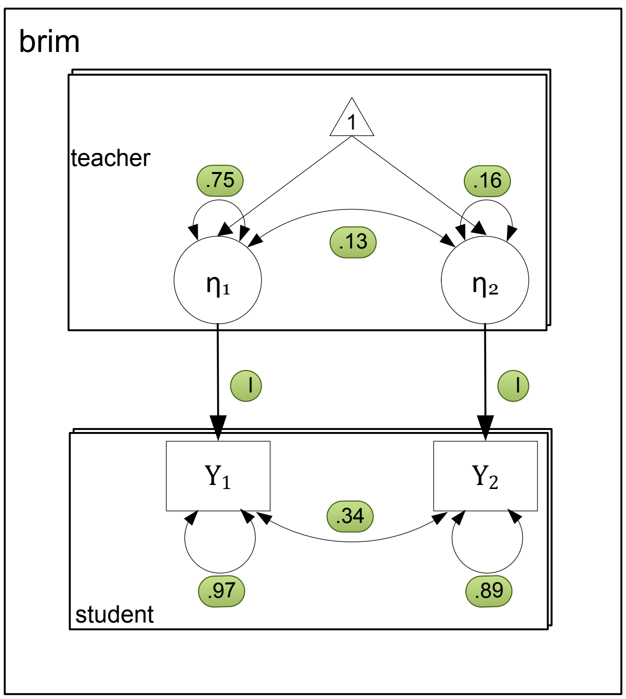
(3) All elements of the across-level factor-loading matrix (lambda:Λ) are fixed to known values. All values in lambda_patten matrix are 0. The corresponding value matrix indicates that the diagonal elements are fixed to 1.0 and off-diagonal elements are fixed to 0.0.

1.3.7 DO COMPUTE

If all went well above and xxM did not produce any error messages, then our model object *brim* has all the information it needs to estimate the model parameters. We can begin estimation by issuing a simple command:

```
brim <- xxMRun( brim )
```

For the particular problem, the final parameter estimates are:



1.4 BIVARIATE RANDOM INTERCEPTS MODEL: CODE LISTING

In this section, we gather code for fitting the bivariate-random intercepts model in xxM as well as in other software packages such as SAS and Mplus.

1.4.1 SAS: PROC MIXED

SAS assumes that the dataset is in an univariate format with two columns of IDs: teacher and vars and a single column with dependent variables.

```
Proc Mixed data = brim covtest;
 CLASS teacher vars;
 MODEL y = vars / solution noint;
 RANDOM vars / subject = teacher type = un;
 REPEATED vars/ subject = student(teacher) type = un;
RUN;
```

1.4.2 MPLUS

```
DATA:
  File = 'brim.dat';
VARIABLE:
  NAMES are y1 y2 teacher;
  CLUSTER = teacher;
MODEL:
 %WITHIN%
  y1 y2;
  y1 with y2;
 %BETWEEN%
  y1 y2;
  y1 with y2;
  [y1 y2];
```

1.4.3 XXM

Complete code for the xxM specification is repeated here.

```
library(xxM)
brim <- xxmModel( levels = c( "student", "teacher") )
brim <- xxmSubmodel( model = “brim”, level = “student”, parents =
c(“teacher”), ys = c(“y1”, “y2”), xs =, etas = , data = student )
brim <- xxmSubmodel( model = “brim”, level = “teacher”, parents = , ys
= ,
     xs = , etas = c( “eta1”, “eta2” ), data = teacher )
```

```
theta_pattern <- matrix( c(1,1,1,1), 2,2 )
theta_value   <- matrix( c(1.1,.2,.2,2.3), 2,2 )
psi_pattern <- matrix( c(1,1,1,1), 2,2 )
psi_value   <- matrix( c(.1,.05,.05,.2), 2,2 )
alpha_pattern <- matrix( c(1,1) , 2,1)
alpha_value <- matrix( c(1.1,2.1) , 2,1)
lambda_pattern <- matrix( c(0,0,0,0), 2,2 )
lambda_value   <- matrix( c(1,0,0,1), 2,2 )
brim <- xxmWithinMatrix( model = “brim”, level = “student”, type =
“theta”,
     pattern = theta_pattern, value = theta_value)
brim <- xxmWithinMatrix( model = “brim”, level = “teacher”, type =
“psi”,
     pattern = psi_pattern, value = psi_value)
brim <- xxmWithinMatrix( model = “brim”, level = “teacher”, type =
“alpha”,
     pattern = alpha_pattern, value = alpha_value)
brim <- xxmBetweenMatrix( model = “brim”, parent = “teacher”, child =
     “student”, type = “lambda”,  pattern = lambda_pattern, value =
     lambda_value)
brim <- xxmRun( brim )
brim <- xxmCI( brim )
xxmSummary( brim )
est <- xxmGet( brim, what=”estimates” )
lik <- xxmGet( brim, what=”fit” )
brim <- xxmFree( brim )
```

1.5 COMPLETE XXM MODEL

xxM is a modeling framework inspired by LISREL and generalized to a model that is defined as a network of SEM sub-models, in which variables across levels are also related by LISREL-like matrices.

1.5.1 OBSERVED DEPENDENT VARIABLES VS. EXOGENOUS INDEPENDENT VARIABLES

In ordinary multiple and multilevel regression, observed predictors are assumed to be strictly exogenous, fixed, and without measurement error. In SEM, with possible latent independent variables the distinction is frequently blurred. xxM follows the regression tradition of making an explicit and concrete distinction between a modeled observed dependent variable and an exogenous predictor.

1.5.2 WHAT IS A LATENT VARIABLE IN XXM?

xxM blurs the distinction between an unobserved latent variable in a SEM sense in which a latent variable is defined by multiple hypothesized observed indicators and a random-effect in a multilevel modeling sense in which the latent variable is thought of as an unobserved variable measured at a higher clustering level on an observed variable measured at a lower level. Indeed in the current example, random-intercepts for clustered data were conceptualized as teacher latent variables and measured by observed indicators at the student level.

1.5.3 WITHIN-MATRICES

Possible student and teacher sub-models may include the matrices in the following table. The bolded within matrices were entered in the previous xxM code. The rows of the table represent model dependent variables. There may be observed and/or latent dependent variables at any level. As in ordinary SEM model, variables may have means and covariances (or intercepts and residual covariances). SEM models allow for directional relationships among observed and latent variables. With two possible types of dependent variables (latent and observed) and two possible types of independent variables (latent and exogenous observed), there are four total directional relationships. Mathematically, all four represent linear effect of a variable on another variable. Conceptually, however, we make a distinction between measurement and structural relationships. Regression of an observed variable on a latent variable, when the observed variable is thought of as an indicator of the latent variable, is called a measurement model (Λ). The remaining three matrices (B, K & Γ) are part of the structural model.

Teacher submodel	***Mean/intercept (Residual) covariance***	**Latent Variable**	**Observed Exogenous variable**
Latent Variable	α, Ψ	B	Γ
Observed Dependent-variable	ν, Θ	Λ	K

Student submodel	***Mean/intercept (Residual) covariance***	**Latent Variable**	**Observed Exogenous variable**

Latent Variable	α, Ψ	B	Γ
Observed Dependent-variable	ν, Θ	Λ	K

1.5.4 ACROSS-MATRICES: TEACHER TO STUDENT

Across levels matrices capture the influence of a higher level variable on a lower level variable. Hence, only the four directional matrices are allowed. The lambda matrix (in bold) was entered into the current model.

	Teacher Latent Variable	**Teacher Observed Exogenous variable**
Student Latent Variable	B	Γ
Student Observed Dependent variable	Λ	K

2 RANDOM SLOPES MODEL

The model description so far is very much like a SEM model for multiple levels. In this chapter, we examine conventional random-slopes model for two level data. Mehta and West (2000) demonstrated how a longitudinal model with random-slopes for time may be estimated within an SEM context. Mehta and Neale (2005) extended the idea to random-slopes model for the case of clustered data. Another example of fitting a random slopes model can be found in Branum-Martin (2013).

2.1 RANDOM SLOPES MODEL

2.1.1 TWO-LEVEL EQUATIONS (MULTILEVEL MODELING REPRESENTATION)

$$y_{ij} = b_{0j} + b_{1j} \times x_{ij} + e_{ij}$$

$$b_{0j} = \gamma_{00} + u_{0j}$$

$$b_{1j} = \gamma_{10} + u_{1j}$$

$$e \sim N(0, \theta_{11})$$

$$u \sim N(0, \Psi)$$

In a random slopes model, the effect of a level-1 independent variable (x_{ij}) on a level-1 dependent variable (y_{ij}) varies randomly across level-2 units. Such models are also known as the random-coefficients model, as the regression-coefficients (b_{0j}& b_{1j}) vary randomly. Such random-effects are unobserved or latent variables at level-2. In xxM, random-effects are treated as latent variables. Using the SEM notation for a latent variable (η) the above equations may be written as:

$$y_{ij} = 1_{ij} \times \eta_{0j} + x_{ij} \times \eta_{1j} + e_{ij}$$

$$\eta_{0j} = \gamma_{00} + u_{0j}$$

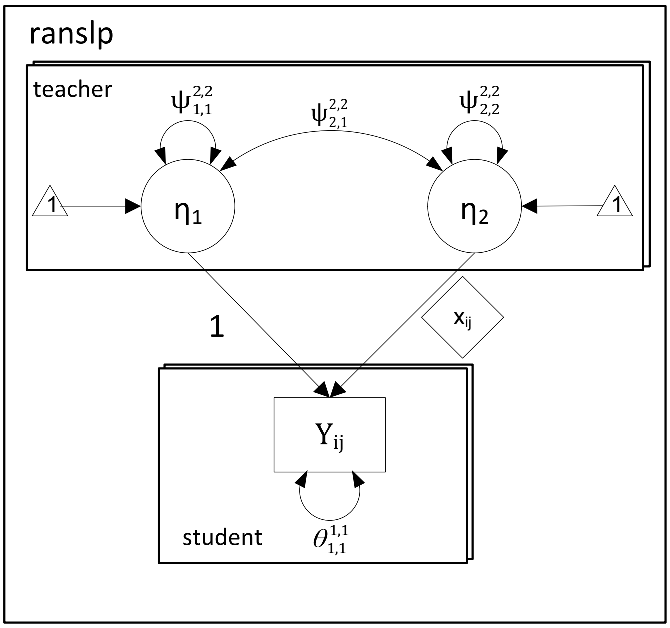
$$\eta_{1j} = \gamma_{10} + u_{1j}$$

Note that the coefficients are random or latent variables, whereas the predictors (1_{ij} & x_{ij}) are fixed. We can use a SEM measurement model to capture the directional relationship between random-effects at level-2 and observed variables at level-1, in which the factor loadings are fixed to level-1 values of fixed predictors. With this mind-set, the factor-loading matrix (Λ) has a single row and two columns:

$$\Lambda = [1.0 \quad x_{ij}].$$

The first column is fixed to 1.0, whereas the second column is fixed to student-specific values of x_{ij}. The idea of fixing factor-loadings for specifying multilevel random slopes was introduced in the context of growth curve modeling in Mehta and West (2000) and extended to clustered data and general multilevel case in Mehta and Neale (2005). The following path-diagram has a one-to-one correspondence to the first and second level equations for a random-slope model.

2.1.2 PATH DIAGRAM



2.1.3 TWO-LEVEL EQUATIONS (N-LEVEL SEM REPRESENTATION)

In the previous chapter, we used the simplest representation of a bivariate mixed-effects model. In this chapter, we will use a proper representation of a multilevel SEM model that generalizes to models with any number of levels. We begin by rewriting the conventional random-slopes model:

$$y_i^{(1)} = 1_{i,j}^{(1,2)} \times \eta_{1,j}^{(2)} + x_{i,j}^{(1,2)} \times \eta_{2,j}^{(2)} + e_i^{(1)}$$

More generally, we can use a matrix representation:

$$y_i^{(1)} = \Lambda_{i,j}^{(1,2)} \times \eta_j^{(2)} + e_i^{(1)}$$

where,

$$\Lambda_{ij}^{(1,2)} = [1.0 \quad x_{ij}].$$

$$e \sim N\left(0, \Theta^{(1,1)}\right)$$

$$\eta \sim N\left(\alpha, \Psi^{(2,2)}\right)$$

2.1.4 XXM MODEL MATRICES

Matrices used in the above equations are described at length below.

2.1.4.1 LEVEL-1: WITHIN MATRICES

In this example, we will use superscripts for associating each within-matrix to a given level. The rules of superscripts (for levels) correspond to the rules for subscripts (for variables).

2.1.4.1.1 RESIDUAL COVARIANCE MATRIX

At level-1, we have a single dependent variable and hence a single parameter level-1 residual variance ($\theta_{1,1}$). Hence, the residual covariance or theta matrix is a (1×1) matrix:

$$\Theta^{1,1} = \left[\theta_{1,1}^{1,1}\right].$$

The residual covariance matrix includes a superscript $\{1,1\}$ to indicate that the matrix belongs to level-1. In principle it is possible to define covariance between variables across *two different levels.* For example, residual-covariance between the 3rd observed variable at level-5 and the 4th observed variable at level-2 may be represented as: $\theta_{4,3}^{2,5}$. In this case, the superscripts indicate the levels of the respective variables in the subscript.

2.1.4.2 LEVEL-2: WITHIN MATRICES

At level-2, we have two latent variables for the intercept and the slope and hence we have two latent means and a covariance matrix. In SEM, the latent variable mean vector is called alpha (α) and the latent variable covariance matrix is called psi (Ψ)

2.1.4.2.1 LATENT MEANS

With two latent variables, the latent variable mean matrix is a (2×1) matrix:

$$\alpha^1 = \begin{bmatrix} \alpha_1^1 \\ \alpha_2^1 \end{bmatrix}$$

α_1^1 is the mean of the intercept and is the mean of the slope parameter or the average effect of x_{ij} on y_{ij}. In the parlance of mixed-effects models, the latent means represent the fixed-effects of intercepts and slopes, respectively.

2.1.4.2.2 LATENT FACTOR COVARIANCE MATRIX

Latent covariance matrix is a (2×2) matrix with two variances and a single covariance:

$$\Psi^{2,2} = \begin{bmatrix} \psi_{1,1}^{2,2} & \\ \psi_{2,1}^{2,2} & \psi_{2,2}^{2,2} \end{bmatrix}$$

$\psi_{1,1}^{2,2}$ is the variance of the intercept factor representing variability in the intercept of y_{ij} across level-2 units and $\psi_{2,2}^{2,2}$ is the variance of the slope parameter representing variability in the effect of x_{ij} on y_{ij}. Finally, $\psi_{2,1}^{2,2}$ is the covariance between the intercept and slope factors. In the parlance of mixed-effects models, the psi matrix represents the covariance among the random-effects.

2.1.4.3 ACROSS LEVEL MATRICES: TEACHER TO STUDENT

As described above, we need to capture the effect of level-2 intercept and slope factors on the level-1 dependent variable using a factor-loading matrix with fixed parameters.

The factor-loading matrix ($\Lambda^{1,2}$) has a single row and two columns (1×2):

$\Lambda^{1,2} = [1.0 \quad x_{ij}]$.

The first column is fixed to 1.0, whereas the second column is fixed to student-specific values of x_{ij}. Note that we use superscripts to indicate that the factor-loading matrix connects latent variables at level-2 with observed variables at level-1. In fact, for an xxM model with multiple levels and observed and latent variables at each level, *all* model matrices require superscripts to uniquely associate the matrix with the level. In order to keep things simple, we have avoided superscripts for simple models considered so far. Very quickly it will become apparent that superscripts are necessary for our sanity.

2.2 CODE LISTING

2.2.1 SAS PROC MIXED

```
Proc Mixed data = ranslp covtest;
CLASS teacher;
MODEL y = x/s;
RANDOM Intercept x/subject = teacher G type = UN;
RUN;
```

SAS code for a random-slopes model uses a CLASS statement to identify the level-2 units, in this case teacher. The MODEL statement estimates the fixed-effects (α). The RANDOM statement specifies that the level-1 intercepts and the effect of level-1 predictor x_{ij} is allowed to vary across "subject" (i.e., the level-2 units). The covariance among the random-effects is freely estimated (specified by a "type = UN") and the covariance matrix is referred to as the G matrix. The G matrix corresponds to the xxM Ψ matrix. Finally, like all regression models, Proc Mixed estimates the residual variance of level-1 dependent variable ($\theta_{1,1}$) by default. The important thing to note is that there is one-to-one correspondence between the parameters estimated in Proc Mixed and SEM.

2.2.2 MPLUS

Mplus allows random-slopes model to be estimated for two level data.

```
TITLE: Random Slopes
DATA: File is ranslp.dat;
VARIABLE:
  Names = y x teacher;
  Within= x;
  Cluster = teacher;
ANALYSIS:
  TYPE = TWOLEVEL RANDOM;
MODEL:
%WITHIN%
s | y ON x;
y;
%BETWEEN%
y s;
y with s;
[y s];
```

At level-1, the vertical bar in “s | y ON x;” is a command to treat the regression of y on x as random at level-2. The name “s” on the left side of the bar is the name of the random-effect, in this case the random slope. At level-2, “y” represents the intercept of the corresponding level-1 dependent variable. The same five parameters are estimated at level-2: means of the intercepts and slopes (“[y s];”), their respective variances (“ y s;”) and their covariance (“y with s;”).

2.2.3 XXM

The following code for the random-slopes model is nearly identical to the bivariate random-intercepts model presented earlier. In this case, there is a single level-1 dependent variable and a single level-1 independent variable. The number of matrices is the same as before; however, their dimensions are different as we have a single as opposed to two level-1 dependent variables. Practically, speaking the only new element is that the second factor-loading is fixed to x_{ij}. This is accomplished by using an additional *label* matrix for the factor-loading matrix. This is what we want to do:

1. We do not wish to estimate any factor-loadings. Factor-loadings are *fixed* to 1 and x_{ij}. Hence, both elements of pattern matrix are zero:

$$\Lambda^{1,2}_{pat} = [0 \quad 0]$$

2. We want to fix the first factor-loading to 1.0 (intercept). We use the value matrix to provide the fixed-value of 1.0 for the first factor-loading. The second factor-loading does not have a single fixed value for every observation. Instead each observation (i) would have its own value for that factor-loading (x_{ij}). Clearly, the value matrix cannot be used for providing individual specific fixed values. Hence, the second element in the value matrix is left as 0.0. xxM ignores it internally.

$$\Lambda^{1,2}_{val} = [1.0 \quad 0.0]$$

3. The job of fixing the factor-loading is left to the *label* matrix. A label matrix is used to assign labels to each parameter within the matrix. Label matrices can be used to impose equality constraints across matrices. Any two parameters with the same label are constrained to be equal. Label matrices are also used for specifying that a specific parameter is to be fixed to data-values. In this case, the first label is irrelevant as that parameter has already been fixed to 1.0. We use a descriptive label *lambda_11* as the first label (something such as *Justin* would have worked as well). The second factor-loading is the one we are interested in. We want to fix the second factor-loading to the observation specific value of the predictor (x_{ij}). This is accomplished by using a two-part label: *levelName.predictorName*. In this case, the predictor is a student level variable. Hence, the first part of the label is *student*. The second part is the actual predictor name, in this case x.

$$\Lambda^{1,2}_{label} = [l_{1,1} \quad x_{ij}]$$

```
lambda_label   <- matrix( c(“lambda_11”,”student.x”), 2, 1 )
```

The complete listing of xxM code for the random-slopes example follows:

```
library(xxm)

data(pcwa.xxm, package="xxm")

ranslp <- xxmModel( levels = c( "student", "teacher") )

ranslp <- xxmSubmodel(model = ranslp, level = "student", parents =
"teacher", ys = "pc", xs = c("wa"), etas = , data = pcwa.student)

ranslp <- xxmSubmodel(model = ranslp, level = "teacher", parents = ,
ys = , xs = , etas = c("int","slp"),data = pcwa.teacher)

alpha_pattern <- matrix(1, 2,1)
```

```
alpha_value <- (matrix(c(443,1), 2,1))

psi_pattern <- matrix(1, 2,2)

psi_value <- (matrix(c(10,.0,.0,.05), 2,2))

lambda_pattern <- matrix(c(0,0), 1,2)

lambda_value <- (matrix(c(1,0), 1,2))

lambda_label <- (matrix(c("l11","student.wa"), 1,2))

theta_pattern <- matrix(1, 1,1)

theta_value <- (matrix(200., 1,1))

ranslp <- xxmWithinMatrix(model = ranslp, level = "student", type =
"theta", pattern = theta_pattern, value = theta_value,,)

ranslp <- xxmWithinMatrix(model = ranslp, level = "teacher", type =
"alpha", pattern = alpha_pattern, value = alpha_value,,)

ranslp <- xxmWithinMatrix(model = ranslp, level = "teacher", type =
"psi", pattern = psi_pattern, value = psi_value,,)

ranslp <- xxmBetweenMatrix(model = ranslp, parent = "teacher",child =
"student", type = "lambda", pattern = lambda_pattern, value =
lambda_value, label = lambda_label,)

ranslp  <-xxmRun(ranslp)

ranslp <- xxmCI(ranslp)

xxmSummary(ranslp)

ranslp <- xxmFree(ranslp)

rm(list=ls())
```

The above code appears to be very complicated. However, as the model becomes increasingly complex with multiple levels, variables, and constraints, the xxM code will remain succinct. Thus, although there is an overhead for simple models, for complex models the matrix notation is a blessing.

3 LATENT GROWTH CURVE MODEL: LONG FORMAT

The model equations and matrices in the present example are identical to the previous. However, there are two key features that make the present model distinct: (1) level-1 observations represent reaction times nested within individuals and (2) the level-1 predictor measures time since the study began (measurement occasion), leading to a latent growth curve model (LGC). We have chosen to structure the data in a manner consistent with the mixed-effects modeling approach to LGC analysis, which also allows us to draw an explicit parallel between the present model and the more general random-slopes model (example 2) in which the level-1 predictor represents another attribute of the level-1 unit. The data for the present example were drawn from the Reisby et al. (1977) example described in Hedeker's (2004) introduction to growth modeling chapter. The outcome is ratings on the Hamilton depression rating scale (Hamilton, 1960). Depressions scores were taken over a period of weeks. A baseline was taken (week 0). Ratings were then taken after a week of the subjects consuming a placebo (week 1) and the following four weeks (week 1-5) participant took a depression drug.

UNIVARIATE LGC MODEL – LONG FORMAT

3.1.1 TWO-LEVEL EQUATIONS

$$HamD_{ij} = 1_{ij} \times \eta_{Intj} + week_{ij} \times \eta_{Slopej} + e_{ij}$$

$$\eta_{Intj} = \gamma_{00} + u_{0j}$$

$$\eta_{Slopej} = \gamma_{10} + u_{1j}$$

As in the previous model, the coefficients are latent variables and the predictors (1_{ij}& $week_{ij}$) are fixed. The following path-diagram has a one-to-one correspondence to the first and second level equations for a random-slopes model.

3.2 NL-SEM REPRESENTATION

$$y_i^{(1)} = 1_{i,j}^{(1,2)} \times \eta_{1,j}^{(2)} + w_{i,j}^{(1,2)} \times \eta_{2,j}^{(2)} + e_i^{(1)}$$

More generally, we can use a matrix representation:

$$y_i^{(1)} = \Lambda_{i,j}^{(1,2)} \times \eta_j^{(2)} + e_i^{(1)}$$

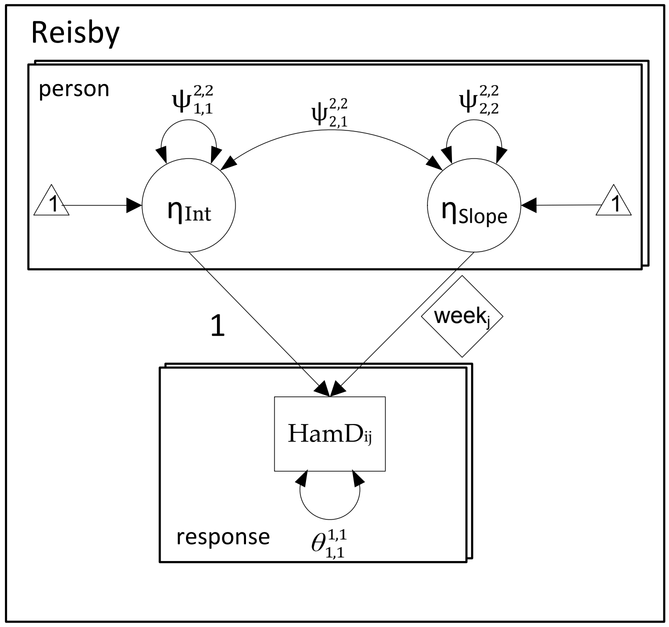
where,

$$\Lambda_{ij}^{(1,2)} = [1.0 \quad w_{ij}].$$

$$e \sim N\left(0, \Theta^{(1,1)}\right)$$

$$\eta \sim N\left(\alpha, \Psi^{(2,2)}\right)$$

3.2.1 PATH DIAGRAM



3.2.2 XXM MODEL MATRICES

3.2.2.1 LEVEL-1: WITHIN MATRICES

3.2.2.1.1 RESIDUAL COVARIANCE MATRIX

As before, we have a single dependent variable and hence a single parameter level-1 residual variance ($\theta_{1,1}$) at level-1. Hence, the residual covariance or theta matrix is a (1×1) matrix:

$$\Theta^{1,1} = [\theta^{1,1}_{1,1}].$$

3.2.2.2 LEVEL-2: WITHIN MATRICES

At level-2, we have two latent variables: Intercept and slope. Hence, we have two latent means and a covariance matrix.

3.2.2.2.1 LATENT MEANS

The latent variable mean matrix is a (2×1) matrix:

$$\alpha^2 = \begin{bmatrix} \alpha_1^2 \\ \alpha_2^2 \end{bmatrix}$$

α_1^2 is the mean of the intercept and α_2^2 is the mean of the slope parameter or the average effect of $week_{ij}$ on $HamD_{ij}$, a measure of depression.

3.2.2.2.2 LATENT FACTOR COVARIANCE MATRIX

The latent covariance matrix is a (2×2) matrix with two variances and single covariance:

$$\Psi^{2,2} = \begin{bmatrix} \psi_{1,1}^{2,2} & \\ \psi_{2,1}^{2,2} & \psi_{2,2}^{2,2} \end{bmatrix}$$

$\psi_{1,1}^{2,2}$ is the variance of the intercept factor representing variability in the intercept of y_{ij} across persons and $\psi_{2,2}^{2,2}$ is the variance of the slope parameter representing between-persons variability in the effect of $week_{ij}$ on $HamD_{ij}$. Finally, $\psi_{2,1}^{2,2}$ is the covariance between the intercept and slope factors.

3.2.2.3 ACROSS LEVEL MATRICES: PERSON TO RESPONSE

As described above, we need to capture the effect of level-2 intercept and slope factors on the level-1 dependent variable using a factor-loading matrix with fixed parameters.

The factor-loading matrix ($\Lambda^{1,2}$) has a single row and two columns (1×2):

$$\Lambda^{1,2} = [1.0 \quad week_{ij}].$$

The first column is fixed to 1.0, whereas the second column is fixed to person-specific values of $week_{ij}$. Collectively, the within and across-level matrices described here specify all of the parameters necessary to model individual differences in over-time trajectories of reaction times across persons. Next, we provide SAS, MPlus, and xxM code for fitting LGCs.

3.3 CODE LISTING

3.3.1 SAS PROC MIXED

```
Proc Mixed data = reisby covtest;
CLASS subject;
MODEL HamD = week/s;
RANDOM Intercept week/subject = subject G type = UN;
RUN;
```

SAS code for a random-slopes model uses a CLASS statement to identify the level-2 units, in this case person. The MODEL statement estimates the fixed-effects (α). The RANDOM statement specifies that the level-1 intercepts and the effect of level-1 predictor $week_{ij}$ is allowed to vary across "subject" (i.e., persons). The covariance among the random-effects (G) is freely estimated (specified by "type = UN"). The G matrix corresponds to the xxM Ψ matrix. Finally, like all regression models, Proc Mixed estimates the residual variance of the level-1 dependent variable ($\theta_{1,1}$) by default. The important thing to note is that there is one-to-one correspondence between the parameters estimated in Proc Mixed and SEM.

3.3.2 MPLUS

Mplus allows latent growth curve model (long version) model to be estimated for two level data.

```
TITLE: Univariate LGC
DATA: File is reisby.dat;
VARIABLE:
  Names = HamD week subject;
  Within= x;
  Cluster = subject;
ANALYSIS:
  TYPE = TWOLEVEL RANDOM;
MODEL:
%WITHIN%
s | HamD ON week;
HamD;
%BETWEEN%
HamD s;
HamD with s;
[HamD s];
```

At level-1, the vertical bar in "s | HamD ON week;" is a command to treat the regression of HamD scores (HamD) on week as random at level-2. The name "s" on the left side of the bar is the name of the random-effect, in this case the random-slope. At level-2, "HamD" represents the intercept of the corresponding level-1 dependent variable. The same five parameters are estimated at level-2: means of the intercepts and slopes ("[HamD s];"), their respective variances ("HamD s;"), and their covariance ("HamD with s;")

3.3.3 XXM

As in the previous example, there is a single level-1 dependent variable and a single level-1 independent variable. The number of matrices is the same as before, and their dimensions are also identical. :

1. We do not wish to estimate any factor-loadings. Factor-loadings are *fixed* to 1.0 and $week_{ij}$. Hence, both elements of pattern matrix are zero:

$\Lambda_{pat} = [0 \quad 0]$

2. We want to fix the first-factor loading to 1.0 (intercept). We use the value matrix to provide the fixed-value of 1.0 for the first factor-loading. The second factor-loading does not have a single fixed value for every observation. Instead each observation (i) would have its own value for that factor-loading (x_{ij}). Clearly, the value matrix cannot be used for providing individual specific fixed values. Hence, the second element in the value matrix is left as 0.0. xxM ignores it internally.

$\Lambda_{val} = [1.0 \quad 0.0]$

3. The job of fixing the factor-loading is left to the *label* matrix. A label matrix is used to assign labels to each parameter within the matrix. Label matrices can be used to impose equality constraints across matrices. Any two parameters with the same label are constrained to be equal. Label matrices are also used for specifying that a specific parameter is to be fixed to data-values. In this case, the first-label is irrelevant as that parameter has already been fixed to 1.0. We use a descriptive label *lambda_11* as the first label (something such as *Justin* would have worked as well). The second factor-loading is the one we are interested in. We want to fix the second factor-loading to the observation specific values of the predictor ($week_{ij}$). This is accomplished by using a two-part label: *levelName.predictorName*. In this case, the predictor is a response level variable. Hence, the first part of the label is *response*. The second part is the actual predictor name, in this case *week*.

$\Lambda_{label} = [l_{1,1} \quad week_{ij}]$

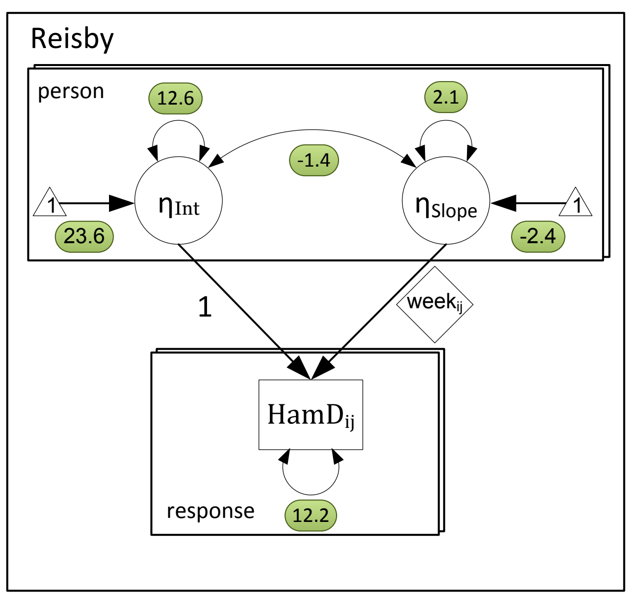
```
lambda_label   <- matrix( c("lambda_11","response.week"), 2, 1 )
```

The complete listing of xxM code for the latent growth curve (long version) example follows.

```
reisby <- xxmModel( levels = c( "response", "subject") )

reisby <- xxmSubmodel( model = reisby, level = "response", parents =
c("subject"), ys = c("hamd"), xs = c("week"), etas = , data = response
)

reisby <- xxmSubmodel( model = reisby, level = "subject", parents = ,
ys = , xs = , etas = c("Int", "Slope"), data = subject)

theta_pattern <- matrix( 1,1,1 )
```

```
theta_value   <- matrix( 20,1,1 )

psi_pattern <- matrix( c(1,1,1,1), 2,2 )

psi_value   <- matrix( c(1,.01,.01,1), 2,2 )

alpha_pattern <- matrix( c(1,1) , 2,1)

alpha_value <- matrix( c(10, -.4) , 2,1)

lambda_pattern <- matrix( c(0,0), 1, 2 )

lambda_value   <- matrix( c(1,0), 1, 2 )

lambda_label   <- matrix( c("lambda_11", "response.week"), 1, 2 )

reisby <- xxmWithinMatrix( model = reisby, level = "response", type =
"theta", pattern = theta_pattern, value = theta_value)

reisby <- xxmWithinMatrix( model = reisby, level = "subject", type  =
"psi", pattern = psi_pattern, value = psi_value)

reisby <- xxmWithinMatrix( model = reisby, level = "subject", type =
"alpha", pattern = alpha_pattern, value = alpha_value)

riesby <- xxmBetweenMatrix( model = riesby, parent = "subject", child
= "response", type = "lambda",  pattern = lambda_pattern, value =
lambda_value, label = lambda_label)

riesby <- xxmRun( riesby )
```

3.4 RESULTS



4 LATENT GROWTH CURVE MODEL – WIDE FORMAT

Next, we provide a mathematically equivalent representation of the previous example. In this version we use conventional SEM formulation of latent growth curves. The data must first be in a wide format, with columns for each week (week 0- week 5) that contain the Hamilton depression ratings for that week. Below are the first two rows of the wide format of the Reisby data:

	subject	HamD0	HamD1	HamD2	HamdD3	HamD4	HamD5
1	101	26	22	18	7	4	3
2	103	33	24	15	24	15	13

4.1 UNIVARIATE LGC MODEL –WIDE FORMAT

4.1.1 EQUATIONS

We begin with the single equation from the previous chapter.

$$HamD_{iw} = 1 \times \eta_{0i} + x_w \times \eta_{1i} + e_{iw}.$$

where $w = 1, 2, 3, 4, 5,$ and 6 and $x_w = 0, 1, 2, 3, 4,$ and 5. η_{0i} is the initial status (Hamilton depression rating at the baseline for each participant) and η_{1i} is the growth rate (trend) for subject i.

If we rewrite equation for the six occasions we get:

$$HamD_{i1} = 1 \times \eta_{0i} + 0 \times \eta_{1i} + e_{i1}$$

$$HamD_{i2} = 1 \times \eta_{0i} + 1 \times \eta_{1i} + e_{i2}$$

$$HamD_{i3} = 1 \times \eta_{0i} + 2 \times \eta_{1i} + e_{i3}$$

$$HamD_{i4} = 1 \times \eta_{0i} + 3 \times \eta_{1i} + e_{i4}$$

$$HamD_{i5} = 1 \times \eta_{0i} + 4 \times \eta_{1i} + e_{i5}$$

$$HamD_{i6} = 1 \times \eta_{0i} + 5 \times \eta_{1i} + e_{i6}$$

4.1.2 NL-SEM REPRESENTATION

$$y_i^{(1)} = 1_{i,i}^{(1,1)} \times \eta_{1,i}^{(1)} + x_{i,i}^{(1,1)} \times \eta_{2,i}^{(1)} + e_i^{(1)}$$

More generally, we can use a matrix representation:

$$y_i^{(1)} = \Lambda_{i,i}^{(1,1)} \times \eta_i^{(1)} + e_i^{(1)}$$

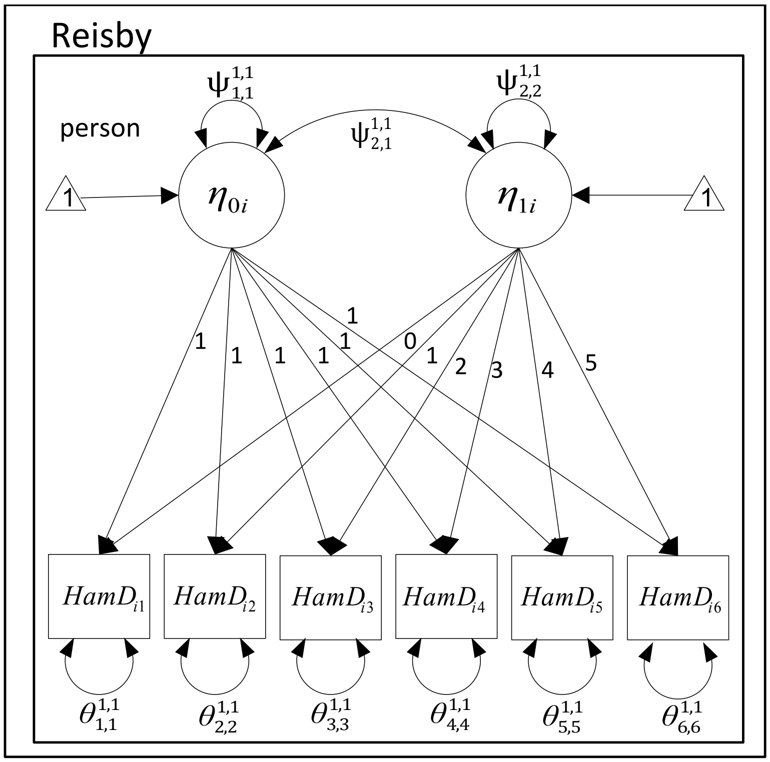
where,

$\Lambda_{i,i}^{(1,1)} = [1.0 \quad x_{ij}].$

$e \sim N\left(0, \Theta^{(1,1)}\right)$

$\eta \sim N\left(\alpha, \Psi^{(2,2)}\right)$

4.1.3 PATH DIAGRAM



4.1.4 XXM MODEL MATRICES

4.1.4.1.1 RESIDUAL COVARIANCE MATRIX

We have six observed variables. Hence, the residual covariance or theta matrix is a (6×6) diagonal matrix:

$$\Theta^{1,1} = \begin{bmatrix} \theta_{1,1}^{1,1} & & & & & \\ & \theta_{2,2}^{1,1} & & & & \\ & & \theta_{3,3}^{1,1} & & & \\ & & & \theta_{4,4}^{1,1} & & \\ & & & & \theta_{5,5}^{1,1} & \\ & & & & & \theta_{6,6}^{1,1} \end{bmatrix}.$$

Unlike the long format, in this case the residual variances at each occasion present themselves for our consideration. We can see them! To keep the model consistent with the one estimated in the long format, we must constrain all 6 diagonal elements to be equal.

4.1.4.1.2 LATENT MEANS

The latent variable mean matrix is a (2×1) matrix:

$$\alpha^1 = \begin{bmatrix} \alpha_1^1 \\ \alpha_2^1 \end{bmatrix}$$

α_1^1 is the mean of η_{0i} (equivalent to the intercept in example 3) and α_2^1 is the mean of η_{1i} (equivalent to the slope in example 3).

4.1.4.1.3 LATENT FACTOR COVARIANCE MATRIX

The latent covariance matrix is a (2×2) matrix with two variances and single covariance:

$$\Psi^{1,1} = \begin{bmatrix} \psi_{1,1}^{1,1} & \\ \psi_{2,1}^{1,1} & \psi_{2,2}^{1,1} \end{bmatrix}$$

$\psi_{1,1}^{1,1}$ is the variance of the initial status representing variability in η_{0i} across persons and $\psi_{2,2}^{1,1}$ is the variance of the growth trend representing variability in η_{1i}. Finally, $\psi_{2,1}^{1,1}$ is the covariance between the two.

4.1.4.1.4 LAMBDA MATRIX

The lambda matrix is 6 X 2. The rows represent the observed variables (HamD1-HamD6) and the columns represent the latent variables. As in the case of the long format, the factor loadings are not estimated. Instead these are fixed to. The first column is fixed to 1.0 and the second column is fixed to person specific values of 'time'.

$$\Lambda^{1,1} = \begin{bmatrix} \lambda_{1,1}^{1,1} & \lambda_{1,2}^{1,1} \\ \lambda_{2,1}^{1,1} & \lambda_{2,2}^{1,1} \\ \lambda_{3,1}^{1,1} & \lambda_{3,2}^{1,1} \\ \lambda_{3,1}^{1,1} & \lambda_{4,2}^{1,1} \\ \lambda_{5,1}^{1,1} & \lambda_{5,2}^{1,1} \\ \lambda_{6,1}^{1,1} & \lambda_{6,2}^{1,1} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 2 \\ 1 & 3 \\ 1 & 4 \\ 1 & 5 \end{bmatrix}.$$

4.1.5 CODE LISTING

4.1.5.1 MPLUS

```
TITLE: Multivariate LGC
DATA: File is riesby.wide.dat;
VARIABLE:
  Names = HamD1-HamD6;
MODEL:
i BY HamD1-HamD6@1;
s BY HamD1@0 HamD2@1 HamD3@2 HamD4@3 HamD5@4 HamD6@5;
[HamD1-HamD6@0];
HamD1-HamD6 (theta);
[i s];
i with s;
```

4.1.5.2 XXM

4.1.5.2.1 RESIDUAL COVARIANCE MATRIX

For the pattern matrix of the residual covariances, we want to estimate the residual variances and we want to fix the covariances. Thus we have the following pattern matrix:

$$\Theta_{pat}^{1,1} = \begin{bmatrix} 1 & & & & & \\ 0 & 1 & & & & \\ 0 & 0 & 1 & & & \\ 0 & 0 & 0 & 1 & & \\ 0 & 0 & 0 & 0 & 1 & \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Since, we want to fix all the covariance to 0, the corresponding values are 0 in the following value matrix:

$$\Theta_{val}^{1,1} = \begin{bmatrix} 1000 & & & & & \\ 0 & 1000 & & & & \\ 0 & 0 & 1000 & & & \\ 0 & 0 & 0 & 1000 & & \\ 0 & 0 & 0 & 0 & 1000 & \\ 0 & 0 & 0 & 0 & 0 & 1000 \end{bmatrix}$$

We want to set all the residual values to be equal. To do so, we use the label matrix and make all the variance the same name, such as "theta." In xxM when parameters have the same name in a label matrix, the parameters will be constrained be equal. The covariances are equal and equal 0, so zeros as labels will work just fine here.

$$\Theta_{lab}^{1,1} = \begin{bmatrix} theta & & & & & \\ 0 & theta & & & & \\ 0 & 0 & theta & & & \\ 0 & 0 & 0 & theta & & \\ 0 & 0 & 0 & 0 & theta & \\ 0 & 0 & 0 & 0 & 0 & theta \end{bmatrix}$$

The following code will specify the theta matrices:

```
th_pat <-diag(1,6)

th_val <-diag(1000,6)

th_lab <- diag(0, 6)

diag(th_lab) <- rep("theta",6)

reisby.wide <- xxmWithinMatrix(reisby.wide,"subject","theta", th_pat,
th_val, th_lab)
```

4.1.5.2.2 LATENT MEANS

We want to freely estimate the latent means, thus the alpha pattern matrix is as follows:

$$\alpha_{pat}^{1} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

4.1.5.2.3 LATENT FACTOR COVARIANCE MATRIX

We also want to freely estimate the variances and the covariance in psi matrix, thus the psi pattern matrix is as follows:

$$\Psi_{pat}^{1,1} = \begin{bmatrix} 1 & \\ 1 & 1 \end{bmatrix}$$

4.1.5.2.4 LAMBDA MATRIX

The pattern lambda matrix is 6 X 2 and is filled with all zeros, because we are fixing all the loading factors as depicted in the diagram. The complete lambda pattern matrix is as follows:

$$\Lambda_{pat}^{1,1} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

The lambda matrix contains all one's in the first column to connect the six observed variables with the initial status (η_{0i}) variable. The second column connects the observed variables to the growth trend variable (η_{1i}). The complete lambda value matrix is as follows:

$$\Lambda_{val}^{1,1} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 2 \\ 1 & 3 \\ 1 & 4 \\ 1 & 5 \end{bmatrix}$$

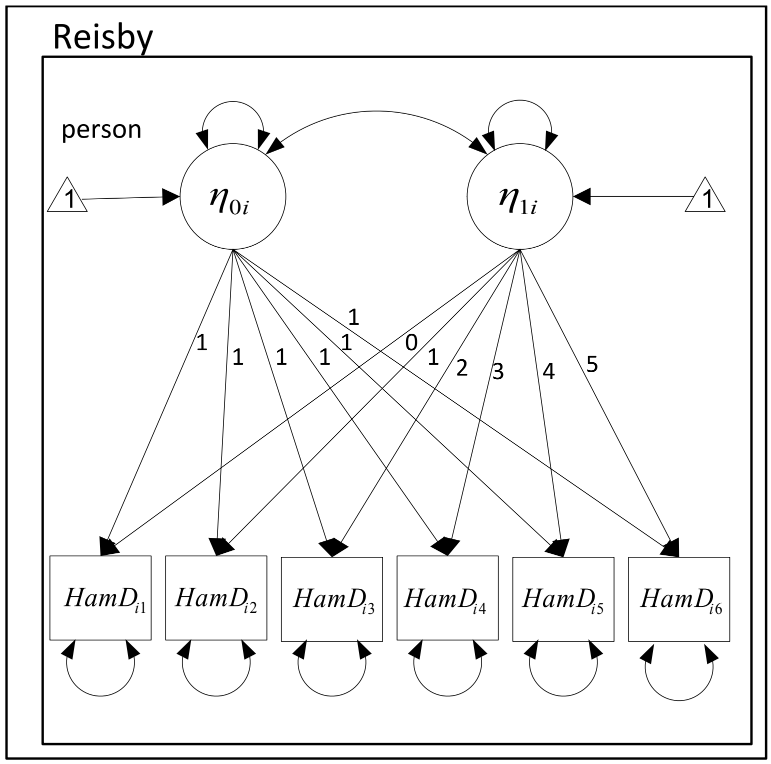
4.1.5.2.5 COMPLETE XXM CODE

```
#level-1 matrices
th_pat <-diag(1,6)
th_val <-diag(1000,6)
th_lab <- diag(0, 6)
diag(th_lab) <- rep("theta",6)


ps_pat <- matrix(1,2,2)
ps_val <- diag(c(10,1),2)
al_pat <- matrix(1,2,1)
al_val <- matrix(c(250, 10),2,1)
```

```
ly_pat <-matrix(0,6,2)

one <- rep(1,6)

week <- seq(from = 0, to = 5, by = 1)

ly_val <-matrix(c(one, week),6,2)


##xxmSubmodel(model, level, parents,  ys, xs, latent, data)

reisby.wide <- xxmModel("subject")


reisby.wide <- xxmSubmodel(model = reisby.wide,

                     level = "subject",

                     parents = ,

                     ys =
c("HamD1","HamD2","HamD3","HamD4","HamD5","HamD6"),

                     xs = ,

                     etas = c("Intercept", "Slope"),

                     data = reisby.wide)

#level 1

reisby.wide <- xxmWithinMatrix(reisby.wide,"subject","theta", th_pat,
th_val, th_lab)

reisby.wide <- xxmWithinMatrix(reisby.wide,"subject","psi", ps_pat,
ps_val)

reisby.wide <- xxmWithinMatrix(reisby.wide,"subject","alpha", al_pat,
al_val)

reisby.wide <- xxmWithinMatrix(reisby.wide, "subject","lambda",
ly_pat, ly_val)


system.time(reisby.wide <- xxmRun(reisby.wide))

reisby.wide<- xxmFree(reisby.wide)
```

4.1.5.3 RESULTS

Because constrained the residual variances to be equal, all the thetas will be the same.



5 BIVARIATE CROSS-CLASSIFIED MODEL

The previous example illustrated a research design in which a single outcome variable was measured from students who were simultaneously nested within primary and secondary schools, leading to a cross-classified dependency structure. The current example is also a cross-classified model, but with two observed outcome variables at the lowest level. This design allows us to examine the relationship between these variables at each level of the model.

5.1 MOTIVATING EXAMPLE

Our example draws on a common research paradigm in social and cognitive psychology known as the stimulus-sampling designs. In these studies, participants (raters) respond to a series of randomly sampled stimuli (targets). Specifically, responses are simultaneously nested within raters and targets, leading to a cross-classified dependency structure. Treating these sources of non-independence as random effects allows the results to generalize to the larger populations from which these stimuli were drawn (Judd, Westfall, & Kenny, 2012). In the present example, we use data collected from a sample of 243 undergraduate students who evaluated the symmetry (S) and physical attractiveness (PA) of male and female faces in photographs (Langner et al., 2010). Our initial model features a decomposition of SYM and PA ratings into rater- and target-specific variance components and latent correlations between PA and SYM for each of these sources. Response-specific residuals will also be allowed to correlate.

Unlike the standard analytic approach which confounds these distinct sources of variability in photo ratings, the current approach answers precise research questions that are specific to each class of variance component. Specifically, the correlation between rater variance components describes the extent to which individuals who evaluate all photographs as consistently more or less symmetrical, tend to also rate all photographs as more or less attractive. Moreover, the correlation between target variance components expresses the degree to which target photographs that are evaluated by all raters as consistently more or less symmetrical, are also rated as more or less attractive. Finally, the correlation between response-specific residuals reflects idiosyncratic associations between symmetry and attractiveness.

5.2 BIVARIATE CROSS-CLASSIFIED RANDOM INTERCEPTS MODEL

In this case, each rating is simultaneously nested within raters (participants) and targets (photographs). As a result, the symmetry and attractiveness rating is subject to three distinct sources of influence: (1) rater-effects, (2) target-effects, and (3) idiosyncratic effects, which are confounded with measurement error. The following model makes the idea of two sources of systematic influence explicit.

5.2.1 MLM NOTATION

For a response i, provided by rater u and evaluating target v, the level-1 equations are:

$$SYM_{i(u,v)} = \nu_{SYM} + 1 \times \eta^{R}_{SYM,u} + 1 \times \eta^{T}_{SYM,v} + e_{SYM,i(u,v)}$$

$PA_{i(u,v)} = \nu_{PA} + 1 \times \eta^{R}_{PA,u} + 1 \times \eta^{T}_{PA,v} + e_{PA,i(u,v)}$

$e \sim N(0, \Theta)$

$\eta^R \sim N(0, \Psi^R)$

$\eta^T \sim N(0, \Psi^T)$

η^R and η^T are the unobserved latent variables representing the effects of rater and target, respectively, on symmetry and attractiveness responses.. The actual effect for a given response i depends on the specific combination of the rater and the target (u & v) involved in the evaluation. It is assumed that the effects sources of influence are independent (uncorrelated).

The model has eleven parameters:

1. Grand-mean or the intercept for symmetry (ν_{SYM}) and attractiveness (ν_{PA}).
2. Response level residual variances (θ_{SYM} & θ_{PA})and covariance ($\theta_{SYM,PA}$)
3. Rater level latent factor variances (ψ^R_{SYM} & ψ^R_{PA}) and covariance ($\psi^R_{SYM,PA}$) for symmetry and attractiveness.
4. Target level latent factor variances (ψ^T_{SYM} & ψ^T_{PA}), and covariance ($\psi^T_{SYM,PA}$) for symmetry and attractiveness.

5.2.2 NL-SEM REPRESENTATION

The above equations using superscripts *R* and *T* clarify the effects of raters and targets on multivariate responses. However, it is useful to think of these models in more general terms. The general notation is useful in understanding more complex models with observed and latent variables at multiple levels. For this reason, we will use numeric superscripts.

The model has three levels:

(1) Level 1: *response*
(2) Level 2: rater
(3) Level 3: target

The above equation can be re-written using numeric superscripts to specify the level information as:

$Y^1_{pi} = \nu^1_p + 1^{1,2}_{i,u} \times \eta^2_{p,u} + 1^{1,3}_{i,v} \times \eta^3_{p,v} + e^1_{p,i}$

Here, we use superscript 1 for the dependent variable Y_p measured at the lowest level (response) and superscripts *2* and *3* for the corresponding random-intercepts for rater (2) and target (3) levels, respectively. For two dependent variables, we can re-write the above equation as:

$Y^1_{1i} = \nu^1_1 + 1^{1,2}_{i,u} \times \eta^2_{1,u} + 1^{1,3}_{i,v} \times \eta^3_{1,v} + e^1_{1i}$

$Y^1_{2i} = \nu^1_2 + 1^{1,2}_{i,u} \times \eta^2_{2,u} + 1^{1,3}_{i,v} \times \eta^3_{2,v} + e^1_{2i}$

We could entirely the variable subscripts be rewriting the above equation as a matrix expression.

$$Y_i^1 = \nu^1 + \Lambda_{i,u}^{1,2} \times \eta_u^2 + \Lambda_{i,v}^{1,3} \times \eta_v^3 + e_i^1$$

We can now express distributional assumptions regarding random-effects and residuals as:

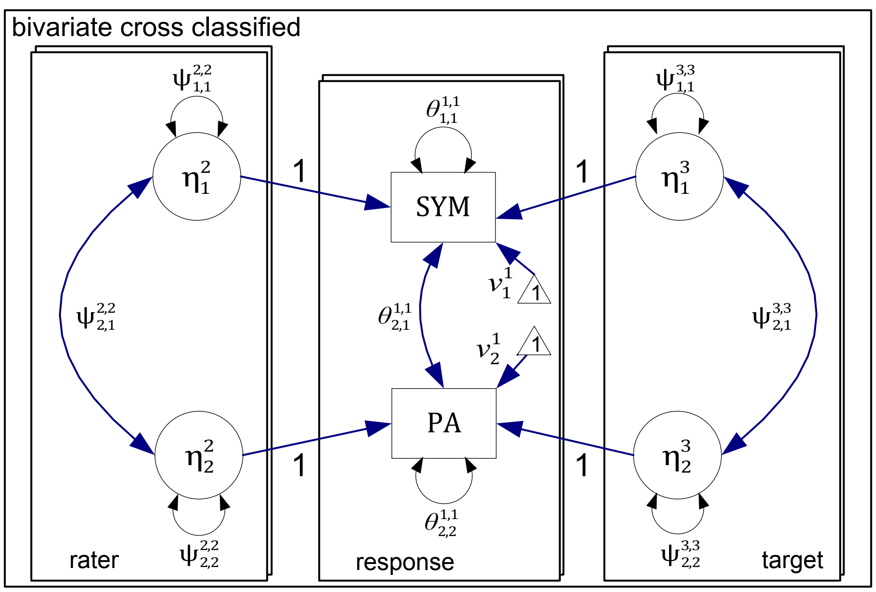
$$e^1 \sim N(0, \Theta^{1,1})$$

$$\eta^2 \sim N(0, \Psi^{2,2})$$

$$\eta^3 \sim N(0, \Psi^{3,3})$$

5.2.3 PATH DIAGRAM

The above model can be conceptualized as a three level xxM model shown in the following diagram. In this case, the three levels are response, rater, and target. The response level is nested simultaneously within rater and target levels. For each level, there is a separate sub-model delineated by a rectangle.



At the response level, we have two observed dependent variables and no latent variables. The remaining two levels each have two latent variables representing rater and target effects for symmetry and attractiveness. Each of these latent variables influences the response level outcome as indicated by the directional arrow. Consistent with the above equations, the coefficient (factor-loading) linking each latent variable to the observed response is fixed to 1.0.

5.2.4 XXM MODEL MATRICES

The above model can be described more conveniently in matrix notation. As described earlier, we will use super-scripts to indicate level and subscripts to indicate variables. Within-level matrices for each of the three levels (response, rater, and target) are presented first, followed by across-level matrices connecting the rater (level-2) and target (level-3) levels with the response level (level-1).

5.2.4.1 LEVEL-1 (RESPONSE): WITHIN MATRICES

Level-1 is the response level. There are two parameter matrices: (1) theta (Θ): the residual covariance matrix and (2) nu (ν): the observed variable intercepts.

5.2.4.1.1 LEVEL-1 RESIDUAL COVARIANCE MATRIX

With two observed dependent variables SYM and PA at level-1, specification of the response level is rather straightforward. Hence, the residual covariance or theta matrix is 2×2:

$$\Theta^{1,1} = \begin{bmatrix} \theta_{1,1}^{1,1} & \\ \theta_{2,1}^{1,1} & \theta_{2,2}^{1,1} \end{bmatrix}$$

Superscripts make it clear that we are referring to the level-1 covariance matrix. The two subscripts indicate SYM and PA.

5.2.4.1.2 LEVEL-1 OBSERVED VARIABLE INTERCEPT

Because the rater and target level effects are expressed in deviation terms, mean-structure will be modeled at the response level by estimating intercepts for SYM and PA, using a nu (ν) matrix. It is important to keep in mind that intercepts are fixed-effects or constants that are not tied to individual responses, raters, or targets. Rather, the intercepts reflect the grand-mean of the dependent variables across all levels:

$$\nu^1 = \begin{bmatrix} \nu_1^1 \\ \nu_2^1 \end{bmatrix}$$

where ν_1^1 represents the grand-mean of symmetry (SYM), and ν_2^1 reflects the average attractiveness (PA) rating across all responses. Note that the superscripts indicate that the intercepts are at level-1.

5.2.4.2 LEVEL-2 (RATER): WITHIN MATRICES

There are two latent variables at the rater level (η_{SYM}^R & η_{PA}^R) with means of zero and a variance-covariance structure described in the following psi (Ψ) matrix.

5.2.4.2.1 LEVEL-2 RATER LATENT FACTOR COVARIANCE MATRIX

The rater level latent covariance matrix is 2 x 2:

$$\Psi^{2,2} = \begin{bmatrix} \psi_{1,1}^{2,2} & \\ \psi_{2,1}^{2,2} & \psi_{2,2}^{2,2} \end{bmatrix}$$

$\psi_{1,1}^{2,2}$ is the variance of rater SYM, $\psi_{2,2}^{2,2}$ is the variance of rater PA, and $\psi_{2,1}^{2,2}$ is the covariance between rater SYM and PA.

5.2.4.3 LEVEL-3 (TARGET): WITHIN MATRICES

There are also two latent variables at the target level, (η_{SYM}^{T} & η_{PA}^{T}) with means of zero and a variance-covariance structure described in the following psi (Ψ) matrix.

5.2.4.3.1 LEVEL-3 LATENT FACTOR COVARIANCE MATRIX

The target level latent covariance matrix is 2 x 2:

$$\Psi^{3,3} = \begin{bmatrix} \psi_{1,1}^{3,3} & \\ \psi_{2,1}^{3,3} & \psi_{2,2}^{3,3} \end{bmatrix}$$

$\psi_{1,1}^{3,3}$ is the variance of *target* SYM, $\psi_{2,2}^{3,3}$ is the variance of *target* PA, and $\psi_{2,1}^{3,3}$ is the covariance between *target* SYM and PA.

Raters and photographs (targets) each have their own sub-model. We now need to connect latent variables for these levels with the corresponding observed variable at the response level. In order to do so, we need *two separate* factor-loading matrices – one connecting the rater latent variables with the response outcomes and the second connecting the target level latent variables with the same response outcomes. This is specified by two separate across-level factor-loading matrices.

5.2.4.4 ACROSS-LEVEL MATRICES: LEVEL-2 (RATER) TO LEVEL-1 (RESPONSE)

If you look closely, the equations and diagrams clearly indicate:

(1) The latent variable η_{SYM}^{R} in the rater level is measured by observed variable SYM. The corresponding coefficient is 1.0.

(2) The latent variable η_{SYM}^{R} is not measured by PA. Hence, the corresponding coefficient is implicitly 0.0. For this reason, there is no path in the diagram from η_{SYM}^{R} to PA.

(3) Similarly, the latent variable η_{PA}^{R} in the rater level is measured by observed variable PA. The corresponding coefficient is 1.0.

(4) The latent variable η_{PA}^{R} for *rater* is not measured by SYM. Hence, the corresponding coefficient is implicitly 0.0.

We can succinctly represent this idea by specifying the value factor-loading matrix connecting the two *rater* latent variables (η_{SYM}^{R} & η_{PA}^{R}) with the corresponding response variables (SYM & PA) as follows:

$$\Lambda_{val}^{1,2} = \left| \begin{array}{cc} & \begin{array}{cc} \eta_{SYM}^{R} & \eta_{PA}^{R} \end{array} \\ \begin{array}{c} SYM \\ PA \end{array} & \begin{bmatrix} 1.0 & 0.0 \\ 0.0 & 1.0 \end{bmatrix} \end{array} \right|$$

The rows indicate the two dependent variables and the columns indicate the two latent independent variables. While the above matrix is sufficient for the current purpose, in general we prefer to use superscripts and subscripts to indicate levels and variables.

$$\Lambda^{1,2}_{val} = \begin{array}{c c} & \begin{array}{cc} \eta^2_1 & \eta^2_2 \end{array} \\ \begin{array}{c} Y^1_1 \\ Y^1_2 \end{array} & \begin{bmatrix} 1.0 & 0.0 \\ 0.0 & 1.0 \end{bmatrix} \end{array}$$

Superscripts make it clear that the effect of first latent variable at level-2 (rater, SYM) on first observed variable at level-1 (response, SYM) is 1.0. Similarly, the effect of second latent variable at level-2 (rater, PA) on second observed variable at level-1 (response, PA) is 1.0. The remaining two effects are obviously zero. Notice that the factor loading matrix ($\Lambda^{1,2}$) includes superscripts to indicate that this matrix defines measurement relationship across levels 1 and 2.

In general, elements of across-level matrices have the following structure.

$$\Lambda^{1,2} = \begin{array}{c c} & \begin{array}{cc} \eta^2_1 & \eta^2_2 \end{array} \\ \begin{array}{c} Y^1_1 \\ Y^1_2 \end{array} & \begin{bmatrix} \lambda^{1,2}_{1,1} & \lambda^{1,2}_{1,2} \\ \lambda^{1,2}_{2,1} & \lambda^{1,2}_{2,2} \end{bmatrix} \end{array}$$

Note that each element has superscripts indicating which two levels are being connected and subscripts indicating which two variables are being connected. In this case, we are not estimating any factor-loadings. As such the specification of rater level to the *response* level factor-loading matrix is:

$$\Lambda^{1,2} = \begin{bmatrix} 1.0 & 0.0 \\ 0.0 & 1.0 \end{bmatrix}.$$

5.2.4.5 ACROSS-LEVEL MATRICES: LEVEL-3 (TARGET) TO LEVEL-1 (RESPONSE)

Factor-loading matrix connecting target latent variables with response observed variables is identical to the corresponding *rater* to *response* matrix.

$$\Lambda^{1,3} = \begin{bmatrix} 1.0 & 0.0 \\ 0.0 & 1.0 \end{bmatrix}.$$

It is now apparent that with multiple levels, we need explicit superscripts to clearly indicate which two levels are being connected.

5.2.4.6 SUMMARY OF MODEL MATRICES

The following table provides a complete summary of pattern matrices:

	Matrix	Pattern
Level 1: Θ	$\Theta^1 = \begin{bmatrix} \theta^{1,1}_{1,1} & \\ \theta^{1,1}_{2,1} & \theta^{1,1}_{2,2} \end{bmatrix}$	$\Theta^{1,1} = \begin{bmatrix} 1 & \\ 1 & 1 \end{bmatrix}$

Level 1: ν	$\nu^1 = \begin{bmatrix} \nu_1^1 \\ \nu_2^1 \end{bmatrix}$	$\nu^1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$
Level 2: Ψ	$\Psi^{2,2} = \begin{bmatrix} \psi_{1,1}^{2,2} & \\ \psi_{2,1}^{2,2} & \psi_{2,2}^{2,2} \end{bmatrix}$	$\Psi^{2,2} = \begin{bmatrix} 1 & \\ 1 & 1 \end{bmatrix}$
Level 3: Ψ	$\Psi^{3,3} = \begin{bmatrix} \psi_{1,1}^{3,3} & \\ \psi_{2,1}^{3,3} & \psi_{2,2}^{3,3} \end{bmatrix}$	$\Psi^{3,3} = \begin{bmatrix} 1 & \\ 1 & 1 \end{bmatrix}$
Level 2 → Level 1: Λ	$\Lambda^{1,2} = \begin{bmatrix} \lambda_{1,1}^{1,2} & \lambda_{1,2}^{1,2} \\ \lambda_{2,1}^{1,2} & \lambda_{2,2}^{1,2} \end{bmatrix}$	$\Lambda^{1,2} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}.$
Level 3 → Level 1: Λ	$\Lambda^{1,3} = \begin{bmatrix} \lambda_{1,1}^{1,3} & \lambda_{1,2}^{1,3} \\ \lambda_{2,1}^{1,3} & \lambda_{2,2}^{1,3} \end{bmatrix}$	$\Lambda^{1,3} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$

The important thing to note is that all parameters for the within-level matrices are freely estimated, whereas the two across-level factor-loading matrices have no free parameters.

5.3 CODE LISTING

5.3.1 SAS PROC MIXED

```
Proc Mixed data = faces covtest;
  CLASS rater target response SYM_PA;
  MODEL: y = SYM PA / noint s;
  RANDOM SYM PA / subject = rater type = un;
  RANDOM SYM PA / subject = target type = un;
  REPEATED SYM_PA / subject = response type = un r rcorr;
RUN;
```

Because SAS PROC MIXED is designed to fit univariate mixed-effects models, we must 'trick' the procedure into fitting a multivariate model. This involves creating a SAS data file (faces.sas7bdat) containing a single outcome variable (y) containing both SYM and PA values for each response. Additionally, a variable denoting the outcome variable (SYM_PA) is added as the sole fixed predictor, which SAS converts into a dummy vector. The RANDOM statements identify rater and target levels as two independent levels with independent random effects. These statements estimate variance of respective rater and target levels random intercepts for each outcome variable. The REPEATED statement allows response level residuals for SYM and PA to covary.

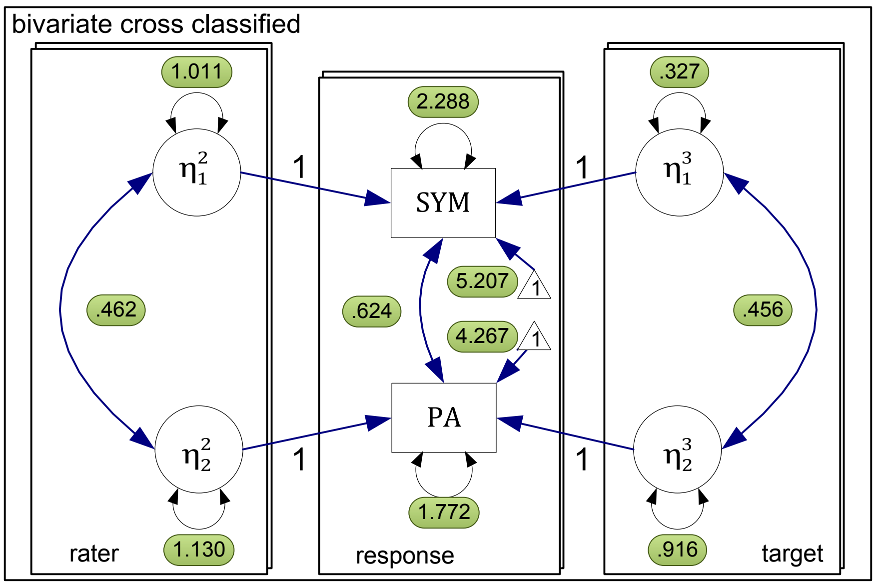
5.3.2 XXM

The following code closely follows the xxM model description presented earlier.

```
library(xxm)

faces <- xxmModel(levels = c("response","rater","target"))

faces <- xxmSubmodel(model = faces, level = "response", parents =
c("rater","target"), ys = c("SYM", "PA"), xs =, etas =, data =
response)

faces<- xxmSubmodel(model = faces, level = "rater", parents = ,
ys =, xs =, etas = c("rater_SYM", "rater_PA"), data = rater)

faces <- xxmSubmodel(model = faces, level = "target", parents = ,
ys = , xs =, etas = c("target_SYM", "target_PA"), data = target)



resp_th_pat <- matrix(c(1,1,1,1),2,2)

resp_th_val <- matrix(c(2,.0,.0,2),2,2)

resp_nu_pat <- matrix(c(1,1),2,1)

resp_nu_val <- matrix(c(5,5),2,1)

faces <- xxmWithinMatrix(model = faces, level = "response", type =
"theta",                    pattern = resp_th_pat, value =
resp_th_val)

faces <- xxmWithinMatrix(model = faces, level = "response", type =
"nu",                     pattern = resp_nu_pat, value = resp_nu_val)



rater_psi_pat <- matrix(c(1,1,1,1),2,2)

rater_psi_val <- matrix(c(.5,.25,.25,.5),2,2)

faces <- xxmWithinMatrix(model = faces, level = "rater", type = "psi",
pattern = rater_psi_pat, value = rater_psi_val)

target_psi_pat <- matrix(c(1,1,1,1),2,2)

target_psi_val <- matrix(c(.5,.25,.25,.5),2,2)

faces <- xxmWithinMatrix(model = faces, level = "target", type =
"psi",                     pattern = target_psi_pat, value =
target_psi_val)

rater_resp_la_pat <- matrix(c(0,0,0,0),2,2)

rater_resp_la_val <- matrix(c(1,0,0,1),2,2)

faces <- xxmBetweenMatrix(model = faces, parent = "rater", child =
“response", type = "lambda", pattern = rater_resp_la_pat, value =
rater_resp_la_val)
```

```
target_resp_la_pat <- matrix(c(0,0,0,0),2,2)

target_resp_la_val <- matrix(c(1,0,0,1),2,2)

faces <- xxmBetweenMatrix(model = faces, parent = "target", child =
"response", type = "lambda", pattern = target_resp_la_pat, value =
target_resp_la_val)

faces <- xxmRun( faces )
```

5.4 RESULTS



6 TWO LEVEL CONFIRMATORY FACTOR ANALYSIS

We now illustrate a multilevel confirmatory factor analysis (CFA) model (see Mehta, 2013).

6.1 MOTIVATING EXAMPLE

Data from the present example were drawn from a large-scale study in which indicators of verbal ability were measured for 1,141 students nested within 163 teachers. We will use xxM to fit unrestricted CFA models at the student and teacher levels in an effort to explain the common covariance among these indicators at each level.

6.2 TWO-LEVEL CFA MODEL

In this case, each indicator of reading ability varies across students and teachers, and performance on these indicators is correlated within each level. Estimating a common latent factor at each level of analysis allows us to parsimoniously explain the common variance/covariance among the indicators.

6.2.1 SCALAR REPRESENTATION

For clarity, we present scalar equations. However, it will become apparent that the number of subscripts that we have to consider increases very rapidly. The matrix equation presented later is succinct and clear.

6.2.1.1 STUDENT SUBMODEL (LEVEL-1)

The measurement model for each indicator of reading ability at the student level can be presented as:

$$y_{pi}^{1} = \nu_{p}^{1} + \lambda_{p,1}^{1,1} \times \eta_{1i}^{1} + e_{pi}^{1}$$

where, y_{pij} is p^{th} observed indicator for student i, nested within teacher j. The superscript 1 is for the student level.

$$\eta_{1i}^{1} \sim N\left(0, \psi_{1,1}^{1,1}\right)$$

$$e_{pi}^{1} \sim N\left(0, \theta_{p,p}^{1,1}\right)$$

The student sub-model has the following parameters:

1. $(p-1)$ factor-loadings ($\lambda_{p,1}^{1,1}$) with the first factor loading being fixed to 1.0 for scale identification.
2. Residual variance for each of the p observed indicators ($\theta_{p,p}$)
3. Single latent variance ($\psi_{1,1}^{1,1}$).
4. Measurement intercepts for each of the p observed indicators (ν_p).

6.2.1.2 TEACHER SUBMODEL (LEVEL-2) AND ACROSS LEVELS

The teacher level has a single latent variable with mean of zero and unknown residual variance ($\psi_{1,1}^{2,2}$). The superscript 2 is for the teacher level, and the generic teacher-level equation is nearly identical to the student equation:

$$y_{pij}^{1} = \lambda_{p,1}^{1,2} \times \eta_{1j}^{2}$$

where, y_{pij} is p^{th} observed indicator for student i, nested within teacher j. It is important to note that the student level reading outcomes serve as indicators of the reading ability factor at the teacher level. As a result, the student level measurement intercepts and residual variances carry-over from the previous equation. Hence, we need only a single teacher level reading ability latent variable:

$$\eta_{1j}^{2} \sim N\left(0, \psi_{1,1}^{2,2}\right)$$

Because we have already specified the mean structure and residual variances for the observed indicators at the student level, only two ‘types’ of parameter remain:

1. $(p-1)$ factor-loadings ($\lambda_{p,1}^{1,2}$) with the first factor-loading being fixed to 1.0 for scale identification.
2. Single latent variance ($\psi_{1,1}^{2,2}$).

6.2.2 XXM MODEL MATRICES

With multivariate outcomes, the scalar representation that we have used until now becomes cumbersome. The matrix equations are much easier to understand. For the current model, we can describe multivariate outcome at the student level with a single measurement model that defines latent variables at level-1 and at level-2.

The measurement model for each indicator of reading ability at the student level can be presented as:

$$y_i^1 = \nu^1 + \Lambda_{i,i}^{1,1} \times \eta_i^1 + +\Lambda_{i,j}^{1,2} \times \eta_j^2 + e_i^1 \ .$$

6.2.2.1 STUDENT SUBMODEL (LEVEL-1)

6.2.2.1.1 FACTOR-LOADING MATRIX (LAMBDA)

$$\Lambda_{\text{pattern}}^{1,1} = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \end{bmatrix}, \Lambda_{\text{value}}^{1,1} = \begin{bmatrix} 1.0 \\ 1.1 \\ 0.9 \\ 0.8 \end{bmatrix}, \Lambda_{\text{label}}^{1,1} = \begin{bmatrix} \lambda_{1,1}^{1,1} \\ \lambda_{2,1}^{1,1} \\ \lambda_{3,1}^{1,1} \\ \lambda_{4,1}^{1,1} \end{bmatrix}$$

The first factor-loading is fixed to 1.0. Hence, we need to fix the first parameter in the pattern matrix. The value at which it is being fixed is specified in the value matrix. In this case the first factor-loading is being fixed to a value of 1.0.

6.2.2.1.2 OBSERVED RESIDUAL COVARIANCE MATRIX (THETA)

$$\Theta^{1,1}_{\text{pattern}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \Theta^{1,1}_{\text{value}} = \begin{bmatrix} 1.1 & 0.0 & 0.0 & 0.0 \\ 0.0 & 2.1 & 0.0 & 0.0 \\ 0.0 & 0.0 & 1.3 & 0.0 \\ 0.0 & 0.0 & 0.0 & 1.5 \end{bmatrix}$$

The residual covariance matrix is a diagonal matrix, meaning we are only estimating residual variances. Residual covariances are all fixed to 0.0. Again we use a pattern and a value matrix to fix all off-diagonal elements to 0.0.

6.2.2.1.3 LATENT COVARIANCE MATRIX (PSI)

$$\Psi^{1,1}_{\text{pattern}} = [1], \Psi^{1,1}_{\text{value}} = [1.1]$$

6.2.2.1.4 OBSERVED VARIABLEL INTERCEPTS (NU)

$$\nu^{1}_{\text{pattern}} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}, \nu^{1}_{\text{value}} = \begin{bmatrix} 1.1 \\ 2.1 \\ 1.3 \\ .71 \end{bmatrix}.$$

6.2.2.2 TEACHER SUBMODEL (LEVEL-2)

6.2.2.2.1 FACTOR-LOADING MATRIX (LAMBDA)

$$\Lambda^{1,2}_{\text{pattern}} = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \end{bmatrix}, \Lambda^{1,2}_{\text{value}} = \begin{bmatrix} 1.0 \\ 1.1 \\ .9 \\ .8 \end{bmatrix}, \Lambda^{1,2}_{\text{label}} = \begin{bmatrix} \lambda^{1,2}_{1,1} \\ \lambda^{1,2}_{2,1} \\ \lambda^{1,2}_{3,1} \\ \lambda^{1,2}_{4,1} \end{bmatrix}$$

It is important to note that in the current specification $\Lambda^{1,1}_{\text{label}} \neq \Lambda^{1,2}_{\text{label}}$, suggesting that the factor-loadings at each level are to be uniquely estimated. Modifying these matrices so that elements corresponding to common indicators share the same label (e.g., $\lambda^{1,1}_{2,1}, \lambda^{1,2}_{2,1} \rightarrow \lambda_{2,1}$) places equality constraints on loadings to the same indicator at the student and teacher level. The next example illustrates this concept in greater detail.

6.2.2.2.2 LATENT COVARIANCE MATRIX (PSI)

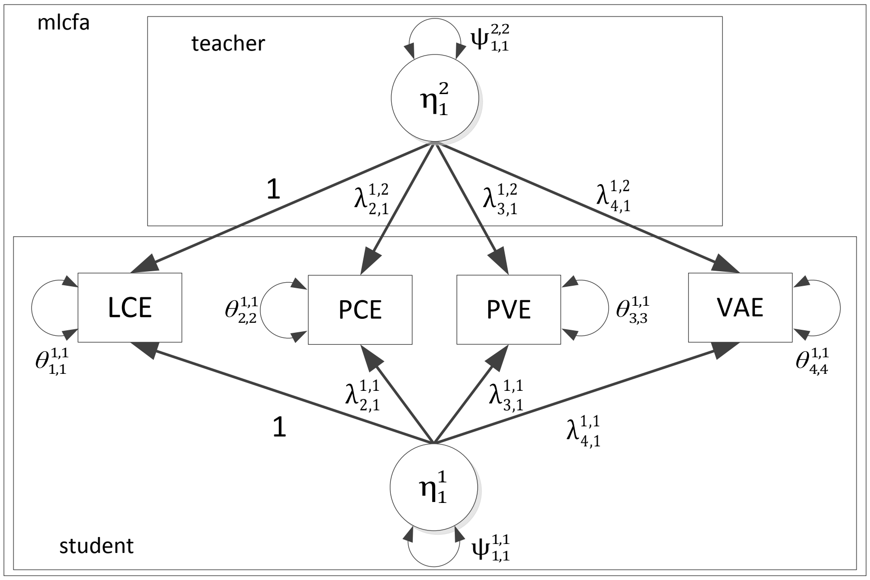
$$\Psi^{2,2}_{\text{pattern}} = [1], \Psi^{2,2}_{\text{value}} = [0.05]$$

6.2.3 MODEL MATRICES SUMMARY

The following table provides a complete summary of parameter matrices:

	Matrix	Pattern
Level 1: Θ	$\Theta^{1,1} = \begin{bmatrix} \theta_{1,1}^{1,1} & & & \\ \theta_{2,1}^{1,1} & \theta_{2,2}^{1,1} & & \\ \theta_{3,1}^{1,1} & \theta_{3,2}^{1,1} & \theta_{3,3}^{1,1} & \\ \theta_{4,1}^{1,1} & \theta_{4,2}^{1,1} & \theta_{4,3}^{1,1} & \theta_{4,4}^{1,1} \end{bmatrix}$	$\Theta^{1,1} = \begin{bmatrix} 1 & & & \\ 0 & 1 & & \\ 0 & 0 & 1 & \\ 0 & 0 & 0 & 1 \end{bmatrix}$
Level 1: ν	$\nu^1 = \begin{bmatrix} \nu_1^1 \\ \nu_2^1 \\ \nu_3^1 \\ \nu_3^1 \end{bmatrix}$	$\nu^1 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$
Level 1: Λ	$\Lambda^{1,1} = \begin{bmatrix} \lambda_{1,1}^{1,1} \\ \lambda_{2,1}^{1,1} \\ \lambda_{3,1}^{1,1} \\ \lambda_{4,1}^{1,1} \end{bmatrix}$	$\Lambda^{1,1} = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \end{bmatrix}$
Level 1: Ψ	$\Psi^{1,1} = \left[\psi_{1,1}^{1,1}\right]$	$\Psi^{1,1} = [1]$
Level 2 → Level 1: Λ	$\Lambda^{1,2} = \begin{bmatrix} \lambda_{1,1}^{1,2} \\ \lambda_{2,1}^{1,2} \\ \lambda_{3,1}^{1,2} \\ \lambda_{4,1}^{1,2} \end{bmatrix}$	$\Lambda^{1,2} = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \end{bmatrix}.$
Level 2: Ψ	$\Psi^{2,2} = \left[\psi_{1,1}^{2,2}\right]$	$\Psi^{2,2} = [1]$

6.2.4 PATH DIAGRAM



6.3 CODE LISTING

6.3.1 MPLUS

Mplus allows confirmatory factor analyses to be estimated for two level data.

```
TITLE: MLCFA
DATA: File is mlcfa.dat;
VARIABLE:
  Names = lce pce pve vae teacher;
  Within = ;
  Cluster = teacher;
ANALYSIS:
  TYPE = TWOLEVEL;
MODEL:
%WITHIN%
Eta_stu BY lce@1 pce* pve* vae*;
Eta_stu*;
%BETWEEN%
Eta_teach BY lce@1 pce* pve* vae*;
```

```
Eta_teach*;
[lce pce pve vae];
```

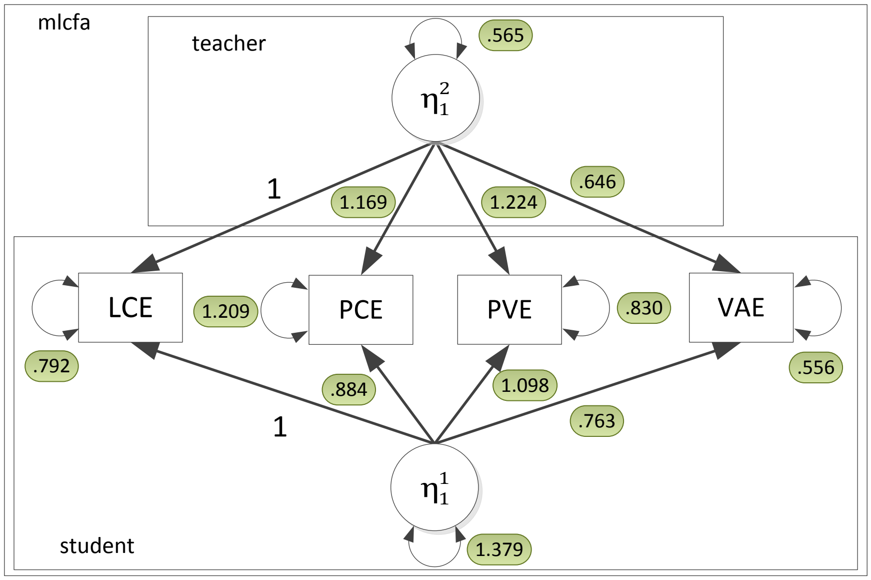
Although the specification described in the earlier equations models mean structure at the student level, Mplus only allows means to be modeled at the between level. Regardless, the model-implied mean structure is identical.

6.3.2 XXM

```
library(xxm)
data(mlcfa.xxm)
#Student: factor-loading matrix
ly11_pat <- matrix(c(0,1,1,1),4,1)
ly11_val <- matrix(c(1,1.1,.9,.8),4,1)
ly11_lab <- matrix(c("l1y1","l1y2","l1y3", "l1y4"),4,1)
#Student: factor-covariance matrix
ps1_pat <- matrix(1,1,1)
ps1_val <- matrix(.1,1,1)
#Student: observed residual-covariance matrix
th1_pat <- diag(1,4)
th1_val <- diag(c(2.68,2.98,3.26,1.57),4)
#Student: "grand-means" NU
nu1_pat <- matrix(1,4,1)
nu1_val <- matrix(c(46.0, 46.99, 46.41, 47.97),4,1)
#Teacher model matrices
#Teacher: factor-covariance matrix
ps2_pat <- matrix(1,1,1)
ps2_val <- matrix(.1,1,1)
#Teacher -> Student factor-loading matrix
ly12_pat <- matrix(c(0,1,1,1),4,1)
ly12_val <- matrix(c(1,1.1,.9,.8),4,1)
```

```
ly12_lab <- matrix(c("l12y1","l12y2","l12y3", "l12y4"),4,1)

mlcfa <- xxmModel(levels = c("student", "teacher"))

### Submodel: Student

mlcfa <- xxmSubmodel(model = mlcfa, level = "student", parents =
c("teacher"), ys = c("LCE", "PCE", "PVE", "VAE"), xs = ,etas =
c("Eta_y_Stu"), data = mlcfa.student)

### Submodel: Teacher

mlcfa <- xxmSubmodel(model = mlcfa, level = "teacher", parents = , ys
= , xs = , etas = c("Eta_y_Tea"), data = mlcfa.teacher)

## Student within matrices (lambda, psi, theta and nu)

mlcfa <- xxmWithinMatrix(model = mlcfa, level = "student", "lambda",
pattern = ly11_pat, value = ly11_val,)

mlcfa <- xxmWithinMatrix(model = mlcfa, level = "student", "psi",
pattern = ps1_pat, value = ps1_val,)

mlcfa <- xxmWithinMatrix(model = mlcfa, level = "student", "theta",
pattern = th1_pat, value = th1_val,)

mlcfa <- xxmWithinMatrix(model = mlcfa, level = "student", "nu",
pattern = nu1_pat, value = nu1_val,)

mlcfa <- xxmWithinMatrix(model = mlcfa, level = "teacher", type =
"psi", pattern = ps2_pat, value = ps2_val,)

## Teacher->Student lambda matrix

mlcfa <- xxmBetweenMatrix(model = mlcfa, parent = "teacher", child =
"student", type = "lambda", pattern = ly12_pat, value = ly12_val,)

mlcfa <- xxmRun(mlcfa)

mlcfa <- xxmCI(mlcfa)

summary <- xxmSummary(mlcfa)

summary

mlcfa <- xxmFree(mlcfa)
```

6.4 RESULTS



Mean structure is not illustrated in this diagram. The student ν matrix was estimated as:

$$\nu^1 = \begin{bmatrix} \nu_{LCE} \\ \nu_{PCE} \\ \nu_{PVE} \\ \nu_{VAE} \end{bmatrix} = \begin{bmatrix} 46.073 \\ 47.057 \\ 46.477 \\ 48.006 \end{bmatrix}.$$

7 TWO LEVEL CONFIRMATORY FACTOR ANALYSIS WITH A RANDOM SLOPE

The present example combines aspects of the models in chapter 3 (random slopes model) and chapter 7 (two-level confirmatory factor analysis).

7.1 MOTIVATING EXAMPLE

Data were simulated for a population model with five observed variables at level-1, four of which serve as indicators for latent factors at levels 1 and 2. The level-1 latent construct is regressed on the remaining observed variable and this coefficient is allowed to vary randomly across level-2 units (i.e. random slope).

7.2 CONDITIONAL TWO-LEVEL CFA WITH A RANDOM SLOPE

As in the previous example, each observed indicator ($y1 - y4$) varies across levels 1 and 2, and these indicators are correlated within each level. Estimating a common latent factor at each level of analysis allows us to parsimoniously explain the common variance/covariance among the indicators. Moreover, the latent factor at level-1 (η_1^1) is regressed on the exogenous observed predictor ($x1$). This effect varies randomly at level-2, leading to two latent variables at level-2, corresponding to the correlated intercept (η_1^2) and slope (η_2^2) factors.

7.2.1 SCALAR REPRESENTATION

7.2.1.1 LEVEL-1 MODEL

The measurement model for each level-1 indicator can be presented as:

$$y_{pi}^1 = \nu_p^1 + \lambda_{p,1}^{1,1} \times \eta_{1i}^1 + e_{pi}^1$$

where, y_{pi} is p^{th} observed indicator for observation i.

$$\eta_{1i}^1 = \beta_{1,1}^{1,2} \times \eta_{1j}^2 + \beta_{1,2}^{1,2} \times \eta_{2j}^2 + \xi_{1i}^1$$

$$\xi_{1i}^1 \sim N\left(0, \psi_{1,1}^{1,1}\right)$$

$$e_i^1 \sim N(0, \Theta^{1,1})$$

The level 1 model hypothesizes following parameters:

1. $(p-1)$ factor loadings $(\lambda_{p,1}^{1,1})$. The first factor-loading is fixed to 1.0 for scale identification.
2. Residual variance for each of the p observed indicators ($\theta_{p,p}^{1,1}$).
3. Latent-on-Latent regression coefficients ($\beta_{1,1}^{1,2}$ & $\beta_{1,2}^{1,2}$) linking the level-2 random intercept (η_{1j}^2) and slope (η_{2j}^2) to the level-1 latent factor (η_{1ij}^1).

4. A single latent residual variance ($\psi_{1,1}^{1,1}$). Note that η_{1ij}^{1} is influenced by η_{1j}^{2} and η_{2j}^{2} at level 2.
5. Measurement intercepts for each of the p observed indicators (ν_p^1).

7.2.1.2 LEVEL-2 MODEL

The teacher level has a two latent variables corresponding to the random intercept (η_1^2) and slope (η_2^2):

$\eta^2 \sim N(\alpha^2, \Psi^{2,2})$

7.2.2 XXM MODEL MATRICES

Again, matrix equations for generic observations within each level are succinct and clear. For this model, there is a single measurement model, at level -1. In addition, there is an across-level regression involving levels 1 and 2. These two sets of equations can be succinctly described by the following two equations:

$$y_i^1 = \nu^1 + \Lambda_{i,i}^{1,1} \times \eta_i^1 + e_i^1$$

$$\eta_i^1 = B_{i,j}^{1,2} \times \eta_j^2 + \xi_i^1$$

The within and across-level matrices in the above two equations as well as the covariances among the residuals is presented next.

7.2.2.1 LEVEL-1

7.2.2.1.1 FACTOR-LOADING MATRIX (LAMBDA)

$$\Lambda_{\text{pattern}}^{1,1} = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \end{bmatrix},$$

$$\Lambda_{\text{value}}^{1,1} = \begin{bmatrix} 1.0 \\ 1.1 \\ 0.9 \\ 0.8 \end{bmatrix}.$$

First factor loading is fixed to 1.0. Hence, we need to fix the first parameter in the pattern matrix. The value at which it is being fixed to is specified in the value matrix. In this case the first factor-loading is being fixed to a value of 1.0.

7.2.2.1.2 OBSERVED RESIDUAL COVARIANCE MATRIX (THETA)

$$\Theta_{\text{pattern}}^{1,1} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$\Theta^{1,1}_{\text{value}} = \begin{bmatrix} 1.1 & 0.0 & 0.0 & 0.0 \\ 0.0 & 2.1 & 0.0 & 0.0 \\ 0.0 & 0.0 & 1.3 & 0.0 \\ 0.0 & 0.0 & 0.0 & 1.5 \end{bmatrix}$$

Residual covariance matrix is a diagonal matrix, meaning we are only estimating residual variances. Residual covariances are all fixed to 0. Again we use a pattern and a value matrix to "fix" all off-diagonal elements to 0.0.

7.2.2.1.3 LATENT RESIDUAL COVARIANCE MATRIX (PSI)

$$\Psi^{1,1}_{\text{pattern}} = [1],$$

$$\Psi^{1,1}_{\text{value}} = [1.1]$$

7.2.2.1.4 OBSERVED VARIABLE INTERCEPTS (NU)

$$\nu^{1}_{\text{pattern}} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix},$$

$$\nu^{1}_{\text{value}} = \begin{bmatrix} 1.1 \\ 2.1 \\ 1.3 \\ 0.7 \end{bmatrix}.$$

7.2.2.1.5 LATENT ON LATENT REGRESSION COEFFICIENT MATRIX (BETA)

$$\mathrm{B}^{1,2}_{pattern} = [0 \quad 1],$$

$$\mathrm{B}^{1,2}_{value} = [1.0 \quad 0.0]$$

The level-1 latent variable is regressed on the intercept and slope factors at level-2. Estimating a random slope for the level-1 predictor X requires that we specify a label matrix $\mathrm{B}^{1,2}_{label}$ that tells xxM to use subject-specific values for the regression of η^1_1 on X.

$$\mathrm{B}^{1,2}_{label} = ["dog" \quad "l1.X"]$$

The first element of the label matrix (1, 1) corresponds to the regression of η^1_1 on η^2_1, which identifies the level-1 random intercept for the latent factor, and the second element (1, 2) corresponds to the regression of η^1_1 on η^2_2, which identifies the random slope of η^1_1 on X. The label provided in the first element is arbitrary, and could have just as well been *fido*, *cat*, or any other character string. In contrast, the label provided for the second element (corresponding to the random slope) is *not arbitrary*, as it must indicate the name of the lower-level predictor dataset and variable in the format: *datasetName.variableName*. The presence of a period (.) in the label matrix causes xxM to search the specified dataset for the predictor variable and insert observation-specific values for the factor-loading. This specification identifies the random slope factor η_2^2, which allows the regression of η_1^1 on X to vary across level-2 subjects. In the current example, the value $\beta_{1,2}^{1,2}$ = "$l1.X$" tells xxM to estimate a random slope for variable X found in dataset $l1$.

7.2.2.2 LEVEL-2

7.2.2.2.1 LATENT COVARIANCE MATRIX (PSI)

$$\Psi_{\text{pattern}}^{2,2} = \begin{bmatrix} 1 & \\ 1 & 1 \end{bmatrix},$$

$$\Psi_{\text{value}}^{2,2} = \begin{bmatrix} 0.1 & \\ 0.0 & 0.1 \end{bmatrix}$$

There are only latent variables at level-2, corresponding to the intercept and slope factors.

7.2.2.2.2 LATENT MEAN MATRIX

$$\alpha_{\text{pattern}}^{2} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$\alpha_{\text{value}}^{2} = \begin{bmatrix} 0.0 \\ 0.5 \end{bmatrix}$$

Mean structure for the random intercept is modeled at level-1, therefore the level-2 intercept for level-1 latent variable (α_1^2) is not identified and must be fixed to 0.0. α_2^2 represents the mean of the slope factor, or the fixed regression coefficient for X.

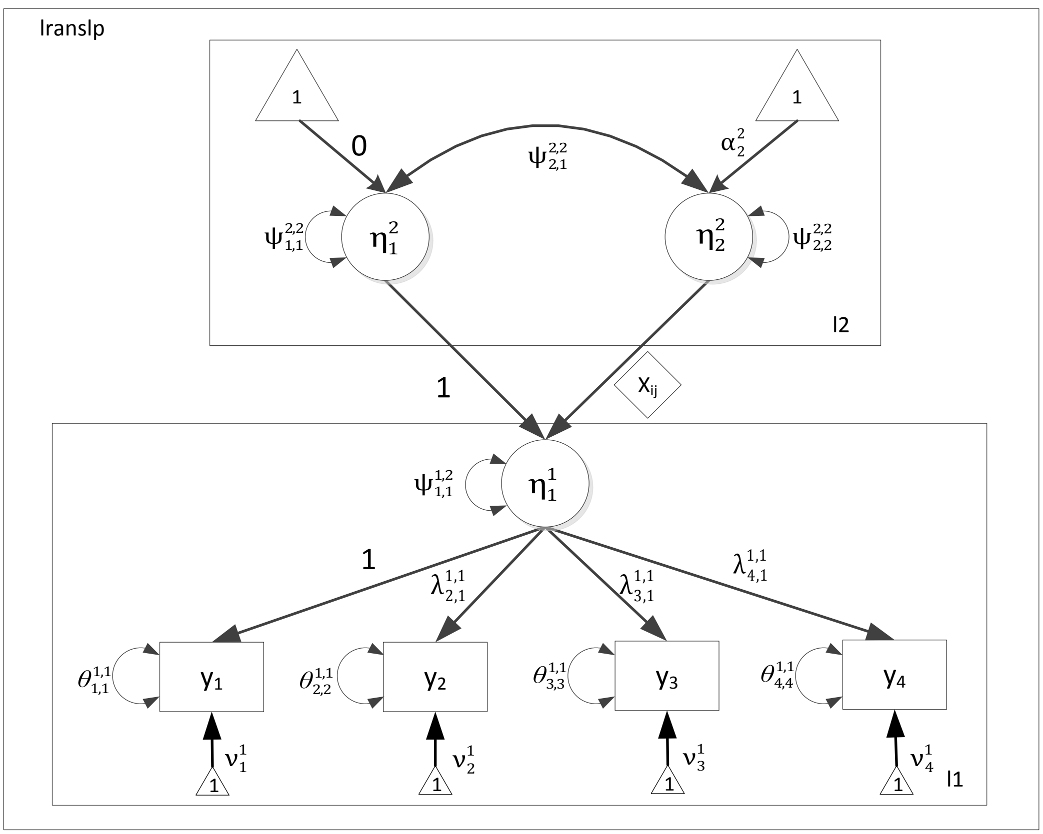
7.2.3 MODEL MATRICES SUMMARY

The following table provides a complete summary of parameter matrices:

	Matrix	Pattern
Level 1: Θ	$\Theta^{1,1} = \begin{bmatrix} \theta_{1,1}^{1,1} & & & \\ \theta_{2,1}^{1,1} & \theta_{2,2}^{1,1} & & \\ \theta_{3,1}^{1,1} & \theta_{3,2}^{1,1} & \theta_{3,3}^{1,1} & \\ \theta_{4,1}^{1,1} & \theta_{4,2}^{1,1} & \theta_{4,3}^{1,1} & \theta_{4,4}^{1,1} \end{bmatrix}$	$\Theta^{1,1} = \begin{bmatrix} 1 & & & \\ 0 & 1 & & \\ 0 & 0 & 1 & \\ 0 & 0 & 0 & 1 \end{bmatrix}$
Level 1: ν	$\nu^1 = \begin{bmatrix} \nu_1^1 \\ \nu_2^1 \\ \nu_3^1 \\ \nu_4^1 \end{bmatrix}$	$\nu^1 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$

Level 1: Λ	$\Lambda^{1,1} = \begin{bmatrix} \lambda^{1,1}_{1,1} \\ \lambda^{1,1}_{2,1} \\ \lambda^{1,1}_{3,1} \\ \lambda^{1,1}_{4,1} \end{bmatrix}$	$\Lambda^{1,1} = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \end{bmatrix}$
Level 1: Ψ	$\Psi^{1,1} = [\psi^{1,1}_{1,1}]$	$\Psi^{1,1} = [1]$
Level 2 → Level 1: B	$B^{1,2} = [\beta^{1,2}_{1,1} \quad \beta^{1,2}_{1,2}]$	$B^{1,2} = [1 \quad 1]$
Level 2: Ψ	$\Psi^{2,2} = \begin{bmatrix} \psi^{2,2}_{1,1} & \\ \psi^{2,2}_{2,1} & \psi^{2,2}_{2,2} \end{bmatrix}$	$\Psi^{2,2} = \begin{bmatrix} 1 & \\ 1 & 1 \end{bmatrix}$
Level 2: α	$\alpha^2 = \begin{bmatrix} \alpha^2_1 \\ \alpha^2_2 \end{bmatrix}$	$\alpha^2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$

7.2.4 PATH DIAGRAM

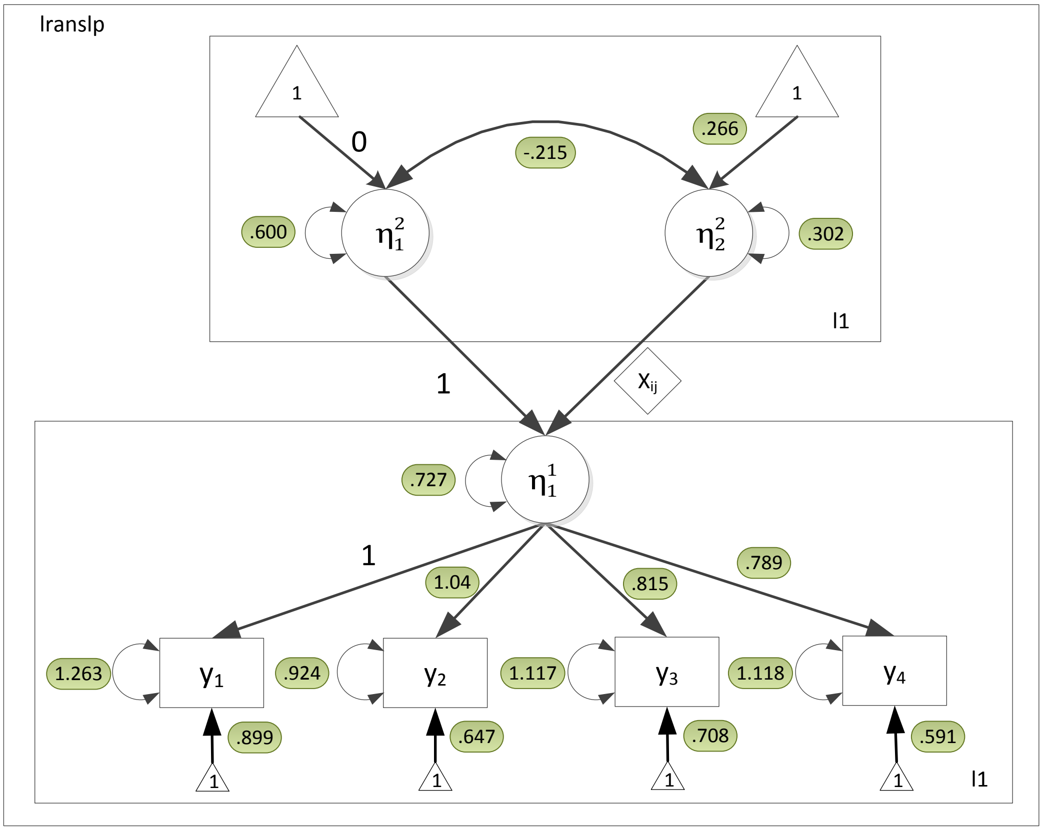


7.3 CODE LISTING

7.3.1 XXM

```
library(xxm)
data(lranslp.xxm)
#l1: Factor-Loading Matrix
ly1_pat <- matrix(c(0,1,1,1),4,1)
ly1_val <-  matrix(c(1,1,1,1),4,1)
#l1:latent variable-covariance matrix
ps1_pat <- matrix(1,1,1)
ps1_val <- matrix(1,1,1)
#l1:observed residual variable-covariance matrix
th1_pat <- matrix(c(1,0,0,0,
                    0,1,0,0,
                    0,0,1,0,
                    0,0,0,1),4,4,byrow=TRUE)
th1_val <- matrix(c(1,0,0,0,
                    0,1,0,0,
                    0,0,1,0,
                    0,0,0,1),4,4,byrow=TRUE)
#l1: observed variable intercepts
nu1_pat <- matrix(c(1,1,1,1),4,1)
nu1_val <- matrix(c(.9,.7,.7,.6),4,1)
#l2: latent variable covariance matrix
ps2_pat <- matrix(1,2,2)
ps2_val <- matrix(c(.2,.01,.01,.2),2,2)
#l2: intercepts for level 2 slope variables
al2_pat <- matrix(c(0,1),2,1)
al2_val <- matrix(c(0,.2),2,1)
#l2 -> l1 factor loading matrix
be12_pat <- matrix(c(0,0),1,2)
```

```
be12_val <- matrix(c(1,0),1,2)

be12_label  <- matrix(c("one","l1.x"), 1, 2)

lranslp <- xxmModel(levels = c("l1","l2"))

### Submodel: l1

lranslp <- xxmSubmodel(model = lranslp, level = "l1", parents = "l2",
ys = c("y1","y2","y3","y4"), xs = "x", etas = "fw", data = l1)

### Submodel: l2

lranslp <- xxmSubmodel(model = lranslp, level = "l2", parents = , ys =
, xs =, etas = c("int","slp"), data = l2)

#l1 within matrices (lambda, psi, theta)

lranslp <- xxmWithinMatrix(model = lranslp, level = "l1", type =
"lambda", pattern = ly1_pat, value = ly1_val)

lranslp <- xxmWithinMatrix(model = lranslp, level = "l1", type =
"psi", pattern = ps1_pat, value = ps1_val)

lranslp <- xxmWithinMatrix(model = lranslp, level = "l1", type =
"theta", pattern = th1_pat, value = th1_val)

lranslp <- xxmWithinMatrix(model = lranslp, level = "l1", type = "nu",
pattern = nu1_pat, val = nu1_val)

#l2 within matrices (psi, alpha)

lranslp <- xxmWithinMatrix(model = lranslp, level = "l2", type =
"psi", pattern = ps2_pat, value = ps2_val)

lranslp <- xxmWithinMatrix(model = lranslp,level = "l2", type =
"alpha", pattern = al2_pat, value = al2_val)

##l2->l1 loading matrix (beta)

lranslp <- xxmBetweenMatrix(model = lranslp, parent = "l2", child =
"l1", type = "beta", pattern = be12_pat, value = be12_val, label
=be12_label)

lranslp <- xxmRun(lranslp)

lranslp <- xxmCI(lranslp)

summary <- xxmSummary(lranslp)

summary

lranslp <- xxmFree(lranslp)
```


lranslp
1
1
0
.266
-.215
.600
η_1^2
η_2^2
.302
l1
1
x_{ij}
.727
η_1^1
1
.789
1.04
.815
1.263
y_1
.924
y_2
1.117
y_3
1.118
y_4
.899
.647
.708
.591
1
1
1
1
l1

8 THREE LEVEL HIERARCHICAL MODEL WITH OBSERVED AND LATENT VARIABLES AT MULTIPLE LEVELS

We now consider a general xxM model for three level data with observed and latent variables at multiple levels.

8.1 MOTIVATING EXAMPLE

We may have multiple indicators of student reading achievement, teacher quality, and school resources. In this case, students are nested within teachers and teachers are nested within schools. We are interested in examining the effects of latent teacher quality and school resources on latent student achievement.

In this case, we have four observed indicators of student achievement at level-1, no observed variables at the teacher level, and three measures of school resources at level-3.

8.2 THREE LEVEL RANDOM INTERCEPTS MODEL WITH LATENT REGRESSION

Very simply, these models are complex. It is easier to visualize these models than describe them in terms of equations or matrices. One thing to keep in mind that xxM is intended to be flexible so as to allow a model to be specified in the most 'natural' fashion. The following model can be specified in several different equivalent ways. The simplest expression of the model is presented here.

8.2.1 SCALAR REPRESENTATION

8.2.1.1 STUDENT SUBMODEL (LEVEL-1)

The measurement model for the student achievement can be presented as:

$$y^1_{pi} = \nu^1_p + \lambda^{1,1}_{p,1} \times \eta^1_{1i} + e^1_{pi}$$

where, y_{pi} is p^{th} observed indicator for student i, nested within teacher j . The superscript 1 is for the student level. Conditional on level 2 teacher latent variable, the residual of level-1 latent variable is distributed normally:

$$\eta^1_{1i} \mid \eta^2_{1j} \sim N\left(0, \psi^{1,1}_{1,1}\right)$$

$$e^1_{pi} \sim N\left(0, \theta^{1,1}_{p,p}\right)$$

The student model hypothesizes the following parameters:

3. $(p-1)$ factor loadings ($\lambda^{1,1}_{p,1}$) with the first factor loading being fixed to 1.0 for scale identification.
4. Residual variance for each of the p observed indicators($\theta^{1,1}_{p,p}$)

5. Single latent residual variance ($\psi_{1,1}^{1,1}$). Note: The student level latent variable is regressed on teacher level latent factor. As a result, ($\psi_{1,1}^{1,1}$) is the conditional or residual variance. This is discussed later.
6. Measurement intercepts for each of the p observed indicators (ν_p^1).

8.2.1.2 TEACHER SUBMODEL (LEVEL-2)

The teacher level has a single latent variable with zero mean and unknown residual variance ($\psi_{1,1}^{2,2}$). The superscript 2 is for the teacher level. Conditional on level 3 school latent variable, the residual of level-2 or teacher random-effect is distributed normally:

$$\eta_{1,j}^2 \mid \eta_{1,k}^3 \sim N\left(0, \psi_{1,1}^{2,2}\right)$$

8.2.1.3 SCHOOL SUBMODEL (LEVEL-3)

The school level has two latent variables each with a mean of zero and unknown variances.

The *second* school level latent variable is the school-resource factor measured by three school level indicators. School level measurement model for the school-resource factor is:

$$y_{pk}^3 = \nu_p^3 + \lambda_{p,2}^{3,3} \times \eta_{2,k}^3 + e_{pk}^3$$

$$\eta_{2,k}^3 \sim N\left(0, \psi_{2,2}^{3,3}\right)$$

$$e_{pk}^3 \sim N\left(0, \theta_{p,p}^{3,3}\right)$$

At the school level the first latent variable (school level random-intercept of student outcome) is regressed on school level latent resource factor:

$$\eta_{1k}^3 = \beta_{1,2}^{3,3} * \eta_{2k}^3 + \xi_k^3$$

$$\xi_k^3 \sim N\left(0, \psi_{1,1}^{3,3}\right)$$

The structural model states that school level variability in student achievement is predicted by school resources. So far, our description has been limited to within-level models only. Latent variables representing random-intercepts for student achievement at the teacher and school levels were presented, but these have not yet been defined. Clearly, we need to ‘link’ the latent student achievement factor (η_{1i}^1) with the corresponding teacher level intercept (η_{1j}^2). Similarly, we need to connect the school level intercept of student achievement to the student level achievement factor. There are many ways of specifying such links. Here we use a mediated effect approach. The effect of the school level intercept for student achievement on student level achievement is mediated by the teacher effect. In other words, we are envisioning regression among latent variables across levels.

8.2.1.4 TEACHER TO STUDENT EFFECTS

$$\eta_{1i}^{1} = \beta_{1,1}^{1,2} * \eta_{1j}^{2} + \xi_{1,i}^{1}$$

$$\xi_{1,i}^{1} \sim N\left(0, \psi_{1,1}^{1,1}\right)$$

Note:

1. The dependent variable is a level-1 latent variable (student achievement). The independent variable is a level-2 latent variable (teacher intercept of latent student achievement). This is reflected in the respective superscripts.
2. The superscript for the regression coefficient ($\beta_{1,1}^{1,2}$) indicates that the dependent variable is a level-1 variable and the independent variable is a level-2 variable.
3. The subscript for the regression coefficient is (1,1) meaning the first latent variable at level-1 is being regressed on the first latent variable at level-2. With single latent variables at both levels, this seems like overkill. However, with multiple variables, superscripts and subscripts become a necessary evil.
4. We return to the level-1 variance for the student achievement factor ($\psi_{1,1}^{1,1}$). This was incompletely specified in the student submodel.

8.2.1.5 SCHOOL TO TEACHER EFFECTS

$$\eta_{1j}^{2} = \beta_{1,1}^{2,3} \times \eta_{1k}^{3} + \xi_{1,j}^{2}$$

$$\xi_{1,j}^{2} \sim N\left(0, \psi_{1,1}^{2,2}\right)$$

Note:

1. The dependent variable is a level-2 latent variable (teacher intercept of latent student achievement). The independent variable is a level-3 latent variable (school intercept of latent student achievement). This is reflected in the respective superscripts.
2. The superscript for the regression coefficient ($\beta_{1,1}^{2,3}$) indicates that the dependent variable is a level-2 variable and the independent variable is a level-3 variable.
3. The subscript for the regression coefficient is (1, 1) meaning the first latent variable at level-2 is being regressed on the first latent variable at level-3. In this case, we have two latent variables at level-3. Hence, we could in principle have two latent regressions coefficients ($\beta_{1,1}^{2,3}$ & $\beta_{1,2}^{2,3}$). Subscripts make it clear which latent variables are involved.
4. We return to the level-2 variance for the teacher achievement intercept ($\psi_{1,1}^{2,2}$). This was incompletely specified in the teacher sub-model.

8.2.2 XXM MODEL MATRICES

For the current model, two levels include measurement models, (a) student, and (b) teacher. There are two across-level regressions: (a) Teacher to Student, and (b) School to Teacher. The following four matrix equations, succinctly represents all of these within- and across-levels effects.

8.2.2.1 STUDENT LEVEL MEASUREMENT EQUATION

$$y_i^1 = \nu^1 + \Lambda_{i,i}^{1,1} \times \eta_i^1 + e_i^1$$

8.2.2.1 SCHOOL LEVEL MEASUREMENT EQUATION

$$y_i^3 = \nu^3 + \Lambda_{i,i}^{3,3} \times \eta_i^3 + e_i^3$$

8.2.2.1 TEACHER -> STUDENT EQUATION

$$\eta_i^1 = B_{i,j}^{1,2} \times \eta_j^2 + \xi_i^1$$

8.2.2.1 SCHOOL-> TEACHER EQUATION

$$\eta_i^2 = B_{i,j}^{2,3} \times \eta_j^3 + \xi_i^2$$

The matrices used in these equations and the covariances among the residuals at each level are now described to complete the model specification.

8.2.2.2 STUDENT SUBMODEL (LEVEL-1)

8.2.2.2.1 FACTOR-LOADING MATRIX (LAMBDA)

$$\Lambda_{\text{pattern}}^{1,1} = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \end{bmatrix}, \Lambda_{\text{value}}^{1,1} = \begin{bmatrix} 1.0 \\ 1.1 \\ 0.9 \\ 0.8 \end{bmatrix},$$

The first factor-loading is fixed to 1.0. Hence, we need to fix the first parameter in the pattern matrix. The actual value at which the parameter is to be fixed is specified in the value matrix. In this case the first factor-loading is being fixed to a value of 1.0.

8.2.2.2.2 OBSERVED RESIDUAL COVARIANCE MATRIX (THETA)

$$\Theta_{\text{pattern}}^{1,1} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \Theta_{\text{value}}^{1,1} = \begin{bmatrix} 1.1 & 0 & 0 & 0 \\ 0 & 2.1 & 0 & 0 \\ 0 & 0 & 1.3 & 0 \\ 0 & 0 & 0 & 1.5 \end{bmatrix}$$

Residual covariance matrix is a diagonal matrix, meaning we are only estimating residual variances. Residual covariances are all fixed to 0.0. Again we use a pattern and a value matrix to fix all off-diagonal elements to 0.0.

8.2.2.2.3 LATENT (RESIDUAL) COVARIANCE MATRIX (PSI)

$\Psi^{1,1}_{\text{pattern}} = [1], \Psi^{1,1}_{\text{value}} = [1.1]$

8.2.2.2.4 OBSERVED VARIABLE INTERCEPTS (NU)

$$\nu^{1}_{\text{pattern}} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}, \nu^{1}_{\text{value}} = \begin{bmatrix} 1.1 \\ 2.1 \\ 1.3 \\ 0.71 \end{bmatrix}.$$

8.2.2.3 TEACHER SUBMODEL (LEVEL-2)

8.2.2.3.1 LATENT (RESIDUAL) COVARIANCE MATRIX (PSI)

$\Psi^{2,2}_{\text{pattern}} = [1], \Psi^{2,2}_{\text{value}} = [0.05]$

8.2.2.4 SCHOOL SUBMODEL (LEVEL-3)

8.2.2.4.1 FACTOR-LOADING MATRIX (LAMBDA)

$$\Lambda^{3,3}_{\text{pattern}} = \begin{bmatrix} 0 & 0 \\ 0 & 1 \\ 0 & 1 \end{bmatrix}, \Lambda^{3,3}_{\text{value}} = \begin{bmatrix} 0.0 & 1.0 \\ 0.0 & 1.1 \\ 0.0 & 0.9 \end{bmatrix}$$

There are three observed and two latent variables at level-3. Hence the factor-loading matrix is 3×2. The first latent variable is the school level random-intercept of the teacher-level random-intercept of student achievement. Clearly, the first latent variable *cannot* have school-level latent indicators. Hence, the first column is zero in both pattern and value matrices. The second latent variable is the school-resource factor measured by all three level-3 indicators. As always, the first factor loading is fixed to 1.0 to identify the latent measurement scale.

8.2.2.4.2 OBSERVED RESIDUAL COVARIANCE MATRIX (THETA)

$$\Theta^{3,3}_{\text{pattern}} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \Theta^{3,3}_{\text{value}} = \begin{bmatrix} 1.1 & 0.0 & 0.0 \\ 0.0 & 2.1 & 0.0 \\ 0.0 & 0.0 & 1.3 \end{bmatrix}$$

8.2.2.4.3 OBSERVED VARIABLE INTERCEPTS (NU)

$$\nu^3_{\text{pattern}} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}, \nu^3_{\text{value}} = \begin{bmatrix} 1.1 \\ 2.1 \\ 0.7 \end{bmatrix}.$$

8.2.2.4.4 LATENT VARIABLE REGRESSION MATRIX (BETA)

$$\mathrm{B}^{3,3}_{pattern} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \mathrm{B}^{3,3}_{value} = \begin{bmatrix} 0.0 & 0.4 \\ 0.0 & 0.0 \end{bmatrix}.$$

There are two latent variables at level-3 and the first latent variable is regressed on the second. Hence, element is freely estimated. The other three elements are fixed to zero.

8.2.2.4.5 LATENT VARIABLE (RESIDUAL) COVARIANCE MATRIX (PSI)

$$\Psi^{3,3}_{pattern} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \Psi^{3,3}_{value} = \begin{bmatrix} 0.3 & 0.0 \\ 0.0 & 0.7 \end{bmatrix}.$$

Like the theta matrix, the psi matrix is a diagonal matrix. $\psi^{3,3}_{1,1}$ is the variance of the first latent variable (student achievement random intercept) and represents the variance in the intercept factor *unexplained* by the school-resource factor. $\psi^{3,3}_{1,1}$ is the unconditional variance of the school-resource factor.

8.2.2.5 TEACHER TO STUDENT EFFECTS

8.2.2.5.1 LATENT VARIABLE REGRESSION MATRIX (BETA)

$$\mathrm{B}^{1,2}_{pattern} = [0], \mathrm{B}^{1,2}_{value} = [1.0]$$

This matrix links the teacher latent random-intercept variable with the student latent achievement variable. As indicated earlier, the value is fixed to 1.0. Note that the superscript has two elements, the first element refers to the lower level (student) and the second element refers to the higher level (teacher). This is always true for all linking matrices.

8.2.2.6 SCHOOL TO TEACHEREFFECTS

8.2.2.6.1 LATENT VARIABLE REGRESSION MATRIX (BETA)

$$\mathrm{B}^{2,3}_{pattern} = [0 \quad 0], \mathrm{B}^{2,3}_{value} = [1.0 \quad 0.0]$$

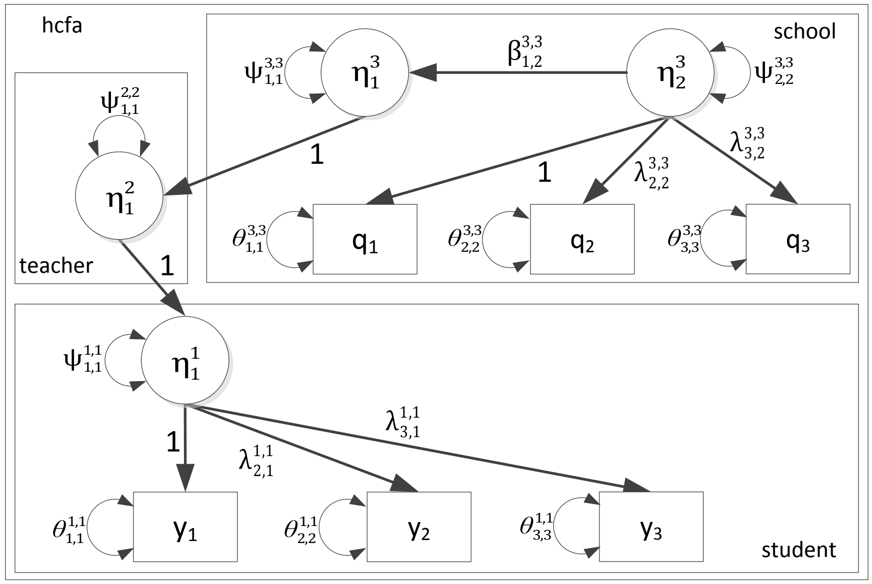
This matrix links the school latent random-intercept of student achievement with the teacher latent intercept of achievement variable. There is a single latent variable at level-2 (teacher random-intercept of student achievement), but two latent variables at level-3 (school random-intercept of student achievement and school-resources). Only the school random-intercept of student achievement influences the teacher random intercept of student achievement. Hence, the first element is fixed to 1.0 and second element is fixed to 0.0.

8.2.3 MODEL MATRICES SUMMARY

The following table provides a complete summary of parameter matrices:

	Matrix	Pattern
Level 1: Θ	$\Theta^{1,1} = \begin{bmatrix} \theta_{1,1}^{1,1} & & \\ \theta_{2,1}^{1,1} & \theta_{22}^{1,1} & \\ \theta_{3,1}^{1,1} & \theta_{3,2}^{1,1} & \theta_{3,3}^{1,1} \end{bmatrix}$	$\Theta^{1,1} = \begin{bmatrix} 1 & & \\ 0 & 1 & \\ 0 & 0 & 1 \end{bmatrix}$
Level 1: ν	$\nu^1 = \begin{bmatrix} \nu_1^1 \\ \nu_2^1 \\ \nu_3^1 \end{bmatrix}$	$\nu^1 = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$
Level 1: Λ	$\Lambda^{1,1} = \begin{bmatrix} \lambda_{1,1}^{1,1} \\ \lambda_{2,1}^{1,1} \\ \lambda_{3,1}^{1,1} \end{bmatrix}$	$\Lambda^{1,1} = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}$
Level 1: Ψ	$\Psi^{2,2} = [\psi_{1,1}^{2,2}]$	$\Psi^{2,2} = [1]$
Level 2 → Level 1: B	$\mathrm{B}^{1,2} = [\beta_{1,1}^{1,2}]$	$\mathrm{B}^{1,2} = [0]$.
Level 2: Ψ	$\Psi^{2,2} = [\psi_{1,1}^{2,2}]$	$\Psi^{2,2} = [1]$
Level 3: Θ	$\Theta^{3,3} = \begin{bmatrix} \theta_{1,1}^{3,3} & & \\ \theta_{2,1}^{3,3} & \theta_{2,2}^{3,3} & \\ \theta_{3,1}^{3,3} & \theta_{3,2}^{3,3} & \theta_{3,3}^{3,3} \end{bmatrix}$	$\Theta^{3,3} = \begin{bmatrix} 1 & & \\ 0 & 1 & \\ 0 & 0 & 1 \end{bmatrix}$
Level 3: ν	$\nu^3 = \begin{bmatrix} \nu_1^3 \\ \nu_2^3 \\ \nu_3^3 \end{bmatrix}$	$\nu^3 = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$
Level 3 → Level 2: B	$\mathrm{B}^{2,3} = [\beta_{1,1}^{2,3}]$	$\mathrm{B}^{2,3} = [0]$
Level 3: Λ	$\Lambda^{3,3} = \begin{bmatrix} \lambda_{1,1}^{3,3} \\ \lambda_{2,1}^{3,3} \\ \lambda_{3,1}^{3,3} \end{bmatrix}$	$\Lambda^{3,3} = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}$
Level 3: B	$\mathrm{B}^{3,3} = \begin{bmatrix} \beta_{1,1}^{3,3} & \\ \beta_{2,1}^{3,3} & \beta_{2,2}^{3,3} \end{bmatrix}$	$\mathrm{B}^{3,3} = \begin{bmatrix} 0 & \\ 1 & 0 \end{bmatrix}$
Level 3: Ψ	$\Psi^{3,3} = \begin{bmatrix} \psi_{1,1}^{3,3} & \\ \psi_{2,1}^{3,3} & \psi_{2,2}^{3,3} \end{bmatrix}$	$\Psi^{3,3} = \begin{bmatrix} 1 & \\ 0 & 1 \end{bmatrix}$

8.2.4 PATH DIAGRAM



8.3 CODE LISTING

This model cannot be estimated using currently available SEM software.

8.3.1 XXM

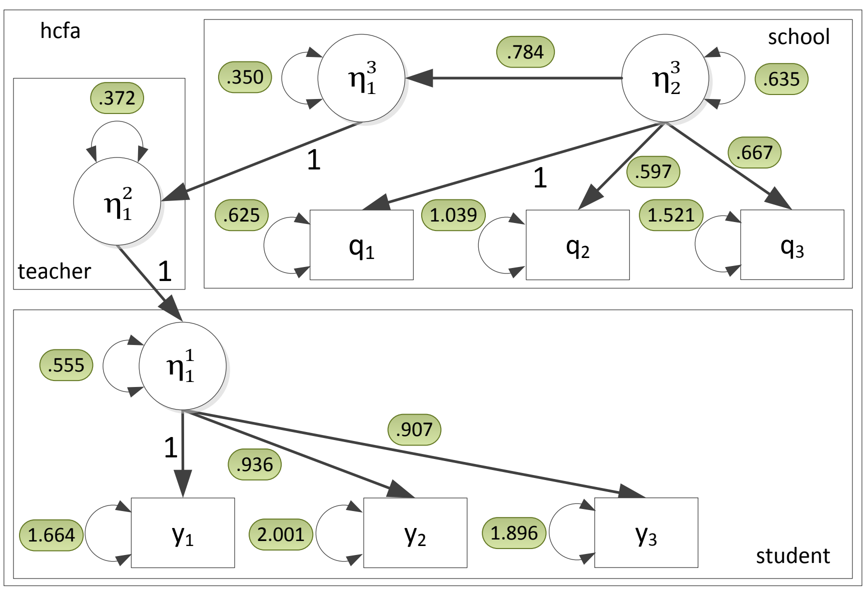
```
library(xxm)
data(hcfa.xxm)


#Student: factor-loading matrix
ly1_pat <- matrix(c(0,1,1),3,1)
ly1_val <- matrix(c(1,1.1,.9),3,1)
ly1_lab <- matrix(c("ly1","ly2","ly3"),3,1)
#Student: factor-covariance matrix
ps1_pat <- matrix(1,1,1)
ps1_val <- matrix(.498,1,1)
#Student: observed residual-covariance matrix
th1_pat <- diag(1,3)
th1_val <- diag(c(2.727,2.990,2.854),3)
```

```
#Student: "grand-means" NU
nu1_pat <- matrix(1,3,1)
nu1_val <- matrix(c(.526,.571,.592),3,1)
#Teacher model matrices
#Teacher: factor-covariance matrix
ps2_pat <- matrix(1,1,1)
ps2_val <- matrix(.1333 ,1,1)
# School model matrices
#School: factor-loading matrix
ly3_pat <- matrix(c(0,0,0, 0,1,1),3,2)
ly3_val <- matrix(c(0,0,0, 1,.9,1.1),3,2)
#School: factor-covariance matrix
ps3_pat <- matrix(c(1,0,0,1),2,2)
ps3_val <- matrix(c(.1335,0,0,.1402),2,2)
#School: observed residual-covariance matrix
th3_pat <- diag(1,3)
th3_val <- diag(c(1.787,1.937,2.418),3)
#School: "grand-means/intercepts" NU
nu3_pat <- matrix(1,3,1)
nu3_val <- matrix(c(.129,.144,.081),3,1)
#School: "Latent Factor Regression" Beta
be3_pat <- matrix(c(0,0,1,0),2,2)
be3_val <- matrix(c(0,0,.3,0),2,2)
#Teacher -> Student matrices
be12_pat <- matrix(0,1,1)
be12_val <- matrix(1,1,1)
#School - > teacher matrices
be13_pat <- matrix(0,1,2)
```

```
be13_val <- matrix(c(1,0),1,2)




hcfa <- xxmModel(levels = c("student", "teacher", "school"))

hcfa <- xxmSubmodel(model = hcfa, level = "student", parents =
c("teacher"), ys = c("y1","y2","y3"), xs = ,etas = c("Eta_y_Stu"),
data = hcfa.student)

hcfa <- xxmSubmodel(model = hcfa, level = "teacher", parents =
c("school"), ys = , xs = , etas = c("Eta_y_Tea"), data = hcfa.teacher)

hcfa <- xxmSubmodel(model = hcfa, level = "school", parents = , ys =
c("q1", "q2", "q3"), xs = , etas = c("Eta_y_Sch", "Eta_q_Sch"), data =
hcfa.school)

## Student within matrices (lambda, psi, theta and nu)

hcfa <- xxmWithinMatrix(model = hcfa, level = "student", "lambda",
pattern = ly1_pat, value = ly1_val,)

hcfa <- xxmWithinMatrix(model = hcfa, level = "student", "psi",
pattern = ps1_pat, value = ps1_val,)

hcfa <- xxmWithinMatrix(model = hcfa, level = "student", "theta",
pattern = th1_pat, value = th1_val,)

hcfa <- xxmWithinMatrix(model = hcfa, level = "student", "nu", pattern
= nu1_pat, value = nu1_val,)

## Teacher within matrices (psi)

hcfa <- xxmWithinMatrix(model = hcfa, level = "teacher", type = "psi",
pattern = ps2_pat, value = ps2_val,)

## School within matrices (psi)

hcfa <- xxmWithinMatrix(model = hcfa, level = "school", type =
"lambda", pattern = ly3_pat, value = ly3_val,)

hcfa <- xxmWithinMatrix(model = hcfa, level = "school", type = "psi",
pattern = ps3_pat, value = ps3_val,)

hcfa <- xxmWithinMatrix(model = hcfa, level = "school", type =
"theta", pattern = th3_pat, value = th3_val,)

hcfa <- xxmWithinMatrix(model = hcfa, level = "school", type = "nu",
pattern = nu3_pat, value = nu3_val,)

hcfa <- xxmWithinMatrix(model = hcfa, level = "school", type = "beta",
pattern = be3_pat, value = be3_val,)
```

```
## Teacher->Student beta matrix

hcfa <- xxmBetweenMatrix(model = hcfa, parent = "teacher", child =
"student", type = "beta", pattern = be12_pat, value = be12_val,)

## School-> Teacher beta matrix

hcfa <- xxmBetweenMatrix(model = hcfa, parent = "school", child =
"teacher", type = "beta", pattern = be13_pat, value = be13_val,)

hcfa <- xxmRun(hcfa)

hcfa <- xxmCI(hcfa)

summary <- xxmSummary(hcfa)

summary

hcfa <- xxmFree(hcfa)
```

8.4 RESULTS



Mean structure is not illustrated in this diagram. The student and school ν matrices were estimated as:

$$\nu^1 = \begin{bmatrix} \nu_1^1 \\ \nu_2^1 \\ \nu_3^1 \end{bmatrix} = \begin{bmatrix} .526 \\ .571 \\ .592 \end{bmatrix}, \ \nu^3 = \begin{bmatrix} \nu_1^3 \\ \nu_2^3 \\ \nu_3^3 \end{bmatrix} = \begin{bmatrix} .129 \\ .144 \\ .081 \end{bmatrix}$$