ASSESS THREE-LEVEL-CROSS-CLASSIFIED MODEL IN LME4 PACKAGE

by

Ashraf F. Demian

A thesis submitted to the Graduate College of
Texas State University in partial fulfillment
of the requirements for the degree of
Master of Science
with a Major in Mathematics
August 2019

Committee Members:

Alexander White, Chair

Qiang Zhao

Shuying Sun

FAIR USE AND AUTHOR'S PERMISSION STATEMENT

Fair Use

Duplication Permission

DEDICATION

I would like to dedicate this work to my mother Soad, whose dreams for me have resulted in this achievement and without her loving upbringing and nurturing; I would not be where I am today.

I would also like to dedicate this work to my family my wife Heather, my daughter Emily and my parents in law, Russell and Becky for motivating and supporting me.

ACKNOWLEDGEMENTS

First and foremost, I would like to thank God for giving me the strength to complete this research. That is without his blessing, this achievement would not be possible. I would also like to acknowledge my thesis advisor Dr. White at Texas State University who was always willing to assist me whenever I needed help. Additionally, he taught me latex which I used to write my thesis with and the package tidyverse, which helped me to simulate the data of the project.

I would also like to thank the committee members Dr. Sun and Dr. Zhao at Texas State University for teaching me regression analysis, statistical methods and experimental design which strongly supported my research. I also appreciate their time to review my work.

I would also like to give a special thanks to my family for the spiritual and financial support.

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

Figure   **Page**

ABSTRACT

The use of the hierarchical linear model is increasingly common in the medical and educational research. In education, the data structure requires more than an ordinary linear model in order to obtain a precise outcome. A leading example is the student's achievements in math while in elementary school. In this example, the test score is nested within both the teacher's and student's characteristics and the student and the teacher are nested within the school's characteristics. Additionally, the student and the teacher are cross-classified. The three-level-cross-classified model explains the student's achievement based on the teacher's effect on a student's characteristics and the school's characteristics. The research investigates the capability of the R package *lme4* to fit the three-level-cross-classified model.

I.   INTRODUCTION

Introduction to Hierarchical Data

Many studies in the social sciences involve nested (hierarchical) data. One typical example from education is students nested in classes. Variables are measured in two levels with the student's characteristics in level one and the teacher's variables in level two. In another example from management, researchers are interested in the characterizations of both workers and firms and how these characteristics affect how business decisions are made (Raundenbush and Bryk, 2002) where the workers' variables are measured in level one and vary within the firms.

The data may be structured in more than two levels. For example, educational studies contain three levels in which the students are nested in classes and the classes are nested in schools. The predictors, in this example, are measured in three levels where the variables for students, teachers and schools are measured in levels one, two and three, respectively. The data structure may be more complicated if two sets of characteristics are included within a single level. For example, if student A lives in neighborhood 20 and attends school 1 which is located in neighborhood 10. Meanwhile, student 2 goes to the same school 1, but lives in neighborhood 20 and student 3 attends school 2 and lives in neighborhood 10. We say that level three of the data are cross-classified where the students vary within both schools and neighborhoods (Raundenbush and Bryk, 2002).

Prior to advent of more powerful computing tools in the 1990s, researchers often ignored the true multilevel structure of the data and used standard multiple regression techniques when analyzing the data. Raundenbush and Bryk (2002) described the two common approaches. The first one is to assign all higher level variables (teacher and school characteristics) to the individual level and run the

model at the individual level (student). Note an observed teacher or school characteristic is repeated for all students in a single class. Since it is not reasonable to assume that student performance in a class is independent from that of other students in the class, these data violate the assumptions of tradition multiple linear regression. This approach ignores the fact that though we have a large sample of students, we have a relatively smaller sample of teachers and even smaller sample of schools. The other strategy is to aggregate the individual level variables to the higher level and do the analysis at the higher level. We use class averages of the student variables in a model run at the class level. The problem here is we lose the within-group information and about 80 - 90 % of the variation before running the model. As a result, the model variables will be strongly correlated yields to inaccurate interpretation.

Using a traditional multivariate linear model requires the four assumptions: linearity, normality, homoscedasticity and independence (Raundenbush and Bryk, 2002). Except in rare cases, hierarchical data do not satisfy the independence assumption. For example, the characteristics of students in the same class are similar unlike in different classes where the students may have different characteristics. Therefore, there is a need for a suitable model that solves the independence issue. One approach to resolving this issue is variance component models (Raundenbush and Bryk, 2002). That is, the individual components are all independent and the group components are independent between groups but strongly correlated within groups.

The Impact of Nesting on Estimation

In order to demonstrate the importance of taking nesting into account, we consider the sample (grand) mean from a simple case of nested data (students within schools) and investigate the impact of the correlation within the group on variance.

2

Effective Sample Size

Suppose we have J schools each with n students for a total of $N = nJ$ where y_{ij} is the test score for student i in school j for $i = 1, \ldots, n$ and $j = 1, \ldots, J$. Furthermore, let y_{ij} have mean μ and variance σ^2 for $i = 1, \ldots, n$, with $cor(y_{ij}, y_{i',j'}) = \rho$ when $j = j'$ and 0 otherwise. Then the typical estimate of the population mean is:

$$\hat{\mu} = \frac{1}{N} \sum_{j=1}^{J} \sum_{i=1}^{n} y_{ij}, \tag{1.1}$$

and

$$\begin{aligned}
\text{Var}(\hat{\mu}) &= \frac{1}{N^2} \text{Var}(\sum_{j=1}^{J} \sum_{i=1}^{n} y_{ij}) \\
&= \frac{1}{N^2} \left(\sum_{j=1}^{J} \sum_{i=1}^{n} \sigma^2 + \sum_{j=1}^{J} n(n-1)\sigma^2 \rho \right)
\end{aligned} \tag{1.2}$$

since

$$\text{Cov}(y_{ij}, y_{ij'}) = 0 \quad j \neq j'$$
$$\text{Cov}(y_{ij}, y_{i'j}) = \sigma^2 \rho \quad i \neq i'$$

Thus,

$$\begin{aligned}
\text{Var}(\hat{\mu}) &= \frac{\sum_{i=1}^{n} \sum_{j=1}^{J} \sigma^2 + \sum_{j=1}^{J} n(n-1)\sigma^2 \rho}{(N)^2} \\
&= \frac{\sigma^2 + (n-1)\sigma^2 \rho}{N} \\
&= \frac{\sigma^2 (1 + (n-1)\rho)}{N}
\end{aligned} \tag{1.3}$$

If $\rho = 1$; the students in a school may have the same test score, then the variance

3

increases as shown in Equation 1.4

$$
\begin{aligned}
\text{Var}(\hat{\mu}) &= \frac{\sigma^2(1 + (n-1)\rho)}{nJ} \\
&= \frac{\sigma^2(1 + (n-1))}{nJ} \\
&= \frac{n\sigma^2}{nJ} \\
&= \frac{\sigma^2}{J}.
\end{aligned}
\tag{1.4}
$$

Intuitively, since $\rho = 1$, each additional individual observation in a school provides no new information about μ, so though there are nJ students, effectively it is as if the sample is J, the number of schools.

On the other hand, if $\rho = 0$ as assumed in traditional models, Equation 1.4 becomes

$$
\begin{aligned}
\text{Var}(\hat{\mu}) &= \frac{\sigma^2(1 + (n-1)\rho)}{nJ} \\
&= \frac{\sigma^2}{nJ}.
\end{aligned}
\tag{1.5}
$$

To capture the effect of ρ on the variance, we define the **effective sample size** (n_{eff}):

$$
n_{eff} = \frac{nJ}{1 + (n-1)\rho}
\tag{1.6}
$$

Thus, when $\rho = 0$,

$$
n_{eff} = Jn.
\tag{1.7}
$$

The effective sample size increases as the number of students per class increases. On

the other hand if $\rho = 1$,

$$n_{eff} = \frac{Jn}{n} = J, \tag{1.8}$$

meaning that the effective sample size is the same no matter the number of student per class. Figure 1 shows how the effective sample decreases as ρ increases when there are $n = 100$ students in $J = 16$ schools.
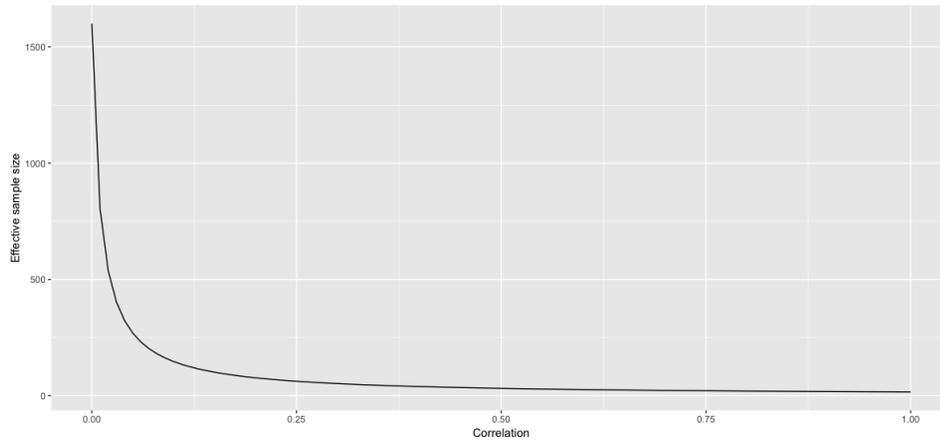
Figure 1: The relationship between Effective Sample Size and Correlation

Individual level model vs. Aggregate Model

To see the role of effective sample size, we consider two common traditional approaches described by Raundenbush and Bryk (2002):

- **Individual Level Mode:** The first technique is to ignore the group structure and treat all variables as level one variables. In the simple example described above, this implies using one-sample t-methods. Hence, the (incorrect) estimated standard error of the sample mean will be

$$\widehat{s.e.(\widehat{\mu})} = \frac{S}{\sqrt{N}} \tag{1.9}$$

where S is the sample standard deviation. However, in the case where $\rho > 0$, this provides biased estimate of the true variance derived in Equation 1.2.

$$E\left[\frac{S^2}{N}\right] = \frac{\frac{N}{N-1}\frac{n_{eff}-1}{n_{eff}}\sigma^2}{N} < \frac{\sigma^2}{n_{eff}} \tag{1.10}$$

Aggregate Model: The other technique is to aggregate the individual level variables to the higher level and do the analysis at the higher level. In this case, we use one-sample t-methods with the J group means. Since the data are assumed be independent between groups, the group means, $\overline{Y}_{.j}$, are independent with mean μ and variance $\sigma^2 \frac{1+(n-1)\rho}{n}$. The estimated standard error will be

$$\widehat{s.e.(\hat{\mu})} = \frac{S_g}{\sqrt{N}} \tag{1.11}$$

where S_g is the standard deviation of the J group means. In this simple case, the estimate of the true variance is unbiased,

$$E\left[\frac{S_g^2}{N}\right] = \frac{\sigma^2}{n_{eff}}. \tag{1.12}$$

However, if the group sizes are not equal, weights should be used when computing S_g. Furthermore, we lose the within-group information and (often about 80 - 90 % of the variation before running the model). For more complicated linear models with predictors at level one, the loss of variation produces strong correlation between variables and leads to inaccurate interpretations.

Data Description

We now present the particular motivating example for the assessment of the implementation of the Three-level Cross-classified **Hierarchical Linear Model (HLM)** in lme4. Guyot et al. (2017) and Galindo and Newton (2017) investigated the effect of professional development on the growth in mathematics achievement for children enrolled in elementary schools. For that purpose, the researchers selected 14,750 pupils nested in 499 classes where the classes are nested within 35 elementary schools. Figure 2 shows the data design.
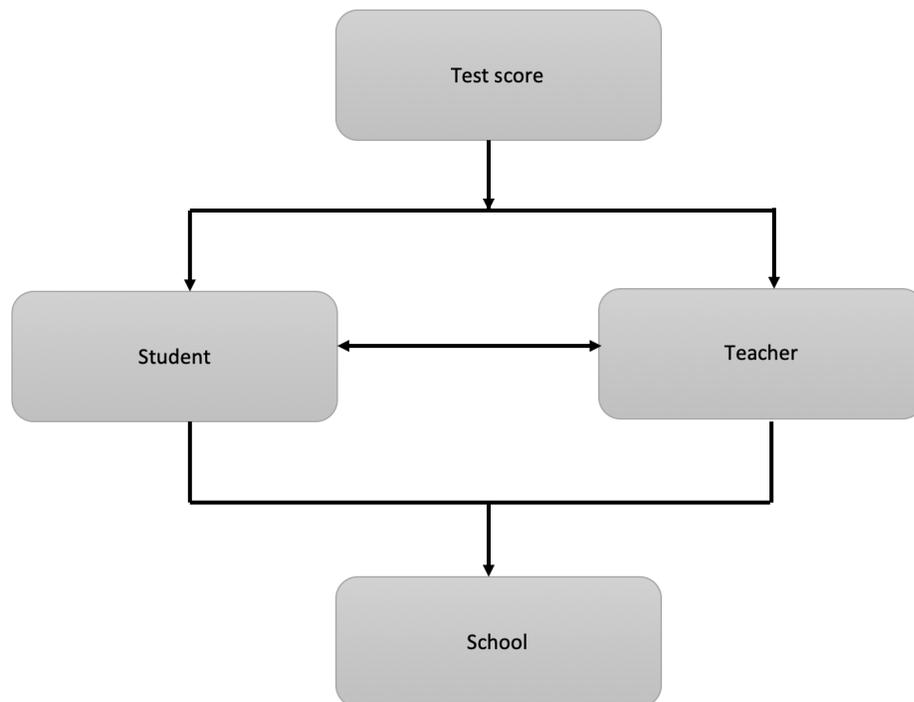
Figure 2: The Data Structure in Three Levels

The test score in Figure 2 is nested within the teacher and student

characteristics where teacher and student are cross classified in level two and both teacher and student are nested within schools in level three. The test score was observed through three academic years. The variable y_{tijk} represents the test score in year t for student i in class j and school k where, $t = 1, 2, 3$, $i = 1, \ldots, n$, $j = 1, \ldots, J$ and $k = 1, \ldots, 35$. Level one of the data describe the test score in term of time. On the other hand, level two explains the (teacher and student) characteristics.

Data Structure

The researchers described the data in three levels to indicate effects of students, teachers and schools on the test score. The structure of the data were as follows

- **Level 1 - Test** is the mathematical achievement of each student for each academic year of the study.

- **Level 2 - Student** is the individual characteristics of each student (e.g., bilingual, special education, diversity).

- **Level 2 - Teacher** describes the instruction mathematical knowledge for teaching at the baseline and each of the subsequent years.

- **Level 3 - School** indicates the characteristics of each school during this study.

Analyzing the Data

Since our data are educational, we do not expect the data to satisfy the assumptions of classical multivariate regression which presents several challenges to the data analyst.

- **The data are nested:** since it has a hierarchical structure contained in three levels creating several types of correlation within the response observations. First, achievement is measured repeatedly over time for a single student. Secondly, the teacher and the fellow students in a class have a large impact on student learning. Finally, students are nested in schools and school culture may impact the learning of all students in a school.

- **The nesting structure is complicated:** since the teacher and students are cross-classified in level-two. This complexity adds challenges to the numerical techniques used in estimating hierarchical models.

- **The change of the test-score:** In this particular study, the standardized test used by the school district was changed during the study. This complicates year to year comparisons needed to assess growth.

- **Measuring teacher quality is notoriously difficult:** In this study, the quality of teaching was assessed via periodic classroom observations which were video taped and scored according a rubric. Due to several factors, including that the observed lessons occurred at different times of the year and over different topics, the measure may not be reliable.

- **Data are missing:** We do not have complete data for each student nor each teacher. The lme4 package used to analyze the data require complete data. Hence the sample used in a particular analysis depends upon: (1) which variables are included, (2) complicated comparison between possible models and (3) leading to potential bias in the estimates.

In this study we will focus on the first two of these challenges: the nested and cross-classified structure of the data.

<div align="center">**Statistical Software**</div>

Specialized software packages are required in order to estimate the HLM parameters. In this research, I will analyze the data (by using R and HLM7 software).

R Implementation

R is an environment for statistical computing and graphics which is similar to the S language environment developed by Bell Laboratories. In R, I mainly will use the lme4 package to fit HLM and analyze the fixed and random effects. In lme4, users can do any data processing in R environment (McCoach et al., 2018).

HLM7 Implementation

In addition to R, I will use HLM7 package which was developed by Stephen Raudenbush, Cheong Congdon, Du Toit and produced by Scientific Software International (SSI) (McCoach et al., 2018). The main reason of using the HLM7 package is because of its ease of use, allowing use the data and is able to specify the models. Moreover, it is important in cross-classified hierarchical linear model.

Lme4 vs HLM7

- **The package lme4:** LME4 is an R package created by Douglas Bates who shared in creating the package NLME. In this project, I will use lme4 version 1.1-21 which was released on March 5th, 2019. The package lme4 requires that the user is knowledgeable with R code. In term of Estimation, the lme4 function (lmer) uses both **Full Maximum Likelihood (FML)** and **Restricted Maximum Likelihood (MLR)** (default) for estimation with Powel's (2009) BOBYQA (as the default algorithm to estimate the variance

components matrix. Due to the sensitivity of the computational algorithm (BOBYQA), the package lme4 shows convergence warning messages. I have experienced convergence issues with the package lme4 that may annoy the user. I will talk in details about these messages.

- **The package HLM7:** The other package I will use is HLM7 which is released by Scientific Software International by (Raudenbush, Bryk,Cheong, Congdon, & Du Toit, 2011). HLM7 allows users to specify models, within a user graphical user interface (GUI). For level -one, the users must select level-one predictors and specify all the ID variables that related to the higher levels. (e.g., student, school, etc). If level-two is a cross-classified level, the user has to choose the Id variables for the row and the column (e.g., student and teacher) where student and teacher are cross-classified.

    In term of estimation, HLM7 has several techniques such as MLR and FML. In terms of estimation methods, and uses both of the EM algorithm and Fisher scoring to compute the estimates (McCoach et al., 2018) HLM7 does not experience convergence issues as in lme4 since HLM7 has as "*automatic fix-up*" method (McCoach et al., 2018).

Statement of Purpose

A matter of concern is the growth of child academic achievement in math in elementary schools. The data observed contain three levels: level-one includes the students' test scores across three years while level-two contains the characteristics of the teachers and students involved. Additionally, level-three describes the schools where teachers and students are nested within. Based on the data structure, the data are hierarchically designed and the teacher and the student are cross-classified. As a result, the research will shed the light onto the cross-classified-three-level models. That will include the nature of the hierarchical data structure and how to fit hierarchical linear model to the data. Additionally, the research will discuss the estimation theory in term of fixed and random effects of the hierarchical linear models. In this thesis, I will do the following:

- Simulate a three-level-cross-classified data by R. I will test the accuracy of the package lme4 in estimating the model's parameters. I will also consider having a balanced, complete and large data in order to facilitate the model estimation. This data include the test score in level one, the teacher and the student predictors in level-two and school predictor in level-three. Additionally, the teacher and the student are cross-classified in level-two.

- Fit the cross-classified model that match the simulated data by using the function lmer in the package lme4.

- Study the correlation between the cross-classified errors of level-two (the variation of the teacher characteristics and the effect of the teaching quality on the student' test score).

- Assess the relationship between the cross-classified random coefficients in

level-two with the one in level-three.

- Study the convergence and singularity warning messages given by the package lme4 and examine these messages in term of fixed and random effect estimation.

- Compare the results of the models with issues (convergence or singularity) with their estimates in HLM7

- Examine the effect of removing the level-three random effect of the student predictor.

II.   REVIEW OF LITERATURE

Hierarchical Linear Models

History and motivation

Many studies in the social sciences require a hierarchical data structure (e.g., educational and medical research). In order to analyze this kind of data, a special kind of model must be considered. One level linear model would not be enough to estimate the effects of the individuals' characterizations. The early generation of hierarchical applications involved three objectives: (1) enhanced estimation of effects within individuals' units, (2) the formulation and testing of hypothesis about cross-level effects, (3) partitioning variance-covariance components.

Hierarchical Data Design

In social science and education, the data are hierarchically designed. A leading example is that the students are in classes. There are variables describing students and variables describing the classes. The class variables could also be collected with the students' variables such as the number of students in a class or a student's test score in a class. The class variable could describe the teacher, if the class has only one teacher. Further, the hierarchical design could include the classes in schools, schools in districts and so on (Raundenbush and Bryk, 2002).

Why Multilevel Model?

The matter of interests is the student academic achievement in math. The students are nested in classes where each class has one teacher and classes are nested within schools. In order to track the students academic achievement, the

investigators recorded the student's test score for three consecutive years. Suitable models must be fitted in order to evaluate the student's achievement. Since the test score (the response variable) is nested within the individuals' (students and teachers) characteristics, whereas students and teachers are nested within different schools, simple model (one level model) is insufficient to provide the accurate outcome,because we usually have some variables describing individuals. Yet, in real life, these individuals could be grouped into some units (e.g., students' characteristics) and their values go to a higher level of individuals. This is the idea of the hierarchical model, where the structure of the data are built on multi-level models. Similarly, in the education example, the data are hierarchically designed. Raundenbush & Bryk (2002) looked at the hierarchical structure from three foci: (1) The annual individual growth of students for the academic year, (2) the effects of student's characteristics and teacher's experience affecting the test score, and (3) the school where both student and teacher are nested within. The Hierarchically Linear Model (HLM) describes the test as the response variable and both student and teacher as the interacted individual's characteristics. For example, Students are described by their explanatory variables, such as diversity, English Language Learner (ELL), Special Education (Sp.Ed), etc. Moreover, the students who are grouped in a class are influenced by their teacher's characteristics. That is, the teacher's units are fixed effects for the response variable (test score) in level one. Both teachers and students are described by the school characteristics which are in the third level. The school is described by some other variables such as federal lunch reimbursement rates (Raundenbush & Bryk, 2002).

2-Level Model

In order to understand the multilevel model, we start by fitting a simple linear regression model. The model's dependent variable y is expressed as a function

of an explanatory variable (predictor) X multiplied by its coefficient β_1 and added to the intercept β_0 (The sample mean when the predictor $X = 0$) and the random error ϵ as follows:

$$Y_i = \beta_0 + \beta_1 X_i + \epsilon_i, \tag{2.1}$$

where

β_0 the intercept of the model

β_1 the fixed effect of the predictor X

X the predictor of the model

ϵ the error within the observations and $\epsilon \sim N(0, \sigma^2)$

Notice that the intercept is common for all the individuals. However, when the individuals are grouped together with some conditions (e.g., students in a class, players on teams, etc.), the intercepts differ from one group to another. Therefore, The intercept vary among the difference of the clusters.

The unconditial model

A leading example of the unconditional model is the student academic performance. The student is nested in a classroom. In this study, Finch W. Holmes and Kelley (2014) assumed that the error of level one is normally distributed with constant variance and level two residuals are independent across classrooms. Additionally, level two fixed effects are independent of level one residuals with a normally distributed multivariate with a constant covariance matrix. Finch W. Holmes and Kelley (2014) stated the equation of level 1 would be as follows

$$Y_{ij} = \beta_{0j} + \epsilon_{ij}, \tag{2.2}$$

where

Y_{ij} the test score of student i in a class j

β_{0j} the average of the test score in class j

ϵ_{ij} the error within the students in class j and $\epsilon_{ij} \sim N(0, \sigma^2)$.

Because the mean of the test score β_{0j} varies from one class to another, we should represent the intercept of the test score in terms of the student characteristics and represent the level 2 as follows:

$$\beta_{0j} = \gamma_{00} + U_{0j}, \tag{2.3}$$

where

γ_{00} the grand mean of all students in all classrooms j

U_{0j} the error between classes' test score average and $U_{0j} \sim N(0, \tau_{00}^2)$.

By substituting Equation 2.3 in Equation 2.2, the mixed model is

$$Y_{ij} = \gamma_{00} + U_{0j} + \epsilon_{ij}. \tag{2.4}$$

where γ_{00} is the model's fixed effect and (U_{0j} and ϵ_{ij}) are the random effects of the model.

17

The unconditional growth model

In the previous model we did not represent any predictor in level one. In growth model, we add the time as a predictor in level one. That is, time affects the student academic performance. Magnusson (2015) explained the unconditional equation as follows

Level 1

$$Y_{ij} = \beta_{0j} + \beta_{1j}t_{ij} + \epsilon_{ij} \tag{2.5}$$

Level 2

$$\beta_{0j} = \gamma_{00} + U_{0j} \tag{2.6a}$$

$$\beta_{1j} = \gamma_{10} + U_{1j} \tag{2.6b}$$

where

t_{ij} the time of student ij

β_{1j} the effect of the time for student i in class j

U_{0j} the unique increment to the intercept associated with class j

U_{1j} the unique increment to the slope associated with class j

where

$$\begin{pmatrix} U_{0j} \\ U_{1j} \end{pmatrix} \sim N\left(0, \tau^2\right)$$

and

$$\boldsymbol{\tau^2} = \begin{pmatrix} \tau_{00}^2 & \tau_{01}^2 \\ \tau_{10}^2 & \tau_{11}^2 \end{pmatrix}$$

The conditional growth model

It is possible that the individuals' characteristics affect level one intercept $\beta_0 j$. For example, the class' predictor W may affect the initial test score $(t = 0)$. Magnusson (2015) stated the conditional model as follows

Level 1

$$Y_{ij} = \beta_{0j} + \beta_{1j}t_{ij} + \epsilon_{ij} \tag{2.7}$$

Level 2

$$\beta_{0j} = \gamma_{00} + \gamma_{01}W_j + U_{0j} \tag{2.8a}$$

$$\beta_{1j} = \gamma_{10} + U_{1j} \tag{2.8b}$$

where,

W_j The class characteristics j

Notice that the individual characteristics may also affect the student's progress β_{1j} (the slope of time) meaning that equation 2.8b becomes

$$\beta_{1j} = \gamma_{10} + \gamma_{11}W_j + U_{1j} \tag{2.9}$$

Thus, the model in Equation 2.7 can be written as follows

$$Y_{ij} = \gamma_{00} + \gamma_{01}W_j + U_{0j} + \gamma_{10}t_{ij} + \gamma_{11}W_jt_{ij} + U_{1j}t_{ij} + \epsilon_{ij} \qquad (2.10)$$

Where γ_{00}, γ_{01} and γ_{10} are called the fixed effects and U_{0j}, U_{1j} and ϵ_{ij} are called the random effects of the model.

3-Level Model

The previous example showed that the test is nested within the students' characteristics. That is, the pupils have different academic performances that are based on their characteristics (e.g., ELL, gender, etc.). That example may be extended to three levels if the students are nested within schools. In this case, their academic performance would vary based on the schools' characteristics(e.g., public school, private school, etc.). The model in this case is no longer explained in two but three levels where the school is the third level. Magnusson (2015) extended the second level as follows

Level 1

$$Y_{ijk} = \beta_{0jk} + \beta_{1jk}t_{ijk} + \epsilon_{ijk} \qquad (2.11)$$

Level 2

$$\beta_{0jk} = \gamma_{00k} + \gamma_{01k}W_{jk} + U_{0jk} \qquad (2.12a)$$

$$\beta_{1jk} = \gamma_{10k} + U_{1jk} \qquad (2.12b)$$

Level 3

$$\gamma_{00k} = \delta_{000} + \delta_{001}Z_k + V_{00k} \tag{2.13a}$$

$$\gamma_{01k} = \delta_{010} + V_{01k} \tag{2.13b}$$

$$\gamma_{10k} = \delta_{100} + V_{10k} \tag{2.13c}$$

where,

i the index of the student and $i = 1, \ldots, n$.

j the index of the class (teacher) and $j = 1, \ldots, J$.

k the index of the school and $k = 1, \ldots, K$

δ_{000} the grand mean of students' test score over the schools

δ_{001} the effect of the school characteristic on the test score of student$_{ijk}$

Z_k the predictor of the school characteristic

δ_{100} the student's improvement over the school k

V_{00k} the random effect of the initial test score among schools

V_{01k} the error of the class characteristic across schools

V_{10k} the error of student improvement effect within the schools

where,

$$\begin{pmatrix} V_{00k} \\ V_{01k} \\ V_{10k} \end{pmatrix} \sim N\left(0 \;,\; \mathbf{\Phi}\right)$$

where

$$\mathbf{\Phi} = \begin{pmatrix} \Phi_{00k}^2 & 0 & 0 \\ 0 & \Phi_{11}^2 & 0 \\ 0 & 0 & \Phi_{11}^2 \end{pmatrix}$$

The mixed model is

$$Y_{ijk} = \delta_{000} + \delta_{001}Z_k + \delta_{010}W_{jk} + \delta_{100}t_{ijk} + V_{00k}$$
$$+ U_{0jk} + V_{10k}t_{ijk} + U_{1jk} + \epsilon_{ijk}, \tag{2.14}$$

where $\delta_{000}, \delta{001}, \delta_{010}$ and δ_{100} are the fixed effects and U_{0jk}, U_{1jk} , V_{00k}, V_{10k} and ϵ_{ijk} are the random effects of the model.

Cross-Classified Model

I have explained some ideas about 2-level model where the individuals of level one are nested in level two units (organizations). Additionally, I clarified the 3-level model where the organizations are nested in higher level (e.g., district). However, sometimes level two units may cross classify (e.g., school and neighborhood contributions to the student education). Raundenbush & Bryk (2002) investigated a sample of 2,310 students are nested within 524 neighborhoods and 17 schools where there are six students living in neighborhood 259 attending school 10 and one student living in the same neighborhood and attending school 11, and two other students attend school 16 (p.373). In this case, Raundenbush & Bryk (2002) considered steps for analyzing the data:

- estimate the components of variance outcomes that are within neighborhoods, between schools and between students within (neighborhood by school).

- estimate the residual components between schools, between neighborhood and within the cells after taking into account student, neighborhood and school characteristics.

- estimate random effects associated with particular neighborhoods or schools (p.375).

Unconditional 2-level-cross-classified model

Raundenbush & Bryk (2002) presented the unconditional cross-classified model that represents the above mentioned study as follows

Level 1: Students are nested within cross-classified cells

$$Y_{ijk} = \pi_{0jk} + e_{ijk} \qquad e_{ijk} \sim N(0, \sigma^2) \tag{2.15}$$

where,

Y_{ijk} the attainment of student i in neighborhood j and school k

π_{0jk} the mean of the test score of students who live in neighborhood j and attend school k

e_{ijk} the random effect of students' test scores that varies around the test score mean π_{0jk}.

where the indices i, j, and k represents the students, neighborhood and schools as follow

$i \quad = 1, \ldots, n_j;$

$j \quad = 1, \ldots, J;$ and

$k \quad = 1, \ldots K.$

Level 2: Between cross-classified cells

$$\pi_{0jk} = \theta_0 + b_{00j} + c_{00k} + d_{0jk} \tag{2.16}$$

where

$$\begin{cases} b_{00j} \sim N(0, \tau_{b00}), \\[2ex] c_{00k} \sim N(0, \tau_{c00}), \\[2ex] d_{0jk} \sim N(0, \tau_{d00}), \end{cases}$$

and

θ_0 the grand mean of all the students test score;

b_{00j} the random effect of neighborhood j;

c_{00k} the random effect of school k; and

d_{0jk} the random interaction effect, that is, the deviation of the cell mean

      predicted by the grand mean.

The mixed model is

$$Y_{ijk} = \theta_0 + b_{00j} + c_{00k} + d_{0jk} + e_{ijk} \tag{2.17}$$

where the model is called two-way analysis of variance with random row effects b_{00j}; column effects c_{00k}; two-way interaction effect and within students variation e_{ijk} (Raundenbush and Bryk, 2002). In the above mentioned model, Raundenbush and Bryk (2002, p.378) listed three types of correlation coefficients:

- the correlation between the test score of two students who have the same school and neighborhood:

$$\text{corr}(Y_{ijk}, Y_{i'jk}) = \rho_{bcd} = \frac{\tau_{b00} + \tau_{c00} + \tau_{d00}}{\tau_{b00} + \tau_{c00} + \tau_{d00} + \sigma^2};$$ (2.18)

- the correlation between two students test scores who live in the same neighborhood but attend two different schools:

$$\text{corr}(Y_{ijk}, Y_{ijk'}) = \rho_b = \frac{\tau_{b00}}{\tau_{b00} + \tau_{c00} + \tau_{d00} + \sigma^2};$$ (2.19)

- the correlation between two students' test scores that attend the same school but live in two different neighborhood:

$$\text{corr}(Y_{ijk}, Y_{ij'k}) = \rho_c = \frac{\tau_{00c}}{\tau_{b00} + \tau_{c00} + \tau_{d00} + \sigma^2}.$$ (2.20)

It is also useful considering the reliability with which the neighborhood affects the students who attend a specific school:

$$\text{reliability}[\hat{b}_{00j} + \hat{d}_{0jk}|c_{00k}] = \frac{\tau_{b00} + \tau_{d00}}{\tau_{b00} + \tau_{d00} + \sigma^2/n_{jk}}$$ (2.21)

and the reliability with which the school affects the students who lives in a specific neighborhood:

$$\text{reliability}[\hat{c}_{00k} + \hat{d}_{0jk}|b_{00j}] = \frac{\tau_{b00} + \tau_{c00}}{\tau_{c00} + \tau_{d00} + \sigma^2/n_{jk}}$$ (2.22)

A leading example of the reliability is a student taking the Graduate Record Examinations (GRE) test. The test score does not state the test's difficulty nor the student's percentile among the other students if the student takes the test alone. However, if the student is tested with students in the same class, the reliability ratio

25

explains the percentile of the student's test score within the class. Further if the student takes the test with students in different classes that are nested in a school, the reliability ratio represents the student's performance among all schools' students.

Conditional 3-level-cross-classified model

The idea of the unconditional 2-level-cross-classified model may be developed to conditional three-level-cross-classified model by adding predictors in level 1, 2, and 3 respectively.

To make the presentation simpler, I use simulated data, (I have created and explained in Chapter 3). Consider a study of the student achievement in elementary school where the test score (level 1) is nested within the teacher and student characteristics (level 2) in which teacher and student are nested within schools characteristics (level 3). In this simulation, I represent four predictors: X that is a level-1 categorical variable that represents the time of the test, W_1 is a level-2 categorical variable that explains the student's ELL, W_2 is a continuous variable that represents the teacher's MQI score, and Z is a level-3 categorical variable that explains the school's treatment.

Level 1 or "within cells" (Raundenbush and Bryk, 2002, p.377)

$$Y_{t(ij)k} = \beta_{0(ij)k} + \beta_{1(ij)k}X_{t(ij)k} + e_{t(ij)k}, \qquad \text{where } e_{t(ij)k} \sim N(0, \sigma^2), \qquad (2.23)$$

Level 2 or "between cells" (Raundenbush and Bryk, 2002, p.377)

$$\beta_{0(ij)k} = \gamma_{00k} + (\gamma_{01ik} + c_{1jk})W_{1ik} + \gamma_{02jk}W_{2jk} + b_{0ik} + c_{0jk} \qquad (2.24a)$$

$$\beta_{1(ij)k} = \gamma_{10k} \qquad (2.24b)$$

where,

$$b_{0ik} \sim N(0, \tau_{st}^2)$$

,

$$\begin{pmatrix} c_{0jk} \\ c_{1jk} \end{pmatrix} \sim N\left(0, \boldsymbol{\tau_{tch}^2}\right)$$

and

$$\boldsymbol{\tau_{tch}^2} = \begin{pmatrix} \tau_{tch00}^2 & \tau_{tch01}^2 \\ \tau_{tch10}^2 & \tau_{tch11}^2 \end{pmatrix}$$

and level 3 is

$$\gamma_{00k} = \delta_{000} + \delta_{001} Z_k \tag{2.25a}$$

$$\gamma_{01ik} = \delta_{010} + d_{01ik} \tag{2.25b}$$

$$\gamma_{02jk} = \delta_{020} \tag{2.25c}$$

$$\gamma_{10jk} = \delta_{100} + d_{10jk} \tag{2.25d}$$

where,

$$\begin{pmatrix} d_{01ik} \\ d_{10jk} \end{pmatrix} \sim N\left(0, \boldsymbol{\Phi^2}\right)$$

27

and

$$\mathbf{\Phi^2} = \begin{pmatrix} \Phi_{00}^2 & 0 \\ 0 & \Phi_{11}^2 \end{pmatrix}$$

The combined model is

$$
\begin{aligned}
y_{t(ij)k} =\ & \delta_{000} + \delta_{001}Z_k + (\delta_{010} + c_{1jk} + d_{01ik})W_{1ik} \\
& + \delta_{020}W_{2jk} + (\delta_{100} + d_{10jk})X_{t(ij)k} \\
& + b_{0ik} + c_{0jk} + \epsilon_{t(ij)k}
\end{aligned}
\tag{2.26}
$$

Notice that the coefficients of level 2 (δ_{010}, c_{1jk} and d_{01}) are the effects on the student ELL where,

δ_{010} is the average difference between the ELL= 1 vs ELL =0 in test score after accounting for other variables in the model.

c_{ijk} is the teacher's effect on the test score of students with ELL in class jk

d_{01} is the teacher's effect on the test score of students with ELL in school k.

Notice, when the student has different teacher every year, the teacher's effect on the test score of students with ELL would vary. That is, the cross-classified data provide us more information to understand the effect of the teacher on the student's characteristics.

Estimation Theory

One-way ANOVA with fixed effects

Consider the following equation:

$$Y_{ij} = \mu + \alpha_j + e_{ij} \qquad for \sum_{j=1}^{J} \alpha_j = 0 \qquad\qquad (2.27)$$

where

μ is the overall mean,

α_j is a parameter unique to the ith treatment,

e_{ij} is a random error component.

      For the fixed effect hypothesis test, we assume

α_j is not random

Y_{ij} are all independent

e_{ij} is the error and $e_{ij} \sim N(0, \sigma^2)$

      Montgomery (2012) stated the one-way ANOVA hypothesis test of the experiment:

$H_0 : \alpha_1 = \alpha_2 = \cdots = \alpha_J = 0 \qquad\qquad H_a : \alpha_j \neq 0$ for at least one j

      We also can write $\mu + \alpha_j$ as

$$\mu + \alpha_j = \mu_j \qquad\qquad (2.28)$$

where the number of the parameters is $j + 2$. Then, the one-way ANOVA hypothesis test becomes

$H_0 : \mu_1 = \mu_2 = \cdots = \mu_J \qquad\qquad H_a : \mu_i \neq \mu_j$ for at least one pair(i, j)

Where the variance and covariance components are knowm (Montgomery, 2012).

If we have the same sample size for all j, then,

$$\hat{\mu} = \overline{Y_{..}}$$
$$= \frac{1}{IJ} \sum_{j=1}^{J} \sum_{i=1}^{I} y_{ij} \tag{2.29}$$

and if we have a different sample size, then

$$\hat{\mu} = \overline{Y_{..}}$$
$$= \sum_{j=1}^{J} I_j \overline{Y_{.j}} / \sum_{j=1}^{J} I_j \tag{2.30}$$

One-Way ANOVA with random effects

Consider the unconditional 2-level hierarchical linear model:

$$Y_{ij} = \beta_{0j} + r_{ij} \tag{2.31}$$

and assume that r_{ij} is normally distributed with a mean zero and a constant variance σ^2. In this level the intercept β_{0j} is the mean outcome of the jth unit (Raundenbush and Bryk, 2002). That is, $\beta_{0j} = \mu_{Y_j}$. The level 2 model for the one-way is

$$\beta_{0j} = \gamma_{00} + u_{0j}, \tag{2.32}$$

where γ_{00} is the overall mean and u_{0j} is the random effect that is associated with the unit j where $u_{0j} \sim N(0, \tau_{00})$.

Substituting Equation 2.32 into Equation 2.31, Raundenbush and Bryk

(2002) provided the mixed model equation

$$Y_{ij} = \gamma_{00} + u_{0j} + r_{ij} \tag{2.33}$$

which is the one-way ANOVA.

Therefore, the hypothesis test for the random effects is:

H_0: $\quad \tau_{00} = 0$

H_a: $\quad \tau_{00} \neq 0$

Notice the number of parameters in Equation 2.27 is $J + 1$ while, the number of parameters in Equation 2.33 is 1. It is clear that testing and estimating one parameter is easier than $J + 1$.

Based on Equation 2.33, the variance of the outcome is

$$\begin{aligned} \mathrm{Var}(Y_{ij}) &= \mathrm{Var}(\gamma_{00} + u_{0j} + r_{ij}) \\ &= \tau_{00} + \sigma^2 \end{aligned} \tag{2.34}$$

Using One-way ANOVA is recommended, because it is a simple method to analyze the data specially the hierarchically designed. Furthermore, it provides more information about the variability of each level of the model. Raundenbush and Bryk (2002) suggested "the correlation as a useful parameter for one-way ANOVA , because it measures the proportion of the variance in the outcome that is between level-2 units"(p.24).

$$\rho = \frac{\tau_{00}^2}{\tau_{00}^2 + \sigma^2} \tag{2.35}$$

31

In order to estimate the fixed effects of a model, there are two cases: The variance-covariance are known, and the other case the variances are assumed unknown.

One-Way ANOVA in HLM

Fixed effect: Point Estimation

A matter of interest is investigating the student's math performance among J schools. Raundenbush and Bryk (2002) studied three "types of estimation: (1) One-way random effect ANOVA model, (2) examining the means as outcomes of the model and (3) considering the random coefficient regression model (pp.39-40)".

In order to compare with other models, consider the study of student math performance in several schools as follows

Level 1

$$y_{ij} = \beta_{0j} + \epsilon_{ij} \tag{2.36}$$

where y_{ij} is the test score for student i in school j. $i = 1, \ldots, n_j$ and $j = 1, \ldots, J$ and $\epsilon_{ij} \sim N(0, \sigma^2)$.

Raundenbush and Bryk (2002) averaged about the n_j observations as follows

$$\bar{Y}_{\cdot j} = \beta_{0j} + \bar{\epsilon}_{ij} \tag{2.37}$$

where

$$\bar{\epsilon}_{ij} = \Sigma_{i=1}^{n_j} \epsilon_{ij}/n_j$$

$$\mathrm{Var}(\bar{\epsilon}_{ij}) = \sigma^2/n_j = V_j$$

where $\bar{Y}_{.j}$ is an estimator for β_{0j}.

The level two model is

$$\beta_{0j} = \gamma_{00} + u_{0j} \tag{2.38}$$

where $u_{0j} \sim N(0, \tau_{00})$

Substituting Equation 2.38 in Equation 2.37, the combined model yields

$$\bar{Y}_{.j} = \gamma_{00} + u_{0j} + \bar{\epsilon}_{ij} \tag{2.39}$$

and the variance of $\bar{Y}_{.j}$ is

$$\begin{aligned}
\mathrm{Var}(\bar{Y}_{.j}) &= \mathrm{Var}(u_{0j}) + \mathrm{Var}(\bar{\epsilon}_{ij}) \\
&= \tau_{00} + V_j \\
&= \Delta_j
\end{aligned}$$

If the sample sizes are equaled for all level two units, then the variance component will be Δ and the estimate of the grand mean $\hat{\gamma}_{00}$ is

$$\hat{\gamma}_{00} = \sum \bar{Y}_{.j}/J; \tag{2.40}$$

while, if the sample sizes are not equaled, then the variance components will remain Δ_j and $\hat{\gamma}_{00}$ is

$$\hat{\gamma}_{00} = \sum \Delta_j^{-1} \bar{Y}_{.j} / \sum \Delta_j^{-1} \tag{2.41}$$

which is the estimator of the fixed coefficient γ_{00} and this estimator is called the maximum likelihood estimator and the confidence interval of $\widehat{\gamma}_{00}$ is

$$95\%\text{CI}(\gamma_{00}) = \widehat{\gamma}_{00} \pm 1.96\sqrt{\sum \Delta_j^{-1}}. \tag{2.42}$$

If we look at the estimate of $\hat{\mu}$ in Equation 2.30 and the estimate of $\hat{\gamma}_{00}$ in Equation 2.41, we see that we replace I_j in the first equation with Δ_j^{-1} in the second equation. where

$$\Delta_j = \tau_{00} + \frac{\sigma^2}{I_j} \tag{2.43}$$

Since the ordinary estimation $\hat{\mu}$ is only one level (not hierarchical), then $\tau_{00} = 0$. Additionally, all the observations in the samples have the same measure. That is, $\sigma^2 = 1$. Therefore, Equation 2.43 becomes,

$$\begin{aligned}
\Delta_j^{-1} &= \frac{1}{\tau_{00} + \dfrac{\sigma^2}{I_j}} \\
&= \frac{1}{0 + \dfrac{1}{I_j}} \\
&= I_j
\end{aligned} \tag{2.44}$$

and that explains the comparison between the estimate of $\hat{\mu}$ and $\hat{\gamma}_{00}$.

Random Effect Estimation

In the above mentioned lines, I assumed that the variance and covariances components are known. However, in HLM, it is rare to find balanced and known variances. In order to estimate the variance components, a numerical method must be used. One of these methods is the maximum likelihood estimator. There are

34

three different types of estimation to solve this problem: **Full Maximum Likelihood (MLF)**, **Restricted Maximum Likelihood (MLR)** and Bayes estimation. In this research, I focused on maximum likelihood and restricted maximum likelihood (Raundenbush and Bryk, 2002, P.52).

Full maximum likelihood(MLF)

Consider the parameters $(\gamma, \sigma^2$ and $\tau)$, where γ represents the fixed effect coefficients, σ^2 and τ represent random effect coefficients of level 1 and 2 respectively. The basic idea of Maximum likelihood is to select estimates of γ, σ^2 and τ for which the likelihood of the observed Y is a maximum (Raundenbush and Bryk, 2002). These estimates are consistent and asymptotically efficient. Raundenbush and Bryk (2002) stated that "for a large sample , the MLF estimators are approximately unbiased with minimum variance"(p.52).

Maximum likelihood is useful when the sample size is large and there are many groups in the second level (Boedeker, 2017). Additionally, if it is desired to estimate a function of parameters, we simply plug in maximum likelihood estimator as shown in the example

$$\hat{\gamma}_{00} = \Sigma\Delta_j^{-1}\bar{Y}_{.j}/\Sigma\Delta_j^{-1}$$

where Δ_j is known. If each Δ_j is unknown, the result of $\hat{\gamma}_{00}$ is an estimator of Δ_j. One other benefit is that as long the sample size increases, the sampling distribution becomes normal and with a variance can be readily estimated (Raundenbush and Bryk, 2002).

Restricted maximum likelihood(MLR)

Restricted maximum likelihood differs from the full maximum likelihood because, MLR's variance and covariance are conditional. Raundenbush and Bryk (2002) showed that the estimator $\hat{\sigma}^2$ is bias downwards as compared with σ^2. Consider the multiple regression model

$$y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \cdots + \beta_J X_{Ji} + r_i \tag{2.45}$$

where $i = 1, 2, \ldots, n$ and $r_i \sim N(0, \sigma^2)$.If $\beta_0, \beta_1, \ldots, \beta_J$ are known, then the maximal likelihood of σ^2 is

$$\widehat{\sigma^2} = \sum \widehat{r_i^2}/n. \tag{2.46}$$

If the model parameters are unknown, which is true in most real case studies, then

$$\widehat{r_i} = Y_i - \widehat{\beta_0} - \widehat{\beta_1} X_{1i} - \widehat{\beta_2} X_{2i} - \cdots - \widehat{\beta_J} X_{Ji} \tag{2.47}$$

and the usual unbiased estimator is

$$\hat{\sigma}^2 = \sum \hat{r}_i^2/(n - J - 1) \tag{2.48}$$

Raundenbush and Bryk (2002) noticed that "If the number of parameters is small, then the correction will have a little effect as J increases the estimator $\widehat{\sigma^2}$ becomes negatively biased (p.53)".

The parameter estimation in MLR is performed in three steps: (1) find the transformation of the data which is invariant to the fixed effects, (2) use the maximum likelihood of the transformed data to find unbiased estimate of the variance components and (3) use the variance components to find the generalized

least square $\widehat{\delta}$.

Consider the linear model

$$Y = X\delta + Zb + \epsilon \quad \text{where} \quad Y \sim N(X\delta, H) \tag{2.49}$$

and

Y is the $(N \times 1)$ response vector,

X is the designed matrix $(N \times K)$ that includes the fixed effects,

Z includes the random effects, and

H is the variance components matrix including σ^2, τ^2, and ϕ^2

Consider the density likelihood function

$$\mathcal{L}(Y|\delta, \sigma, \tau) = \frac{1}{(2\pi)^{-\frac{n}{2}} det^{-\frac{1}{2}}[H]} e^{\frac{1}{2}(Y - \delta_{00}1)^T H^{-1}(Y - \delta_{00}1)} \tag{2.50}$$

- **Step 1:** We try to find a vector a is orthogonal to all the columns of X and define $A = \begin{bmatrix} a_1 a_2 \ldots a_{N-P} \end{bmatrix}$, then $A^T X = 0$ and $E(A^T Y) = 0$. Zhang (2015) stated the transformed data formula

$$A^T Y = A^T(X\delta + Zb + \epsilon) = A^T \epsilon \sim N(0, A^T H A) \tag{2.51}$$

- **Step 2:** We estimate θ by maximizing the log likelihood function $\mathcal{L}(\theta|A^T Y)$ where θ contains the variance components σ^2, τ^2 and Φ^2.

- **Step 3:** We use the variance components to find the generalized least square

$$\widehat{\delta} = (X^T H X)^{-1} X^T H Y \tag{2.52}$$

37

where H is the covariance matrix including σ^2, τ^2 and Φ^2.

Likelihood Ratio Test (LRT)

A likelihood ratio test (deviance test) is a statistical hypothesis that is used to compare the full model with the null or reduced model in order to test the significance of the model parameters (Montgomery and Geoffrey, 2012). This comparison computed by subtracting the model with smaller deviance out of the model of a larger deviance (Newsom, 2017). The model with a smaller deviance has more parameters than the model with a larger deviance. Bolin (2017) stated the ratio test formula as follows

$$L = \arg \max_\theta L(\theta|y) \tag{2.53a}$$

$$D(p) = -2(\log(Lp) - \log(L)) \tag{2.53b}$$

Where Lp is the Likelihood of the parameter p. The difference (the ratio value is the $\chi^2_{0.05,1}$, because there only one parameter is different in the two models (Newsom, 2017). In order to check the parameter effect significance, we check the χ^2 value of the difference with $\chi^2_{0.05,1}$.

Alternatively, the variance components does not have an asymptotic normal distribution. The difference again is $\chi^2_{0.05,1}$, because only one parameter changes. The test is one-tailed test (Newsom, 2017), because the variance can not be negative.

<div align="center">

Motivating Example

</div>

As I mentioned the motivated example in Chapter 1, that the data have several challenges. In this research I will focus on the following:

- **The data is nested:** The test score in Figure 2 is nested within the individuals level (the teacher and student) and the individuals are nested

within schools. That is, the initial test score's average variates from class to another. Moreover, the school test score average variates from school to another. As I explained in Chapter 1, neither the individual nor the aggregate model properly explains the data results. Thus, an ordinary multiple variables linear regression does not explain the data structure. In order to solve this problem, we need to consider the hierarchical linear model

- **Cross-Classified data:** In this chapter, I focused on the problem of cross-classified data. The student may have three different teachers in the three academic years. For example, student 1, student 2, and student 3 may have one teacher in year 1 and three teachers in year 2. While three students may have three teachers in year 1 and one teacher in year 2. Therefore, students and teachers are relatively cross-classified. For illustration, Table 1 shows how nine students are Cross-Classified with nine teachers during three years.

Table 1: Teacher and Student are Cross-Classified

	Year1			Year2			Year3		
	tch1	tch2	tch3	tch4	tch5	tch6	tch7	tch8	tch9
std1	x			x			x		
std2	x				x		x		
std3	x					x		x	
std4		x		x				x	
std5		x			x			x	
std6		x				x	x		
std7			x	x					x
std8			x		x				x
std9			x			x			x

- **Not all the students attended the three years:** As I mentioned before, understanding the teacher's effect of the test score for a student with a specific

characteristic is challenging. The reason of that is the nature of the data where not all the students experienced the cross classification. For example, some students were in the fifth grade in the first year of the experiment, while others were in the third grade in the last year of the experiment. So, the number of the students who have a perfect cross-classification with their teachers is small compared to total number of the observations. For that purpose, I simulated data that were similar to our motivated example in order to understand that problem.

Fitting Hierarchical Linear model in R

The next step after understanding HLM models is to fit these models by using a software package. In this research, I focused on R as it is an open resource and sufficient for the data analysis.

Fitting an Ordinary linear model

In order to fit an ordinary linear model in R, we use the function lm with the response and explanatory variables as follows

$$\text{model1} = \text{lm(y} \sim \text{x, data} = \text{data1)}$$

Fitting a HLM in R

Fitting a HLM is different from the OLM as we need to install a specific package to fit the multilevel model. There are two different packages that are able to perform a multilevel model: nlme, and lme4. They are both can fit a basic and advanced multilevel models. The package lme4 may be slightly newer and may provide more flexibility with the models than the nlme package.

- **The package nlme:** The package nlme is one of the packages that fit and compare Gaussian linear and nonlinear mixed-effects models. The current version 3.1-140 was created by Jose Pinheiro and Douglas Bates and released on 05-01-2019. The package nlme uses the function lme in order to fit multilevel model. The nlme package has a good variety of variance and correlation structure and access several distributions. However, the package speed is slow with large data and random effects, and it has a poor performance with estimating cross-classified random effects, and does not deal with multivariate data.

- **The package lme4:** The lme4 package is a project created by Douglas Bates (one of the co-authors of nlme). Firstly, the package seems to be faster than nlme. In addition, it works with cross-classified random effects better than the nlme package. One the other hand, the package does not work with covariance and correlation structures yet.

    Since the lme4 package works better in cross-classified model (the research topic), I chose the lme4 for this project. I used the motivating example to explain fitting multilevel models in R.

Fitting two-level model in lme4

The matter of interest is the test score of students that are nested in classes (teachers) and the teachers and students are contained in schools. In order to use lme4 in multilevel models, we must know the fixed and random effects with their relevant levels. Consider the unconditional two-level model where level-1 describes the test score of student ij and level-2 explains the variation of the students among classes (teachers), then mixed model is

$$y_{ij} = \gamma_{00} + \epsilon_{ij} + u_{0j} \tag{2.54}$$

The R command in lme4 is

$$\text{lmer}(y \sim 1 + (1|\text{tch}), \text{data} = \text{data1})$$

Suppose that the test score has predictor X and level level-2 is effected by the predictor W, then level 1 of the conditional mixed model is

$$y_{ij} = \gamma_{00} + \gamma_{01}X_{ij} + \gamma_{10}$$
$$+ u_{0j} + u_{ij}W_{0j} + \epsilon_{ij} \tag{2.55}$$

Notice that the first line of Equation 2.55 is the fixed effects, while the second line includes the random effects. The R command in lme4 is

$$\text{lmer}(y \sim X + W + (X|\text{tch}), \text{data=data1})$$

Notice that the predictors X and W have fixed coefficients γ_{01} and γ_{10} respectively, while the predictor X has a random coefficient $_{1j}$ at the teacher level.

Fitting three-level model in lme4

Suppose we extend the classes to be in schools where the schools variate based on their characteristics (e.g., public, private and so on) and include the school categorical variable Z. Then, mixed model is

$$y_{ijk} = \delta_{000} + \delta_{001}Z_k + \delta_{01k}W_{0jk} + \delta_{100}X_{ijk}$$
$$+ d_{00k} + u_{0jk} + \epsilon_{ijk} + d_{01k}W_{0jk} + d_{10k}X_{ijk} + u_{1jk}X_{ijk} \tag{2.56}$$

The R command in lme4 is

$$\text{lmer}(y \sim X + W + Z + (X|\text{tch}) + (X + W|\text{sch}), \text{data=data1})$$

Fitting cross-classified model in lme4

Consider the three-level-cross-classified model where X is a categorical variable explaining the time of the test, W_1 is a categorical variable describing the student's ELL, W_2 is a continuous variable that explain the teacher's MQI test score and Z is a categorical variable explains whether the school has treatment or not. Recall the mixed model in Equation 2.26

$$
\begin{aligned}
y_{t(ij)k} = {}& \delta_{000} + \delta_{001}Z_k + \delta_{010}W_{1ik} + \delta_{020}W_{2jk} + \delta_{100}X_{t(ij)k} \\
& + d_{00k} + b_{0ik} + c_{0jk} + \epsilon_{t(ij)k} \\
& + d_{01ik}W_{1ik}d_{10jk}X_{t(ij)k} \\
& + c_{1jk}W_{1ijk}.
\end{aligned}
\tag{2.57}
$$

Notice that the cross-classified random effect is c_{1jk} that affects the student's predictor W_{1ik}. Then the model in Equation 2.57 can be fitted in lme4 as folows

$$\text{lmer}(y \sim X + W1 + W2 + Z + (1|\text{std}) + (W1|\text{tch}) + (X+W1|\text{sch}), \text{data=data1})$$

Estimation theory in package lme4

The most common estimation methods for estimate the variance components are FML and MLR. In general the maximum likelihood (ML) technique is to find the coefficients' values that maximize the likelihood of the data. The full maximum likelihood (FML) selects the estimates of G, T and S^2 in order to estimate γ, τ and σ^2. On the other hand, "The restricted maximum likelihood (MLR) maximize the joint likelihood of these parameters for a fixed value of the sample data" (Raundenbush and Bryk, 2002, p.52).

- **BOBYQA algorithm:** The Bound Optimization BY Quadratic Approximation (BOBYQA) (Powell, 2009) algorithm "is a Quasi-Newton approach to estimation that is implemented in the lme4 package"(McCoach et al., 2018, p.8).

The function lmer in lme4 package uses "Both FML and MLR (the default) in terms of the computational algorithms, lmer utilizes Powell's (2009) BOBYQA by default, while offering other optimizers to estimate T" (McCoach et al., 2018, p.11).

III.   METHODOLOGY

The Statement of The Research

The matter of concern is assessing the three-level-cross-classified model in *lme4*. In order to accomplish that goal, three steps must be performed: (1) ensure that the package *lme4* can perform that model, (2) simulate cross-classified structured data to determine the accuracy of the package's estimation, and (3) understand the relationship between the model's parameters.

Since the study focuses on the cross-classified models, the research covered the behavior of fixed and random coefficients of the cross-classified predictors. In order to check the accuracy, true fixed and random values must be proposed.

Simulation Model

The matter of interest is a three-level-cross-classified model where the intercept and slope are randomly variates though level-two and level-three explanatory variables. In addition, level-two predictors are cross-classified. For that purpose, I used the package lme4 to obtain the estimate of the parameters (fixed and random).

In order to check the accuracy of lme4 in estimating the model parameters, I simulated a data set that is designed to fit a three-level-cross-classified model and compared lme4 output with the assigned parameters' true values.

Simulation Data Structure

I simulated three-level-cross-classified data that explains the student achievement in elementary school in three consecutive years. The data consist of 30

schools. Each school includes 500 students that are nested within 15 classes (teachers). I assume that each classroom has only one teacher. The test score was recorded for three years. The data were designed to be in three levels:

- Level 1 : explains the time of the test score and includes the predictor X (the year of the test) that takes the values 1,2 and 3

- Level 2: describes the effects of the teachers and students characteristics on the test score. It includes two predictors: (1) A student's categorical variable W_1 that explains the student characteristics (e.g., ELL) and takes the values 0 and 1, and (2) A teacher's continuous variable W_2 that represents a teacher test (e.g., MQI). The reason of choosing two different types of variables is to match the real-life data.

- Level 3: shows the effect of the school on the test score. It contains one categorical variable (Z) that takes the values 0 and 1. I choose it to be categorical, because the the variable Z represents the treatment.

In order to get the best results, I simulated large data containing 27,000 observations, balanced and complete.

In order to match the real-life data, I created five groups of students that are distributed on three grades such that each group contains 100 students. Table 2 shows the groups distribution of the three years.

Table 2: 500 Students are Distributed in Five Groups

Year	Third grade	Fourth grade	Fifth grade
Year 1	001:100	101:200	201:300
Year 2	301:400	001:100	101:200
Year 3	401:500	301:400	001:100

The table shows 300 students in year 1 that are distributed on the three grades. By following the students in the second year, we see that the student 201 :

student 300 are out of the experiment, because they only attended one year. similarly student 401:500 they attended only one year (year 3). Thus, the table contains the five groups: Group 1 contains 100 students attended the three years, group 2 and group 4 contains 200 students that attended two years and group 3 and group 5 contains 200 students attaining only one year. Therefore, the total observations of each school is 900 and the total observation of the data are 27,000.

I also distributed the fifteen teachers of each school on the three groups such that each grade has five teachers. For the purpose of cross-classification, I randomized the teachers on the students. That is, five students may have the same teacher in the first year and five different teachers in the second year or vice versa. I also randomized the teacher's pattern to be different across the 30 schools.

The simulation model structure

As I mentioned above, I designed the model to have three levels including cross-classification between the student and the teacher in level two. Equation 3.1 describes level-1 of the model.

Level-1:

$$Y_{t(ij)k} = \beta_{0(ij)k} + \beta_{1(ij)k}X_{t(ij)k} + e_{t(ij)k}, \tag{3.1}$$

where,

t is the index of the time where $t = 1, 2, 3$,

i is the index of student and $i = 1, \ldots, 500$,

j is the index of the teacher and $j = 1, \ldots, 15$,

k is the index of the school and $k = 1, \ldots 30$,

$y_{t(ij)k}$ is the test score,

$\beta_{0(ij)k}$ is the initial test score of student ij of level-1,

$\beta_{1(ij)k}$ is the student ij improvement though the time,

$X_{t(ij)k}$ is the time predictor and

$\epsilon_{t(ij)k}$ is the random effect of level-1 and $\epsilon \sim N(0, \sigma^2)$.

Level 2 contains the students and teachers characteristics and explains the cross-classification between them.

$$\beta_{0(ij)k} = \gamma_{00k} + (\gamma_{01ik} + c_{1jk})W_{1ik} + \gamma_{02jk}W_{2jk} + b_{0ik} + c_{0jk} \tag{3.2a}$$

$$\beta_{1(ij)k} = \gamma_{10k} \tag{3.2b}$$

where,

48

γ_{00k} is the group mean,

γ_{02ik} is the fixed effect coefficient of W_{1ik},

W_{1ik} is the student categorical variable,

γ_{02jk} is the fixed effect coefficient of W_{2jk},

W_{2jk} is a continuous predictor

γ_{10k} is the improvement of student ijk,

b_{0ik} is the student's random effect,

c_{0jk} is the teacher's random effect ,

c_{1jk} is the random effect of the teacher on the student achievement.

where

$$b_{0ik} \sim N(0, \tau_b)$$

and

$$\begin{pmatrix} c_{0jk} \\ c_{1jk} \end{pmatrix} \sim N(0, \boldsymbol{\tau_c^2})$$

and

$$\boldsymbol{\tau_c^2} = \begin{pmatrix} \tau_{c00}^2 & \tau_{c01}^2 \\ \tau_{c10}^2 & \tau_{c11}^2 \end{pmatrix}$$

and Level-3:

$$\gamma_{000k} = \delta_{000} + \delta_{0001} Z_k, \tag{3.3a}$$

$$\gamma_{01ik} = \delta_{010} + d_{01jk}, \tag{3.3b}$$

$$\gamma_{02jk} = \delta_{020}, \tag{3.3c}$$

$$\gamma_{100k} = \delta_{100} + d_{100k}. \tag{3.3d}$$

where

δ_{000} is the grand mean,

δ_{001} is the fixed slope of the predictor Z,

Z is the school variable ,

δ_{010} is the fixed effect of the $W1$,

δ_{020} is the fixed effect of $W2$,

γ_{100} is the fixed effect of X

d_{01jk} is the random effect of the student at the school level,

d_{100k} is the random effect of the X at the school level,

where

$$\begin{pmatrix} d_{01jk} \\ d_{10jk} \end{pmatrix} \sim N(0, \mathbf{\Phi^2})$$

and

$$\mathbf{\Phi^2} = \begin{pmatrix} \Phi_{00}^2 & 0 \\ 0 & \Phi_{11}^2 \end{pmatrix}$$

50

Thus, the mixed model becomes:

$$Y_{t(ij)k} = \delta_{000} + e_{t(ij)k} + b_{0ik} + c_{0jk} + \delta_{001}Z_k$$
$$+ (\delta_{010} + c_{1jk} + d_{01k})W1_{ik} + \delta_{020}W2_{ik} \qquad (3.4)$$
$$+ (\delta_{100} + d_{10k})X_{t(ij)k} + \epsilon_{t(ij)k}$$

For refining the results, I simulated the data fifty times such that the response variable $Y_{t(ij)k}$ variates based on the variances of $(\epsilon, b_0, c_0, c_1, d_{01}$ and $d_{10})$. I used the R software package to simulate the data.

The choices of fixed and random effects

In order to create the fifty simulations in R, I chose simple values for the fixed and random effects after several trials to have a simple three-level-cross-classified model so that lme4 can perform the model and has accurate estimates for the model's coefficients. Table 3 shows the selected values for the simulation.

Table 3: The True Fixed and Random Coefficients

Fixed Effects		Random Effects	
Parameter	value	Parameter	value
δ_{000}	3.0	σ	0.03
δ_{001}	8.0	τ_{st}	0.05
δ_{010}	3.5	τ_{tch00}	0.1
δ_{020}	0.9	τ_{tch11}	0.17
δ_{100}	2.0	ϕ_{00}	0.05
		ϕ_{11}	0.02

Therefore, Equation 3.4 becomes

$$Y_{t(ij)k} = 3 + e_{t(ij)k} + b_{0ik} + c_{0jk} + 8Z_k$$
$$+ (3.5 + c_{1jk} + d_{01k})W1_{ik} + 0.9W2_{ik} \tag{3.5}$$
$$+ (2 + d_{10k})X_{t(ij)k}$$

where,

$$\epsilon_{tijk} \sim (0, (0.03)^2),$$

$$b_{0ik} \sim N(0, (0.05)^2),$$

$$\begin{pmatrix} c_{0jk} \\ c_{1jk} \end{pmatrix} \sim N(0, \tau_c^2),$$

and

$$\tau_c^2 = \begin{pmatrix} (0.1)^2 & 0 \\ 0 & (0.17)^2 \end{pmatrix},$$

$$\begin{pmatrix} d_{01jk} \\ d_{10jk} \end{pmatrix} \sim N(0, \Phi^2)$$

and

$$\Phi^2 = \begin{pmatrix} (0.05)^2 & 0 \\ 0 & (0.02)^2 \end{pmatrix}$$

In order to check the model's coefficient estimates accuracy, I computed the mean square error and the relative bias as shown in Equations 3.6, 3.7

$$\text{MSE}(\theta) = \frac{1}{n} \sum_{i=1}^{N} (\hat{\theta}_i - \theta)^2 \tag{3.6}$$

52

$$\text{Relative bias}(\theta) = \frac{(\hat{\theta}_i - \theta)}{\theta} \tag{3.7}$$

where

θ is the model coefficient

$\hat{\theta}$ is the model coefficient estimate

N is the number of the simulations

Based on the data design and the coefficients choices, several questions must be answers such as

- the effect of the teacher variances on the model's estimates.

- The correlation between the teacher's variances c_0 and c_1 and its effect on the model's estimates.

The Effects of Teacher's Variances on The Model's Estimates

Different teacher's variations

In order to investigate the above studies, I have simulated eight different types of simulations such that each type contains 50 simulations. Since my goal is to study the effect of the random effects specially the cross-classified random variations, I kept the fixed coefficient, level-1 error and student error and level-3 variation values as the same in the first 50 simulations. For the purpose of understanding the effect of the correlation between teacher's variances. I split the eight simulation-types into two groups: one is with a small correlation value and the other group is with a large value. For each group, I selected two values for each variance (τ_{00} and τ_{11}) one is small and the other is large. Table 4 shows the selected values of $\tau_{tch00}, \tau_{tch11}$ and their correlation for each simulation-type.

53

Table 4: The Teacher's Variance in Eight Simulation-Types

Type	τ_{tch00}	τ_{tch11}	Correlation
Type 1	0.10	0.17	-0.03
Type 2	0.10	0.87	-0.03
Type 3	0.92	0.17	-0.03
Type 4	0.92	0.87	-0.03
Type 5	0.10	0.17	-0.75
Type 6	0.10	0.87	-0.75
Type 7	0.92	0.17	-0.75
Type 8	0.92	0.87	-0.75

- **Why negative correlation:** Notice that the teacher's variations (c_0, c_1) are no longer independent but correlated. That is, if c_1 is small, then the test score of students with ELL is closer to the test score of students without ELL. As a results there will be a gap in test score between classes meaning that the variation between classes are larger and c_0 is large. On the other hand, if the c_1 is large, then the diffidence in the test score between the students with ELL and the students without ELL is large meaning that the gap between the classes is smaller. That is, c_0 is small. Based on this concept, I chose the correlation to be negative.

I assigned $\rho = -0.03$ for the first four simulation-types and $\rho = -0.75$ to the second four simulation-types. Based on the following correlation formula in Equation 3.8 (Wackerly et al., 2008)

$$\rho(c_0, c_1) = \frac{\text{cov}(c_0, c_1)}{\sqrt{\text{var}(c_0)\text{var}(c_1)}}, \tag{3.8}$$

The variance-covariance matrix τ_c^2 of the eight types as follows,

Type 1

$$\tau_c^2 = \begin{pmatrix} (0.1)^2 & -0.00051 \\ -0.00051 & (0.17)^2 \end{pmatrix},$$

Type 2

$$\tau_c^2 = \begin{pmatrix} (0.1)^2 & -0.00261 \\ -0.00261 & (0.87)^2 \end{pmatrix},$$

Type 3

$$\tau_c^2 = \begin{pmatrix} (0.92)^2 & -0.004692 \\ -0.004692 & (0.17)^2 \end{pmatrix},$$

Type 4

$$\tau_c^2 = \begin{pmatrix} (0.93)^2 & -0.024012 \\ -0.024012 & (0.87)^2 \end{pmatrix},$$

Type 5

$$\tau_c^2 = \begin{pmatrix} (0.1)^2 & -0.01275 \\ -0.01275 & (0.17)^2 \end{pmatrix},$$

Type 6

$$\tau_c^2 = \begin{pmatrix} (0.1)^2 & -0.06525 \\ -0.06525 & (0.87)^2 \end{pmatrix},$$

Type 7

$$\tau_c^2 = \begin{pmatrix} (0.92)^2 & -0.1173 \\ -0.1173 & (0.17)^2 \end{pmatrix},$$

Type 8

$$\tau_c^2 = \begin{pmatrix} (0.92)^2 & -0.6003 \\ -0.6003 & (0.87)^2 \end{pmatrix}.$$

The eight simulation-types give a greater chance to check many aspects in cross-classified model such as the accuracy of fixed and random effect estimates, the correlation between c_0 and c_1, the effects of level one and level two variations on level-3 random effect estimate. The matter of interest is particularly the relationship between the cross classified random effect in level-two with the corresponding one in level-three.

The Package LME4 Efficiency

The first step I have done is to check the efficiency of lme4. McCoach et al. (2018) examined the ability of lme4 in fitting two-level hierarchical linear model. In this step I extended the capability test to include the change of c_0 and c_1. Based on the previous results given by the first 50 simulations, I expect that lmer is capable to perform the model with the random effect changes.

Coefficient Estimates Accuracy

The advantage of simulating the data are having a measure of coefficient estimate. By simulating the data, I provide the true coefficient values that assists determining how close the model's coefficients from their true values. In our desired

model, there are eleven parameters (five fixed and six random). There is a great chance to get clear results, because we have large, balanced and complete data. Additionally. In order to understand the estimate accuracy in term of c_0 and c_1 change, I performed the following tasks:

- In each simulation-type, I computed the average and the standard deviation of each parameter's estimate,

- per each parameter, I compared the MSE and the relative bias through the eight simulation types by plotting the histogram of each parameter,

- I performed the previous task based on the correlation in order to determine the effect of the correlation between c_0 and c_1 on the model's coefficient estimates.

The Effect of Teacher Variances on Level-Three Variances

Based on the results of the eight simulation-types, I found that the shape of level three random effect estimate $\hat{\phi}_{00}$ is right skewed in a way that does not follow neither normal nor chi square distributions. As a result, there was a need to investigate this problem and understand whether there is a relationship between the level-two variance τ_{tch11} and level-three random effect estimate $\hat{\phi}_{00}$ since both c_1 and $d01$ are coefficients of the cross-classified predictor W_1 in level two. In order to find out that effect, I simulated more models.

Firstly, I found a gap between the two different values for τ_{11} in the first eight simulation-types (0.17, 0.87) so, I added a third value of c_1 where $\tau_{11} = 0.63$, because I needed to understand the behavior of $\hat{\phi}_{00}$ in term of the change of τ_{00}. Since the focus in on the behavior of $\hat{\phi}_{00}$, I had to chose many values for ϕ_{00} (six different values of ϕ_{00} per each one value of τ_{tch00}) and remained the rest of the model's parameters as shown in Table 5:

Table 5: The Unchanged Parameters Values

Parameter	δ_{000}	δ_{001}	δ_{010}	δ_{020}	δ_{100}	σ	τ_{st}	τ_{tch00}	ϕ_{11}
Value	3.0	8.0	3.5	0.9	2.0	0.03	0.05	0.1	0.02

Notice that the that $\phi_{00} = 0.05$ was used already selected twice with $\tau_{tch11} = 0.17$ and $\tau_{tch11} = 0.87$ in the first eight simulation-types, so I just needed to simulate sixteen more simulation-types. Table 6 shows each of the sixteen simulation-types with their τ_{tch11} and ϕ_{00} values.

Table 6: The New ϕ_{00} Values Corresponding to τ_{tch11} Values

Type	τ_{tch11}	ϕ_{00}	Type	τ_{tch11}	ϕ_{00}
Type 9	0.17	0.000	Type 17	0.63	0.400
Type 10	0.17	0.010	Type 18	0.63	0.640
Type 11	0.17	0.014	Type 19	0.63	0.860
Type 12	0.17	0.025	Type 20	0.87	0.073
Type 13	0.17	0.150	Type 21	0.87	0.150
Type 14	0.63	0.000	Type 22	0.87	0.300
Type 15	0.63	0.053	Type 23	0.87	0.500
Type 16	0.63	0.200	Type 24	0.87	0.750

The main goal of simulating more data are to understand the relationship between c_1 and $d_0 1$. In order to accomplish that goal, I ran the total of 1,200 models and performed the following:

- for each value of τ_{tch11}, I plot six histograms in one shape to shows the change of the $\hat{\phi}_{00}$ shape with the increase of ϕ_{00},

- test the shape normality by using Shapiro-Wilk test (Shapiro and Wilk, 1965) by using the test formula:

$$W = \Big(\sum_{i=1}^{n} a_i y_i \Big)^2 / \sum_{i=1}^{n} \big(y_i - \bar{y} \big)^2 \tag{3.9}$$

where

y_i is the ith order statistics

$\bar{y}$ is the sample mean

a_i $= (a_1, \ldots, a_n) = \dfrac{m^T V^{-1}}{(m^T V^{-1} V^{-1} m)^{1/2}}$

m $= (m_1, \ldots, m_n)$ are the expected value of the order statistics of

    independent and identically distributed random variables

V is the covariance matrix of these order statistics (Razali et al., 2011).

We test the normality by setting the null hypothesis,

H_o :   The sample comes from a normal population,

H_a :   The sample does not come from a normal population,

If the p-value is less than 0.05, we reject the hypothesis test and conclude that the sample does not come from a normal distribution.

Further study to understand the effect of the level-two cross-classified random effect (c_1) on the level-three cross-classified random effect (d_{01}) is to compute the the ratio as Raundenbush and Bryk (2002) stated

$$\text{reliability}(\phi_{00}) = \frac{\phi_{00}^2}{\tau_{tch11}^2 + \phi_{00}^2} \tag{3.10}$$

Based on that number, I performed the following tasks

- Find the ratio number when the shape of $\hat{\phi}_{00}$ begins to change from right skewed to symmetric by testing the normality with Shapiro-Wilk test,

- check the relationship between the ratio numbers and the normality.

One of the challenges I have experienced with the study of the behavior of $\hat{\phi}_{00}$ was the model convergence and singularity warning messages.

The Model Convergence and Singularity Effects

Although I have provided complete, balanced and large data, the package lme4 showed convergence or singularity warning messages while fitting the 1,200 models. One of the tasks I had to do, before stating the model's results, was to check the effect(s) of the warning message on the estimate's accuracy, because only 20% of the models converged, while 80% of the models I fit had convergence or singularity warning messages.

For most of the models output, the package lme4 showed two types of convergence model warning message:

The first warning message shows that the max of the gradient exceeded the tolerance which is 0.002.

```
    Warning message:
In checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
  Model failed to converge with max|grad| = 0.00287919
  (tol = 0.002, component 1)
```

The second message states two warnings: the first is similar to the first warning message, while the second states that there is one negative eigenvalue in the Hessian matrix.

```
    Warning messages:
1: In checkConv(attr(opt, "derivs"), opt$par, ctrl =
control$checkConv,:
  unable to evaluate scaled gradient
2: In checkConv(attr(opt, "derivs"), opt$par, ctrl =
control$checkConv,:
  Model failed to converge: degenerate  Hessian with 1 negative
```

```
eigenvalues
```

These two messages refer to the complexity of the model because the model includes six random effects. The lme4 takes long time to scale the estimated gradient at the estimate appropriately, McCoach et al. (2018) clarified that "lme4 scales gradients by the inverse Cholesky factor of the Hessian. The disadvantage of this approach is that it requires estimation of the Hessian" (p.12). The lme4 package's author Bates et al. (2015) suggested using alternate estimation methods such as the Nelder-Mead simplex algorithm (Nedler and Mead, 1965).

I have tried different methods based on the package's community. However, our model was more complex. The method I have used was how much these warning messages affect the model's estimates.

The package lme4 also showed a singularity warning message, when I chose a small value (close to zero)for ϕ_{00}. The model's estimates always has a very small random effect that is close to zero. Based on the results, I have two methods: Firstly, I increased the true value of the random effect and secondly, I removed the small random effect from the model. The following line shows the singularity warning message

```
boundary (singular) fit: see ?isSingular
```

Although the package states a warning message for almost each model, the model's coefficient estimates were so close to their true values. As a results, I had to perform more tests in order to make the right decision whether to consider these warning messages or ignore them.

The effect of convergence and singularity messages on $\hat{\phi}_{00}$

In order to monitor the effect of the warning messages on the $\hat{\phi}_{00}$, I selected simulation-type 8 where $\tau_{tch11} = 0.87$ and $\phi_{00} = 0.05$ and simulation-type 10 where

$\tau_{tch11} = 0.17$ and $\phi_{00} = 0.01$ and fit the 100 models. I split each of the two simulation-types into three groups based on the issue of each model in the group (convergent, non convergent or singular) then I performed the following tasks:

- compute the mean and the standard deviation of $\hat{\phi}_{00}$ in each convergent model coefficient estimates,

- compute the mean and the standard deviation of $\hat{\phi}_{00}$ in each non-convergent model coefficient estimates,

- compute the mean and the standard deviation of $\hat{\phi}_{00}$ in each singular model coefficient estimates.

- compare the mean and the standard deviation of $\hat{\phi}_{00}$ the three cases (convergent, non-convergent and singular model) in each simulation-type,

- plot a scatter plot containing the estimate of six groups to compare the estimate of ϕ_{00}.

Based on the results of the above tasks, I did not find any difference in $\hat{\phi}_{00}$ between the convergent, non-convergent or singular model. However, that was insufficient to ignore the warning messages. Thus, there is also a need to check the fixed effect estimates in term of the model's issues.

Non-convergent and singular models in HLM7

In order to ensure that the convergence and singularity issues do not affect the model's estimates, I decided to test these issues by utilizing another software. Because HLM7 is one of the software packages that have "auto-fix" convergence (McCoach et al., 2018), I selected that package to compare with the estimate of the cross-classified predictor W_1 coefficients $(\hat{\delta}_{010}, \hat{\tau}_{tch11})$. Based on the results given in Chapter 4 Table 15, I selected eight models to compare each model's estimates in

the packages lme4 and HLM7 respectively. Table 7 explains the eight models selection based on the model issue, τ_{tch11} and ϕ_{00} values.

Table 7: Compare The Convergence and Singularity in *HLM7*

Model	Model issue	τ_{tch11}	ϕ_{00}	Model	Model issue	τ_{tch11}	ϕ_{00}
1	convergent	0.17	0.01	5	non-convergent	0.17	0.15
2	non-convergent	0.17	0.01	6	convergent	0.87	0.05
3	singular	0.17	0.01	7	non-convergent	0.87	0.05
4	convergent	0.17	0.15	8	singular	0.87	0.05

The Fixed Effects in Non-Convergent and Singular Models

Further study is to understand the effect of the model's warning messages on the estimate of the fixed effect coefficients. In order to perform that test, I selected the same two simulation-types and split each simulation-type into three groups based on the model's issue (convergent, non-convergent or singular) as I have done in the previous section. I performed the following tasks on each of the three groups

- compute the mean and the standard deviation of the fixed effect estimates of the convergent models in each of the two simulation-types.,

- compute the mean and the standard deviation of the fixed effect estimates of the non-convergent models in each of the two simulation-types,

- compute the mean and the standard deviation of the fixed effect estimates of the singular models in each of the two simulation-types,

- plot a scatter plot for each fixed effect estimate to compare the fixed effect estimate in term of the model's issue.

63

The Effect of ϕ_{00} on The Model Estimation

I have covered the behavior of $\hat{\phi}_{00}$, the relationship between τ_{tch11} and the shape of $\hat{\phi}_{00}$ as they are the variation of the c_1 and d_{01} respectively. It was necessary to understand the this relationship since both c_1 and d_{01} are the random coefficients of the cross-classified variable W_1. Furthermore, I explained the effects of model warning message on $\hat{\phi}_{00}$ and the estimate of the fixed effects in order to confirm the accuracy of the model results. Now, it is interesting to know the effect of the random variable d_{01} on the model's estimate. In order to check that effect, I refit the desired model without W_1 as a random effect in level-three. That is,

Level-3:

$$\gamma_{000k} = \delta_{000} + \delta_{0001}Z_k, \tag{3.11a}$$

$$\gamma_{01ik} = \delta_{010}, \tag{3.11b}$$

$$\gamma_{02jk} = \delta_{020}, \tag{3.11c}$$

$$\gamma_{100k} = \delta_{100} + d_{100k}. \tag{3.11d}$$

Thus, the mixed model becomes:

$$
\begin{aligned}
Y_{t(ij)k} = {} & \delta_{000} + e_{t(ij)k} + b_{0ik} + c_{0jk} + \delta_{001}Z_k \\
& + (\delta_{010} + c_{1jk})W1_{ik} + \delta_{020}W2_{ik} \\
& + (\delta_{100} + d_{10k})X_{t(ij)k}
\end{aligned}
\tag{3.12}
$$

and the model in R is

lmer(y$\sim$ X+W1+W2+Z+(1|std)+(W1|tch)+(0+X||sch), data =data1)

Notice that the predictor W1 is no longer a school's random effect. In order to determine the effect of the random effect c_1 with the absence of the level-three

random effect d_{01}, fit the model in three simulation-types such that each simulation-type had different τ_{tch11}. I performed the following tasks:

- compute the mean and the standard deviation of each of the fixed and random effect estimate in each of the three groups.

- compare the outcome of task one with those with the existence of d_{01},

- plot each model parameter's estimate in term of d_{01},

- determine the effect of the random variable d_{01} based on the results given in task two and three.

The above study opened the gate to investigate the effect of c_{01} on the model estimate. For that purpose, I ran the previous model without c_1. That is, level 2

$$\beta_{0(ij)k} = \gamma_{00k} + \gamma_{01ik}W_{1ik} + \gamma_{02jk}W_{2jk} + b_{0ik} + c_{0jk} \tag{3.13a}$$

$$\beta_{1(ij)k} = \gamma_{10k} \tag{3.13b}$$

and level 3

$$\gamma_{000k} = \delta_{000} + \delta_{0001}Z_k, \tag{3.14a}$$

$$\gamma_{01ik} = \delta_{010}, \tag{3.14b}$$

$$\gamma_{02jk} = \delta_{020}, \tag{3.14c}$$

$$\gamma_{100k} = \delta_{100} + d_{100k}. \tag{3.14d}$$

Thus, the mixed model becomes:

$$Y_{t(ij)k} = \delta_{000} + e_{t(ij)k} + b_{0ik} + c_{0jk} + \delta_{001}Z_k$$
$$+ \delta_{010}W1_{ik} + \delta_{020}W2_{ik} \qquad (3.15)$$
$$+ (\delta_{100} + d_{10k})X_{t(ij)k}$$

and the model in R is

lmer(y~ X+W1+W2+Z+(1|std)+(1|tch)+(0+X||sch), data =data1)

Based on the results given by fitting the 100 models, I compared the estimate of δ_{010} in the three model types. I used this comparison to understand the effect of the cross-classified random effects on the model's parameters estimates.

After finishing these methods, I collected the results in tables and graphs. Chapter 4 explains the results I have found through the methodology.

IV.   RESULTS

Simulation Results

Based on the 50 simulations I have mentioned in Chapter 3, page 52 the package lme4 can perform a three-level-cross-classified model. The lme4 package output shows that the package can provides a list of fixed effect estimates with their std.error, degree of freedom and t-value and the correlation between the fixed effect parameters. Additionally, lme4 provides an estimate of each random effect with the correlation value if the random variable is correlated with another. In addition, the package shows each random variable's level. The output of the first model is listed below to show the package lme4 results.

```
Linear mixed model fit by REML. t-tests use Satterthwaite's method
['lmerModLmerTest']
Formula: y1 ~ X1 + W1 + W2 + Z1 + (1 | std) + (W1 | tch) +
(0 + X1 + W1 ||sch)
   Data: data


REML criterion at convergence: -81612.4


Scaled residuals:
     Min       1Q   Median       3Q      Max
-3.00888 -0.46966  0.00088  0.47022  2.88741


Random effects:
 Groups   Name          Variance  Std.Dev.  Corr
```

```
std        (Intercept) 2.501e-03 5.001e-02

tch        (Intercept) 9.979e-03 9.990e-02

           W1              2.845e-02 1.687e-01 -0.05

sch        X1              4.323e-04 2.079e-02

sch.1      W1              2.069e-09 4.549e-05

Residual                   8.959e-04 2.993e-02
```

Number of obs: 27000, groups: std, 15000; tch, 450; sch, 30

Fixed effects:

```
              Estimate Std. Error        df t value Pr(>|t|)

(Intercept) 2.976e+00  2.575e-02 4.488e+02  115.58   <2e-16 ***

X1          2.006e+00  3.814e-03 2.900e+01  525.94   <2e-16 ***

W1          3.510e+00  8.003e-03 4.557e+02  438.57   <2e-16 ***

W2          9.080e-01  1.066e-02 4.470e+02   85.16   <2e-16 ***

Z1          8.010e+00  9.639e-03 4.772e+02  831.02   <2e-16 ***

---
```

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:

```
   (Intr) X1     W1     W2

X1 -0.002

W1 -0.012  0.000

W2 -0.967  0.000  0.000

Z1 -0.231  0.000  0.000  0.059
```

The R output shows that the package lme4 lists the random effect first based on each level, the number of each group (e.g., std, 15000) followed by the fixed effect including(the estimate, the standard error, the degree of freedom and

the t-value) and finally, the correlation of the fixed effects.

The t-value does not come with the package lme4. The package "lmertest" must be installed with the package "lme4" in order to display the t-value.

The Package Estimate Accuracy

Measuring the package's accuracy was not difficult, because we already had the model's coefficient true values. The results of fitting the one of the first fifty models are listed in Table 8.

Table 8: The Fixed and Random Effect Estimates in One Model

Fixed Effects			Random Effects		
Parameter	true	Estimate	Parameter	true	Estimate
δ_{000}	3.0	2.9761	σ	0.03	0.0299
δ_{001}	8.0	8.0103	τ_{st}	0.05	0.0500
δ_{010}	3.5	3.5099	τ_{tch00}	0.10	0.0999
δ_{020}	0.9	0.9079	τ_{tch11}	0.17	0.1636
δ_{100}	2.0	2.0059	ϕ_{00}	0.05	0.0422
			ϕ_{11}	0.02	0.0208

Based on the results listed in the lme4 output and Table 8, I found the following:

- **The fixed effects estimates:** The estimate of all the fixed effects are close to the true values. For example, $\hat{\delta}_{000}$ is 2.9761 in which is close to the $\delta_{000} = 3$. That is, the estimate of the grand mean of the test score is close to its true value. Similarly the coefficients $\delta_{001}, \delta_{010}, \delta_{020}$ and δ_{100} also are close to their true values.

- **The random effects:** The estimate of most of the random effect coefficients are also close to the true values. For example, $\hat{\sigma} = 0.0299$ while the true value $\sigma = 0.03$. Similarly, the estimate of level-two random effects $\hat{\tau}_{std00}$, $\hat{\tau}_{tch00}$ and

level-three random effects $\hat{\phi}_{11}$. However, the estimate of the cross-classified random effects are not that close to the true values. For example, in level-two, $\hat{\tau}_{tch11} = 0.1636$ is slightly far from the true value $\tau_{tch11} = 0.17$, and in level-three $\hat{\phi}_{00} = 0.05$ is a little farther from its true value $\phi_{00} = 0.0422$.

Over all, the results of one model are fine. However, they are insufficient to determine the accuracy of the package in estimating the model parameters. As a result, I fitted the fifty models and averaged each parameter's estimate and each parameter's standard deviation. Additionally, I computed the mean square error (MSE) for each of the model's estimates as shown in Table 9 where,

$$\text{avg}((\hat{\theta})) = \frac{1}{50} \sum_{i=1}^{50} \hat{\theta}_i \tag{4.1a}$$

$$\text{avg}(\text{sd}(\hat{\theta})) = \frac{1}{50} \sum_{i=1}^{50} sd(\hat{\theta}_i) \tag{4.1b}$$

$$\text{MSE}(\hat{\theta}) = \frac{1}{50} \sum_{i=1}^{50} (\hat{\theta}_i - \theta)^2 \tag{4.1c}$$

Table 9: The Average of Fixed and Random Effect Estimates

	Fixed Effects Estimate				Random Effects Estimate		
θ	$\text{avg}(\hat{\theta})$	$\text{avg}(\text{sd}(\hat{\theta}))$	$\text{MSE}(\hat{\theta})$	θ	$\text{avg}(\hat{\theta})$	$\text{avg}(\text{sd}(\hat{\theta}))$	$\text{MSE}(\hat{\theta})$
δ_{000}	2.9972	0.0278	0.00080	σ	0.0300	0.0002	3.43e-08
δ_{001}	8.0015	0.0094	8.86e-05	τ_{st}	0.0500	0.0003	1.09e-07
δ_{010}	3.5000	0.0134	0.00020	τ_{tch00}	0.0997	0.0034	1.18e-05
δ_{020}	0.9008	0.0113	0.00010	τ_{tch11}	0.1724	0.0063	4.45e-05
δ_{100}	2.0004	0.0035	1.24e-05	ϕ_{00}	0.0457	0.0143	0.0002
				ϕ_{11}	0.0197	0.0024	5.60e-06

The fixed effects analysis in fifty simulations

Based on the listed results in Table 9, the average of the fixed effects estimate is close to their true values meaning that most of the estimate of fixed

effects in the fifty models are close to their true values. on the other hand, the standard deviation of each fixed effect parameter varies. For example, the average of the $\text{sd}(\hat{\delta}_{000})$ is 0.0278, while the average of the $\text{sd}(\hat{\delta}_{100})$ is 0.0035 meaning that the confidence interval of $\text{sd}(\hat{\delta}_{100})$ is smaller than the confidence interval of $\text{sd}(\hat{\delta}_{000})$. In order to clarify that error, I computed the MSE of each parameters.

Based on the Mean square error results, we see that the $\text{MSE}(\delta_{000})$ is greater than the $\text{MSE}(\delta_{100})$. Overall, since the MSE of all the fixed effect estimates are almost zero, I conclude that the fixed effect estimates in this 50 simulations are precise and accurate.

The random effects analysis in fifty simulations

Based on the results in Table 9, the estimate of level-one random effect $(\hat{\sigma}) = 0.03$ which is equaled to the $\sigma = 0.03$. By looking at mean square error of σ ($\text{MSE}(\sigma)$= 3.43e-08), the error is almost zero. Level-two random effects also have precise estimates, because their MSE numbers that are close to zero. On the other hand, for level-three random effects, $\hat{\phi}_{11}$ has a precise estimate, while $\hat{\phi}_{00}$ is slightly far from its true value.

Overall, the estimates are precisely good. However, the true random effect values are very small to state the accuracy of the model in lme4 package. In order to confirm the results, I have simulated eight simulation-types including 400 simulations as I mentioned in Chapter 3.

The Estimate Accuracy with Different Simulations Conditions

As I mentioned above, we needed to examine more model conditions in order to test lme4 package accuracy in assessing the three-level-cross-classified models. I have simulated eight simulation-types, as mentioned in Chapter 3 and computed the average and the standard deviation of each parameter's estimate.

71

The fixed effect estimates in the eight simulation-types

As I computed the statistics of the first fifty simulations, I did the same for the eight simulation-types. Notice that the first simulation-type is the first fifty simulations that were mentioned in the previous section. Since the matter of concern is the cross-classified model, I focused on the coefficients of the cross-classified predictor W_1 ($\delta_{010}, \tau_{tch11}$ and ϕ_{00}). Table 11 compares the three coefficient estimates within the eight simulation-types

Based on the given results in Table 11, we see that when the correlation is -0.03, the average of $\hat{\delta}_{010} = 3.500$ and when the correlation is -0.75, the average of of $\hat{\delta}_{010} = 3.5014$. Similarly, the estimate of $\tau_{tch11} = 0.\hat{0}063, 0.0065$ with the correlation values -0.03 and -0.075 respectively. Figure 8, 11 and 12 confirm the results given in Table 11. So, we can conclude

- the correlation between τ_{tch00} and τ_{tch11} does not affect the estimate of model's parameter estimates specially the coefficients of the cross-classified predictor W_1,

- most of the estimates are satisfying except the estimate of level-three cross-classified random effect $\hat{\phi}_{00}$ that has a right skewed shape as shown in Figure 12

The Relationship Between Level Two and Level Three Variances

Since the matter of interest is the cross-classified models, we are interested in understanding the relationship between level-two random effect τ_{tch11} and level-three random effect ϕ_{00}.

Based on the sixteen simulations I have created and mentioned in Chapter 3, I plot three histograms with three different choices of τ_{tch11} 0.17, 0.63 and 0.87

Table 10: The Teacher's Variance in Eight Simulation-Types

Type	δ_{000}	δ_{001}	δ_{010}	δ_{020}	δ_{100}	σ	τ_{st00}	τ_{tch00}	τ_{tch11}	ϕ_{00}	ϕ_{11}	$\mathrm{corr}(\tau_{tch00}, \tau_{tch11})$
Type 1	3.00	8.00	3.50	0.90	2.00	0.03	0.05	0.10	0.17	0.05	0.02	-0.03
Type 2	3.00	8.00	3.50	0.90	2.00	0.03	0.05	0.10	0.87	0.05	0.02	-0.03
Type 3	3.00	8.00	3.50	0.90	2.00	0.03	0.05	0.92	0.17	0.05	0.02	-0.03
Type 4	3.00	8.00	3.50	0.90	2.00	0.03	0.05	0.92	0.87	0.05	0.02	-0.03
Type 5	3.00	8.00	3.50	0.90	2.00	0.03	0.05	0.10	0.17	0.05	0.02	-0.75
Type 6	3.00	8.00	3.50	0.90	2.00	0.03	0.05	010	0.87	0.05	0.02	-0.75
Type 7	3.00	8.00	3.50	0.90	2.00	0.03	0.05	0.92	0.17	0.05	0.02	-0.75
Type 8	3.00	8.00	3.50	0.90	2.00	0.03	0.05	0.92	0.87	0.05	0.02	-0.75

Table 11: The Average of Fixed and Random Effect Estimates

sim-type	$\hat{\delta}_{010}$			$\hat{\tau}_{tch11}$			$\hat{\phi}_{00}$		
	avg($\hat{\theta}$)	avg(sd($\hat{\theta}$))	MSE($\hat{\theta}$)	avg($\hat{\theta}$)	avg(sd($\hat{\theta}$))	MSE($\hat{\theta}$)	avg($\hat{\theta}$)	avg(sd($\hat{\theta}$))	MSE($\hat{\theta}$)
type-1	3.5000	0.0134	2.28e-08	0.1724	0.0063	2.97e-04	0.0457	0.01431	9.05e-04
type-2	3.4950	0.0391	1.27e-03	0.8623	0.0294	2.96e-03	0.0480	0.0595	2.06e-04
type-3	3.4992	0.0140	3.01e-05	0.1725	0.0070	3.14e-04	0.0494	0.0124	1.88e-05
type-4	3.5019	0.039	1.77e-04	0.8602	0.0318	4.81e-03	0.0356	0.0436	1.04e-02
type-5	3.5014	0.0123	1.03e-04	0.1742	0.0065	8.95e-04	0.0450	0.0121	1.24e-03
type-6	3.5062	0.0401	1.89e-03	0.8678	0.0264	2.52e-04	0.0576	0.0447	2.90e-03
type-7	3.4980	0.0120	2.02e-04	0.1723	0.0060	2.77e-04	0.0490	0.0083	4.76e-05
type-8	3.5024	0.0425	2.86e-04	0.8688	0.0276	6.75e-05	0.0597	0.0919	4.68e-03

respectively. For each of the three values, I chose six different values of ϕ_{00} in order to understand the shape of $\hat{\phi}_{00}$ histogram in term of change of both ϕ_{00} and τ_{tch11}.

The behavior of $\hat{\phi}_{00}$ when τ_{tch11} is small

Based on the plot in Figure 4 that explains the shape of $\hat{\phi}_{00}$ with $\tau_{tch11} = 0.17$, I found the following:

- the histogram shape is right skewed when $\phi_{00} = 0$,

- The shape turns symmetric with normal distribution as ϕ_{00} gets larger (e.g., $\phi_{00} = 0.15$).

On the other hand, the standard deviation behaves differently with the change of the value of ϕ_{00}:

- the standard deviation is One-Way small (the shape is right skewed) when $\phi_{00} = 0$,

- the standard deviation increases when $\phi_{00} = 0.01$,

- The standard deviation gets larger, when ϕ_{00} increases.

It is bizarre that the shape does not go around zero (the true value of ϕ_{00} when $\phi_{00} = 0$, but goes to the right side (positive side). The probability of the shape's normality was discussed in Chapter 3 and the results are listed at the end of this chapter.

The behavior of $\hat{\phi}_{00}$ when τ_{tch11} is moderate

In the first experiment, I have set $\tau_{tch11} = 0.17$. which is a small value for τ_{tch11}. In this experiment, I monitored the behavior of $\hat{\phi}_{00}$ with a larger value ($\tau_{tch11} = 0.63$).

Based on figure 5, $\hat{\phi}_{00}$ shape shows similar behaviors as in case 1:

75

- the shape is right skewed when $\phi_{00} = 0$,

- the shape starts to be symmetric, when $\phi_{00} = 0.053$,

- the shape symmetry improves when ϕ_{00} increases.

The standard deviation $\hat{\phi}_{00}$ behavior is similar to case 1 meaning the shape of $\hat{\phi}_{00}$ starts with a small standard deviation that gets smaller when $\phi_{00} = 0.053$ then gets larger when the value of ϕ_{00} increases.

The behavior of $\hat{\phi}_{00}$ when τ_{tch11} is large

In order to complete the behavior of $\hat{\phi}_{00}$ in term of the change of ϕ_{00} true value, I had to choose a larger value for the variance of the cross-classified random effect in level-two ($\tau_{tch11} = 0.87$). Figure 6 displays six histogram of six different values of ϕ_{00} where

- the shape is right skewed with outliers when $\phi_{00} = 0$,

- the shape is normal, with the increase of the value of ϕ_{00}.

On the other hand, the histogram of $\hat{\phi}_{00}$ standard deviation is different:

- the standard deviation is small when $\phi_{00} = 0.05$,

- the standard deviation gets larger when $\phi_{00} = 0.073$ unlike the other two cases when the standard deviation gets smaller at the shape breaking point.

Based on the ratio formula

$$Ratio = \frac{\phi_{00}^2}{\tau_{tch11}^2 + \phi_{00}^2}, \tag{4.2}$$

when the ratio gets larger, the shape of $\hat{\phi}_{00}$ becomes closer to the true value with a small standard deviation and symmetric shape.

76

Table 12: Compare $\hat{\phi}_{00}$ Statistics With τ_{tch00}

ϕ_{00}	avg($\hat{\phi}_{00}$)	avg(sd($\hat{\phi}_{00}$))	MSE($\hat{\phi}_{00}$)
	$\tau_{tch00} = 0.17$		
0.000	0.0103	0.0127	0.00532
0.010	0.0129	0.0131	0.00042
0.014	0.0344	0.0067	0.02090
0.025	0.0209	0.0158	0.00083
0.050	0.100	0.0160	0.12600
0.150	0.151	0.0237	0.00001
	$\tau_{tch00} = 0.63$		
0.000	0.0340	0.0453	0.057600
0.053	0.1300	0.0274	0.298000
0.200	0.2000	0.0480	0.000010
0.400	0.4040	0.0609	0.000670
0.640	0.5440	0.0851	0.462000
0.860	0.8600	0.1050	0.000008
	$\tau_{tch00} = 0.87$		
0.050	0.0597	0.0919	0.00468
0.073	0.1800	0.0601	0.57700
0.150	0.1240	0.0849	0.03260
0.300	0.2800	0.0751	0.01940
0.500	0.4950	0.1020	0.00143
0.750	0.7200	0.0904	0.04490

Based on the results given in the Table 12, the estimate of ϕ_{00} is the best when τ_{tch11} value is small and ϕ_{00} value is large.

Test the normality of $\hat{\phi}_{00}$ shape

In order to check the normality of the eighteen shapes, we use the Shapiro-Wilk test. The results of the test are given in Table 13.

Notice that in Table 13, the p-values are less than 0.05 when $\tau_{tch11} = 0.17$

Table 13: Test The Normality of $\hat{\phi}_{00}$ By Shapiro-Wilk Test

τ_{tch11}= 0.17			τ_{tch11}= 0.63			τ_{tch11}= 0.87		
ϕ_{00}	W	p-value	ϕ_{00}	W	p-value	ϕ_{00}	W	p-value
0.000	0.7836	3.7e-07	0.000	0.7570	1.0e-7	0.050	0.5161	1.3e-11
0.010	0.8402	8.4e-06	0.053	0.9913	0.9718	0.073	0.9122	0.0012
0.014	0.9921	0.98260	0.200	0.9771	0.4381	0.150	0.9392	0.01254
0.025	0.9003	0.0005	0.400	0.9790	0.5123	0.300	0.9375	0.0107
0.050	0.9730	0.3050	0.640	0.9902	0.9497	0.500	0.9767	0.4226
0.150	0.9784	0.4861	0.860	0.9807	0.5802	0.750	0.9761	0.4001

and $\phi_{00} = 0, 0.01$, and 0.025 respectively. Therefore, we reject the null hypothesis and conclude that the sample did not come from a normal population. On the other hand, the p-values are greater than 0.05. Thus when $\phi_{00} = 0.014, 0.05$ and 0.015 respectively. Thus, we fail to reject the null hypothesis test. When we fail to reject the hypothesis test, we refer to the histogram as shown in Figure 4a, and conclude that the shape is normally distributed.

By performing the same test with the other two values of τ_{tch11}, the given results in Table 13, and the histogram shapes in Figure 5a and Figure 6a, we conclude that the shape of $\hat{\phi}_{00}$ becomes normal when ϕ_{00} is large.

While general behavior is as predicted, further study is necessary to fully understand the effect of ϕ_{00} and τ_{tch11} on the distribution of $\hat{\phi}_{00}$.

Convergence and Singularity in Lme4

Although I simulated large, complete and balanced data, the package lme4 shows convergence and singularity warning messages. These messages made concerns about the accuracy of the package model estimates. In order to decide whether to consider or ignore these warning messages, I investigated the messages in term of the estimate of the desired random effect $\hat{\phi}_{00}$ and the fixed effect estimates.

The effect of the model's issues on the cross-classified predictor coefficients

In order to investigate the effect of the warning messages, I chose two values of τ_{tch11} (0.17, 0.87) with five τ_{tch11} I selected two ϕ_{00} values as follows

- Group 1: $\tau_{11} = 0.17$ and $\phi_{00} = 0.01$

- Group 2: $\tau_{11} = 0.17$ and $\phi_{00} = 0.15$

- Group 3: $\tau_{11} = 0.87$ and $\phi_{00} = 0.05$ and

- Group 4: $\tau_{11} = 0.87$ and $\phi_{00} = 0.75$.

and for each group, I fit the model and recorded the convergent, non-convergent, and singular models, the results of the experiment was as shown in Table 14

Table 14: The Convergence and Singularity Issues of 200 models

τ_{11}	ϕ_{00}	Convergent	Convergence issue	Singularity issue
0.17	0.01	7	35	8
0.17	0.15	10	40	0
0.87	0.05	6	41	3
0.87	0.75	5	45	0

Based on the results of Table 14, I found that when the value of ϕ_{00} is close to zero (0.01,0.05) the model may experience singularity issues. In order to analyze warning messages concern, I selected the first three groups to assess the estimate of cross-classified predictor W_1 coefficients ($\delta_{010}, \tau_{tch11}$ and ϕ_{00}). Table 15 shows the results.

Compare the Estimates of Non-Convergent and Singular Models in HLM

Although we had enough to show that the convergence and singularity issues do not affect the model's estimates, checking another statistical package would confirm that statement. As I mentioned in Chapter 3 p.61, I chose the HLM7

Table 15: Compare the Estimates in Convergent, Non-Convergent and Singular Models

$\tau_{tch11} = 0.17$ $\quad \phi_{00} = 0.01$

model issue	convergent			non-convergent			singular		
statistics	avg($\hat{\theta}$)	avg(sd($\hat{\theta}$))	MSE($\hat{\theta}$)	avg($\hat{\theta}$)	avg(sd($\hat{\theta}$))	MSE($\hat{\theta}$)	avg($\hat{\theta}$)	avg(sd($\hat{\theta}$))	MSE($\hat{\theta}$)
δ_{010}	3.4992	0.00681	6.52e-07	3.5013	0.00965	3.91e-05	3.5	0.00690	1.47e-08
τ_{tch11}	0.1682	0.00330	3.32e-06	0.173	0.00587	2.657.e-4	0.169	0.00694	5.8e-07
ϕ_{00}	0.0240	0.00256	1.91e-4	0.0136	0.0135	3.19e-4	5.98e-05	8.51e-05	1.26e-4

$\tau_{tch11} = 0.17$ $\quad \phi_{00} = 0.15$

model issue	convergent			non-convergent			singular		
δ_{010}	3.5	0.0250	2.44e-05	3.5	0.0276	6.70e-4	NA	NA	NA
τ_{tch11}	0.173	0.00592	1.19e-05	0.172	0.00597	0.000144	NA	NA	NA
ϕ_{00}	0.157	0.0171	6.62e-05	0.149	0.0247	0.0000191	NA	NA	NA

$\tau_{tch11} = 0.87$ $\quad \phi_{00} = 0.05$

model issue	convergent			non-convergent			singular		
δ_{010}	3.5	0.0606	4.86e-05	3.5	0.0401	4.10e-05	3.5	0.0241	2.67e-4
τ_{tch11}	0.862	0.0297	4.74e-05	0.868	0.0275	1.67e-4	0.897	0.00640	1.34e-4
ϕ_{00}	0.0440	0.0462	2.62e-05	0.0662	0.0988	8.77e-4	0.00255	0.00279	4.05e-4

package because it has a "auto-fix" convergence technique (McCoach et al., 2018). I selected eight models to check the estimate of non-convergent and singular models in HLM7. Based on the results in Table 16, I found the following:

- in Model 1, the estimate of $\hat{\delta}_{010} = 3.4703$ in HLM and 3.4700 in lme4 where, Model 1 is convergent.

- in Model 2, the estimate of $\hat{\delta}_{010} = 3.603$ in HLM and 3.600 in lme4 where Model 2 is a non-convergent model.

- similarly, in Model 3, the estimate of $\hat{\delta}_{010} = 3.51$ in HLM and 3.514 in lme4, where Model 3 is singular.

Table 16 shows that the estimates for convergent, non-convergent or singular models are almost identical in both HLM and lme4. Therefore, the convergence and the singularity have no effect of the model's estimates.

The Convergence and Singularity Effect on The Fixed Effect Estimates

In order to ensure that the warning messages (convergence and singularity issues) do not effect the model's estimates, I have checked the fixed effect estimates in term of the convergence and singularity issues. I checked the estimate of the fixed effects ($\hat{\delta}_{000}$, $\hat{\delta}_{001}$, $\hat{\delta}_{010}$, $\hat{\delta}_{020}$ and $\hat{\delta}_{100}$) with two values of τ_{11} and two values of ϕ_{00}. Table 17 explains the estimate's mean and the standard deviation's mean of each fixed effect coefficient.

Based on the results in the table:

- The estimates of the fixed effects are similar in all the model's issues. That is, the convergence and singularity issues do not affect the fixed effect estimates.

- The standard deviation of the fixed effect estimates is similar to the fixed

Table 16: Non-Convergent and Singular models Estimates in *HLM7* and *lme4*

$\tau_{tch11} = 0.87$ $\phi_{00} = 0.05$						
model issue	convergent		non-convergent		singular	
statistics	HLM	lme4	HLM	lme4	HLM	lme4
δ_{010}	3.4703	3.4700	3.6030	3.6000	3.5140	3.5140
τ_{tch11}	0.8366	0.8366	0.8400	0.8420	0.8277	0.8286
ϕ_{00}	0.0728	0.0752	0.0249	0.0205	8.82e-5	1.64e-5
$\tau_{tch11} = 0.17$ $\phi_{00} = 0.15$						
model issue	convergent		non-convergent		singular	
statistics	HLM	lme4	HLM	lme4	HLM	lme4
δ_{010}	3.5316	3.5320	3.4528	3.4530	NA	NA
τ_{tch11}	0.1564	0.1564	0.1634	0.1634	NA	NA
ϕ_{00}	0.1537	0.1565	0.1346	0.1371	NA	NA
$\tau_{tch11} = 0.17$ $\phi_{00} = 0.01$						
model issue	convergent		non-convergent		singular	
statistics	HLM	lme4	HLM	lme4	HLM	lme4
δ_{010}	3.5001	3.5000	3.5023	3.5020	3.4960	3.4960
τ_{tch11}	0.1736	0.1737	0.1622	0.1625	0.1579	0.1574
ϕ_{00}	0.0190	0.0192	0.0008	0.0007	0.0103	1.18e-6

effect estimates. Again neither the convergence nor the singularity messages affect the standard deviation of $\hat{\theta}$.

- If the model experiences singularity issue then $\hat{\phi}_{00} \sim 0$. However, the opposite is not always true. For example, In Figure 15, when $\phi_{00} = 0.05$, there is a convergent model that has $\hat{\phi}_{00} = 0$ and does not experience singularity issue.

The Effect of Removing the Cross-Classified Random Effect

One of the points that may be considered is the effect of removing W_1 from the school level as a random effect. That is, running the model as follows :

```
lmer(y1~X+W1+W2+Z+(1|std)+(W1|tch)+(0+X||sch), data1)
```

I have fitted 150 models and checked the results as showed in Table 18

Table 17: The Average of Fixed Effect Estimates and Standard deviation of 100 Models

$\tau_{11} = 0.17$	Convergent		Non-Conv.		Singular	
$\phi_{00} = 0.01$	Est.	SD	Est.	SD	Est.	SD
δ_{000}	2.9399	0.1591	3.0060	0.1621	2.8802	0.1629
δ_{001}	8.0258	0.0597	7.9803	0.0616	7.9937	0.0588
δ_{010}	3.5082	0.0421	3.4989	0.0446	3.5385	0.0423
δ_{020}	0.9163	0.0645	0.9023	0.0657	0.9395	0.0660
δ_{100}	2.0027	0.0036	1.9999	0.0036	1.9996	0.0043
$\tau_{11} = 0.87$	Convergent		Non-Conv.		Singular	
$\phi_{00} = 0.05$	Est.	SD	Est	SD	Est	SD
δ_{000}	2.9970	0.0253	2.9976	0.0258	2.9962	0.0257
δ_{001}	8.0053	0.0095	7.9998	0.0097	8.0049	0.0096
δ_{010}	3.4992	0.0091	3.501	0.0089	3.4999	0.0080
δ_{020}	0.8998	0.0105	0.9014	0.0107	0.8998	0.0106
δ_{100}	1.9981	0.0036	2.0006	0.0036	2.0010	0.0037

Based on the results in Table 18 and the simulated data, we can conclude the following:

- The level-three random effect ϕ_{00} does not have any effect on neither the cross-classified fixed effect δ_{010} nor the level-two random effect τ_{tch11},

- The cross-classified fixed effect δ_{010} does not change by the level-three random effect ϕ_{00}.

Conclusion

Based on the statistical analysis of the simulated data, I found that the package lme4 can fit three-level-cross-classified models by using the function *lmer*. Furthermore, this package provides a precise results that including the random effects estimates with their correlations. Additionally, the package provides an estimate of fixed effects including their degree of freedom the correlation of fixed effects.

Table 18: Comparison between the Complete and Reduced Models

$\tau_{tch11} = 0.17, \phi_{00} = 0.01$			
Case	δ_{010}	$\hat{\delta}_{010}$	$\mathrm{MSE}(\hat{\delta}_{010})$
$(\delta_{010} + c_{1jk} + d_{01k})W_{1ik}$	3.5	3.501	2.84e-05
$(\delta_{010} + c_{1jk})W_{1ik}$	3.5	3.501	2.84e-05
$(\delta_{010})W_{1ik}$	3.5	3.501	2.84e-05
Case	τ_{tch11}	$\hat{\tau}_{tch11}$	$\mathrm{MSE}(\hat{\tau}_{tch11})$
$(\delta_{010} + c_{1jk} + d_{01k})W_{1ik}$	0.17	0.1728	1.82e-4
$(\delta_{010} + c_{1jk})W_{1ik}$	0.17	0.1719	3.93e-4

Based on the simulated data and under the setting considered, the fixed effects provide perfect estimates and the shape of the relative bias of the fixed effects' bias is symmetric with outlier. Similarly, the random effect coefficients $(\hat{\sigma}, \hat{\tau}_{std00}, \hat{\tau}_{tch00}$ and $\hat{\tau}_{tch11})$ have closer estimates to their true values with symmetric shapes. On the other hand, the estimate $\hat{\phi}_{00}$ is far from the true value with a right skewed shape. However, the shape begins to be symmetric when the ratio between ϕ_{00}^2 and $(\phi_{00}^2 + \tau_{tch11}^2)$ is large. On the other hand, the change of the ratio number does not affect the fixed effect estimates.

The estimation results also showed that the correlation between the cross-classified random effect c_1 and c_0 does not affect the model's coefficient estimates.

The research also found that the convergence and singularity messages given by the package lme4 do not affect the model's estimates. Finally, the study showed that removing level-three-cross-classified random effect does not affect the model's coefficient estimates.

Limitations and Future Research

I have covered the three-level-cross-classified model in the package lme4. McCoach et al. (2018) compared a two-level model in five different packages. A

different study may compare the three-level-cross-classified model in several packages. Although I used R in performing the model, I did not try the package nlme, because it is slow in performing the HLM model. Further study may compare the two packages in term of the cross-classification.

Regarding simulated data, I chose complete, large and balanced data in order to get precise results. However, I did not include missing or unbalanced data, which may be included in further research.

The study covered the convergence and singularity issues by comparing the estimates in convergent and non-convergent models that is based on the simulated data. Further research may try different techniques such as choosing different starting values or different computational algorithm (McCoach et al., 2018).

I studied a three-level-cross-classified model in which the teacher and student were cross-classified in level-two. Different research may include the cross-classification between the school and the neighborhood in level-three (Raundenbush and Bryk, 2002). The cross-classification could also be contained in both level-two and level-three.

APPENDIX SECTION

The Simulated Data Code

```
# The used packages

library(tidyverse)

library(lme4)

library(haven)

library(lmerTest)


n.sch =30

n.std =500

n.tch = 15


#**************

# Fixed effects

#*************

delta000 = 3

delta001  = 8

delta010 =  3.5

delta020= 0.9

delta100 = 2


#==============

# Random effects

#==============

sigma = 0.03   #Std Dev

e.real <- sigma

sigma.sq = 0.0009


#****************************
```

```
# I used two correlation values
#*****************************
rho1 = -.03
rho2 = -0.75


#******************
# The random effect
#******************
tau_small_00 =0.01   #variance
tau_small_00.stdev = sqrt(tau_small_00)


tau_big_00 = 0.85    #variance
tau_big_00.stdev = sqrt(tau_big_00)


tau_small_11 = 0.03 # variance
tau_small_11.stdev = sqrt(tau_small_11)


tau_big_11 = 0.75    # variance
tau_big_11.stdev = sqrt(tau_big_11)


# student variance
tau_st_00 = 0.05    # std.Dev
real = tau_st_00
b0.real <- 0.05
tau_st.sq =  0.0025   #Variance


# School variances
phi11 =0.05    #Std. Dev
d01.real <- phi11
phi11.sq = 0.0025 # variance


phi11_2 = 0.1
```

```
phi11_3 = 0.075

Phi11_4 = 0.2

Phi11_5 = 0.15

Phi11_6 = 0.64

Phi11_7 = 0.75

phi33 =   0.02    #Std. Dev

phi33.sq = 0.0004  #variance


# Compute th coveriance when Rho is small
R1.tau_small_small_10 = rho1 *sqrt(tau_small_00*tau_small_11)

R1.tau_small_big_10 = rho1 *sqrt(tau_small_00*tau_big_11)

R1.tau_big_small_10 = rho1*sqrt(tau_big_00*tau_small_11)

R1.tau_big_big_10 = rho1 *sqrt(tau_big_00*tau_big_11)

R9= rho1 *sqrt(tau_small_00*0.4)
# Compute th coveriance when Rho is large
R2.tau_small_small_10 = rho2 *sqrt(tau_small_00*tau_small_11)

R2.tau_small_big_10 = rho2 *sqrt(tau_small_00*tau_big_11)

R2.tau_big_small_10 = rho2*sqrt(tau_big_00*tau_small_11)

R2.tau_big_big_10 = rho2 *sqrt(tau_big_00*tau_big_11)


var.sim1 <- c(tau_small_00, R1.tau_small_small_10,tau_small_11 )

var.sim2 <- c(tau_small_00, R1.tau_small_big_10,tau_big_11 )

var.sim3 <- c(tau_big_00, R1.tau_big_small_10,tau_small_11 )

var.sim4 <- c(tau_big_00, R1.tau_big_big_10,tau_big_11 )

var.sim5 <- c(tau_small_00, R2.tau_small_small_10,tau_small_11 )

var.sim6 <- c(tau_small_00, R2.tau_small_big_10,tau_big_11 )

var.sim7 <- c(tau_big_00, R2.tau_big_small_10,tau_small_11 )

var.sim8 <- c(tau_big_00, R2.tau_big_big_10,tau_big_11 )

var.sim9 <- c(tau_small_00, R9, 0.4 )

case <- cbind(var.sim1, var.sim2, var.sim3, var.sim4, var.sim5,
              var.sim6, var.sim7, var.sim8, var.sim9)
```

```
#==========================================================
# Create a function to generate the variable of students
#==========================================================
StdInScl <- function(){
  std.code <- c(seq( 1,300),seq(301,400),seq(1,100),
                seq(101,200), seq(401,500),seq(301,400),
                seq(1,100))
  W.sample <- sample(c(0,1), replace = TRUE, size = 500)
  b0 <- rnorm(500, 0, tau_st_00)
  for(i in 1:49){
    b0 =cbind(b0, rnorm(500, 0, tau_st_00))
  }
  std_chr <- c(W.sample[1:400], W.sample[1:200],
               W.sample[401:500], W.sample[301:400],
               W.sample[1:100])
  std_b0 <- rbind(b0[1:400,], b0[1:200,],
                  b0[401:500,], b0[301:400,],
                  b0[1:100,])
  group <- rep(c(1,2,3,4,1,2,5,4,1), each=100)
  data.frame(group, cbind(std.code, std_chr,
                          std_b0))
}


l=StdInScl()
p =n.sch-1
q = n.sch*1000+100000
for(i in 1:p){
  l = rbind(l,StdInScl())
}
sch.serial <-  seq(101000, q, by=1000)
l$std.code <- l$std.code + rep(sch.serial, each = 900)
```

```
std.table <- data.frame(cbind(std =l$std.code, W1=l$std_chr,
                sch=rep(sch.serial,group =l$group, each =900),
                l[,4:53]))
for( i in 1:50){
  names(std.table)[3+i] <- paste("b0_", i, sep="")
}
write.csv(data.frame(std.table), "std.table0507.csv")
std.HLM <- data.frame(unique(std.table[,1:3]))
write_sav(std.HLM, "level2std_30_0507.sav")


#=====================================================
# A function to generate the variables of the teachers
#=====================================================


#=========================================
# A function of correlated teacher variance
#=========================================
tch.var <- function(a, b, c )
{
  Sigma1t <- matrix(c(a ,b, b,c),2,2)
  mvrnorm((n.tch), mu=c(0,0),Sigma = Sigma1t)
}

rm(sss, ssb, sbs, sbb, bss, bsb, bbs, bbb)
TchInSch <- function()
{
  tchr <- seq(1,15)
  tch.chr <- runif(15, 1.5,3)
  sss <- tch.var(case[1,1], case[2,1],case[3,1] )
  ssb <- tch.var(case[1,2], case[2,2],case[3,2] )
  sbs <- tch.var(case[1,3], case[2,3],case[3,3] )
  sbb <- tch.var(case[1,4], case[2,4],case[3,4] )
```

90

```
bss <- tch.var(case[1,5], case[2,5],case[3,5] )

bsb <- tch.var(case[1,6], case[2,6],case[3,6] )

bbs <- tch.var(case[1,7], case[2,7],case[3,7] )

bbb <- tch.var(case[1,8], case[2,8],case[3,8] )

sm <- tch.var(case[1,9], case[2,9],case[3,9])

for(i in 1:49)

{

sss <- cbind(sss,tch.var(case[1,1], case[2,1],case[3,1] ))

ssb <- cbind(ssb,tch.var(case[1,2], case[2,2],case[3,2] ))

sbs <- cbind(sbs, tch.var(case[1,3], case[2,3],case[3,3] ))

sbb <- cbind(sbb, tch.var(case[1,4], case[2,4],case[3,4] ))

bss <- cbind(bss,tch.var(case[1,5], case[2,5],case[3,5] ))

bsb <- cbind(bsb,tch.var(case[1,6], case[2,6],case[3,6] ))

bbs <- cbind(bbs,tch.var(case[1,7], case[2,7],case[3,7] ))

bbb <- cbind(bbb,tch.var(case[1,8], case[2,8],case[3,8] ))

sm <-  cbind(sm, tch.var(case[1,9], case[2,9],case[3,9]))

}

x=cbind(sss, ssb, sbs, sbb, bss, bsb, bbs, bbb, sm)


tchr_W <- cbind(tch=tchr, W2=tch.chr, x)

tch3 <- tchr_W[1:5,]

tch4 <- tchr_W[6:10,]

tch5 <- tchr_W[11:15,]

tch_W <- rbind(tchr_W, tch3[sample(nrow(tch3)),],

        tch4[sample(nrow(tch4)),],tch5[sample(nrow(tch5)),],

        tch3[sample(nrow(tch3)),],  tch4[sample(nrow(tch4)),],

        tch5[sample(nrow(tch5)),])

tch_W[rep(seq_len(nrow(tch_W)), each=20),]


}


rm(ll)
```

```r
ll =data.frame(TchInSch())

for(i in 1:p){

  ll = rbind(ll,TchInSch())

}


sch.serial <-  seq(101000,q, by=1000)

ll$tch <- ll$tch + rep(sch.serial, each = 900)

tch.table <- data.frame(cbind(tch=ll$tch, W2=ll$W2,

                                  sch=rep(sch.serial,

            each =900), ll[,3:902]))

tch.table.1 <- tch.table[,1:103]


for( i in 1:50){

  names(tch.table.1)[2+2*i] <- paste("c0_", i, sep="")

  names(tch.table.1)[3+2*i] <- paste("c1_", i, sep="")

}

write.csv(tch.table.1, "tch.table.sim1.csv")

tch.table.2  <- cbind(tch.table[,1:3],tch.table[,104:203])

for( i in 1:50){

  names(tch.table.2)[2+2*i] <- paste("c0_", i, sep="")

  names(tch.table.2)[3+2*i] <- paste("c1_", i, sep="")

}

write.csv(tch.table.2, "tch.table.sim2.csv")

tch.table.3  <- cbind(tch.table[,1:3],tch.table[,204:303])

for( i in 1:50){

  names(tch.table.3)[2+2*i] <- paste("c0_", i, sep="")

  names(tch.table.3)[3+2*i] <- paste("c1_", i, sep="")

}

write.csv(tch.table.3, "tch.table.sim3.csv")


tch.table.4  <- cbind(tch.table[,1:3],tch.table[,304:403])

for( i in 1:50){
```

```r
  names(tch.table.4)[2+2*i] <- paste("c0_", i, sep="")

  names(tch.table.4)[3+2*i] <- paste("c1_", i, sep="")

}

write.csv(tch.table.4, "tch.table.sim4.csv")


tch.table.5  <- cbind(tch.table[,1:3],tch.table[,404:503])

for( i in 1:50){

  names(tch.table.5)[2+2*i] <- paste("c0_", i, sep="")

  names(tch.table.5)[3+2*i] <- paste("c1_", i, sep="")

}

write.csv(tch.table.5, "tch.table.sim5.csv")


tch.table.6  <- cbind(tch.table[,1:3],tch.table[,504:603])

for( i in 1:50){

  names(tch.table.6)[2+2*i] <- paste("c0_", i, sep="")

  names(tch.table.6)[3+2*i] <- paste("c1_", i, sep="")

}

write.csv(tch.table.6, "tch.table.sim6.csv")


tch.table.7  <- cbind(tch.table[,1:3],tch.table[,604:703])

for( i in 1:50){

  names(tch.table.7)[2+2*i] <- paste("c0_", i, sep="")

  names(tch.table.7)[3+2*i] <- paste("c1_", i, sep="")

}


write.csv(tch.table.7, "tch.table.sim7.csv")


tch.table.8  <- cbind(tch.table[,1:3],tch.table[,704:803])

for( i in 1:50){

  names(tch.table.8)[2+2*i] <- paste("c0_", i, sep="")

  names(tch.table.8)[3+2*i] <- paste("c1_", i, sep="")

}
```

```
write.csv(tch.table.8, "tch.table.sim8.csv")


tch.HLM <- data.frame(unique(tch.table[,1:3]))

write_sav(tch.HLM, "level2tch_30_0507.sav")


tch.table.9  <- cbind(tch.table[,1:3],tch.table[,804:903])

for( i in 1:50){

  names(tch.table.8)[2+2*i] <- paste("c0_", i, sep="")

  names(tch.table.8)[3+2*i] <- paste("c1_", i, sep="")

}

write.csv(tch.table.8, "tch.table.sim9.csv")


tch.HLM <- data.frame(unique(tch.table[,1:3]))

write_sav(tch.HLM, "level2tch_30_0507.sav")


#=======================

# Create the school table

#=======================

sch.serial <-  rep(seq(101000,q, by=1000), each =900)

set.seed(5)

Z_1 <- sample(c(0,1), replace = TRUE,size=n.sch)

Z1 <- rep(Z_1, each=900)

set.seed(6)

Z_2 <- sample(c(0,1), replace = TRUE,size=n.sch)

Z2 <- rep(Z_2, each=900)


d01.05 <- rep(rnorm(n.sch, 0, phi11), each= 900)

for(i in 1:49){

  d01.05 <- cbind( d01.05,rep(rnorm(n.sch, 0, phi11),

              each= 900))

}
```

```
d01.1 <- rep(rnorm(n.sch, 0, phi11_2), each= 900)

for(i in 1:49){

   d01.1 <- cbind( d01.1,rep(rnorm(n.sch, 0, phi11_2),

                             each= 900))

}


d01.075 <- rep(rnorm(n.sch, 0, phi11_3), each= 900)

for(i in 1:49){

   d01.075 <- cbind( d01.075,rep(rnorm(n.sch, 0, phi11_3),

          each= 900))

}


d01.2 <- rep(rnorm(n.sch, 0, Phi11_4),

          each= 900)

for(i in 1:49){

   d01.2 <- cbind( d01.2,rep(rnorm(n.sch, 0, Phi11_4),

          each= 900))

}


d01.15 <- rep(rnorm(n.sch, 0, Phi11_5), each= 900)

for(i in 1:49){

   d01.15 <- cbind( d01.15,rep(rnorm(n.sch, 0, Phi11_5),

             each= 900))

}


d01.64 <- rep(rnorm(n.sch, 0, Phi11_6), each= 900)

for(i in 1:49){

   d01.64 <- cbind( d01.64,rep(rnorm(n.sch, 0, Phi11_6),

                       each= 900))

}

d01.75 <- rep(rnorm(n.sch, 0, Phi11_7), each= 900)

for(i in 1:49){
```

```r
  d01.75 <- cbind( d01.75,rep(rnorm(n.sch, 0, Phi11_7),
                              each= 900))
}


d01.014 <- rep(rnorm(n.sch, 0, 0.014), each= 900)
for(i in 1:49){
  d01.014 <- cbind( d01.014,rep(rnorm(n.sch, 0, 0.014),
                                each= 900))
}
write.csv(d01.014,"d01.014.csv")


d01.073 <- rep(rnorm(n.sch, 0, 0.073), each= 900)
for(i in 1:49){
  d01.073 <- cbind( d01.073,rep(rnorm(n.sch, 0, 0.073),
                                each= 900))
}
write.csv(d01.073,"d01.073.csv")


d01.073 <- rep(rnorm(n.sch, 0, 0.073), each= 900)
for(i in 1:49){
  d01.073 <- cbind( d01.073,rep(rnorm(n.sch, 0, 0.073),
                                each= 900))
}
write.csv(d01.073,"d01.073.csv")



d01.053 <- rep(rnorm(n.sch, 0, 0.053), each= 900)
for(i in 1:49){
  d01.053 <- cbind( d01.053,rep(rnorm(n.sch, 0, 0.053),
                                each= 900))
}
write.csv(d01.053,"d01.053.csv")
```

```
d01.7 <- rep(rnorm(n.sch, 0, 0.7), each= 900)
for(i in 1:49){
  d01.7 <- cbind( d01.7,rep(rnorm(n.sch, 0, 0.7),
                            each= 900))
}
write.csv(d01.7,"d01.7.csv")
d01.96 <- rep(rnorm(n.sch, 0, 0.957), each= 900)
for(i in 1:49){
  d01.96 <- cbind( d01.96,rep(rnorm(n.sch, 0, 0.957),
                              each= 900))
}
write.csv(d01.96,"d01.96.csv")
d01.57 <- rep(rnorm(n.sch, 0, 0.57), each= 900)
for(i in 1:49){
  d01.57 <- cbind( d01.57,rep(rnorm(n.sch, 0, 0.57),
                              each= 900))
}
write.csv(d01.57,"d01.57.csv")
d01.78 <- rep(rnorm(n.sch, 0, 0.783), each= 900)
for(i in 1:49){
  d01.78 <- cbind( d01.78,rep(rnorm(n.sch, 0, 0.783),
                              each= 900))
}
write.csv(d01.78,"d01.78.csv")


d01.86 <- rep(rnorm(n.sch, 0, 0.86), each= 900)
for(i in 1:49){
  d01.86 <- cbind( d01.86,rep(rnorm(n.sch, 0, 0.86), each= 900))
}
write.csv(d01.86,"d01.86.csv")
d01.5 <- rep(rnorm(n.sch, 0, 0.5), each= 900)
```

```
for(i in 1:49){

  d01.5 <- cbind( d01.5,rep(rnorm(n.sch, 0, 0.5), each= 900))

}

write.csv(d01.5,"d01.5.csv")

d01.4 <- rep(rnorm(n.sch, 0, 0.4), each= 900)

for(i in 1:49){

  d01.4 <- cbind( d01.4,rep(rnorm(n.sch, 0, 0.4), each= 900))

}

write.csv(d01.4,"d01.4.csv")

d01.3 <- rep(rnorm(n.sch, 0, 0.3), each= 900)

for(i in 1:49){

  d01.3 <- cbind( d01.3,rep(rnorm(n.sch, 0, 0.3), each= 900))

}

write.csv(d01.3,"d01.3.csv")




d01.025 <- rep(rnorm(n.sch, 0, 0.025), each= 900)

for(i in 1:49){

  d01.025 <- cbind( d01.025,rep(rnorm(n.sch, 0, 0.025), each= 900))

}

write.csv(d01.025,"d01.025.csv")




d01.01 <- rep(rnorm(n.sch, 0, 0.01), each= 900)

for(i in 1:49){

  d01.01 <- cbind( d01.01,rep(rnorm(n.sch, 0, 0.01), each= 900))

}

write.csv(d01.01,"d01.01.csv")


d01.64 <- rep(rnorm(n.sch, 0, Phi11_6), each= 900)

for(i in 1:49){

  d01.64 <- cbind( d01.64,rep(rnorm(n.sch, 0, Phi11_6), each= 900))
```

```
}
write.csv(d01.64,"d01.64.csv")


d01 <- (cbind(d01.05,d01.1, d01.075, d01.2, d01.15,

                d01.64, d01.75))




d10 <- rep(rnorm(n.sch,0, phi33), each =900)
for(i in 1:49){
  d10 <- cbind( d10,rep(rnorm(n.sch, 0, phi33),

                       each= 900))
}
sch.table <- data.frame(sch=sch.serial, Z1, Z2, d01, d10)


names(sch.table)[4:353] <- paste("d01_", 1:350, sep="")
names(sch.table)[354:403] <- paste("d10_", 1:50, sep="")


write.csv(sch.table, "sch.table.csv")
sch.HLM <- data.frame(unique(sch.table[,1:3]))
write_sav(sch.HLM, "level3sch_30_0507.sav")


time <- rep(seq(1:3), each=300)
X1 <- rep(time,n.sch)


#=====================================
# Create a data frame without y variable
#=====================================
planedata <- data.frame(std= std.table$std, tch=tch.table$tch,

            sch=sch.table$sch, X1, W1 =std.table$W1,

            W2 =tch.table$W2, Z1=sch.table$Z1, Z2=sch.table$Z2)
data <- planedata
```

```
write.csv(planedata," planedata.csv")

#===================
# Generate the data
#===================
Y1=delta000 + delta001*data$Z1+ delta010*data$W1+
  delta020*data$W2 + delta100*data$X1 + sch.table[,4]*data$W1+
  sch.table[,354]*data$X1+
  tch.table.1[,5]*data$W1+std.table[,4]+
  tch.table.1[,4]+ rnorm(n.sch*900, 0, sigma)
y =0
for (i in 1:49){
  y = delta000 + delta001*data$Z1+ delta010*data$W1+
    delta020*data$W2 +
    delta100*data$X1 + sch.table[,i+4]*data$W1+
    sch.table[,i+354]*data$X1+
    tch.table.1[,2*i+5]*data$W1+std.table[,i+4]+
    tch.table.1[,2*i+4]+ rnorm(n.sch*900, 0, sigma)
  Y1 =cbind(Y1,y)
}
tch.table.2 <- read.csv("tch.table.sim2.csv") %>%
  dplyr::select(2:104)


Y2=delta000 + delta001*data$Z1+ delta010*data$W1+
  delta020*data$W2 + delta100*data$X1 +
  sch.table[,4]*data$W1+sch.table[,354]*data$X1+
  tch.table.2[,5]*data$W1+std.table[,4]+
  tch.table.2[,4]+ rnorm(n.sch*900, 0, sigma)
y =0
for (i in 1:49){
  y = delta000 + delta001*data$Z1+ delta010*data$W1+
    delta020*data$W2 + delta100*data$X1 +
    sch.table[,i+4]*data$W1+sch.table[,i+354]*data$X1+
```

```
      tch.table.2[,2*i+5]*data$W1+std.table[,i+4]+

      tch.table.2[,2*i+4]+ rnorm(n.sch*900, 0, sigma)

    Y2 =cbind(Y2,y)

}


Y3=delta000 + delta001*data$Z1+ delta010*data$W1+

  delta020*data$W2 + delta100*data$X1 +

  sch.table[,4]*data$W1+sch.table[,354]*data$X1+

  tch.table.3[,5]*data$W1+std.table[,4]+tch.table.3[,4]+

  rnorm(n.sch*900, 0, sigma)

y =0

for (i in 1:49){

  y = delta000 + delta001*data$Z1+ delta010*data$W1+

    delta020*data$W2 + delta100*data$X1 +

    sch.table[,i+4]*data$W1+sch.table[,i+354]*data$X1+

    tch.table.3[,2*i+5]*data$W1+std.table[,i+4]+

    tch.table.3[,2*i+4]+ rnorm(n.sch*900, 0, sigma)

  Y3 =cbind(Y3,y)



}


Y4=delta000 + delta001*data$Z1+ delta010*data$W1+

  delta020*data$W2 + delta100*data$X1 +

  sch.table[,4]*data$W1+sch.table[,354]*data$X1+

  tch.table.4[,5]*data$W1+std.table[,4]+

  tch.table.4[,4]+ rnorm(n.sch*900, 0, sigma)

y =0

for (i in 1:49){

  y = delta000 + delta001*data$Z1+ delta010*data$W1+

    delta020*data$W2 + delta100*data$X1 +

    sch.table[,i+4]*data$W1+sch.table[,i+354]*data$X1+
```

```
    tch.table.4[,2*i+5]*data$W1+std.table[,i+4]+

    tch.table.4[,2*i+4]+ rnorm(n.sch*900, 0, sigma)

  Y4 =cbind(Y4,y)



}



Y5=delta000 + delta001*data$Z1+ delta010*data$W1+

  delta020*data$W2 + delta100*data$X1 +

  sch.table[,4]*data$W1+sch.table[,354]*data$X1+

  tch.table.5[,5]*data$W1+std.table[,4]+

  tch.table.5[,4]+ rnorm(n.sch*900, 0, sigma)

y =0

for (i in 1:49){

  y = delta000 + delta001*data$Z1+ delta010*data$W1+

    delta020*data$W2 + delta100*data$X1 +

    sch.table[,i+4]*data$W1+sch.table[,i+354]*data$X1+

    tch.table.5[,2*i+5]*data$W1+std.table[,i+4]+

    tch.table.5[,2*i+4]+ rnorm(n.sch*900, 0, sigma)

  Y5 =cbind(Y5,y)



}



Y6=delta000 + delta001*data$Z1+ delta010*data$W1+

  delta020*data$W2 + delta100*data$X1 +

  sch.table[,4]*data$W1+sch.table[,354]*data$X1+

  tch.table.6[,5]*data$W1+std.table[,4]+

  tch.table.6[,4]+ rnorm(n.sch*900, 0, sigma)

y =0

for (i in 1:49){

  y = delta000 + delta001*data$Z1+ delta010*data$W1+

    delta020*data$W2 + delta100*data$X1 +
```

```
    sch.table[,i+4]*data$W1+sch.table[,i+354]*data$X1+

    tch.table.6[,2*i+5]*data$W1+std.table[,i+4]+

    tch.table.6[,2*i+4]+ rnorm(n.sch*900, 0, sigma)

  Y6 =cbind(Y6,y)



}



Y7=delta000 + delta001*data$Z1+ delta010*data$W1+

  delta020*data$W2 + delta100*data$X1 +

  sch.table[,4]*data$W1+sch.table[,354]*data$X1+

  tch.table.7[,5]*data$W1+std.table[,4]+

  tch.table.7[,4]+ rnorm(n.sch*900, 0, sigma)

y =0

for (i in 1:49){

  y = delta000 + delta001*data$Z1+ delta010*data$W1+

    delta020*data$W2 + delta100*data$X1 +

    sch.table[,i+4]*data$W1+sch.table[,i+354]*data$X1+

    tch.table.7[,2*i+5]*data$W1+std.table[,i+4]+

    tch.table.7[,2*i+4]+ rnorm(n.sch*900, 0, sigma)

  Y7 =cbind(Y7,y)



}

Y8=delta000 + delta001*data$Z1+ delta010*data$W1+

  delta020*data$W2 + delta100*data$X1 +

  sch.table[,4]*data$W1+sch.table[,354]*data$X1+

  tch.table.8[,5]*data$W1+std.table[,4]+

  tch.table.8[,4]+ rnorm(n.sch*900, 0, sigma)

y =0

for (i in 1:49){

  y = delta000 + delta001*data$Z1+ delta010*data$W1+

    delta020*data$W2 +  delta100*data$X1 +

    sch.table[,i+4]*data$W1+sch.table[,i+354]*data$X1+
```

```
      tch.table.8[,2*i+5]*data$W1+std.table[,i+4]+

      tch.table.8[,2*i+4]+ rnorm(n.sch*900, 0, sigma)

    Y8 =cbind(Y8,y)



}



Y9=delta000 + delta001*data$Z1+ delta010*data$W1+

  delta020*data$W2 + delta100*data$X1 +

  sch.table[,4]*data$W1+sch.table[,354]*data$X1+

  tch.table.9[,5]*data$W1+std.table[,4]+

  tch.table.9[,4]+rnorm(n.sch*900, 0, sigma)

y =0

for (i in 1:49){

  y = delta000 + delta001*data$Z1+ delta010*data$W1+

    delta020*data$W2 + delta100*data$X1 +

    sch.table[,i+4]*data$W1+sch.table[,i+354]*data$X1+

    tch.table.9[,2*i+5]*data$W1+std.table[,i+4]+

    tch.table.9[,2*i+4]+ rnorm(n.sch*900, 0, sigma)

  Y9 =cbind(Y9,y)



}



Y10=delta000 + delta001*data$Z1+ delta010*data$W1+

  delta020*data$W2 + delta100*data$X1 +

  sch.table[,54]*data$W1+sch.table[,354]*data$X1+

  tch.table.2[,5]*data$W1+std.table[,4]+

  tch.table.2[,4]+ rnorm(n.sch*900, 0, sigma)

y =0

for (i in 1:49){

  y = delta000 + delta001*data$Z1+ delta010*data$W1+

    delta020*data$W2 +  delta100*data$X1 +
```

```
      sch.table[,i+54]*data$W1+sch.table[,i+354]*data$X1+

      tch.table.2[,2*i+5]*data$W1+std.table[,i+4]+

      tch.table.2[,2*i+4]+ rnorm(n.sch*900, 0, sigma)

    Y10 =cbind(Y10,y)




}



Y11=delta000 + delta001*data$Z1+ delta010*data$W1+delta020*data$W2 +

  delta100*data$X1 + sch.table[,104]*data$W1+sch.table[,354]*data$X1+

  tch.table.2[,5]*data$W1+std.table[,4]+

  tch.table.2[,4]+ rnorm(n.sch*900, 0, sigma)

y =0

for (i in 1:49){

  y = delta000 + delta001*data$Z1+ delta010*data$W1+

    delta020*data$W2 + delta100*data$X1 +

    sch.table[,i+104]*data$W1+sch.table[,i+354]*data$X1+

    tch.table.2[,2*i+5]*data$W1+std.table[,i+4]+

    tch.table.2[,2*i+4]+ rnorm(n.sch*900, 0, sigma)

  Y11 =cbind(Y11,y)




}



Y12=delta000 + delta001*data$Z1+ delta010*data$W1+

  delta020*data$W2 + delta100*data$X1 +

  sch.table[,154]*data$W1+sch.table[,354]*data$X1+

  tch.table.2[,5]*data$W1+std.table[,4]+

  tch.table.2[,4]+ rnorm(n.sch*900, 0, sigma)

y =0

for (i in 1:49){

  y = delta000 + delta001*data$Z1+ delta010*data$W1+
```

```
      delta020*data$W2 + delta100*data$X1 +

      sch.table[,i+154]*data$W1+sch.table[,i+354]*data$X1+

      tch.table.2[,2*i+5]*data$W1+std.table[,i+4]+

      tch.table.2[,2*i+4]+ rnorm(n.sch*900, 0, sigma)

   Y12 =cbind(Y12,y)



}

Y13=delta000 + delta001*data$Z1+ delta010*data$W1+

delta020*data$W2 + delta100*data$X1 +

sch.table[,204]*data$W1+sch.table[,354]*data$X1+

   tch.table.2[,5]*data$W1+std.table[,4]+

   tch.table.2[,4]+ rnorm(n.sch*900, 0, sigma)

y =0

for (i in 1:49){

   y = delta000 + delta001*data$Z1+ delta010*data$W1+

      delta020*data$W2 + delta100*data$X1 +

      sch.table[,i+204]*data$W1+sch.table[,i+354]*data$X1+

      tch.table.2[,2*i+5]*data$W1+std.table[,i+4]+

      tch.table.2[,2*i+4]+ rnorm(n.sch*900, 0, sigma)

   Y13 =cbind(Y13,y)



}
# d01=0, c1=0
Y14=delta000 + delta001*data$Z1+ delta010*data$W1+

   delta020*data$W2 + delta100*data$X1 +

   sch.table[,354]*data$X1+

   std.table[,4]+tch.table.1[,4]+

   rnorm(n.sch*900, 0, sigma)

y =0

for (i in 1:49){
```

```
  y = delta000 + delta001*data$Z1+ delta010*data$W1+
    delta020*data$W2 +  delta100*data$X1 +
    sch.table[,i+354]*data$X1+
    std.table[,i+4]+tch.table.1[,2*i+4]+
    rnorm(n.sch*900, 0, sigma)
  Y14 =cbind(Y14,y)



}
# d01= 0.05 , c1=0
Y15=delta000 + delta001*data$Z1+ delta010*data$W1+
  delta020*data$W2 + delta100*data$X1 +
  sch.table[,4]*data$W1+sch.table[,354]*data$X1+
  std.table[,4]+tch.table.1[,4]+
  rnorm(n.sch*900, 0, sigma)
y =0
for (i in 1:49){
  y = delta000 + delta001*data$Z1+ delta010*data$W1+
    delta020*data$W2 + delta100*data$X1 +
    sch.table[,i+4]*data$W1+sch.table[,i+354]*data$X1+
    std.table[,i+4]+tch.table.1[,2*i+4]+
    rnorm(n.sch*900, 0, sigma)
  Y15 =cbind(Y15,y)



}


Y16=delta000 + delta001*data$Z1+ delta010*data$W1+
  delta020*data$W2 + delta100*data$X1 +
  sch.table[,54]*data$W1+sch.table[,354]*data$X1+
  std.table[,4]+tch.table.1[,4]+
  rnorm(n.sch*900, 0, sigma)
```

```
y =0

for (i in 1:49){

  y = delta000 + delta001*data$Z1+ delta010*data$W1+

    delta020*data$W2 +  delta100*data$X1 +

    sch.table[,i+54]*data$W1+sch.table[,i+354]*data$X1+

    std.table[,i+4]+tch.table.1[,2*i+4]+

    rnorm(n.sch*900, 0, sigma)

  Y16 =cbind(Y16,y)




}

Y17=delta000 + delta001*data$Z1+ delta010*data$W1+

  delta020*data$W2 + delta100*data$X1 +

  sch.table[,204]*data$W1+sch.table[,354]*data$X1+

  std.table[,4]+tch.table.1[,4]+ rnorm(n.sch*900, 0, sigma)

y =0

for (i in 1:49){

  y = delta000 + delta001*data$Z1+ delta010*data$W1+

    delta020*data$W2 +  delta100*data$X1 +

    sch.table[,i+204]*data$W1+sch.table[,i+354]*data$X1+

    std.table[,i+4]+tch.table.1[,2*i+4]+

    rnorm(n.sch*900, 0, sigma)

  Y17 =cbind(Y17,y)




}

Y18=delta000 + delta001*data$Z1+ delta010*data$W1+

  delta020*data$W2 + delta100*data$X1 +

  sch.table[,354]*data$X1+

  tch.table.1[,5]*data$W1+std.table[,4]+

  tch.table.1[,4]+ rnorm(n.sch*900, 0, sigma)

y =0
```

```
for (i in 1:49){

  y = delta000 + delta001*data$Z1+ delta010*data$W1+

    delta020*data$W2 +

    delta100*data$X1 +sch.table[,i+354]*data$X1+

    tch.table.1[,2*i+5]*data$W1+std.table[,i+4]+

    tch.table.1[,2*i+4]+ rnorm(n.sch*900, 0, sigma)

  Y18 =cbind(Y18,y)




}

Y19=delta000 + delta001*data$Z1+ delta010*data$W1+

  delta020*data$W2 + delta100*data$X1 +

  sch.table[,354]*data$X1+

  tch.table.9[,5]*data$W1+std.table[,4]+tch.table.9[,4]+

  rnorm(n.sch*900, 0, sigma)

y =0

for (i in 1:49){

  y = delta000 + delta001*data$Z1+ delta010*data$W1+

    delta020*data$W2 +  delta100*data$X1 +

    sch.table[,i+354]*data$X1+tch.table.9[,2*i+5]*data$W1+

    std.table[,i+4]+tch.table.9[,2*i+4]+

    rnorm(n.sch*900, 0, sigma)

  Y19 =cbind(Y19,y)




}

Y20=delta000 + delta001*data$Z1+ delta010*data$W1+

  delta020*data$W2 + delta100*data$X1 +

  sch.table[,354]*data$X1+tch.table.4[,5]*data$W1+std.table[,4]+

  tch.table.4[,4]+ rnorm(n.sch*900, 0, sigma)

y =0

for (i in 1:49){
```

```
  y = delta000 + delta001*data$Z1+ delta010*data$W1+
    delta020*data$W2 + delta100*data$X1 +
    sch.table[,i+354]*data$X1+
    tch.table.4[,2*i+5]*data$W1+std.table[,i+4]+
    tch.table.4[,2*i+4]+ rnorm(n.sch*900, 0, sigma)
  Y20 =cbind(Y20,y)




}


Y21=delta000 + delta001*data$Z1+ delta010*data$W1+
  delta020*data$W2 + delta100*data$X1 +
  sch.table[,54]*data$W1+sch.table[,354]*data$X1+
  tch.table.1[,5]*data$W1+std.table[,4]+tch.table.1[,4]+
  rnorm(n.sch*900, 0, sigma)
y =0
for (i in 1:49){
  y = delta000 + delta001*data$Z1+ delta010*data$W1+
    delta020*data$W2 + delta100*data$X1 +
    sch.table[,i+54]*data$W1+sch.table[,i+354]*data$X1+
    tch.table.1[,2*i+5]*data$W1+std.table[,i+4]+tch.table.1[,2*i+4]+
    rnorm(n.sch*900, 0, sigma)
  Y21 =cbind(Y21,y)



}


Y22=delta000 + delta001*data$Z1+ delta010*data$W1+
  delta020*data$W2 + delta100*data$X1 +
  sch.table[,54]*data$W1+sch.table[,354]*data$X1+
  tch.table.9[,5]*data$W1+std.table[,4]+
  tch.table.9[,4]+ rnorm(n.sch*900, 0, sigma)
```

```
y =0

for (i in 1:49){

  y = delta000 + delta001*data$Z1+ delta010*data$W1+

    delta020*data$W2 + delta100*data$X1 +

    sch.table[,i+54]*data$W1+sch.table[,i+354]*data$X1+

    tch.table.9[,2*i+5]*data$W1+std.table[,i+4]+

    tch.table.9[,2*i+4]+ rnorm(n.sch*900, 0, sigma)

  Y22 =cbind(Y22,y)



}



Y23=delta000 + delta001*data$Z1+ delta010*data$W1+

  delta020*data$W2 + delta100*data$X1 +

  sch.table[,204]*data$W1+sch.table[,354]*data$X1+

  tch.table.1[,5]*data$W1+std.table[,4]+tch.table.1[,4]+

  rnorm(n.sch*900, 0, sigma)

y =0

for (i in 1:49){

  y = delta000 + delta001*data$Z1+ delta010*data$W1+

    delta020*data$W2 + delta100*data$X1 +

    sch.table[,i+204]*data$W1+sch.table[,i+354]*data$X1+

    tch.table.1[,2*i+5]*data$W1+std.table[,i+4]+

    tch.table.1[,2*i+4]+ rnorm(n.sch*900, 0, sigma)

  Y23 =cbind(Y23,y)



}



Y24=delta000 + delta001*data$Z1+ delta010*data$W1+

  delta020*data$W2 + delta100*data$X1 +

  sch.table[,204]*data$W1+sch.table[,354]*data$X1+
```

```
    tch.table.9[,5]*data$W1+std.table[,4]+

    tch.table.9[,4]+ rnorm(n.sch*900, 0, sigma)

y =0

for (i in 1:49){

  y = delta000 + delta001*data$Z1+ delta010*data$W1+

    delta020*data$W2 + delta100*data$X1 +

    sch.table[,i+204]*data$W1+sch.table[,i+354]*data$X1+

    tch.table.9[,2*i+5]*data$W1+std.table[,i+4]+

    tch.table.9[,2*i+4]+ rnorm(n.sch*900, 0, sigma)

  Y24 =cbind(Y24,y)




}


Y25=delta000 + delta001*data$Z1+ delta010*data$W1+

  delta020*data$W2 + delta100*data$X1 +

  sch.table[,154]*data$W1+sch.table[,354]*data$X1+

  tch.table.9[,5]*data$W1+std.table[,4]+

  tch.table.9[,4]+ rnorm(n.sch*900, 0, sigma)

y =0

for (i in 1:49){

  y = delta000 + delta001*data$Z1+ delta010*data$W1+

    delta020*data$W2 + delta100*data$X1 +

    sch.table[,i+154]*data$W1+sch.table[,i+354]*data$X1+

    tch.table.9[,2*i+5]*data$W1+std.table[,i+4]+

    tch.table.9[,2*i+4]+ rnorm(n.sch*900, 0, sigma)

  Y25 =cbind(Y25,y)




}

Y26=delta000 + delta001*data$Z1+ delta010*data$W1+

  delta020*data$W2 + delta100*data$X1 +
```

```
  sch.table[,254]*data$W1+sch.table[,354]*data$X1+

  tch.table.9[,5]*data$W1+std.table[,4]+

  tch.table.9[,4]+ rnorm(n.sch*900, 0, sigma)

y =0

for (i in 1:49){

  y = delta000 + delta001*data$Z1+ delta010*data$W1+

    delta020*data$W2 + delta100*data$X1 +

    sch.table[,i+254]*data$W1+sch.table[,i+354]*data$X1+

    tch.table.9[,2*i+5]*data$W1+std.table[,i+4]+

    tch.table.9[,2*i+4]+ rnorm(n.sch*900, 0, sigma)

  Y26 =cbind(Y26,y)



}

Y27=delta000 + delta001*data$Z1+ delta010*data$W1+

  delta020*data$W2 + delta100*data$X1 +

  sch.table[,304]*data$W1+sch.table[,354]*data$X1+

  tch.table.2[,5]*data$W1+std.table[,4]+

  tch.table.2[,4]+ rnorm(n.sch*900, 0, sigma)

y =0

for (i in 1:49){

  y = delta000 + delta001*data$Z1+ delta010*data$W1+

    delta020*data$W2 + delta100*data$X1 +

    sch.table[,i+304]*data$W1+sch.table[,i+354]*data$X1+

    tch.table.2[,2*i+5]*data$W1+std.table[,i+4]+

    tch.table.2[,2*i+4]+ rnorm(n.sch*900, 0, sigma)

  Y27 =cbind(Y27,y)



}
```

```
Y28=delta000 + delta001*data$Z1+ delta010*data$W1+
   delta020*data$W2 +delta100*data$X1 +
   d01.7[,1] *data$W1+sch.table[,354]*data$X1+
   tch.table.9[,5]*data$W1+std.table[,4]+
   tch.table.9[,4]+ rnorm(n.sch*900, 0, sigma)
y =0
for (i in 1:49){
   y = delta000 + delta001*data$Z1+ delta010*data$W1+
      delta020*data$W2 + delta100*data$X1 +
      d01.7[,i+1]*data$W1+sch.table[,i+354]*data$X1+
      tch.table.9[,2*i+5]*data$W1+std.table[,i+4]+
      tch.table.9[,2*i+4]+ rnorm(n.sch*900, 0, sigma)
   Y28 =cbind(Y28,y)



}


Y29=delta000 + delta001*data$Z1+ delta010*data$W1+
   delta020*data$W2 + delta100*data$X1 +
   d01.96[,1] *data$W1+sch.table[,354]*data$X1+
   tch.table.2[,5]*data$W1+std.table[,4]+
   tch.table.2[,4]+ rnorm(n.sch*900, 0, sigma)
y =0
for (i in 1:49){
   y = delta000 + delta001*data$Z1+ delta010*data$W1+
      delta020*data$W2 + delta100*data$X1 +
      d01.96[,i+1]*data$W1+sch.table[,i+354]*data$X1+
      tch.table.2[,2*i+5]*data$W1+std.table[,i+4]+
      tch.table.2[,2*i+4]+ rnorm(n.sch*900, 0, sigma)
   Y29 =cbind(Y29,y)
```

```
}


Y30=delta000 + delta001*data$Z1+ delta010*data$W1+
  delta020*data$W2 + delta100*data$X1 +
  d01.57[,1] *data$W1+sch.table[,354]*data$X1+
  tch.table.9[,5]*data$W1+std.table[,4]+
  tch.table.9[,4]+ rnorm(n.sch*900, 0, sigma)
y =0
for (i in 1:49){
  y = delta000 + delta001*data$Z1+ delta010*data$W1+
    delta020*data$W2 + delta100*data$X1 +
    d01.57[,i+1]*data$W1+sch.table[,i+354]*data$X1+
    tch.table.9[,2*i+5]*data$W1+std.table[,i+4]+
    tch.table.9[,2*i+4]+ rnorm(n.sch*900, 0, sigma)
  Y30 =cbind(Y30,y)



}


Y31=delta000 + delta001*data$Z1+ delta010*data$W1+
  delta020*data$W2 + delta100*data$X1 +
  d01.78[,1] *data$W1+sch.table[,354]*data$X1+
  tch.table.2[,5]*data$W1+std.table[,4]+
  tch.table.2[,4]+ rnorm(n.sch*900, 0, sigma)
y =0
for (i in 1:49){
  y = delta000 + delta001*data$Z1+ delta010*data$W1+
    delta020*data$W2 + delta100*data$X1 +
    d01.78[,i+1]*data$W1+sch.table[,i+354]*data$X1+
    tch.table.2[,2*i+5]*data$W1+std.table[,i+4]+
    tch.table.2[,2*i+4]+ rnorm(n.sch*900, 0, sigma)
  Y31 =cbind(Y31,y)
```

```
}



Y32=delta000 + delta001*data$Z1+ delta010*data$W1+
  delta020*data$W2 + delta100*data$X1 +
  d01.86[,1] *data$W1+sch.table[,354]*data$X1+
  tch.table.9[,5]*data$W1+std.table[,4]+
  tch.table.9[,4]+ rnorm(n.sch*900, 0, sigma)
y =0
for (i in 1:49){
  y = delta000 + delta001*data$Z1+ delta010*data$W1+
    delta020*data$W2 + delta100*data$X1 +
    d01.86[,i+1]*data$W1+sch.table[,i+354]*data$X1+
    tch.table.9[,2*i+5]*data$W1+std.table[,i+4]+
    tch.table.9[,2*i+4]+ rnorm(n.sch*900, 0, sigma)
  Y32 =cbind(Y32,y)



}



Y33=delta000 + delta001*data$Z1+ delta010*data$W1+
  delta020*data$W2 + delta100*data$X1 +
  d01.96[,1] *data$W1+sch.table[,354]*data$X1+
  tch.table.9[,5]*data$W1+std.table[,4]+
  tch.table.9[,4]+ rnorm(n.sch*900, 0, sigma)
y =0
for (i in 1:49){
  y = delta000 + delta001*data$Z1+ delta010*data$W1+
    delta020*data$W2 + delta100*data$X1 +
    d01.96[,i+1]*data$W1+sch.table[,i+354]*data$X1+
```

```
      tch.table.9[,2*i+5]*data$W1+std.table[,i+4]+

      tch.table.9[,2*i+4]+ rnorm(n.sch*900, 0, sigma)

    Y33 =cbind(Y33,y)




}



Y34=delta000 + delta001*data$Z1+ delta010*data$W1+

  delta020*data$W2 + delta100*data$X1 +

  d01.783[,1] *data$W1+sch.table[,354]*data$X1+

  tch.table.9[,5]*data$W1+std.table[,4]+

  tch.table.9[,4]+ rnorm(n.sch*900, 0, sigma)

y =0

for (i in 1:49){

  y = delta000 + delta001*data$Z1+ delta010*data$W1+

    delta020*data$W2 + delta100*data$X1 +

    d01.783[,i+1]*data$W1+sch.table[,i+354]*data$X1+

    tch.table.9[,2*i+5]*data$W1+std.table[,i+4]+

    tch.table.9[,2*i+4]+ rnorm(n.sch*900, 0, sigma)

  Y34 =cbind(Y34,y)




}



Y35=delta000 + delta001*data$Z1+ delta010*data$W1+

  delta020*data$W2 + delta100*data$X1 +

  d01.5[,1] *data$W1+sch.table[,354]*data$X1+

  tch.table.9[,5]*data$W1+std.table[,4]+

  tch.table.9[,4]+ rnorm(n.sch*900, 0, sigma)

y =0

for (i in 1:49){

  y = delta000 + delta001*data$Z1+ delta010*data$W1+
```

```
    delta020*data$W2 + delta100*data$X1 +

    d01.5[,i+1]*data$W1+sch.table[,i+354]*data$X1+

    tch.table.9[,2*i+5]*data$W1+std.table[,i+4]+

    tch.table.9[,2*i+4]+ rnorm(n.sch*900, 0, sigma)

  Y35 =cbind(Y35,y)



}



Y36=delta000 + delta001*data$Z1+ delta010*data$W1+

  delta020*data$W2 + delta100*data$X1 +

  d01.4[,1] *data$W1+sch.table[,354]*data$X1+

  tch.table.9[,5]*data$W1+std.table[,4]+

  tch.table.9[,4]+ rnorm(n.sch*900, 0, sigma)

y =0

for (i in 1:49){

  y = delta000 + delta001*data$Z1+ delta010*data$W1+

    delta020*data$W2 + delta100*data$X1 +

    d01.4[,i+1]*data$W1+sch.table[,i+354]*data$X1+

    tch.table.9[,2*i+5]*data$W1+std.table[,i+4]+

    tch.table.9[,2*i+4]+ rnorm(n.sch*900, 0, sigma)

  Y36 =cbind(Y36,y)



}



Y37=delta000 + delta001*data$Z1+ delta010*data$W1+

  delta020*data$W2 + delta100*data$X1 +

  d01.3[,1] *data$W1+sch.table[,354]*data$X1+

  tch.table.9[,5]*data$W1+std.table[,4]+

  tch.table.9[,4]+ rnorm(n.sch*900, 0, sigma)

y =0
```

```
for (i in 1:49){

  y = delta000 + delta001*data$Z1+ delta010*data$W1+

    delta020*data$W2 + delta100*data$X1 +

    d01.3[,i+1]*data$W1+sch.table[,i+354]*data$X1+

    tch.table.9[,2*i+5]*data$W1+std.table[,i+4]+

    tch.table.9[,2*i+4]+ rnorm(n.sch*900, 0, sigma)

  Y37 =cbind(Y37,y)




}


Y38=delta000 + delta001*data$Z1+ delta010*data$W1+

  delta020*data$W2 + delta100*data$X1 +

  d01.2[,1] *data$W1+sch.table[,354]*data$X1+

  tch.table.2[,5]*data$W1+std.table[,4]+

  tch.table.2[,4]+ rnorm(n.sch*900, 0, sigma)

y =0

for (i in 1:49){

  y = delta000 + delta001*data$Z1+ delta010*data$W1+

    delta020*data$W2 + delta100*data$X1 +

    d01.2[,i+1]*data$W1+sch.table[,i+354]*data$X1+

    tch.table.2[,2*i+5]*data$W1+std.table[,i+4]+

    tch.table.2[,2*i+4]+ rnorm(n.sch*900, 0, sigma)

  Y38 =cbind(Y38,y)




}

Y39=delta000 + delta001*data$Z1+ delta010*data$W1+

  delta020*data$W2 + delta100*data$X1 +

  d01.3[,1] *data$W1+sch.table[,354]*data$X1+

  tch.table.2[,5]*data$W1+std.table[,4]+

  tch.table.2[,4]+ rnorm(n.sch*900, 0, sigma)
```

119

```
y =0

for (i in 1:49){

  y = delta000 + delta001*data$Z1+ delta010*data$W1+

    delta020*data$W2 + delta100*data$X1 +

    d01.3[,i+1]*data$W1+sch.table[,i+354]*data$X1+

    tch.table.2[,2*i+5]*data$W1+std.table[,i+4]+

    tch.table.2[,2*i+4]+ rnorm(n.sch*900, 0, sigma)

  Y39 =cbind(Y39,y)




}


Y40=delta000 + delta001*data$Z1+ delta010*data$W1+

  delta020*data$W2 + delta100*data$X1 +

  d01.4[,1] *data$W1+sch.table[,354]*data$X1+

  tch.table.2[,5]*data$W1+std.table[,4]+

  tch.table.2[,4]+ rnorm(n.sch*900, 0, sigma)

y =0

for (i in 1:49){

  y = delta000 + delta001*data$Z1+ delta010*data$W1+

    delta020*data$W2 + delta100*data$X1 +

    d01.4[,i+1]*data$W1+sch.table[,i+354]*data$X1+

    tch.table.2[,2*i+5]*data$W1+std.table[,i+4]+

    tch.table.2[,2*i+4]+ rnorm(n.sch*900, 0, sigma)

  Y40 =cbind(Y40,y)




}


Y41=delta000 + delta001*data$Z1+ delta010*data$W1+

  delta020*data$W2 + delta100*data$X1 +

  d01.5[,1] *data$W1+sch.table[,354]*data$X1+
```

```
   tch.table.2[,5]*data$W1+std.table[,4]+

   tch.table.2[,4]+ rnorm(n.sch*900, 0, sigma)

y =0

for (i in 1:49){

  y = delta000 + delta001*data$Z1+ delta010*data$W1+

    delta020*data$W2 + delta100*data$X1 +

    d01.5[,i+1]*data$W1+sch.table[,i+354]*data$X1+

    tch.table.2[,2*i+5]*data$W1+std.table[,i+4]+

    tch.table.2[,2*i+4]+ rnorm(n.sch*900, 0, sigma)

  Y41 =cbind(Y41,y)




}




Y42=delta000 + delta001*data$Z1+ delta010*data$W1+

  delta020*data$W2 + delta100*data$X1 +

  d01.64[,1] *data$W1+sch.table[,354]*data$X1+

  tch.table.2[,5]*data$W1+std.table[,4]+

  tch.table.2[,4]+ rnorm(n.sch*900, 0, sigma)

y =0

for (i in 1:49){

  y = delta000 + delta001*data$Z1+ delta010*data$W1+

    delta020*data$W2 + delta100*data$X1 +

    d01.64[,i+1]*data$W1+sch.table[,i+354]*data$X1+

    tch.table.2[,2*i+5]*data$W1+std.table[,i+4]+

    tch.table.2[,2*i+4]+ rnorm(n.sch*900, 0, sigma)

  Y42 =cbind(Y42,y)




}
```

```
Y43=delta000 + delta001*data$Z1+ delta010*data$W1+

  delta020*data$W2 + delta100*data$X1 +

  d01.025[,1] *data$W1+sch.table[,354]*data$X1+

  tch.table.1[,5]*data$W1+std.table[,4]+

  tch.table.1[,4]+ rnorm(n.sch*900, 0, sigma)

y =0

for (i in 1:49){

  y = delta000 + delta001*data$Z1+ delta010*data$W1+

    delta020*data$W2 + delta100*data$X1 +

    d01.025[,i+1]*data$W1+sch.table[,i+354]*data$X1+

    tch.table.1[,2*i+5]*data$W1+std.table[,i+4]+

    tch.table.1[,2*i+4]+ rnorm(n.sch*900, 0, sigma)

  Y43 =cbind(Y43,y)



}


Y44=delta000 + delta001*data$Z1+ delta010*data$W1+

  delta020*data$W2 + delta100*data$X1 +

  d01.01[,1] *data$W1+sch.table[,354]*data$X1+

  tch.table.1[,5]*data$W1+std.table[,4]+

  tch.table.1[,4]+ rnorm(n.sch*900, 0, sigma)

y =0

for (i in 1:49){

  y = delta000 + delta001*data$Z1+ delta010*data$W1+

    delta020*data$W2 + delta100*data$X1 +

    d01.01[,i+1]*data$W1+sch.table[,i+354]*data$X1+

    tch.table.1[,2*i+5]*data$W1+std.table[,i+4]+

    tch.table.1[,2*i+4]+ rnorm(n.sch*900, 0, sigma)

  Y44 =cbind(Y44,y)
```

```
}


Y45=delta000 + delta001*data$Z1+ delta010*data$W1+
  delta020*data$W2 + delta100*data$X1 +
  d01.014[,1] *data$W1+sch.table[,354]*data$X1+
  tch.table.1[,5]*data$W1+std.table[,4]+
  tch.table.1[,4]+ rnorm(n.sch*900, 0, sigma)
y =0
for (i in 1:49){
  y = delta000 + delta001*data$Z1+ delta010*data$W1+
    delta020*data$W2 + delta100*data$X1 +
    d01.014[,i+1]*data$W1+sch.table[,i+354]*data$X1+
    tch.table.1[,2*i+5]*data$W1+std.table[,i+4]+
    tch.table.1[,2*i+4]+ rnorm(n.sch*900, 0, sigma)
  Y45 =cbind(Y45,y)



}


Y46=delta000 + delta001*data$Z1+ delta010*data$W1+
  delta020*data$W2 + delta100*data$X1 +
  d01.073[,1] *data$W1+sch.table[,354]*data$X1+
  tch.table.2[,5]*data$W1+std.table[,4]+
  tch.table.2[,4]+ rnorm(n.sch*900, 0, sigma)
y =0
for (i in 1:49){
  y = delta000 + delta001*data$Z1+ delta010*data$W1+
    delta020*data$W2 + delta100*data$X1 +
    d01.073[,i+1]*data$W1+sch.table[,i+354]*data$X1+
    tch.table.2[,2*i+5]*data$W1+std.table[,i+4]+
    tch.table.2[,2*i+4]+ rnorm(n.sch*900, 0, sigma)
  Y46 =cbind(Y46,y)
```

```
}


Y47=delta000 + delta001*data$Z1+ delta010*data$W1+
  delta020*data$W2 + delta100*data$X1 +
  d01.053[,1] *data$W1+sch.table[,354]*data$X1+
  tch.table.9[,5]*data$W1+std.table[,4]+
  tch.table.9[,4]+ rnorm(n.sch*900, 0, sigma)
y =0
for (i in 1:49){
  y = delta000 + delta001*data$Z1+ delta010*data$W1+
    delta020*data$W2 + delta100*data$X1 +
    d01.053[,i+1]*data$W1+sch.table[,i+354]*data$X1+
    tch.table.9[,2*i+5]*data$W1+std.table[,i+4]+
    tch.table.9[,2*i+4]+ rnorm(n.sch*900, 0, sigma)
  Y47 =cbind(Y47,y)


}


YY <- cbind(Y1, Y2,Y3, Y4, Y5, Y6, Y7, Y8, Y9,Y10,Y11,
            Y12, Y13, Y14,Y15,Y16, Y17, Y18, Y19, Y20
            , Y21, Y22, Y23, Y24, Y25, Y26,Y27, Y28,
            Y29, Y30, Y31, Y32, Y33, Y34, Y35, Y36,
            Y37, Y38,Y39,Y40,Y41,Y42,Y43,Y44,Y45,
            Y46,Y47)



data <- data.frame(planedata, YY)
names(data)[9:2358] <- paste("y", 1:2350, sep="")


#Save the data for lme4
```

```
write.csv(data, "lme4data.csv")



# Save the data for HLM7

hlmtest <- data %>% dplyr::select(1:4, y1:y2350)

%>%arrange(std)

write_sav(hlmtest, "hlmtest.sav")


hlmstd2 <- data %>% dplyr::select(std, W1)

%>% arrange(std)

hlmstd <- unique(hlmstd2[,1:2] )

write_sav(hlmstd, "hlmstd.sav")



hlmtch2 <- data %>%

dplyr::select(tch,sch, W2)

hlmtch <- unique(hlmtch2[,1:3] )

write_sav(hlmtch, "hlmtch.sav")



hlmsch2 <- data %>%

dplyr::select(sch, Z1)

hlmsch <- unique(hlmsch2[,1:2] )

write_sav(hlmsch, "hlmsch.sav")
```
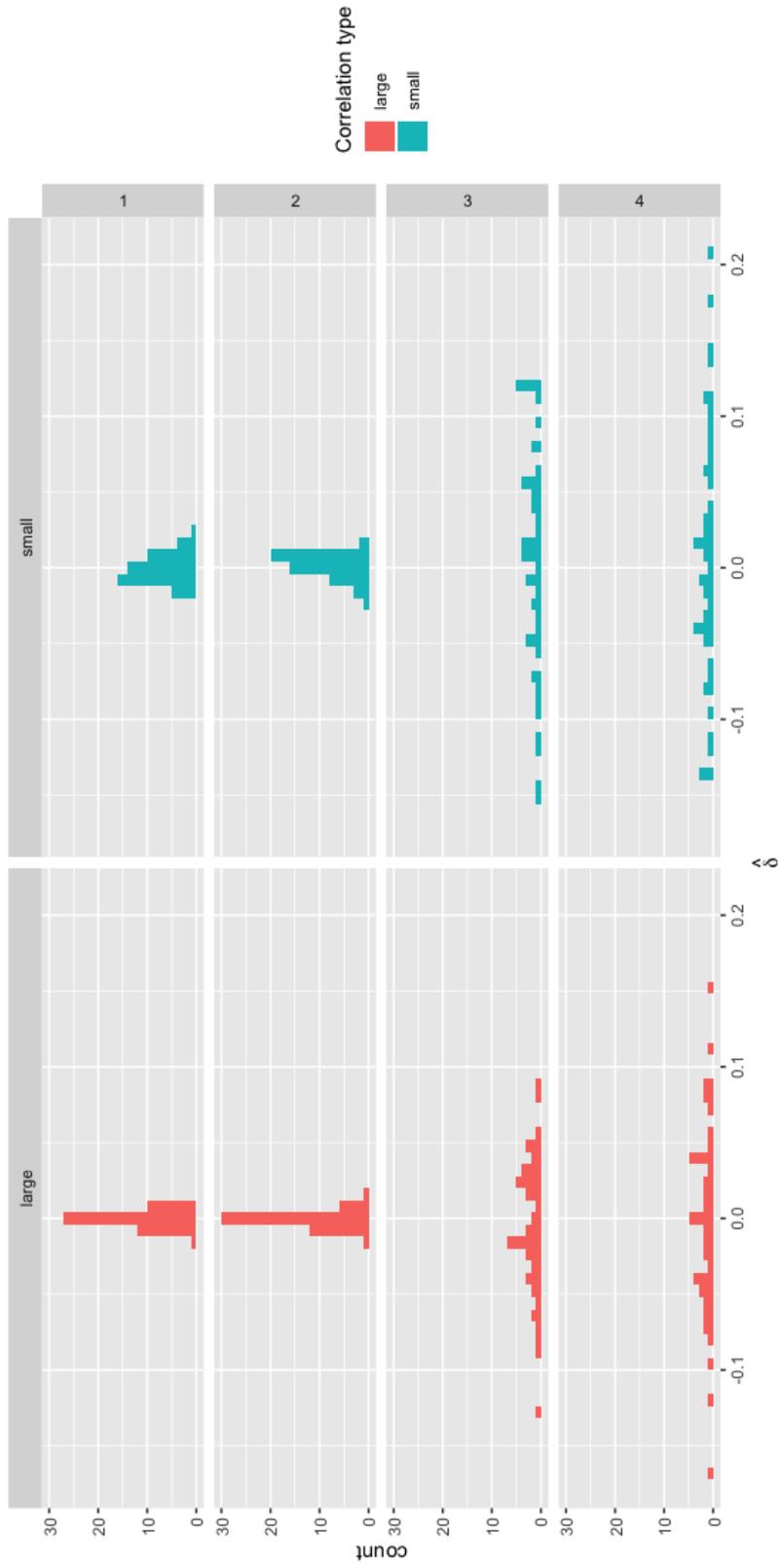
Figure 3: Histogram of the relative bias of $\hat{\delta}_{000}$
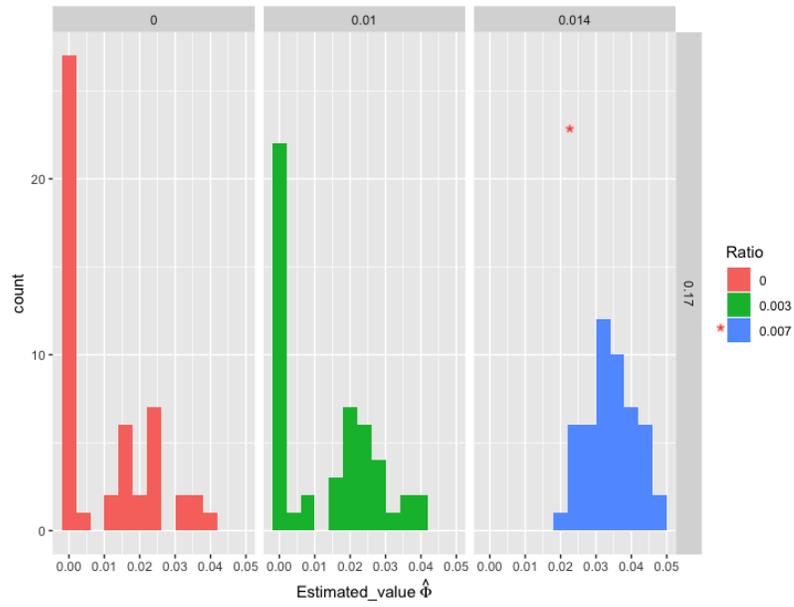
(a) * The ratio $(\frac{\phi_{00}^2}{\tau_{tch11}^2 + \phi_{00}^2}) = 0.007$
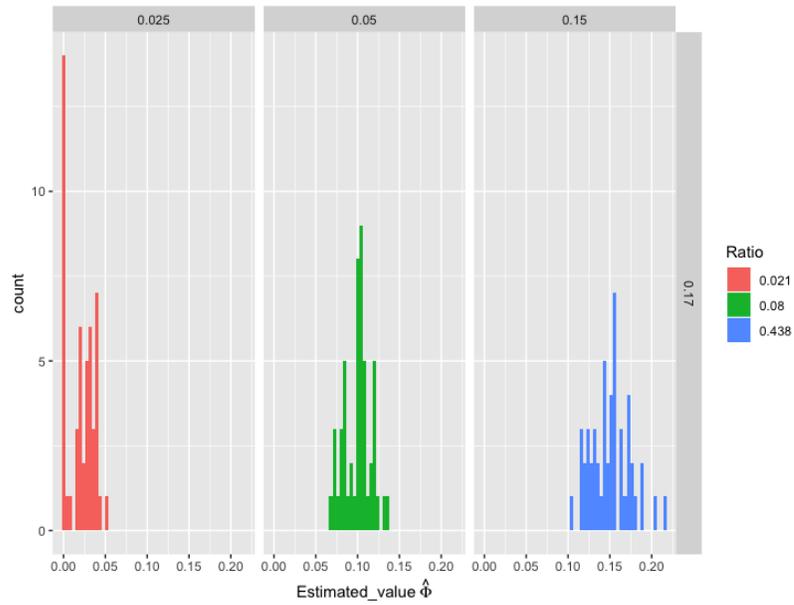
(b) The graph is symmetric when ϕ_{00} is large

Figure 4: Compare six histograms of the $\hat{\phi}_{00}$ with $\tau_{tch=0.17}$
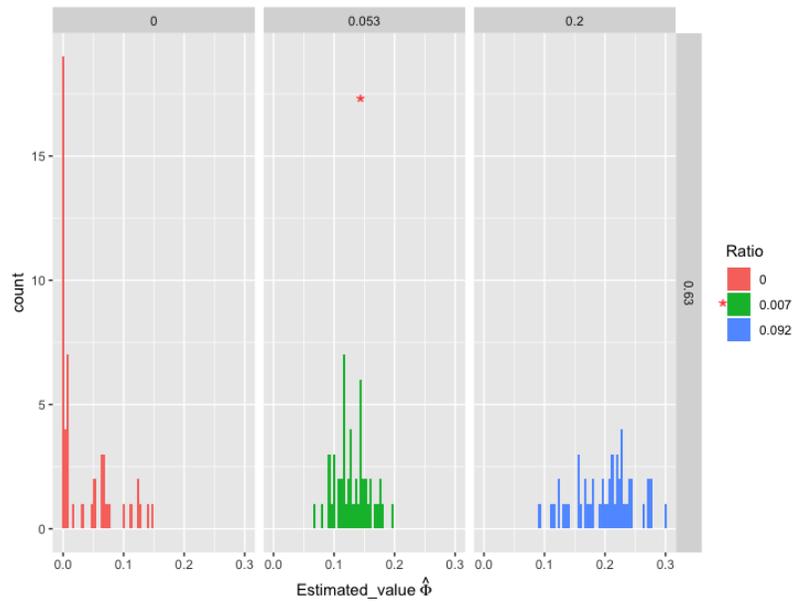
(a) * The ratio $(\dfrac{\phi_{00}^2}{\tau_{tch11}^2 + \phi_{00}^2}) = 0.007$
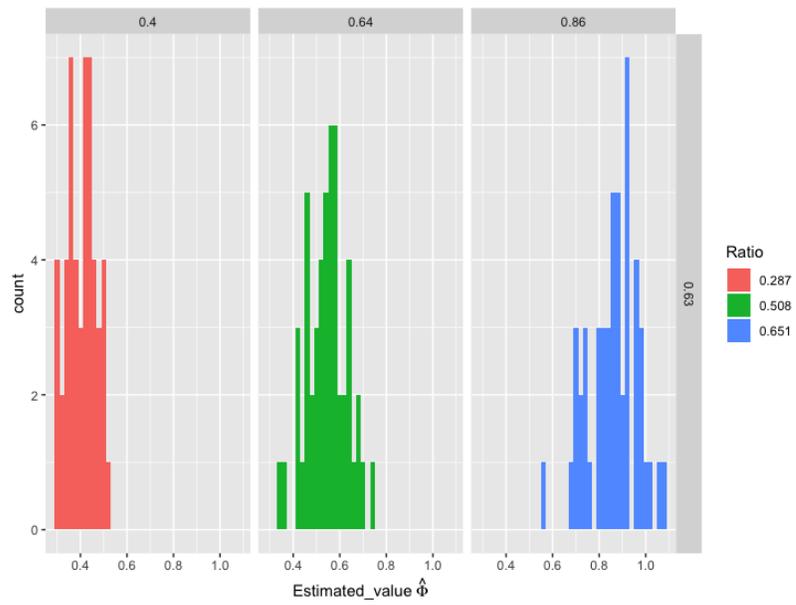
(b) The graph is symmetric when ϕ_{00} is large

Figure 5: Compare six histograms of the $\hat{\phi}_{00}$ with $\tau_{tch=0.63}$

128
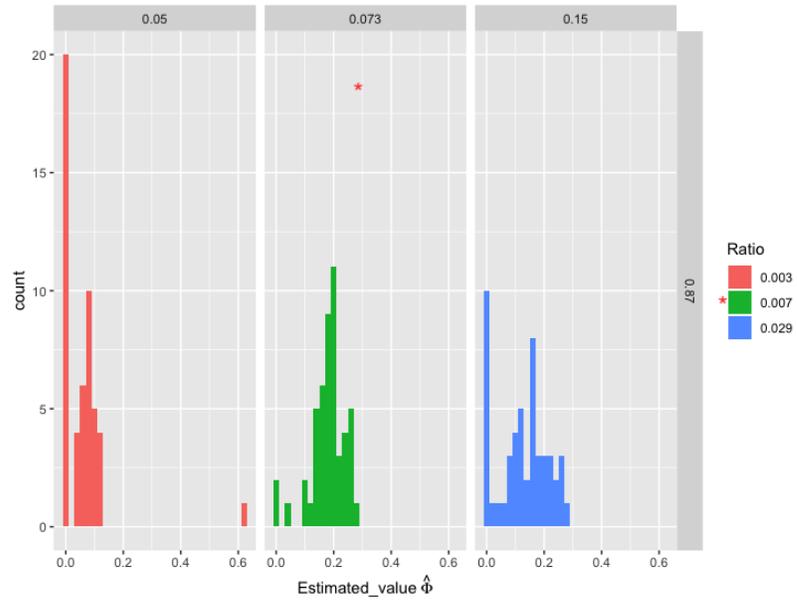
(a) * The ratio $(\dfrac{\phi_{00}^2}{\tau_{tch11}^2 + \phi_{00}^2}) = 0.007$
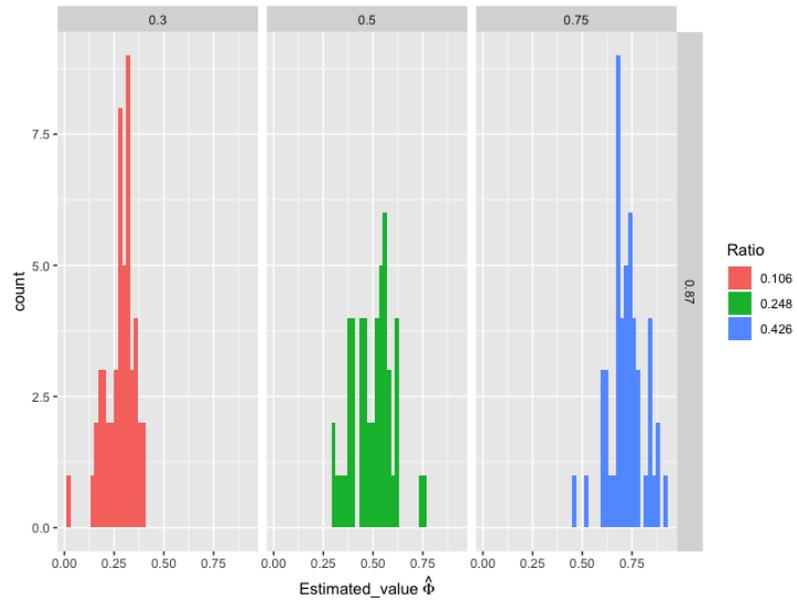
(b) The graph is symmetric when ϕ_{00} is large

Figure 6: Compare six histograms of the $\hat{\phi}_{00}$ with $\tau_{tch=0.87}$
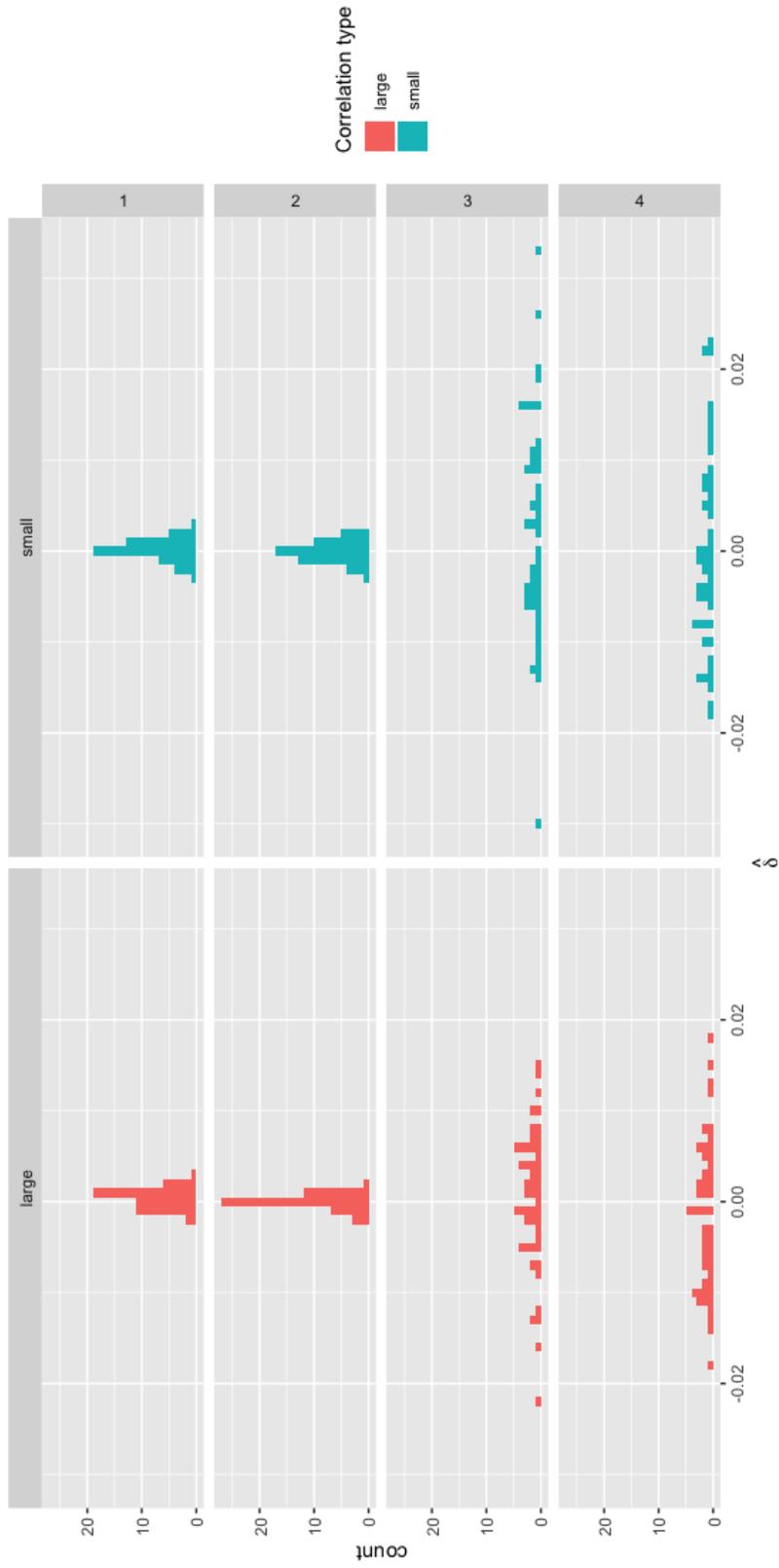
Figure 7: Histogram of the relative bias of $\hat{\delta}_{001}$

130
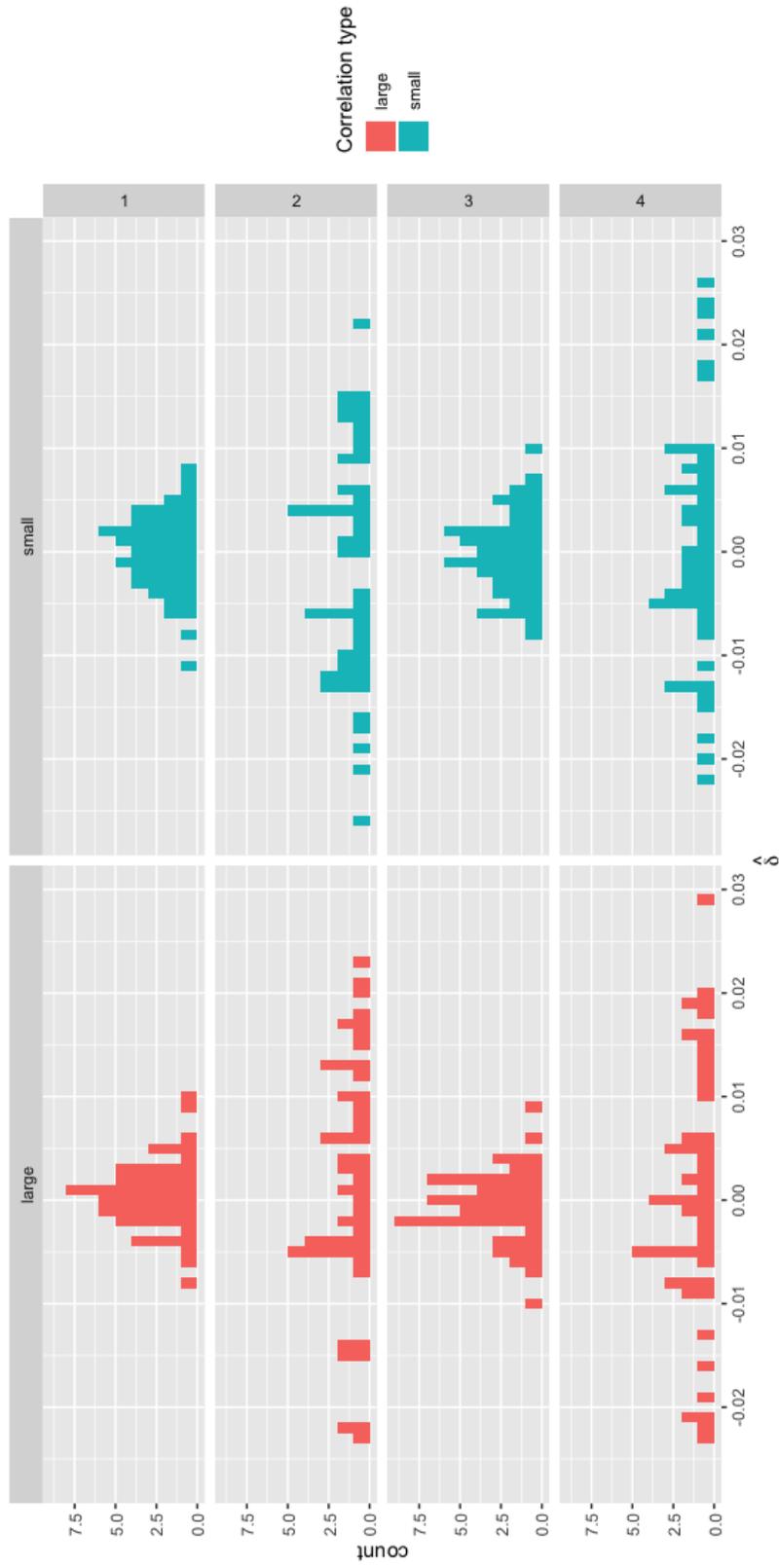
Figure 8: Explain The Relative Bias of δ_{010} with the Correlation between τ_{tch00} and τ_{tch11}
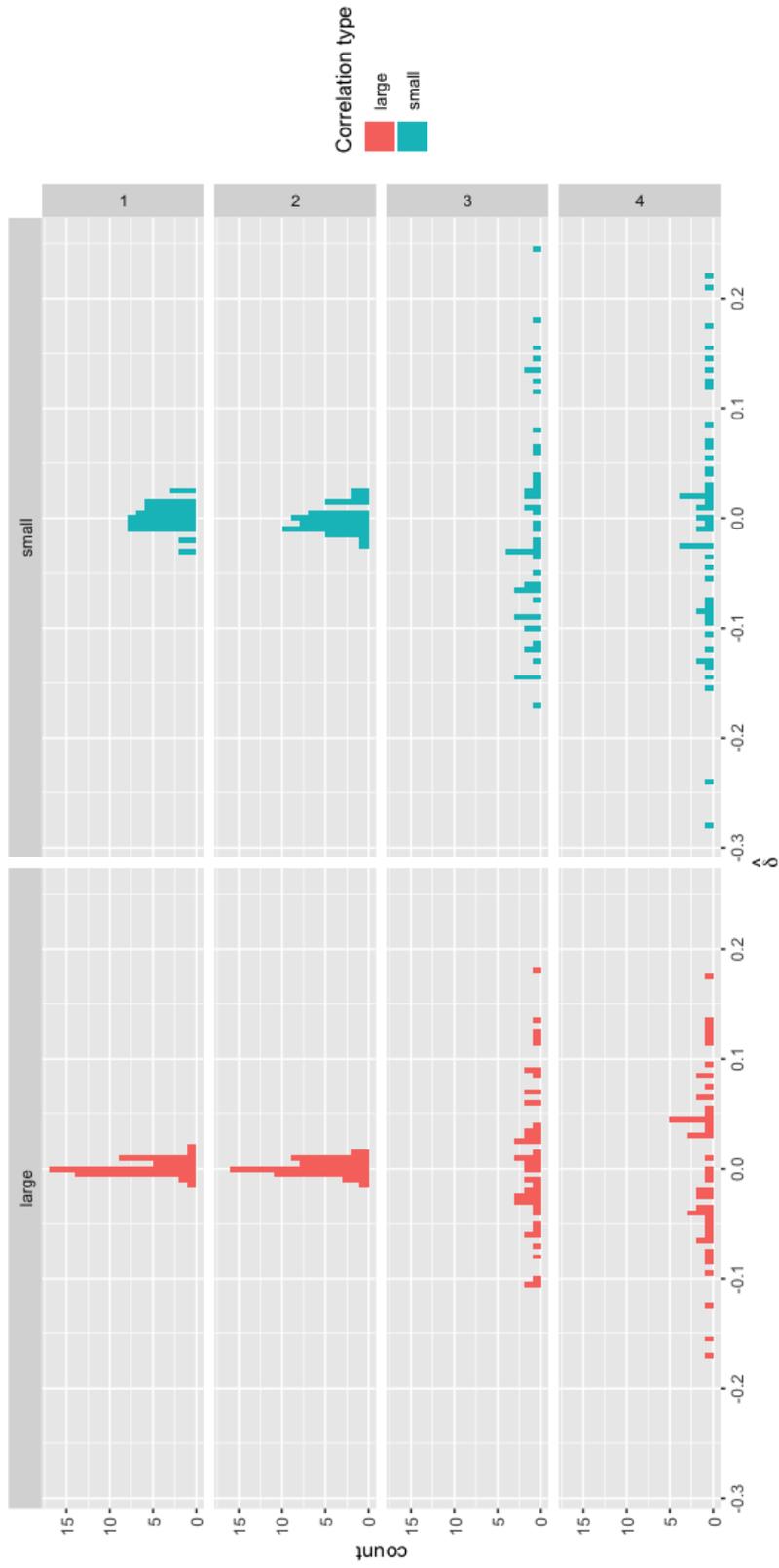
Figure 9: Histogram of the relative bias of $\hat{\delta}_{020}$
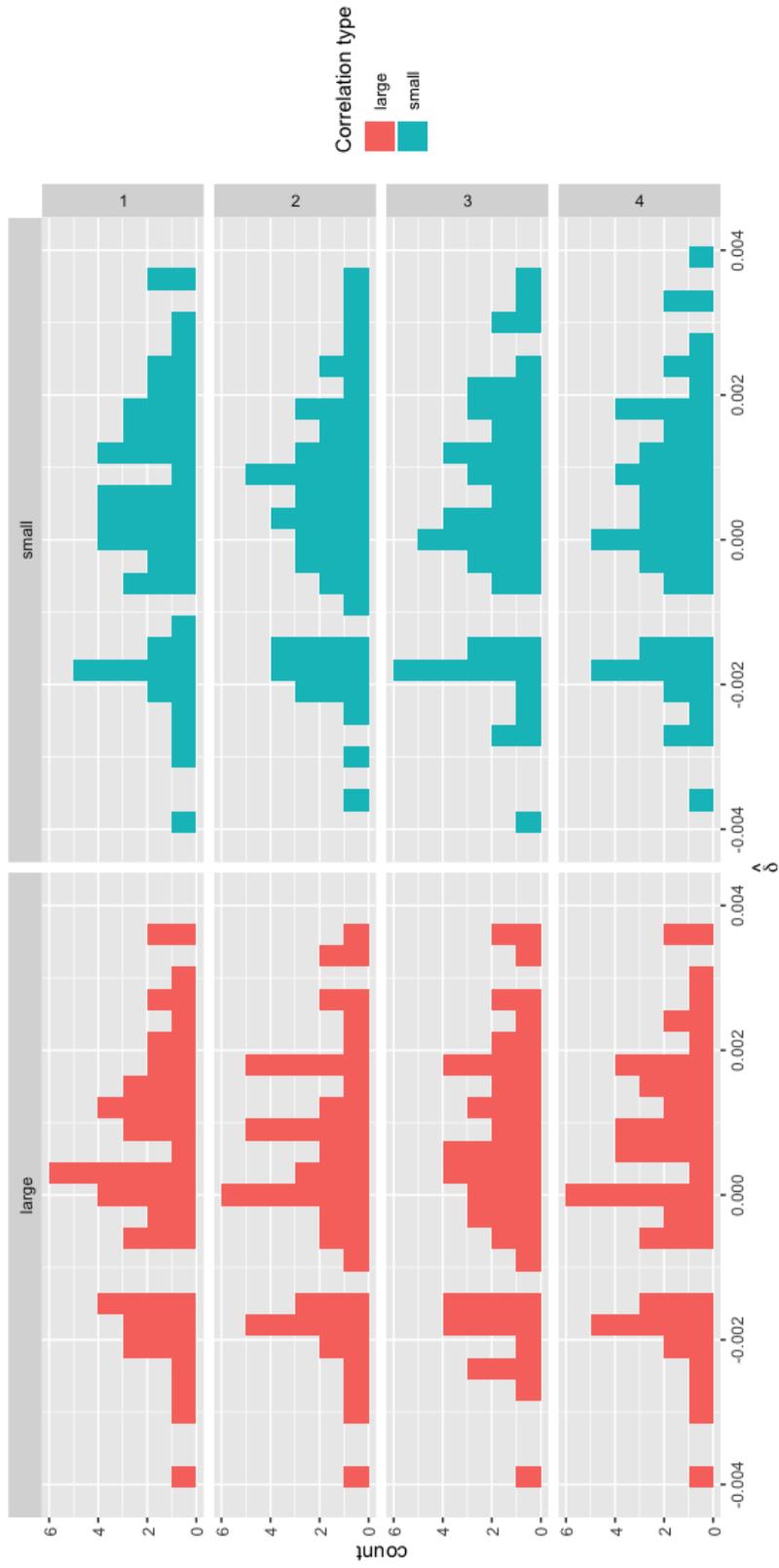
Figure 10: Histogram of the relative bias of $\hat{\delta}_{100}$
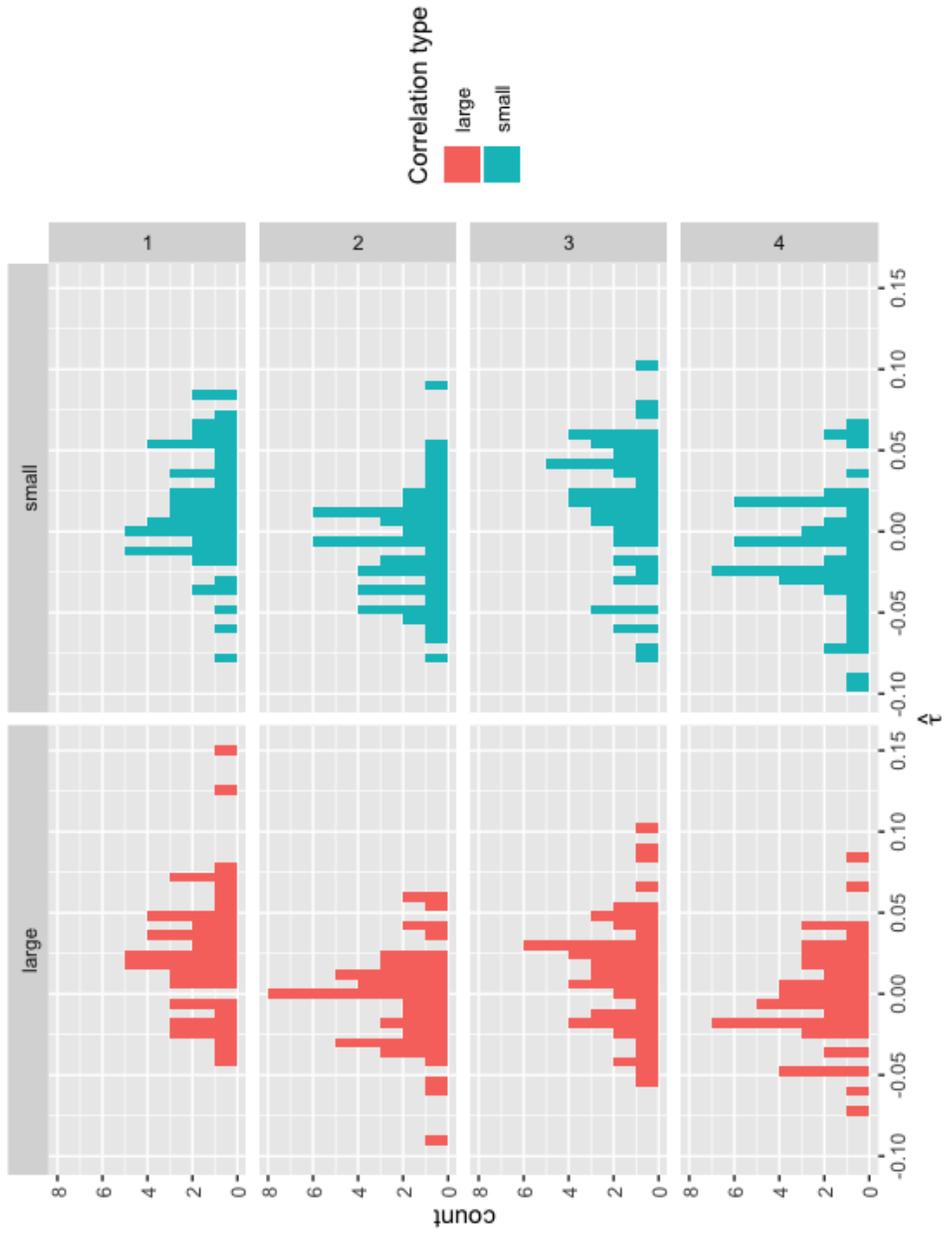
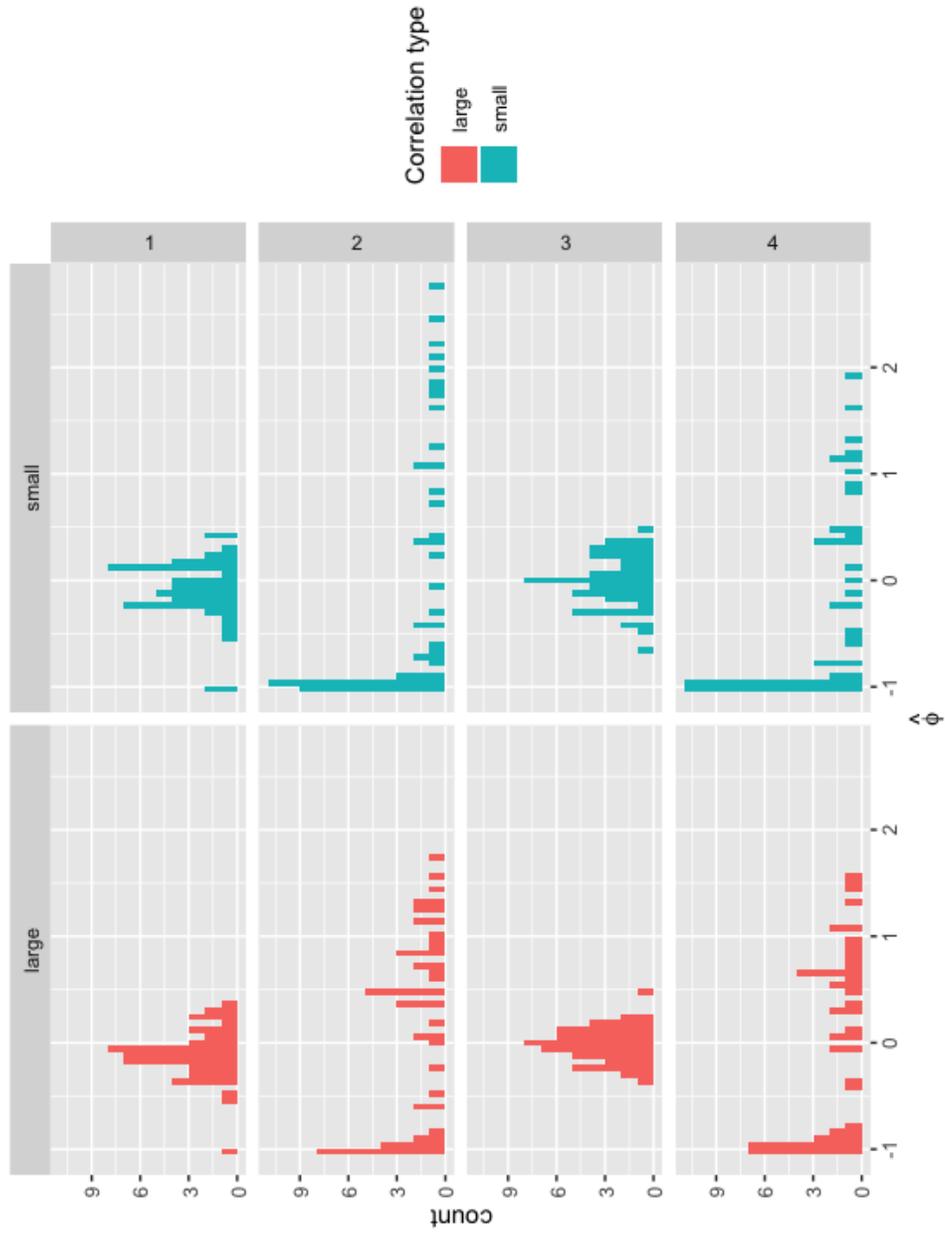Figure 11: Explain The Relative Bias of τ_{tch11} with the Correlation between τ_{tch00} and τ_{tch11}

134

Figure 12: Explain The Relative Bias of ϕ_{00} with the Correlation between τ_{tch00} and τ_{11}
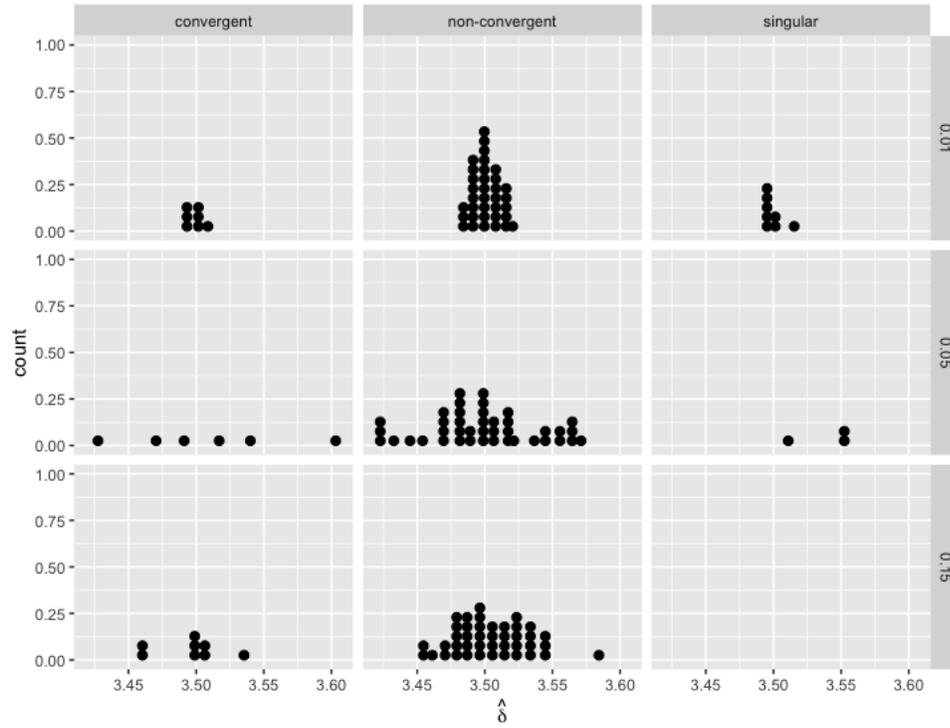For better histogram scale, an outlier has been removed

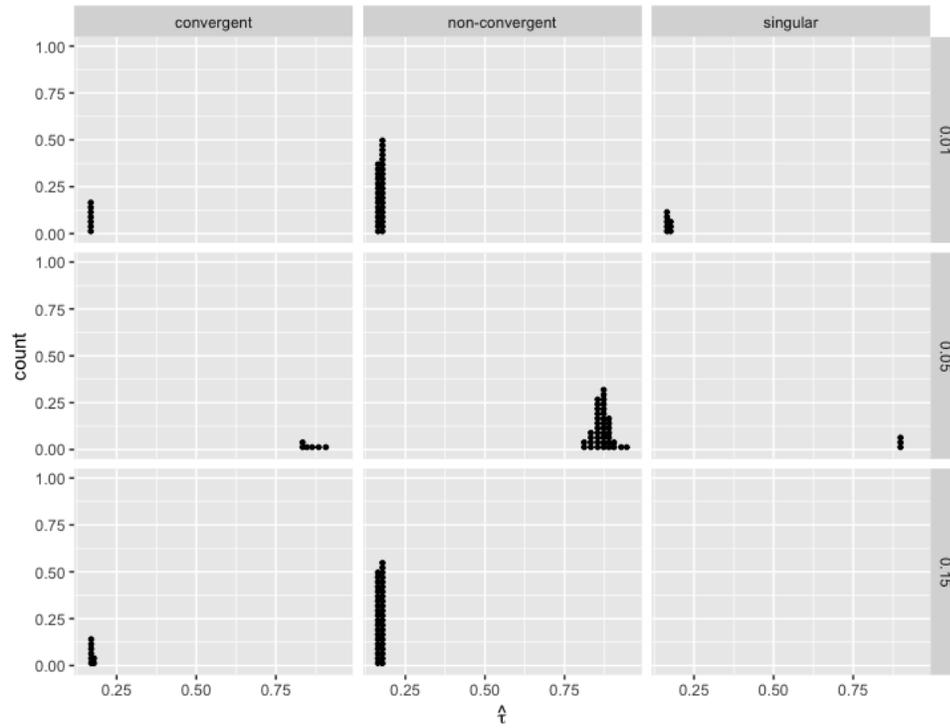Figure 13: The Effect of the Convergence and Singularity on $\hat{\delta}_{010}$

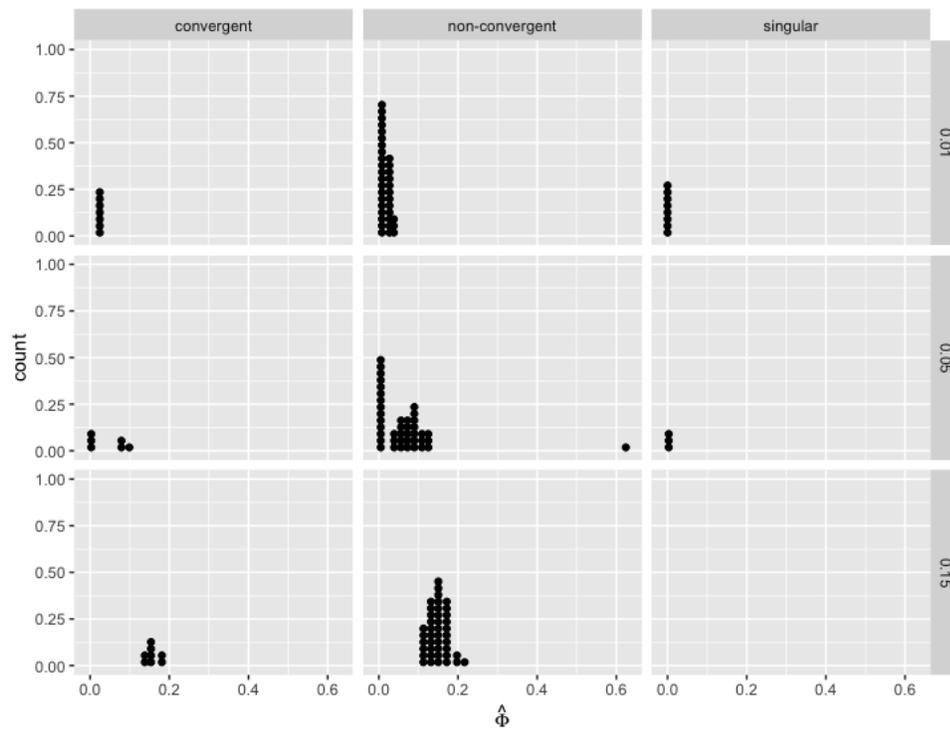Figure 14: The Effect of the Convergence and Singularity on $\hat{\tau}_{tch11}$

Figure 15: The Effect of the Convergence and Singularity on $\hat{\phi}_{00}$

REFERENCES

Bates, D., Mächler, M., Bolker, B., and Walker, S. (2015). Fitting linear mixed-effects models using lme4. *Journal of Statistical Software*, 67(1):1–48.

Boedeker, P. (2017). Hierarchical Linear Modeling with Maximum Likelihood, Restricted Maximum Likelihood, and Fully Bayesian Estimation. *Practical Assessment, Research & Evaluation*, 22(2):1–19.

Bolin, D. (2017). A brief introduction to mixed models. Retrieved from `https://golem.ph.utexas.edu/category/2014/12/effective_sample_size.html`.

Finch W. Holmes, Bolin, J. E. and Kelley, K. (2014). *Multilevel Modeling Using R.* CRC Press Taylor & Francis Group, Boca Raton, FL, USA, 1st edition.

Galindo and Newton, J. (2017). Proceedings of the 39th annual meeting of the North American Chapter of the International Group for the Psychology of Mathematics Education. *Hoosier Association of Mathematics Teacher Educators*, page pp.537.

Guyot, L., Melhuish, K. M., and Fagan, J. (2017). Effect of Teachers' Participation in a Professional Development on Student Achievement: A Longitudinal Large-Scale Study. *Journal of Educational and Behavioral Statistics*.

Magnusson, K. (2015). Using r and lme/lmer: to fit different two- and three-level longitudinal models. Retrieved from `https://rpsychologist.com/r-guide-longitudinal-lme-lmer`.

McCoach, D. B., Rifenbark, G. G., Newton, S. D., Li, X., Kooken, J., Yomtov, D., Gambino, A. J., and Bellara, A. (2018). Does the package matter? a comparison of five common multilevel modeling software packages. *Journal of Educational and Behavioral Statistics*, 43(5):594–627.

Montgomery, D. C. (2012). *Design and Analysis of Experiments.* Wiley, Hoboken, NJ, USA, 8th edition.

Montgomery, Douglas C., P. E. A. and Geoffrey, V. G. (2012). *Intorduction to Linear Regression Analysis.* Wiley, Hoboken, NJ, USA, 5th edition.

Nedler, J. A. and Mead, R. (1965). A Simplex Method for Function Minimization. *The Computer Journal*, pages 1–6.

Newsom (2017). Significance testing in multilevel regression. Retrieved from `http://web.pdx.edu/~newsomj/mlrclass/ho_significance.pdf`.

Powell, M. J. (2009). The bobyqa algorithm for bound constrained optimization without derivatives. *Cambridge NA Report NA2009/06, University of Cambridge, Cambridge*, pages 26–46.

Raundenbush, S. and Bryk, A. S. (2002). *Hierarchical Linear Models.* Sage, Thousand Oaks, CA, USA, 2nd edition.

Razali, N. M., Wah, Y. B., et al. (2011). Power comparisons of shapiro-wilk, kolmogorov-smirnov, lilliefors and anderson-darling tests. *Journal of statistical modeling and analytics*, 2(1):21–33.

Shapiro, S. S. and Wilk, M. B. (1965). An analysis of variance test for normality (complete samples). *Biometrika*, 52(3/4):591–611.

Wackerly, D., Mendenhall III, W., and Scheaffer, R. (2008). *Mathematical Statistics with Applications.* Brooks/Cole, 10 Davis Drive, Belmont,Ca, USA, 7st edition.

Zhang, X. (2015). A tutorial on restricted maximum likelihood estimation in linear regression and linear mixed-effects model. *URL http://statdb1. uos. ac. kr/teaching/multi-grad/ReML. pdf.*