12-1-2018

Bayesian Estimation of Mixture IRT Models using NUTS

Rahab Al Hakmani
Southern Illinois University Carbondale, rehab.hekmani@gmail.com

BAYESIAN ESTIMATION OF MIXTURE IRT MODELS USING NUTS

by

Rahab Al Hakmani

B.Ed., Sultan Qaboos University, Oman, 1999
M.Ed., Sultan Qaboos University, Oman, 2007

A Dissertation
Submitted in Partial Fulfillment of the Requirements for the
Doctor of Philosophy degree

Department of Counseling, Quantitative Methods, and Special Education
in the Graduate School
Southern Illinois University Carbondale
December 2018

DISSERTATION APPROVAL

BAYESIAN ESTIMATION OF MIXTURE IRT MODELS USING NUTS

by

Rahab Al Hakmani

A Dissertation Submitted in Partial

Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy

in the field of Quantitative Methods

Approved by:

Yanyan Sheng, Chair

Todd Headrick

Jennifer Koran

Bhaskar Bhattacharya

David Olive

Graduate School
Southern Illinois University Carbondale
October 18, 2018

AN ABSTRACT OF THE DISSERTATION OF

Rahab Al Hakmani, for the Doctor of Philosophy degree in Quantitative Methods, presented on October 18, 2018, at Southern Illinois University Carbondale.

TITLE: BAYESIAN ESTIMATION OF MIXTURE IRT MODELS USING NUTS

MAJOR PROFESSOR: Dr. Yanyan Sheng

The No-U-Turn Sampler (NUTS) is a relatively new Markov chain Monte Carlo (MCMC) algorithm that avoids the random walk behavior that common MCMC algorithms such as Gibbs sampling or Metropolis Hastings usually exhibit. Given the fact that NUTS can efficiently explore the entire space of the target distribution, the sampler converges to high-dimensional target distributions more quickly than other MCMC algorithms and is hence less computational expensive. The focus of this study is on applying NUTS to one of the complex IRT models, specifically the two-parameter mixture IRT (Mix2PL) model, and further to examine its performance in estimating model parameters when sample size, test length, and number of latent classes are manipulated. The results indicate that overall, NUTS performs well in recovering model parameters. However, the recovery of the class membership of individual persons is not satisfactory for the three-class conditions. Also, the results indicate that WAIC performs better than LOO in recovering the number of latent classes, in terms of the proportion of the time the correct model was selected as the best fitting model. However, when the effective number of parameters was also considered in selecting the best fitting model, both fully Bayesian fit indices perform equally well. In addition, the results suggest that when multiple latent classes exist, using either fully Bayesian fit indices (WAIC or LOO) would not select the conventional IRT model. On the other hand, when all examinees came from a single unified population, fitting MixIRT models using NUTS causes problems in convergence.

ACKNOWLEDGMENTS

Foremost, I would like to express my sincere gratitude and deep appreciation to my advisor Dr. Yanyan Sheng for her continuous support, guidance, patience, and encouragement throughout my entire doctoral study. It was an honor working with her and learning from her knowledge and expertise. I would also like to extend my heartfelt thanks to my other committee members, Dr. Bhaskar Bhattacharya, Dr. David Olive, Dr. Jennifer Koran, and Dr. Todd Headrick. Their insightful comments and suggestions along with their time are very much appreciated.

My sincere thanks also go to my colleagues in the department, Yevgeniy Ptukhin, Meng-I Chang, and Reginald Ziedzor who provided a friendly and supportive environment to learn. A special note of appreciation goes to my friend Fathima Jaffari with whom the doctoral life with its ups and downs was a wonderful experience.

Appreciation is extended to Majid Memari for helping me getting over the hurdle of carrying out the simulations on the computing cluster, BigDawg. Also, I would like to thank Stan developers and members of Stan forums who provided valuable advice with coding using Stan programming language.

Finally, I would like to express my gratitude to my mother, sisters, and brothers for their endless love and for always believing in me.

DEDICATION

To the soul of my father ...

To my beloved mother ...

TABLE OF CONTENTS

CHAPTER PAGE

CHAPTERS

LIST OF TABLES

TABLE PAGE

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Tests, and especially achievement tests are extensively used in different context such as schools, government, and industry. In educational and psychological measurement, test results can be used for various purposes such as screening and selection of individuals, assessing students' learning progress, or evaluating the efficiency of educational systems. The increased awareness of the importance and impact of testing has led to the development of better tests and the improvement of statistical methods for analyzing test scores. Classical test theory (CTT; Novick, 1966) has served the measurement community well for most of the last century. However, problems emerged using CTT have encouraged the development of a modern test theory, namely the item response theory (IRT; Lord, 1980), which has become a fundamental tool for measurement professionals in behavioral sciences (Linden & Hambleton, 1997). IRT provides advantages over CTT that make it applicable in many educational fields, such as test development and equating (Skaggs & Lissitz, 1986), computerized adaptive testing (CAT; Linden & Glas, 2000), and differential item functioning (DIF; Holland & Wainer, 1993). Among the many advantages of IRT over CTT, measurement invariance is one of the more important ones. In contrast to CTT, where item and person characteristics is sample dependent, the corresponding characteristics are invariant in IRT. Specifically, item parameters (e.g., difficulty or discrimination) do not depend on the sample of persons used to calibrate them. For example, an item difficulty parameter will be the same no matter whether this item is administered to a group of high ability or low ability examinees. Likewise, a person ability does not depend on the sample of items used to estimate it and hence ability estimates obtained from different sets of items will be the same. In addition, in IRT, item and person parameters are placed on the same latent continuum. This makes it possible to scale persons relative to items or vice-versa, and hence we can directly compare them. Because of its advantages over CTT, IRT has gained an increased popularity in educational and psychological testing (e.g., Baker & Kim, 2004; De Ayala, 2009; Hambleton & Jones, 1993).

IRT consists of a family of models that specify the probability of a response given person latent trait and item characteristics. Different models exist for different types of response data. Conventional dichotomous IRT models (e.g., Birnbaum, 1969; Lord & Novick, 1968; Lord, 1980; Rasch, 1960) are used when test items require binary responses such as true-false questions or multiple-choice questions that are scored as correct or incorrect. Such IRT models are based on two major assumptions: unidimensionality and local independence. Unidimensionality states that a single unified latent trait is measured by all test items. Although in practice multiple factors affect the response of each person to individual items, the presence of a dominant factor that explains test performance is sufficient for this assumption to be satisfied. Local independence means that when the latent trait influencing test performance is held constant, persons' responses to any pair of test items are independent of each other. This assumption is related to unidimensionality although local independence is a broader or more general concept. When the assumption of unidimensionality is true, the assumption of local independence is obtained and the two concepts are equivalent.

There are three common unidimensional dichotomous IRT models (Birnbaum, 1969; Lord & Novick, 1968; Lord, 1980; Rasch, 1960) that are based on the number of item parameters. The simplest of such conventional IRT models is the one-parameter logistic (1PL) or Rasch model (Rasch, 1960). The model consists of an item difficulty parameter, which is defined as the ability required for a person to have a probability of 0.5 to answer the item correctly. The two-parameter logistic (2PL; Lord & Novick, 1968) model generalizes the Rasch model by adding the discrimination parameter, which is proportional to the slope at the point of the difficulty level. The three-parameter logistic (3PL) model extends the two-parameter model by adding the pseudo-guessing parameter, which is the probability that a person with an extremely low latent ability answers the item correctly.

The conventional IRT models discussed above assume that the observed response data stem from a homogenous population of individuals. This assumption, however, limits their applications in test situations where, for example, a set of test items can be solved with different cognitive strategies. If the population consists of multiple groups of persons, with each group employing a different strategy for the same item, the parameters for this item will be different across these groups (or subpopulations), and consequently, the conventional IRT models cannot be used for the response data. On the other hand, the conventional IRT models may hold when each of the subpopulations employs a common strategy. As a result, mixture IRT (MixIRT) models (Rost, 1997) were developed to capture the presence of these latent classes (i.e. subpopulations) that are qualitatively different but within which a conventional IRT model holds. In the MixIRT modeling framework, persons are characterized by their location on a continuous latent dimension as well as by their latent class membership. Also, each subpopulation has a unique set of item parameters (e.g., difficulty, or discrimination). MixIRT models have become increasingly popular as a technique for investigating various issues in educational and psychological measurement such as identifying items that function differently across latent groups (e.g., Choi, Alexeev & Cohen, 2015; Cohen & Bolt, 2005; De Ayala, Kim, Stapleton, & Dayton 2002; Maij-de Meij, Kelderman, & van der Flier, 2010; Samuelsen, 2005; Shea, 2013;

Wu, et al., 2017) or detecting test speededness (e.g., Bolt, Cohen, & Wollack, 2002; Meyer, 2010; Mroch, Bolt, & Wollack, 2005; Wollack, Cohen, & Wells, 2003)

The first MixIRT model was developed by Rost (1990), which is a one-parameter MixIRT (Mix1PL) model, also known as the mixture Rasch model, for dichotomous data where conventional Rasch model is assumed to hold within each latent class, but different difficulty parameters apply across classes. Individual members within a class can also have different levels of ability. The two-parameter MixIRT (Mix2PL) model and the three-parameter MixIRT (Mix3PL) model extend the Mix1PL model by adding additional item parameters. Specifically, in the Mix2PL model, the conventional 2PL model is assumed to hold for each latent class, but item difficulty and discrimination parameters may differ for different classes. Similarly, in the Mix3PL model, the conventional 3PL model is assumed to hold for each latent class, but item difficulty, discrimination, and guessing parameters may differ for the different classes. In the MixIRT literature, the Mix1PL model (or its hierarchical forms) is the predominant model, while the Mix2PL and Mix3PL models are rarely covered. This could be due in part to the difficulty of model identification caused by the problem of label switching of mixture proportions that is inherent in mixture models in general. Alternatively, model complexity results in difficulty in parameter estimation using conventional methods. Strategies used within the context of the fully Bayesian estimation to solve the problem of exchangeable mixture proportions and hence identify the MixIRT model will be addressed in a later chapter. The conventional IRT models can be seen as special cases of MixIRT models. Stated differently, if only one latent class is retained after fitting a MixIRT model, then it is reduced to a conventional IRT model.

In the IRT literature, many estimation methods have been developed to jointly estimate parameters of IRT models, with the early focus on maximum likelihood (ML; Fisher, 1922) estimation methods, namely the joint maximum likelihood (JML; Birnbaum, 1969), the conditional maximum likelihood (CML; Andersen, 1970), and the marginal maximum likelihood (MML; Bock & Aitkin, 1981). Because these estimators are related to ML, they may result in infinite or implausible parameter estimates in situations where unusual response patterns are encountered such as perfect or zero scores. On the other hand, Bayesian estimation avoids such problems by specifying appropriate prior distributions for parameters. This way, the Bayesian approach can control the parameters to be within a reasonable range. Due to the advanced computational techniques, estimation of IRT models has gradually shifted to the fully Bayesian estimation. While the traditional techniques find a point estimate for each parameter that maximizes the likelihood function, the fully Bayesian estimation via the use of the Markov chain Monte Carlo (MCMC; Hastings, 1970; Metropolis & Ulam, 1949; Metropolis, Rosenbluth, Rosenbluth, Teller, & Teller, 1953) simulation techniques approximates the joint posterior distribution of all model parameters, and hence account for the uncertainty associated with any parameter estimation. The fully Bayesian approach based on MCMC techniques have been successfully used in estimating parameters of various IRT models with different degree of complexity (e.g., Chang, 2017; de la Torre & Douglas, 2004; Johnson & Junker, 2003; Kim, 2001; Kuo, 2015; Patz & Junker, 1999a, 1999b; Sheng & Wikle, 2007; Sheng, 2010).

MCMC methods are a class of algorithms that can be used to simulate random samples from a posterior distribution via constructing a Markov chain that has the desired distribution as its stationary distribution. The idea is to generate sequential random samples such that each random sample is used as a stepping-stone to generate the next random sample. One requirement of the Markov chain is that a sample generated at any state depends on the sample drawn at the previous state, but does not depend on those simulated at any earlier states (Ravenzwaaij, Cassey, & Brown, 2016). MCMC techniques are especially useful in Bayesian inference because it is extremely flexible and can be applied with very complex models. Two main types of MCMC algorithms exist in the literature. They are (1) random walk algorithms such as Gibbs sampling (Geman & Geman, 1984) and Metropolis-Hastings (MH; Hastings, 1970; Metropolis & Ulam, 1949), and (2) non-random walk algorithms such as Hamiltonian Monte Carlo (HMC; Duane, Kennedy, Pendleton, & Roweth, 1987) and its extension, the no-U-turn sampler (NUTS; Hoffman and Gelman, 2011).

Specifically, the Gibbs sampling algorithm proceeds by drawing random samples of each parameter from its full conditional distribution, based on the previously generated values of all the other parameters. Then the joint posterior distribution can be eventually obtained through an adequate number of iterations. In order to use the Gibbs sampler, the full conditional distribution for each parameter should be in closed form. However, in practice, the full conditional distribution may not always be in a closed form or may be difficult to simulate. An alternative algorithm to estimate model parameters is MH. The algorithm works by selecting a proposal or candidate distribution by the current value of the parameters. Then a proposal move to a new point in the parameter space is randomly generated from the proposal distribution and accepted with a certain amount of probability. The whole process repeats for an adequate number of iterations to eventually approximate the joint posterior distribution. Although the MH method can be applied in many situations, finding an appropriate proposal distribution could sometimes be challenging. Both Gibbs sampling and MH algorithms explore the parameter space via inefficient random walks (Neal, 1992). For complicated models with many parameters these methods may require an unacceptably long time to converge to the target posterior distribution.

On the other hand, non-random walk algorithms such as HMC and NUTS avoid the inefficient exploration of the parameter space. Specifically, HMC borrowed its idea from physics to suppress the random walk behavior by means of an auxiliary variable, momentum, that transforms the problem of sampling from a target posterior distribution into the problem of simulating Hamiltonian dynamics, allowing it to move much more rapidly through the posterior distribution (Neal, 2011). The unknown parameter vector is interpreted as the position of a fictional particle. At each iteration, a random momentum vector is generated and the path of the particle is simulated with a potential energy equal to the negative value of the log posterior function. These continuous changes over time are approximated using the leapfrog algorithm (Stan Development Team, 2017). Then, after a Metropolis decision step is applied, the whole process repeats for an adequate number of iterations until convergence is reached.

Although HMC is an effective MCMC technique, it requires specifying the step size and the number of leapfrog steps parameters. Tuning these parameters, and specifically the number of leapfrog steps, requires expertise and a few preliminary runs (Neal, 2011; Hoffman & Gelman, 2011). This difficulty limits the more widespread use of HMC. Therefore, Hoffman and Gelman (2011) introduced NUTS, an extension of HMC that does not require setting the number of leapfrog steps. Using a recursive algorithm, NUTS creates a set of candidate points that spans a wide path of the target posterior distribution, stopping automatically when it starts to double back and retrace its steps (i.e. starts to make a U-turn). Empirically, NUTS performs as efficient as, and sometimes better than, a well-tuned HMC without requiring user interventions. This algorithm is implemented in Stan, an open-source C++ program that performs Bayesian inference (Stan Development Team, 2017).

1.1 Statement of the Problem

Over the past decades, the estimation of IRT and particularly MixIRT models has moved from the traditional maximum likelihood (ML) approach to the fully Bayesian approach via the use of MCMC techniques, whose advantages over ML have been well documented in the IRT literature (e.g., de la Torre, Stark, & Chernyshenko, 2006; Finch & French, 2012; Kim, 2007; Wollack, Bolt, Cohen, & Lee, 2002). Recent developments of MCMC focus on non-random walk MCMCs such as the no-U-turn sampler (NUTS; Hoffman and Gelman, 2011), which can converge to high dimensional posterior distributions more quickly than common random walk MCMC algorithms, and is hence less computational expensive. Moreover, NUTS is a tune-free technique, which makes it easily accessible by practitioners and researchers in behavioral sciences to fit various complex measurement models. Currently, there have been very few studies applying NUTS to IRT models or problems (e.g., Chang, 2017; Grant, Furr, Carpenter, & Gelman, 2016). For example, Chang (2017) fit the 2PL model using NUTS and compared it with Gibbs sampling. The results suggested that NUTS is as effective as Gibbs sampling in estimating model parameters. Also, Grant, et al. (2016) fit a Rasch model and a hierarchical Rasch model using NUTS, and compared it with MH. The results showed that NUTS was generally faster than MH in estimating parameters of the two models. Although MixIRT models have been estimated using random walk MCMC algorithms such as Gibbs sampling or MH (e.g., Cho, Cohen, & Kim, 2013; Huang, 2016; Samuelsen, 2005; Shea, 2013), to date, no research has adopted the non-random walk algorithm, and more specifically NUTS, to fit such complex IRT models. In addition, it is necessary to investigate how NUTS performs in estimating parameters of complex IRT models such as MixIRT models under various test conditions where sample size, test length, mixing proportions, and number of latent classes are taken into consideration.

1.2 Purpose of the Study

In view of the above, the purpose of the study is to implement the non-random walk MCMC algorithm, namely NUTS, to fit the Mix2PL model, and further to examine its performance in estimating model parameters when sample size, test length, and number of latent classes are manipulated. The motive behind this investigation is to add to the literature an evaluation of the NUTS algorithm that has not been fully investigated to estimate complex IRT models, and hence to provide researchers and practitioners with general guidelines on using fully Bayesian estimation via MCMC techniques for estimating complex IRT models.

Monte Carlo simulations are carried out to investigate parameter recovery of the Mix2PL model, the accuracy of determining the number of latent classes, and the performance of the Mix2PL model in comparison to the conventional 2PL model under various conditions where sample size, test length, mixing proportions, and number of latent classes are taken into consideration. It is anticipated that the fully Bayesian estimation can be implemented to fit MixIRT models using NUTS, which can estimate model parameters accurately and efficiently. In addition, the number of latent classes can be accurately determined via the use of Bayesian model fit indices.

1.3 Research Questions

The broad research question is whether NUTS, which can converge to high dimensional posterior distributions efficiently, can be implemented to fit MixIRT models. The specific research questions related to the performance of the algorithm and the accuracy of model parameter estimates are as follows:

1. How does NUTS perform in estimating the Mix2PL model under various test conditions of sample size, test length, and number of latent classes? with respect to the following:

a. The accuracy of recovering model parameters including mixing proportions, class mean ability, class item parameters, person abilities, and class memberships of individual persons.

b. The accuracy of determining the number of latent classes.

2. How does the Mix2PL model compare with the conventional 2PL model under situations where tests involve one or multiple latent classes?

1.4 Definition of Terms

The following are descriptions for some of the important terms used in this study.

1. *Item response theory (IRT)* – A modern test theory, in comparison to classical test theory, that models the probabilistic relationship between person's latent trait and the test at the item level. It is also known as the latent trait theory.
2. *Conventional IRT models* – The unidimensional dichotomous IRT models that are used when test items require binary responses such as true-false, agree-disagree, or from a response that is scored as correct or incorrect. The popular three models are the 1PL, the 2PL, and the 3PL.
3. *Mixture IRT models (MixIRT)* – A combination of latent class analysis model and IRT model where persons are presumed to come from latent subpopulations that are qualitatively different but within which an IRT model holds.
4. *Markov chain Monte Carlo (MCMC)* - A class of algorithms for generating samples from a probability distribution via constructing a Markov chain that has the desired distribution as its stationary distribution. These techniques are very useful in Bayesian inference in order to approximate the joint posterior distribution that cannot be directly calculated.
5. *Random walk MCMC* - A class of MCMC algorithms that explore the parameter space

via random walk behavior, where at each step a proposal move to a new point in the parameter space is accepted or rejected with a certain amount of probability. Gibbs sampling and Metropolis-Hastings algorithms are considered as random walk MCMC methods.

6. *Gibbs Sampling* - one of the simplest MCMC algorithms, which is applicable when the joint posterior distribution is not known explicitly, but the full conditional posterior distribution of each parameter is known. The idea of a Gibbs sampler is to approximate the joint posterior distribution by iteratively generating random samples from the full conditional distribution for each parameter.
7. *Metropolis-Hastings*- An MCMC algorithm that is more general than the Gibbs sampler, which is used when any of the full conditional posterior distributions are not in closed form. The idea of MH is to generate a proposed value from a proposal distribution. Then the proposed value is accepted as the next value in the Markov chain with a certain probability.
8. *Hamiltonian Monte Carlo (HMC*) – An MCMC algorithm that avoid random walk behavior by introducing an auxiliary momentum variable for each parameter in the parameter space. It implements Hamiltonian dynamics so the energy function is the target posterior distribution.
9. *No-U-Turn Sampler (NUTS)* - An adaptation to HMC that eliminates the need to set the number of leapfrog steps *L* that the algorithm takes to generate a proposal state. NUTS creates a set of candidate points that spans a wide path of the target posterior distribution, stopping automatically when it starts to double back and retrace its steps (i.e. starts to make a U-turn).

10. *Fully Bayesian fit indices* - Measures that utilize the joint posterior distribution of parameters in order to evaluate the predictive accuracy of a model. They are valued for comparing different models. The widely applicable (or Watanabe-Akaike) information criterion (WAIC) and the leave-one-out cross-validation (LOO) are considered as fully Bayesian fit indices.
11. *Stan* - An open-source C++ program that performs Bayesian inference using the NUTS algorithm.

1.5 Significance of the Research

The significance of the study lies in that it not only demonstrates the application of a more efficient MCMC algorithm to the more complex MixIRT model, but also provides guidelines to researchers and practitioners on the use of such models under the fully Bayesian framework and on how they compare with the conventional IRT models under situations where latent classes do exist. The successful implementation of NUTS to the Mix2PL model will also help researchers with fitting more complex IRT models using fully Bayesian estimation. Findings from this investigation provide empirical evidence and shed light on the performance of NUTS in fitting more complicated IRT models.

1.6 Delimitation of the Study

The delimitations of this study are as follows.

1. The study focuses on the dichotomous Mix2PL model. Other dichotomous MixIRT models (e.g., Mix1PL, or Mix3PL) are not considered, neither are polytomous MixIRT models where item responses include more than two response categories.
2. This study uses the NUTS algorithm to fit the Mix2PL model. Other non-random walk algorithms such as HMC, or random walk algorithms such as Gibbs sampling or

Metropolis-Hastings are not considered in this study.

3. The study only focuses on simulated data, not real data. It is believed that various combinations of possible test conditions in real life situations can be mimicked using simulation studies, which makes it possible to evaluate the performance of the algorithm.
4. In comparing the Mix2PL model with the conventional 2PL model, the best model that can explain the data adequately will be chosen based on fully Bayesian fit measures including the widely applicable (or Watanabe-Akaike) information criterion (WAIC) and the leave-one-out cross-validation (LOO). Other fit indices such as AIC or BIC are not considered in this study.
5. This study will consider specific combinations of sample size, test length, number of latent classes, and mixing proportions. Values of these factors are chosen such that they reflect real test situations.

1.7 Overview of Subsequent Chapters

The subsequent chapters are organized as follows. Chapter 2 reviews the related literature on the conventional and the mixture IRT models, estimation methods used to estimate IRT models including the MCMC algorithms under the fully Bayesian framework, and related prior research. Chapter 3 describes the procedures of fitting the Mix2PL model using NUTS with simulated datasets. Chapter 4 presents the results of the simulation studies related to the algorithm performance and models comparison. Finally, Chapter 5 summarizes the findings, the implications of this investigation, and directions for future research.

CHAPTER 2

LITERATURE REVIEW

The review of the literature starts with the basic concepts of the item response theory (IRT) modeling framework. Four main sections are included in this chapter. The first section reviews the conventional IRT and mixture IRT models. The second section focuses on the estimation methods used to estimate IRT models. Section three reviews the common Markov chain Monte Carlo (MCMC) methods. The last section reviews prior research estimating conventional IRT Models using the no-U-turn sampler (NUTS) and prior research on estimation of mixture IRT models.

2.1 Item Response Theory Models

Classical test theory (CTT) has been the predominant psychometric method for most of the last century (Gulliksen, 1987) and widely used in educational and psychological testing. It defines a simple additive model such that any test score is comprised of a true score and random error, $X = T + \varepsilon$. In other words, CTT suggests that any mental latent trait T can be known through the examinee's observed score X, such that a normally distributed random error ε exist in everyone's test score. However, its limitations that affect the quality of measurement have led to the development of a new measurement framework that can solve many practical testing issues such as test equating. Item response theory (IRT; Lord, 1980) aims to look beyond the observed test score, at the underlying traits, which produce the test performance. IRT models are measured at item level such that the probability of a correct ($y_{ij} = 1$) response to an item j is a non-linear function of both examinee's latent trait θ_i, and the item parameters ξ_j (e.g., difficulty, discrimination). The general form of IRT models can be expressed as $P(y_{ij} = 1) = f\,(\theta_i, \xi_j)$. IRT has gained an increasing popularity in large-scale educational and psychological testing situations. Bock (1997) noted that IRT is a robust and productive alternative to CTT of test scoring and item analysis. For instance, some applications such as computerized adaptive testing (CAT) are applicable using IRT models, yet cannot reasonably be performed using CTT only.

IRT has been shown to have its advantages over CTT. One of the major limitations of CTT is sample dependency, or as Fan (1998) termed "circular dependency". This means examinees' characteristics and test characteristics cannot be separated. In other words, whether an item is easy or difficult depends on the ability of examinees being measured, and at the same time the ability of examinees depends on whether test items are difficult or easy. Conversely, one of the major advantages of IRT is that it is sample free. This indicates that item parameters (e.g., difficulty or discrimination) do not depend on the ability of examinees used to calibrate them. Hence, item parameter estimates obtained using different groups of examinees will be the same. Similarly, examinee's ability does not depend on the set of items used to estimate it. Therefore, ability estimates obtained from different sets of items will be the same. Technically, this advantage is called the property of invariance of item and ability parameters. This statistical property is the cornerstone of IRT that distinguishes it from CTT.

Another advantage of IRT is related to the standard error of measurement (SEM). In CTT, the SEM is assumed to be the same for all examinees, although this assumption is highly unlikely in practice. For example, test scores for two examinees of different ability levels contain different amount of errors. Furthermore, in the classical framework, the SEM depends on test reliability and variance; that is $SEM = \sigma\sqrt{1 - r_{tt'}}$, where σ is the test standard deviation (SD), and $r_{tt'}$ the reliability estimate. Test reliability is estimated based on the assumption of parallel tests that cannot be satisfied in a strict sense. In contrast, the SEM in IRT is allowed to change given different levels of the latent trait, and hence IRT provides a measure of precision for each ability level. In effect, the SEM for each ability level depends on the information function $I(\theta)$, such that $\mathrm{SEM}\,(\theta)=1/\sqrt{I(\theta)}$.

In IRT, item difficulties and examinee abilities can be placed on the same scale. This advantage makes it possible to scale examinees relative to items or vice-versa. The comparisons between IRT and CTT have been widely reviewed in the literature (for more details see e.g., Thissen & Wainer, 2001; Embreston & Reise, 2000; Hambleton & Jones, 1993).

2.1.1 Unidimensional IRT Models and their Assumptions

IRT models the probabilistic relationship between a person's latent ability (or trait) and the test at the item level. Unidimensional IRT models rely on two major assumptions: unidimensionality and local independence. Unidimensionality means that a single unified latent trait is measured by all test items. This assumption is difficult to satisfy because of the existence of several cognitive, personality, and test taking factors that could affect test performance such as stress, anxiety, and fatigue. However, the presence of a dominant factor (i.e. the ability being measured) that explains test performance is sufficient for this assumption to be satisfied. The assumption of local independence means that when the abilities influencing test performance are all held constant, then examinees' responses to any pair of test items are independent of each other. Local independence is met when all the abilities influencing examinee test performance is specified (i.e. the complete latent space has been specified) (Hambleton, Swaminathan, & Rogers, 1991). This assumption is related to unidimensionality although local independence is a broader or more general concept. Suffice it to say that when the assumption of unidimensionality is true, the assumption of local independence is met and the two concepts are equivalent. Local independence can be mathematically defined as follows:

$$P(y_{i1}, y_{i2},\ldots, y_{iJ} \mid \theta_i) = P(y_{i1} \mid \theta_i)(y_{i2} \mid \theta_i)\ldots(y_{iJ} \mid \theta_i) = \prod_{j=1}^{J} P(y_{ij} \mid \theta_i), \qquad (2.1)$$

where the conditional probability $P(y_{i1}, \dots, y_{iJ})$ of a response pattern on a set of J items by an examinee with ability of θ_i, is equal to the product of the probabilities associated with the examinee's responses to J individual items.

There exist three common unidimensional dichotomously scored IRT models (Rasch, 1960; Lord & Novick, 1968; Birnbaum, 1969; Lord, 1980) that are described based on the number of item parameters, namely the one-parameter logistic (1PL), the two-parameter logistic (2PL), and the three-parameter logistic (3PL) models. Such models have been extensively studied in the IRT literature (e.g., Kang & Cohen, 2007; Maraun, 1993; Sahin & Anil, 2017; Toribio, 2006).

The 1PL model, also known as the Rasch model (Rasch, 1960), is the simplest and one of the most widely used IRT models. The probability of a correct response (Y_{ij} = 1) is defined as:

$$P(Y_{ij}=1|\theta_i,b_j)=\frac{\exp(\theta_i-b_j)}{1+\exp(\theta_i-b_j)}, \tag{2.2}$$

where θ_i is the latent ability of person i $(i=1, \dots, N)$, and b_j is the difficulty parameter for item j $(j=1, \dots, J)$ which is defined as the ability required for a person to have a 0.5 probability to answer the item correctly. Theoretically, a persons' ability levels range from $-\infty$ to $+\infty$ and follows a standard normal distribution with a mean of zero and a unit variance. Given this, about 99.74% of the persons in the population have ability levels range from −3 to 3. Similarly, the range of item difficulties is theoretically from $-\infty$ to $+\infty$ but empirically the parameters range from −2 to 2 when latent abilities are assumed to range from −3 to +3 (Hambleton & Cook, 1977). The larger the value of the item difficulty parameter is, the more difficult the item becomes since it requires a higher ability to answer the item correctly with a probability of 0.50.

The 1PL (Rasch) model assumes that items differ only in difficulty. This means that item characteristics curves (ICCs) are parallel.

In many situations the assumption that items differ only in difficulty is too restrictive and hence the 1PL (Rasch) model is not applicable in such situations. The 2PL model (Lord & Novick, 1968) generalizes the 1PL (Rasch) model where items are allowed to differ in terms of difficulty (b_j) and discrimination (a_j) parameters. In the 2PL model, item characteristics curves (ICCs) can intersect with each other, in contrast to the parallel ICCs in the 1PL (Rasch) model where items are equally discriminating. The 2PL model is defined as follows:

$$P(Y_{ij}=1|\theta_i,b_j,a_j)=\frac{\exp[a_j(\theta_i-b_j)]}{1+\exp[a_j(\theta_i-b_j)]}, \qquad (2.3)$$

where a_j denotes the discrimination parameter for item j, which is defined as the slope of the item characteristic curve (ICC) at the value of the difficulty parameter. Item discrimination can be considered as an indicator of how much information an item provides about the latent ability θ_i. In practice, values of a_j vary from zero to +2 (Hambleton & Cook, 1977). An item with a negative discrimination parameter suggests that persons with greater ability levels are less likely to answer the item correctly. Therefore, such an item should be removed.

For multiple-choice items, it is possible for examinees to randomly guess the answer correctly, which should be taken onto consideration. The 3PL model is an extension of the 2PL model where items differ in difficulty, discrimination, and guessing parameters. The 3PL model is defined as follows:

$$P(Y_{ij}=1|\theta_i,b_j,a_j)=\mathrm{c_j}+(1\text{-}\mathrm{c_j})\frac{\exp[a_j(\theta_i-b_j)]}{1+\exp[a_j(\theta_i-b_j)]}, \qquad (2.4)$$

where c_j denotes the pseudo-guessing parameter for item j, which indicates the probability that a person with an extremely low ability level answers the item correctly. For items with guessing parameters greater than zero, the item difficulty is redefined as the ability required for a person to have a probability of $(1+ c_j)/2$ to answer the item correctly. This means the item difficulty is shifted by the lower asymptote c_j.

2.1.2 Mixture IRT Models

Under many empirical situations, conventional unidimensional IRT models do not explain the data adequately. Specifically, in situations where a mixture of several underlying subpopulations is involved, fitting any conventional IRT models to the data produce biased estimates of model parameters. Mixture item response theory (MixIRT) models can be used to capture the presence of latent classes (i.e. subpopulations) that are qualitatively different but within which a conventional IRT model holds. This model combines the theoretical strength of latent class analysis (LCA) and IRT (Rost, 1990, 1997). Muthén and Asparouhov (2006) applied the MixIRT model to the analysis of tobacco dependence, and found that the MixIRT model fit the data better compared to an LCA or IRT model. Furthermore, Lau (2009) found that use of the MixIRT models led to a better parameter estimation than the conventional IRT models regardless of the proportion of amotivated examinees (i.e. who do not provide meaningful responses to any test items) in low-stakes tests.

In the MixIRT modeling framework, examinees are characterized by both their location on a continuous latent ability as well as by their latent class membership. Conventional IRT models assume all examinees come from the same population. Therefore, a single set of item parameters is appropriate. In contrast, MixIRT models assume that examinees come from multiple subpopulations, with each subpopulation requiring its own unique set of item parameters (i.e. they allow subpopulations to perform differently on the same set of items) (Rost, 1990).

MixIRT models have become increasingly popular in being used as a technique for investigating various issues in educational and psychological measurement. One of these issues is the assessment of the presence of differential item functioning (DIF), which is deemed as one of the fundamental procedures of instrument development and validation in psychometrics. MixIRT models have been extensively used to detect latent DIF (e.g., Aryadoust, 2015; Choi, et al., 2015; Cohen & Bolt, 2005; Cohen, Gregg, & Deng, 2005; DeAyala, Kim, Stapleton & Dayton 2002; Maij-de Meij, et al., 2010; Samuelsen, 2005; Shea, 2013; Wu, et al., 2017) and proved its superiority in explaining sources of DIF beyond those associated with observed variables such as gender or ethnicity that the traditional methods use to compare the functioning of an item across manifest groups. In addition, researchers have expanded DIF analyses by incorporating more complex forms of MixIRT models such as hierarchical MixIRT models. For example, Cho and Cohen (2010) applied a multilevel MixIRT (MMixIRT) model to detect DIF at two different levels: examinee level, and school level. Moreover, Finch and Finch (2013) used multidimensional multilevel mixture IRT (MMMixIRT) models to identify the presence of DIF in a math and language test. The results demonstrated the model provided more complete information regarding the nature of DIF. Also, Bilir (2009) combined the 1PL (Rasch) model for manifest group-DIF and mixture Rasch model for latent class-DIF, and proposed a new mixture (MixIRT-MIMIC) model in order to simultaneously estimate DIF across manifest groups and latent classes. The results showed that MixIRT-MIMIC provides less biased estimates for latent class-DIF and item difficulty parameters when the overlap between the manifest group and the latent class is between 50% and 70%.

Furthermore, another situation where MixIRT models have been employed as a strategy is test speededness. Oshima (1994) noted that administering tests under time constraints may result in poorly estimated item parameters, particularly for items at the end of the test. For example, many researchers used the mixture Rasch model, proposed by Rost (1990) with ordinal constraints to distinguish groups of examinees that are differentially affected by test speededness (e.g., Bolt, et al., 2002; Mroch, et al., 2005). Other researchers such as Meyer (2010) developed the mixture Rasch model with item response time components to detect test speededness and to classify examinee test-taking behavior into either solution behaviors (non-speeded group) or rapid guessing behaviors (speeded group). Moreover, Wollack, et al. (2003) found that the mixture Rasch model for test speededness improved equating and helped prevent scale drift.

In addition, the mixture nominal response (MixNR) IRT model has been proposed by Bolt, Cohen and Wollack (2001) for investigating individual differences between latent classes in the selection of response categories in multiple-choice items. Also, mixture IRT models have been used for detection of latent groups that differ in their use of problem-solving strategies for responding to test items (e.g., Mislevy & Verhelst, 1990).

Rost (1990) proposed a one-parameter MixIRT (Mix1PL) model, also known as the mixture Rasch model, for dichotomous data in which a population is assumed to consist of discrete latent classes. The conventional Rasch model is assumed to hold within each latent class, but different difficulty parameters apply across classes. Individual members within a class can also have different ability levels. Thus, in the mixture Rasch model each examinee is characterized both by a class membership parameter *g*, which determines the relative difficulty ordering of the items for that examinee, and a continuous latent ability parameter $\theta_{ig,}$ which affects the number of items the examinee is expected to answer correctly. Class parameters μ_g

and σ_g denote the mean and standard deviation, respectively, for class-specific abilities θ_{ig}, in class g. If we let Y_{ij} detonate a correct ($Y_{ij} = 1$) or incorrect ($Y_{ij} = 0$) response for person i to item j, the probability of a correct response in the Mix1PL model is defined as follows:

$$P(Y_{ij}=1|\theta_i)=\sum_{g=1}^{G}\pi_g\times P(Y_{ij}=1|\theta_{ig},b_{jg},g)=\sum_{g=1}^{G}\pi_g\times\frac{\exp[(\theta_{ig}-b_{jg})]}{1+\exp[(\theta_{ig}-b_{jg})]}, \tag{2.5}$$

where $g = 1, ...,G$ is the latent class indicator, b_{jg} reflects the difficulty parameter for item j in the gth class, θ_{ig} denotes the ability for person i in class g, and π_g denotes the proportion of persons in each class (i.e., the mixing proportion) in each class with a constraint that all these proportions sum to one. For the purpose of model identification, a sum-to-zero constraint is applied to the item difficulty parameters, such that within each class item difficulty values sum to zero (Rost, 1990). On the other hand, the mean ability for each latent class (μ_g) is allowed to differ in order to account for quantitative differences between classes. More details on identification of MixIRT models will be described in a later chapter.

The two-parameter MixIRT (Mix2PL) model (e.g., Finch & French, 2012) and the three-parameter MixIRT (Mix3PL) model (e.g., Cohen & Bolt, 2005) for dichotomous data can be viewed as an extension of the mixture Rasch model. Similarly, each examinee is parameterized both by a class membership parameter g and a class-specific ability parameter θ_{ig}. In the Mix2PL model, the conventional 2PL IRT model is assumed to hold for each class, but the item difficulty and discrimination parameters may differ across different classes. The probability of a correct response in the Mix2PL model is defined as follows:

$$P(Y_{ij}=1|\theta_i)=\sum_{g=1}^{G}\pi_g\times P(Y_{ij}=1|\theta_{ig},b_{jg},a_{jg},g)=\sum_{g=1}^{G}\pi_g\times\frac{\exp[a_{jg}(\theta_{ig}-b_{jg})]}{1+\exp[a_{jg}(\theta_{ig}-b_{jg})]}, \tag{2.6}$$

where a_{jg} denotes the discrimination parameter for item j in the gth class, θ_{ig} and b_{jg} are as defined in equation (2.5). As in the Mix1PL, the sum-to-one constraint on the mixing proportions and the sum-to-zero constraint on item difficulty parameters within each class are applied.

In the three-parameter MixIRT (Mix3PL) model for dichotomous data, the conventional 3PL IRT model is assumed to hold for each class, but the item difficulty, discrimination, and guessing parameters may differ across different classes. The probability of a correct response in the Mix3PL model is defined as:

$$P(Y_{ij}=1|\theta_i)=\sum_{g=1}^{G}\pi_g \times P(Y_{ij}=1|\theta_{ig},b_{jg},a_{jg},c_{jg},g)=\sum_{g=1}^{G}\pi_g \times c_{jg}+(1-c_{jg})\frac{\exp[a_{jg}(\theta_{ig}-b_{jg})]}{1+\exp[a_{jg}(\theta_{ig}-b_{jg})]}, \qquad (2.7)$$

where c_{jg} denotes the guessing parameter for item j in the gth class, a_{jg}, θ_{ig}, and b_{jg} are as defined in equations (2.5) and (2.6).

If only one class is retained, the mixture IRT models, namely the Mix1PL, Mix2PL, and Mix3PL models reduce to the conventional 1PL, 2PL, and 3PL IRT models, respectively, whose mathematical models are defined in equations (2.2), (2.3), and (2.4). In other words, the conventional IRT models are nested within MixIRT models.

2.2 Parameter Estimation of IRT models

Estimating two sets of parameters (person and item) parameters based on merely a set of response data is one of the crucial steps in applying IRT model. Several estimation techniques have been developed in last decade for estimating IRT models. The early focus was on estimating item and person parameters jointly using maximum likelihood (ML; Fisher, 1922) estimation methods. The three popular ML methods are the joint maximum likelihood (JML; Birnbaum, 1969), the conditional maximum likelihood (CML; Andersen, 1970), and the marginal maximum likelihood (MML; Bock & Aitkin, 1981). Bayesian estimation (e.g., Chib &

Greenberg, 1995) has to be adopted under situations where ML methods fail to produce a solution. The Bayesian approach basically entails combining the likelihood function with prior distributions of parameters to estimate the posterior distribution. The three major ML estimation methods, including the JML, the CML, and the MML, as well as the Bayesian estimation technique are reviewed below.

2.2.1 Joint Maximum Likelihood (JML)

The JML estimation method is based on an iterative two-stage procedure for jointly estimating ability and item parameters. In the first stage, initial values for ability parameters are estimated based on the persons test scores. In the second stage, ability parameters are treated as known and thus item parameters are estimated. These two steps are repeated until both person and item parameters estimates are stable (Si & Schumacker, 2004; Hambleton, et al., 1991). Since person and item parameters are jointly estimated in this method, the assumption of local independence for both items and persons should be satisfied. Stated differently, persons' ability levels are independent of each other, and responses to any pair of items are independent of each other when ability is held constant. Based on this assumption, the joint likelihood function across persons and items is defined as follows:

$$L(\boldsymbol{y}|\boldsymbol{\theta},\boldsymbol{\xi})=\prod_{i=1}^{N}\prod_{j=1}^{J}P_{ij}^{\ y_{ij}}(1-P_{ij})^{1-y_{ij}}\ , \tag{2.8}$$

where **y** is a response matrix of dimension *N* by *J*, P_{ij} is the probability function defined under the appropriate IRT, $\boldsymbol{\theta}$ is a vector of *N* ability parameters, and item parameters $\boldsymbol{\xi}$ is a vector of length *J* for the 1PL model, or a matrix of 2 by *J* for the 2PL model or a matrix of 3 by *J* for the 3PL model.

The JML estimates of persons and item parameters can be obtained by taking the natural logarithm of the likelihood function, shown in equation (2.8), then setting the first derivation of the log likelihood function to zero, and finally solve for ability and item parameters. In order to eliminate the problem of indeterminacy and hence find a unique maximum, a scale for ability parameters is chosen. Usually, a standard normal distribution is chosen for ability values so that item and ability parameters are anchored.

Although the JML estimation method can be easily applied to many IRT models, it has several shortcomings. For instance, ability estimates for persons who get all correct or all incorrect answers do not exist. Similarly, item parameter estimates do not exist for items that are answered correctly, or incorrectly by all examinees. Moreover, for the two- and three- parameter models, the JML estimation method produces inconsistent estimates of item and ability parameters because both parameters are estimated simultaneously (Hambleton, et al., 1991). To solve the problem of inconsistent estimates, an alternative procedure is needed where item parameters can be estimated without any reference to ability parameters. This is achieved by the MML approach discussed below in Section 2.2.3.

2.2.2 Conditional Maximum Likelihood (CML)

Andersen (1970) introduced an ML method based on the conditional distribution given minimal sufficient statistics for the parameters in order to obtain consistent estimates for those parameters. Instead of maximizing the likelihood directly, ability parameters are eliminated from the likelihood equation by considering the conditional distribution given minimal sufficient statistics. Since this technique requires sufficient statistic it is only applicable to the 1PL (Rasch) model, for which the number of correct responses is a sufficient statistic for the ability parameter, and the number of correct responses to an item is a sufficient statistic for the item difficulty parameter (Si & Schumacker, 2004). The likelihood function $L(y_i|\theta_i)$ is replaced by the likelihood function of response pattern s_i for person i , $L(y_i|s_i)$ where s_i is the sufficient statistic or the number of correct responses the person obtained. The likelihood function $L(y_i|s_i)$ can be written as follows:

$$L(y_i\,|\,s_i)=\frac{L(y_i\,|\,\theta_i)}{L(s_i\,|\,\theta_i)}\ . \tag{2.9}$$

As can be seen from equation (2.9), the ability parameter cancels out from the likelihood function. Then, estimates for the difficulty parameter can be found by maximizing the conditional likelihood (or alternatively the log likelihood) function, where:

$$L(b_j)=\prod_{i=1}^{n}L(y_i\,|\,s_i)\ . \tag{2.10}$$

2.2.3 Marginal Maximum Likelihood (MML)

The marginal maximum likelihood (MML) estimation method was developed by Bock and Lieberman (1970), and improved by Bock and Aitkin (1981). MML provides a solution for the problem of the inconsistent estimates resulting from the JML estimation method. This is achieved by treating the ability parameter as a nuisance parameter and factoring it out from the likelihood function. Specifically, MML assumes persons as a random sample from a population with a probability density function $f(\theta)$. Therefore, they can be integrated out of the likelihood function to obtain the marginal likelihood function in terms of the item parameters. That is:

$$L(y\,|\,\xi)=\int L(y\,|\,\theta,\xi)f(\theta)d\theta\ . \tag{2.11}$$

Because the integral cannot be expressed in a closed form, it has to be approximated using a Gaussian quadrature procedure. Once θ has been eliminated from the function, the maximum likelihood estimates of item parameters can be obtained. The resulting item parameter estimates are consistent as the number of persons increases. Then, treating item parameters as known, the maximum likelihood estimates of person parameters can be obtained. Again, the larger the number of items, the better the ability parameters are estimated using MML.

The original MML procedure introduced by Bock and Lieberman (1970) is computationally intensive and is hence impractical for long tests. Bock and Aitken (1981) refined the procedure and introduced an expectation-maximization (EM) algorithm as a procedure for MML estimation. The EM algorithm has two stages, namely, expectation and maximization. In the expectation stage, expected values of the frequencies at quadrature points and expected frequencies of persons passing the items are computed. In the maximization stage, these expected values are used in the marginal likelihood function to engage the maximum likelihood estimation. These two steps go back and forth until the algorithm converges.

One of the disadvantages of the MML estimation method is that the ability parameter is assumed to be normally distributed. However, the normal distribution does not necessarily work for all situations (Johnson, 2007). In addition, the MML method requires integrating out the person parameter to obtain the marginal likelihood function, which is difficult for complex models. Also, it does not take into consideration of the uncertainty of estimating item parameters when computing the uncertainty for estimating θ.

2.2.4 Bayesian Estimation

While traditional techniques of parameter estimation find point estimates of parameters $\boldsymbol{\theta}$, by maximizing the likelihood of the data given those parameters, Bayesian approach finds the joint posterior distribution of the parameters given the data, $f(\boldsymbol{\theta}|\boldsymbol{y})$ (Gelman, Carlin, Stern, Dunson, Vehtari, & Rubin, 2014; Kruschke, 2011). This is done by the application of Bayes' Theorem that combines the prior on parameter values, $f(\boldsymbol{\theta})$ with the likelihood of the data given certain parameter values, $f(\boldsymbol{y}|\boldsymbol{\theta})$, resulting in the posterior distribution of the parameters given the data; $f(\boldsymbol{\theta}|\boldsymbol{y}) = f(\boldsymbol{y}|\boldsymbol{\theta}) \times f(\boldsymbol{\theta}) / f(\boldsymbol{y})$.

In the IRT literature, there are two common types of Bayesian approaches, namely the marginal Bayesian and the fully Bayesian. The marginal Bayesian estimation method proposed by Mislevy (1986) is a simple extension of the MML-EM approach, such that it places a prior distribution for each parameter of the model. However, the fully Bayesian estimation simultaneously obtains posterior estimates for both item and person parameters using Markov chain Monte Carlo (MCMC; Hastings, 1970; Metropolis & Ulam, 1949; Metropolis, Rosenbluth, Rosenbluth, Teller, & Teller, 1953) procedures. The fully Bayesian method has advantages over the marginal Bayesian method for the following reasons. First, in the marginal Bayesian technique, latent ability levels of persons θ_i are treated as random variables and integrated out from the joint likelihood of item and person parameters. However, when a model is complicated, integrating out the latent abilities is not straightforward. The magic of fully Bayesian estimation via the use of Markov chain Monte Carlo (MCMC) techniques is that they do not require the integration step. Instead, by relying on ratios of the posterior probabilities, the integration term cancels out, so the decision to accept or reject a new sample is only based on the likelihood and prior distributions.

Many researchers have found advantages of the fully Bayesian estimation based on MCMC techniques over the maximum likelihood methods in estimating different IRT models. For instance, Finch and French (2012) compared difficulty and discrimination parameters estimation of MixIRT model using the fully Bayesian based on MCMC approach and the MLE method in terms of classification accuracy and estimation bias. The two estimation methods were fitted using the Mplus (Muthén & Muthén, 2011) software. The results showed that fully Bayesian estimation provides a more accurate recovery of group membership across different conditions as well as provides more accurate parameter estimates for data sets with smaller sample sizes and fewer items. Also, de la Torre, Stark and Chernyshenko (2006) estimated the generalized graded unfolding model using fully Bayesian based on MCMC procedure and the marginal maximum likelihood (MML) approach. Results showed that the two methods are comparable in terms of item parameter estimation accuracy. However, fully Bayesian estimation provides reasonable standard error estimates for all items. Furthermore, Wollack, et al. (2002) showed that fully Bayesian estimation can be used as an alternative to marginal maximum likelihood (MML) estimation for more complex and more heavily parameterized IRT models such as nominal response (NR) IRT models.

Due to the advanced computational techniques and the development of MCMC procedure, the fully Bayesian approach have been rapidly developed and applied to estimate different IRT models (e.g., Bolt & Lall, 2003; de la Torre, et al., 2006; Johnson & Sinharay, 2005; Patz & Junker, 1999a; Shea, 2013). Albert (1992) was the first to implement the fully Bayesian estimation via Gibbs sampling algorithm to fit a two-parameter normal ogive (2PNO) model to simulated and real data, using MATLAB (MathWorks, Inc., 1992) program. Since then, many software applications have been developed for fully Bayesian inferences such as WinBUGS (Lunn, Thomas, Best, & Spiegelhalter, 2000), JAGS (Plummer, 2003), OpenBUGS (Thomas, O'Hara, Ligges, & Sturtz, 2006), and recently Stan (Stan Development Team, 2017).

2.3 MCMC Algorithms

The Bayesian method is a general and flexible approach. It could be used with a model having one parameter or one having several of parameters. However, as the number of parameters in the model increases, the traditional numerical methods for estimating the posterior distribution quickly become intractable. Therefore for more sophisticated models, a technique called Markov chain Monte Carlo (MCMC) was developed (Brooks, Gelman, Jones, & Meng, 2011; Gelman, et al., 2014).

In the last decade there has seen an intensive application of MCMC techniques in fitting a variety of measurement models. To date, MCMC has been used for supporting the development of new models that are otherwise computationally intractable, in addition to accurately estimating existing models (Levy, 2009). MCMC techniques have been successfully used in estimating parameters of various complex IRT models (e.g., Chang, 2017; de la Torre & Douglas, 2004; Johnson & Junker, 2003; Kim, 2001; Kuo, 2015; Lamsal, 2015; Patz & Junker, 1999a, 1999b; Sheng & Wikle, 2007; Sheng, 2010).

MCMC is a class of algorithms that use Markov chains for sampling from a probability distribution (e.g., the posterior distribution). An important feature of a Markov chain is its stationary distribution. The stationary state allows one to define the probability for every state of a system at a random time. At each state of the Markov chain, random samples of model parameters are generated from the distribution based on those generated from a previous state. Since early samples may be affected by initial values, they are discarded in the burn-in stage. After the burn-in stage, the quality of the samples becomes approximately stable.

Different MCMC techniques have been developed in the last two decades. The two major ones are random walk algorithms such as Gibbs sampling (Geman & Geman, 1984) and Metropolis-Hastings (MH; Hastings, 1970; Metropolis & Ulam, 1949), and non-random walk algorithms such as Hamiltonian Monte Carlo (HMC; Duane, et al., 1987) and its extension, the no-U-turn sampler (NUTS; Hoffman and Gelman, 2011). A review of these algorithms is presented in the next section.

2.3.1 Random Walk MCMC

The random walk MCMC is one of the most widely used MCMC algorithms for sampling from a posterior distribution. The general method is to randomly sample values of model parameters from approximate distributions where at each step in the simulation the approximate distributions are improved, based on the Metropolis rule, until eventually converging to the target distribution (Gelman, et al., 2014). Since this type of MCMC algorithm explores the distribution via simple random walk proposals, a large number of iterations is needed to sufficiently explore the parameter space. Two of the common random walk MCMC algorithms are discussed below.

Gibbs sampling was introduced by Geman and Geman (1984). In their paper, they discussed optimization to find the posterior mode instead of simulations. Therefore, it took some time for it to be understood that the Gibbs sampler simulated the posterior distribution, thus enabling full Bayesian inference of all kinds (Geyer, 2001). Gelfand and Smith (1990) made the Gibbs sampler very popular among the Bayesian community. The process for Gibbs sampling is a type of random walk through the parameter space. The walk starts at some arbitrary point. Each step in the walk is completely independent of the steps before the current position. This is a special property of the Markov chain where each new sample depends on the one before it, but does not depend on any samples drawn earlier from the posterior distribution.

At each point in the random walk, one of the parameters is simulated from its full conditional distribution and the parameters are cycled through in order. For example, if θ_i has been chosen, Gibbs sampling selects a new value for that parameter by generating a random value from the conditional probability $p(\theta_i|\boldsymbol{\theta}_{j\neq i}, \boldsymbol{y})$. The new value for θ_i along with the unchanged values of $\boldsymbol{\theta}_{j\neq i}$, create the new position in the random walk. The process then repeats. By cycling through these conditional statements, the joint posterior distribution would be eventually reached. All the simulated samples can be considered as those from the joint posterior distribution. Hence, the mean of the posterior distribution can be computed by averaging the generated samples after discarding the burn-in stage. The specific steps of Gibbs sampling is outlined as follows: suppose we are interested in sampling from the posterior $p(\theta_1,\theta_2,\ldots,\theta_p|\boldsymbol{y})$,

1. Choose plausible initial values of the parameters, $\boldsymbol{\theta}^{(0)} = (\theta_1^{(0)}, \theta_2^{(0)}, \cdots, \theta_p^{(0)})$.
2. For each parameter θ_i, draw values from its full conditional distribution given the current values of all other model parameters and the observed data. One cycle is given by sequentially drawing values from:

$$\theta_1^{(l)} \sim p(\theta_1|\theta_2^{(l-1)},\cdots,\theta_p^{(l-1)},\boldsymbol{y})$$

$$\theta_2^{(l)} \sim p(\theta_2|\theta_1^{(l)},\theta_3^{(l-1)},\cdots,\theta_p^{(l-1)},\boldsymbol{y})$$

$$\vdots$$

$$\theta_p^{(l)} \sim p(\theta_p|\theta_1^{(l)},\theta_2^{(l)},\cdots,\theta_{p-1}^{(l)},\boldsymbol{y}). \tag{2.12}$$

3. Repeat step 2 for an adequate large number of L iterations until convergence is reached.

This algorithm generates a sequence of parameter: $(\theta_1^{(0)},\ldots,\theta_p^{(0)}),(\theta_1^{(1)},\ldots,\theta_p^{(1)}),\cdots,(\theta_1^{(L)},\ldots,\theta_p^{(L)})$, where $(\theta_1^{(l)},\ldots,\theta_p^{(l)})$ is approximately a sample from the joint posterior $p(\theta_1,\theta_2,\ldots,\theta_p|\boldsymbol{y})$. In order to use the Gibbs sampler, the full conditional distribution for each parameter should be in closed form. However, in practice, the full conditional distributions may not always be in closed form or may be difficult to simulate. An alternative algorithm to estimate the parameters is Metropolis-Hastings (MH; Hastings, 1970; Metropolis & Ulam, 1949). The Metropolis-Hastings (MH) algorithm was developed by Metropolis, et al. (1953) and generalized by Hastings (1970). The

MH algorithm generalizes the Gibbs sampler since it offers a solution to the problem of sampling from a conditional distribution, from which it is difficult to sample directly.

MH algorithm is also a type of Monte Carlo process that generates a random walk such that each state in the walk only relies on the previous state, but is completely independent of the states before the previous state. Instead of sampling from the full conditional distribution for each parameter, a proposal or candidate distribution is selected by the current value of the parameters. Then a proposal move to a new point in parameter space is randomly generated from a proposal distribution and accepted with a certain amount of probability. The acceptance decision is based on the value of the posterior distribution at the proposed position, relative to the value of the posterior distribution at the current position. In particular, if the posterior distribution is greater at the proposed position than at the current position, the move is definitely accepted. However, if the posterior distribution is less at the proposed position than at the current position, the move is accepted with a probability equal to the ratio of the posterior distributions; $p(\theta_{move}) = p(\theta_{proposed})/p(\theta_{currrent})$. These two possibilities, of the posterior distribution being higher or lower at the proposed position than at the current position, can be expressed as follows:

$$\alpha(\theta^{(l-1)},\theta^{(l)}) = \min(\frac{p(\theta^{(l)})*q(\theta^{(l-1)}|\theta^{(l)})}{p(\theta^{(l-1)})*q(\theta^{(l)}|\theta^{(l-1)})},1) \, . \qquad (2.13)$$

After the probability of accepting the move from $\theta^{(l-1)}$ to $\theta^{(l)}$ is computed according to Equation (2.13), the acceptance decision of the proposal move is conducted by sampling a value from a uniform distribution over the interval [0, 1]. If the sampled value is less than or equal $\alpha(\theta^{(l-1)},\theta^{(l)})$, then the move is accepted. Otherwise, the move is rejected and stayed at the current position. The whole process repeats at the next time iteration. Gibbs sampling could be considered as a special case of the MH algorithm when the probability of accepting the proposal value is always equal to one (Gelman, et al., 2014; Kruschke, 2011). The steps of the MH sampling algorithm are outlined below.

1. Choose plausible initial values of the parameters, $\boldsymbol{\theta}^{(0)} = (\theta_1^{(0)}, \theta_2^{(0)}, \cdots, \theta_p^{(0)})$.
2. For each iteration l = 1, …,L, draw a candidate value, $\boldsymbol{\theta}^{(l)}$ from the proposal distribution $q(\theta^{(l)}|\boldsymbol{\theta}^{(l-1)})$. Then, the proposal value is accepted as the next value with the probability given in equation (2.12). If the proposal move is rejected, the current value will be used as the next value of the Markov chain.
3. Repeat step 2 for a large number of L iterations until convergence is reached.
4. Return the values $\left(\boldsymbol{\theta}^{(0)}, \boldsymbol{\theta}^{(1)}, \ldots, \boldsymbol{\theta}^{(L)}\right)$ for estimating the joint distribution $p(\boldsymbol{\theta})$.

One problem with MH algorithm is that the proposal distribution must be properly tuned to the posterior distribution if the algorithm is to work well. If the proposal distribution is too narrow or too broad, a large proportion of proposed moves will be rejected.

2.3.2 Non-Random Walk MCMC

Both the Gibbs sampling and Metropolis-Hastings algorithms explore the parameter space via inefficient random walks (Neal, 1992). For complicated models with many parameters these methods may require an unacceptably long time to converge to the target posterior distribution. On the other hand, one of the main benefits of Hamiltonian Monte Carlo (HMC; Duane, et al., 1987) and the no-U-turn sampler (NUTS; Hoffman and Gelman, 2011) is their ability to avoid the inefficient exploration of the parameter space via random walks. This advantage has been elaborated in the MCMC literature (see e.g., Hoffman and Gelman, 2011; Neal, 2011). Hamiltonian Monte Carlo (HMC) borrows an idea from physics to suppress the random walk behavior in the Metropolis algorithm by means of an auxiliary variable “momentum”, that transforms the problem of sampling from a target posterior distribution into the problem of simulating Hamiltonian dynamics, allowing it to move much more rapidly through the posterior distribution (Neal, 2011). Duane, et al. (1987) introduced Hamiltonian Monte Carlo (HMC) by applying it to lattice field theory simulations of quantum electroodynamics and called their method a hybrid Monte Carlo. Statistical applications of HMC started with its application to neural network models by Neal (1996), and have received attention in the Bayesian community. It is, however, underappreciated by the psychometric community. A review of the non-random walk MCMC algorithms is presented next.

2.3.2.1 Hamiltonian Monte Carlo (HMC)

Using Hamiltonian dynamics to sample from the target posterior distribution requires translating the density function for this distribution to a potential energy function and introducing a momentum variable $\emptyset_{\mathrm{j}}$ for each parameter θ_i, in the parameter space, which is referred to now as position variables. The Hamiltonian is an energy function for the joint state of position $\boldsymbol{\theta}$ and momentum $\emptyset$, which defines a joint distribution $p(\boldsymbol{\theta}, \emptyset|\boldsymbol{y})$, also known as the canonical distribution. Since the two parameters are independent, their joint distributing is the product of the posterior density $p(\boldsymbol{\theta}|\boldsymbol{y})$ and the momentum density $p(\emptyset)$, which is usually specified as multivariate normal distribution; $p(\boldsymbol{\theta}, \emptyset|\boldsymbol{y}) = p(\emptyset)\mathrm{p}(\boldsymbol{\theta}|\boldsymbol{y})$. Hamiltonian dynamics maintains the properties of time-reversibility and invariant of the joint distribution (see Hoffman and Gelman, 2011 for more details about these properties). Although sampling is carried out using the joint distribution, the interest is only in the simulations of the position parameter $\boldsymbol{\theta}$ whereas the momentum vector $\emptyset$ is introduced only to enable the algorithm to move faster through the parameter space (Gelman et al., 2014).

The HMC algorithm progresses in two steps. The first step changes only the momentum. In the second step, both vector parameters **θ** and ∅ are updated together in a new Metropolis algorithm; a proposed state is either accepted or rejected according to the Metropolis decision rule except that the terms involve not only the relative posterior distributions, but also the momentum at the current and proposed positions. The steps of the HMC algorithm are outlined below.

1. For each iteration, a random candidate value of the momentum vector ∅ is drawn from its posterior distribution, ∅ ~ $N(0, \Sigma)$, where Σ is the covariance matrix of the momentum distribution p(∅).
2. A simultaneous update of (θ, ∅) is conducted using a discrete mimicking of physical dynamics that involves L 'leapfrog' steps with a step size of ε. L and ε are parameters of the algorithm, which need to be tuned to obtain an adequate performance. The L steps proceed as follows:

   (1) Use the gradient of the log-posterior density $p(\boldsymbol{\theta}|\boldsymbol{y})$ to make a half-step of momentum ∅, That is: $\phi \leftarrow \phi + \frac{1}{2}\varepsilon \frac{d \log p(\theta \mid y)}{d\theta}$.

   (2) Now, use the momentum ∅ to update the position parameter $\boldsymbol{\theta}$ as follows: $\theta \leftarrow \theta + \varepsilon \Sigma^{-1} \phi$.

   (3) Again, use the gradient of the log-posterior density to make a half-step of momentum ∅; $\phi \leftarrow \phi + \frac{1}{2}\varepsilon \frac{d \log p(\theta \mid y)}{d\theta}$.

   The stepping begins and ends with a half-step of momentum ∅, while for the L-1 full steps, the updates (1) and (3) can be applied jointly. This leapfrog algorithm is a discrete approximation to physical Hamiltonian dynamics in which both position and momentum evolve in continuous time.

3. The Metropolis acceptance probability for the HMC is defined as follows:

   $\alpha = \min(\frac{p(\theta^{(t-1)} | y) p(\phi^{(t-1)})}{p(\theta^{(t)} | y) p(\phi^{(t)})}, 1)$, where $\theta^{(t-1)}$ and $\emptyset^{(t-1)}$ are the values of position and momentum parameters at the start of the leapfrog process, while $\theta^{(t)}$ and $\emptyset^{(t)}$ are the corresponding values after the *L* steps. If the proposed state is not accepted, the next state is the same as the current state of the Markov chain.

4. Repeat steps 1 and 3 for large number of *N* iterations until convergence is approximately reached.

HMC is a powerful tool, but its performance depends on choosing suitable values for the step size parameter *ε* and the number of leapfrog steps *L*. Tuning these parameters, and specifically *L* requires some expertise and preliminary runs (Hoffman & Gelamn, 2011; Neal, 2011). A poor choice of either of these parameters will greatly decrease the efficiency of HMC. Furthermore, computing the gradient of the log-posterior for a complex model is tedious and sometimes impossible. However, this requirement can be achieved by using automatic differentiation (Griewank and Walther, 2008). To overcome this, Hoffman and Gelman (2011) introduced the no-U-turn Sampler (NUTS) to eliminate the need to set the number of leapfrog steps *L* that the algorithm takes to generate a proposal state. A review of this algorithm follows below.

2.3.2.2.The No-U-Turn Sampler (NUTS)

The no-U-turn sampler (NUTS) is a non-random walk MCMC that is very similar to HMC but it eliminates the need to specify the number of leapfrog steps parameter *L*. In practice, NUTS performs as efficient as, and sometimes better than, a well-tuned HMC without requiring user interventions. Using a recursive algorithm, NUTS creates a set of candidate points that spans a wide path of the target distribution, stopping automatically when it starts to double back and retrace its steps (i.e. starts to make a U-turn). At this point NUTS stops the simulation and samples from the set of points computed during the simulation.

Hoffman and Gelman (2011) introduced a termination criterion that can, to varying degrees of success, indicate when the Hamiltonian dynamics is simulated long enough to yield a sufficient exploration of the canonical distribution $p(\boldsymbol{\theta}, \emptyset|\boldsymbol{y})$. In other words, the termination criterion tells us that running the simulation for more steps will no longer increase the distance between the proposed $\tilde{\theta}$ and the initial θ values of the position parameter. The criterion is based on the dot product of the current momentum value, $\tilde{\phi}$ and the vector from the initial to the current position, $(\tilde{\theta}-\theta)$, which is the derivative, with respect to time, of half the squared distance between the initial and the current position of θ. That is:

$$\frac{d}{dt}\frac{(\tilde{\theta}-\theta)\bullet(\tilde{\theta}-\theta)}{2}=\tilde{\phi}\bullet(\tilde{\theta}-\theta)\,. \tag{2.14}$$

Such a criterion suggests that leapfrog steps will be run until the value of equation (2.14) becomes less than zero. Thus, Hamiltonian dynamics will be simulated until the proposal position $\tilde{\theta}$ begins to move back towards θ. In order to ensure that the time-reversibility property is satisfied, and hence the algorithm converges to the desired posterior distribution, a recursive algorithm for slice sampling proposed by Neal (2003) is used. The recent release of Stan (Stan Development Team, 2017) has adopted some modifications to the original termination criterion. In addition to the generalized termination criterion, the HMC implementation in Stan uses multinomial sampling instead of slice sampling, which provides a notable improvement in performance of NUTS (Betancourt, 2017).

2.4 Prior Research

IRT models have been fitted using a variety of estimation techniques, whether traditional ones that find a point estimate for model parameters or the fully Bayesian estimation based on MCMC algorithms that approximate the target posterior distribution. The following two subsections review some relevant studies that implemented NUTS to estimate the conventional IRT models as well as studies that fitted MixIRT models using different estimation methods, but not including NUTS as it has not been used yet to estimate such complex models.

2.4.1 Conventional IRT Models Using NUTS

Up to date, there have not been many Bayesian IRT studies conducted using NUTS in the IRT literature. Some of these studies are reviewed next. The first three studies compared NUTS with other fully Bayesian or traditional methods in estimating various forms of IRT models while the last two studies took advantage of NUTS to fit complex IRT models and ultimately developed statistical measures. A summary of each study is illustrated below.

Martin-Fernandez and Revuelta (2017) compared the performance of NUTS, Metropolis-Hastings Robins-Monro (MHRM; Cai, 2010a, 2010b), the MML via the EM algorithm, and Gibbs sampling in an estimation of multidimensional item response models. Results indicated that the four estimation methods perform similarly in recovering the parameters of models up to five factors, while the MML-EM had problems recovering models with more dimensions. Also, results showed that NUTS significantly reduced estimation time and converged faster than the Gibbs sampler, and even faster than the MML-EM algorithm in the small sample conditions.

Chang (2017) examined the performance of NUTS via the use of Stan and Gibbs sampling via the use of JAGS, in estimating the 2PL unidimensional model and the 2PL multi-unidimensional model (Sheng & Wikle, 2007), under various test conditions such as test length, sample size, and prior specification. The results indicated that both algorithms recovered item and person ability parameters with similar accuracy and bias. Moreover, in terms of the computational speed, NUTS was faster that Gibbs sampler in fitting the 2PL unidimensional model, yet NUTS was slower than Stan in fitting the 2PL multi-unidimensional model.

Grant et al., (2016) compared NUTS via the use of Stata-Stan and MH via the use of Stata (StataCorp, 2016) in fitting a Rasch model and a hierarchical extension of the Rasch model. The two algorithms were compared based on speed and the number of effective independent samples. The results showed that NUTS was generally more efficient than MH in estimating parameters of the two models.

Copelovitch, Gandrud, and Hallerberg (2015) fit a hierarchical Bayesian IRT model using NUTS to construct an indicator of supervisory data transparency to international institutions. This indicator was used to measure a country's latent willingness to report yearly the minimal credible data about its financial system to international organizations and investors. The results indicated that the level and changes of financial supervisory transparency both influenced sovereign borrowing costs, but this influence was conditional on characteristics of public indebtedness.

Caughey and Warshaw (2014) developed two time-varying measures of citizen and government policy liberalism in the American states over the past half-century. In order to estimate each state's latent policy liberalism, a dynamic hierarchical group-level IRT model using NUTS was applied. The results showed that that state governments are responsive to shifts in public opinion.

In summary, results of previous studies showed that NUTS is generally more efficient that other traditional or fully Bayesian MCMC techniques in fitting IRT models. Furthermore, NUTS has been successfully applied to develop various statistical measures.

2.4.2 Estimation of Mixture IRT Models

The increase in the number of applications of MixIRT models calls for a simultaneous increase in the use of Markov chain Monte Carlo (MCMC) techniques for estimating these models. The advanced computational techniques associated with MCMC algorithms have enabled MixIRT models to be estimated under the fully Bayesian framework. In the MixIRT modeling literature, ML estimation methods have been the traditional method for estimating parameters of MixIRT models using statistical applications such as Mplus and WINMIRA (von Davier, 2001). Some studies related to the estimation of MixIRT using different estimation methods are reviewed below. The first two studies used MixIRT models to identify latent DIF, using traditional ML estimation methods. The next two studies developed MixIRT models for polytomously scored items, using either traditional or the fully Bayesian MCMC methods. The last study evaluated the performance of the Gibbs sampler under various conditions of mixing proportions and priors. A summary of each study is presented below.

Wu et al., (2017) examined latent DIF on physical functioning (PF) and mental health (MH) subscales of the SF-36 scale that is used to measure health status in a diverse population. The two-parameter graded response IRT model with one latent class was compared to the corresponding multi-class models (i.e. two-, three- and four-class) named as a latent variable mixture (LVM) model. The ML method was used to estimate the model parameters using Mplus. The results indicated that the three-class LVM model fit the PF subscale whereas the two-class LVM model fit the MH subscale. For the PF subscale, persons in class two and class one consistently reported greater limitation than those in class one. For the MH subscale, persons in class two reported more health problems than in class one.

Aryadoust (2015) fit a mixture Rasch model to examine DIF in English as a foreign language listening test and investigate its relationship with persons' cognitive and background factors. The WINMIRA was used to estimate model parameters using the CMLE via the EM algorithm. A two-class model was chosen over other models. Class-one comprised of high-ability listeners capable of multitasking whereas class-two comprised low-ability listeners with limited multitasking skills.

Huang (2016) proposed two MixIRT models for rating-scale items by incorporating a random-effect variable into the mixture generalized partial credit model. The proposed models aimed at detecting latent classes from different levels of extreme response style (ERS), which is defined as a consistent and systematic tendency of a person to locate on a limited number of available rating-scale options. Gibbs sampling implemented in WinBUGS was used to obtain model parameter estimates. Results showed that parameters recovered well, as indicated by values of bias and RMSE, with longer tests, larger samples, and more response options in both MixIRT models. In addition, results showed that ignoring mixtures of latent classes led to a decrement in classification accuracy of the response styles.

Maij-de Meij, Kelderman and van der Flier (2008) applied a mixture nominal response (MixNR) model and a mixture partial credit (MixPC) model to Extroversion and Neuroticism scales of the Amsterdam Biographical Questionnaire. The MLE via the EM algorithm was used to estimate parameters of both models using LEM (Vermunt, 1997). The results showed that a three-class MixNR model was identified as the best fitting model, and those latent classes differed with respect to social desirability and ethnic background. Moreover, the results indicted that application of MixIRT models improved the prediction for the Neuroticism scale, but not for the Extroversion scale.

Cho, et al. (2013) used Gibbs sampling, implemented in WinBUGS, for the mixture Rasch model to evaluate the algorithm. Also, effects of several factors on parameter recovery were examined. These include the specification of priors on the mixing proportions, label switching, model selection, and metric anchoring. Moreover, Gibbs sampling was compared to ML estimation method implemented in Mplus, WINMIRA, and LatentGold (Vermunt & Magidson, 2005). Results indicated that the recovery of the number of latent classes and item parameters were very good for different priors specified for the mixing proportions (i.e. Dirichlet prior, the Dirichlet process withstick–breaking prior, and the multinomial logistic regression model with a covariate). In addition, the recovery of item difficulty parameters improved with an increase in test length and with an increase in sample size. With respect to label switching, label switching was not observed within any Gibbs sampling chains (i.e. Type I), but label switching across chains (i.e. Type II), was detected using Gibbs sampling as well as the MLE methods.

In summary, various forms of MixIRT models have been estimated using the traditional ML methods such as MLE and CLME as well as the fully Bayesian MCMC techniques, in particular the Gibbs sampling. The results indicated that the multiple-class MixIRT models were always identified as the best fitting models compared to the one-class or conventional IRT model. In addition, the results showed that using the Gibbs sampler model parameters were recovered well with the increase in test length and sample size.

CHAPTER 3

METHODOLOGY

This chapter describes the methodology that was used to answer the research questions formulated in Chapter 1. It begins by reiterating the research questions. The next two sections describe the procedures of the two Monte Carlo simulation studies that were carried out to investigate the performance of NUTS in estimating the two-parameter mixture (Mix2PL) IRT model under various test conditions and to compare the performance of the Mix2PL model with the conventional 2PL model under situations where one or more latent classes exist.

3.1 Research Questions

The general research question is whether NUTS can be implemented to fit MixIRT models. The specific research questions, related to the performance of the algorithm to recover model parameters and to detect latent classes, are as follows:

1. How does NUTS perform in estimating the Mix2PL model under various test conditions of sample size, test length, and number of latent classes? with respect to the following:
    a. The accuracy of recovering model parameters including mixing proportions, class mean ability, class item parameters, person abilities, and class memberships of individual persons.
    b. The accuracy of determining the number of latent classes.
2. How does the Mix2PL model compare with the conventional 2PL model under situations where tests involve one or multiple latent classes?

Two simulations studies were conducted to address the two research questions, with the first addressing question 1a while the second addressing research questions 1b and 2.

3.2. Model and Prior Specifications

This study focuses the on dichotomous Mix2PL model. In this model, the conventional two-parameter logistic (2PL) IRT is assumed to hold for each latent class, but the item difficulty and discrimination parameters differ for different classes. Moreover, each person is parameterized by a class membership parameter *g* and a class-specific ability parameter θ_{ig}, whereas each item is parameterized by a different set of difficulty and discrimination parameters for each latent class. The probability of a correct (Y_{ij} = 1) response for person *i* to item *j*, in the two-parameter logistic mixture IRT model is defined as follows:

$$P(Y_{ij}=1|\theta_i)=\sum_{g=1}^{G}\pi_g \times \frac{\exp[a_{jg}(\theta_{ig}-b_{jg})]}{1+\exp[a_{jg}(\theta_{ig}-b_{jg})]}, \tag{3.1}$$

where $g = 1, \ldots, G$ is the latent class indicator, b_{jg} and a_{jg} denote the difficulty and discrimination parameters, respectively, for item *j* in the *g*th class, θ_{ig} denotes the ability for person *i* who belongs to class *g*, and π_g denotes the proportion of persons in each class (i.e., the mixing proportion) such that these proportions sum to one. The MixIRT model shown in equation (3.1) is reduced to the conventional 2PL model defined in equation (2.3) in situations where is only one latent class, $g = 1$.

The item difficulty parameter is defined as the ability required for a person to have a probability of 0.5 to answer the item. In practice, item difficulty ranges from -2 to 2 when the latent ability is assumed to range from -3 to +3. The item discrimination parameter is proportional to the slope of the item characteristic curve (ICC) at the value of difficulty parameter. Its values range practically from zero to +2.

In the IRT literature, the effects of sample size and test length on estimation of item parameters have been largely studied. However, as the complexity of the model increases, the discrepancy in findings also increases. For the 2PL IRT model, different combination of sample size and test length were suggested to be sufficient for accurate parameter estimation such as a sample size of 500 with 20 items (Sahin & Anil, 2017; Stone, 1992), or a sample size of 750 with 20 items (Lim & Drasgow, 1990), or a sample size of 500 with 30 items (Hulin, Lissak, & Drasgow, 1982).

In the MixIRT literature, sample size, test length, and number of latent classes appear to affect parameter recovery of the MixIRT model. For instance, Preinerstorfer and Formann (2012) found that increasing both sample size and number of items led to higher accuracy in estimating parameters of the mixture Rasch model. Moreover, Li, Cohen, Kim, and Cho (2009) found that recovery of item difficultly and discrimination parameters in different MixIRT models (i.e. Mix1PL, Mix2PL, Mix3PL) differed based on the number of latent classes, test length, and sample size. Specifically, these parameters were most affected by the number of latent classes such that when the number of latent classes increased the recovery of model parameters was less accurate. Also, their results indicated that root mean square errors (RMSE) decreased as sample size and test length increased. The percentage of correct classifications of class membership for individual persons increased with an increase in test length up to 30 items

The most frequently encountered sample size, and test length in the MixIRT literature are sample size of 500, 1000, and 2000 with test length of 15, 20, 25 and 30. For instance, Bilir, (2009) and Samuelsen (2005) simulated sample size of 500 and 2000 with 20 items, while Meyer (2010) simulated the same sample sizes but with 25 items. Regarding the proportions of persons in each latent class, equal mixing proportions were used by many studies that fitted MixIRT models for different purposes. For example, Bolt, et al. (2001) as well as Cho, et al. (2013) set the mixing proportions for each latent class to be equal. For example, they set π = (0.50, 0.50) in the two-class condition, (0.33, 0.33, 0.33) in the three-class condition, and (0.25, 0.25, 0.25, 0.25) in the four-class condition. Meyer (2010) and Bolt, et al. (2002) specified mixing proportions of $\pi = (0.50, 0.50)$ for the speeded class and the non-speeded class.

Based on the above review, data were generated using the Mix2PL model as defined in equation (3.1) with equal proportions while manipulating three factors: test length ($J = 20$ or 30), number of latent classes ($G = 2$ or 3), sample size in each subpopulation ($n = 250$ or 500). Specifically, for the two-class condition ($G = 2$), the total number of subjects (N) was 500 or 1000; the mixing proportions were $\pi_1 = 0.50$ and $\pi_2 = 0.50$. For the three-class condition ($G = 3$), the total number of subjects was 750 or 1500; the mixing proportions were $\pi_1 = 0.33$, $\pi_2 = 0.33$, and $\pi_3 = 0.33$

A requirement for model identification is that the item difficulty values within each class sum to zero (Rost, 1990). There exist multiple methods to enforce a sum-to-zero constraint on a parameter vector under fully Bayesian estimation using NUTS. The most efficient way is to define the Gth element as the negation of the sum of the elements 1 through G-1. See Stan Development Team (2017) for more details. However, in this parameterization, placing a prior on the transformed difficulty parameter leads to a different posterior than that resulting from the same prior on difficulty parameter in the original parameterization. For example, providing a normal (0, 3) prior on the transformed parameter will produce a different posterior mode than placing the same prior on the parameter itself. Soft centering is an alternative less efficient approach to achieve a symmetric prior. For example, adding a prior such as $b_g \sim N(0, \sigma_g)$ will provide a kind of soft centering of a parameter vector b_g. This approach is only guaranteed to roughly center (Stan Development Team, 2017). Given this, soft centering was used to apply the sum-to-zero constraint on the difficulty parameter in each latent class (i.e., $b_g \sim N(0, 1)$).

As recommended by some studies (e.g., Bolt, et al., 2002; Meyer, 2010), the mean ability for each latent class (μ_g) was allowed to differ in order to account for quantitative differences between classes. Priors and hyperpriors were selected to be comparable to those adopted by others (e.g., Meyer, 2010; Li, et al., 2009; Wollack, et al., 2003; Bolt, et al., 2002). Specifically, normal prior densities were used for person ability parameters $\theta_{ig} \sim N(\mu_g, 1)$, with a standard normal distribution for the hyperparameter μ_g, and a Dirichlet distribution for the mixing-proportion parameters such that $(\pi_1, \ldots, \pi_G) \sim$ Dirichlet(1, …,1). In addition, a truncated normal prior was specified for the class-specific discrimination $a_{jg} \sim N_{(0, \infty)}(0, 1)$ such that only positive draws from a normal distribution are permitted. Further, the sum-to-one constraint on the mixing proportions was achieved by assigning the mixing proportions a unit simplex, which is defined as a vector with non-negative values whose entries sum to one.

The model parameters were generated such that for the two-class condition ($G = 2$), the person ability parameters were generated from a mixture of two subpopulations where $\theta_1 \sim N(-2, 1)$ and $\theta_2 \sim N(2, 1)$; the class-specific item difficulty parameters were generated from a uniform distribution where $b_1 \sim U(-2, 0)$ and $b_2 \sim U(0, 2)$; and the class-specific item discrimination parameters were generated from a uniform distribution where $a_g \sim U(0, 2)$, $g = 1$ or 2. For the three-class condition ($G = 3$), the person ability parameters were generated from a mixture of three subpopulations where $\theta_1 \sim N(-4, 1)$, $\theta_2 \sim N(0, 1)$, and $\theta_3 \sim N(4, 1)$; the class-specific item difficulty parameters were generated from a uniform distribution where $b_1 \sim U(-2, -0.5)$, $b_2 \sim U(-0.5, 0.5)$, and $b_3 \sim U(0.5, 2)$; and the class-specific item discrimination parameters were generated from a uniform distribution where $a_g \sim U(0, 2)$, $g = 1, 2,$ or 3.

Label switching is one of the challenging identification issues in fully Bayesian estimation of mixture IRT models. It occurs when the posterior distribution remains invariant under a permutation of the class indicators. The problem is exacerbated as the number of mixture components increases, leading to $G!$ identical posterior maxima. There are two types of label switching. The first type, which is commonly referred to as label switching, is observed across iterations within a single MCMC chain. This is what commonly referred to as label switching. The second type of switching occurs when the latent classes switch over replications or for different initial values. One of the common strategies to remedy the problem of label switching is to impose an ordinal constraint on the parameters that identifies the components (Stan Development Team, 2017). To avoid the problem of label switching, an ordinal constraint was imposed on the class mean ability (μ_g) parameter as well as the item difficulty parameters (b_g).

The generated model parameters presented above were chosen such that the unified population consisted of a mixture of latent subpopulations that differ on their abilities. Specifically, for the two-class condition, the low ability class had an average latent ability of -2 (2 standard deviations below the mean), while the high ability class had an average ability of 2 (2 standard deviations above the mean); for the three-class condition, the low ability class had a lower mean (4 standard deviations below the mean) and the high ability class had a higher mean (4 standard deviations above the mean) to further differentiate them from the medium class. In addition, values of item difficulties for each class were generated in order to match (or to be around) the class ability. They were also chosen due to the consideration of fitting the mixIRT model using the ordinal constraints imposed on item difficulty parameters, namely, when μ_g and b_g were generated such that the easiest items were simulated for the first low ability while the most difficult items were simulated for the last high ability group and hence there was no overlap on the values of person abilities or item difficulties across the latent classes.

3.3 Convergence Diagnostics

Convergence of the Markov chains was examined using the Gelman-Rubin R statistic (Gelman & Rubin, 1992). This statistic computes the potential scale reduction factor (PSRF). A PSRF value close to 1 indicates model convergence and in practice, the value of 1.1 has been recommended as the threshold to decide whether the model has converged (Gelman, et al., 2014). Based on the values of Gelman-Rubin R statistic, the number of warm-up iterations that should be discarded because of their dependence on the starting values would be determined. Also, the Gelman-Rubin R statistic was used to determine the number of sampling iterations that should be used to estimate the posterior distribution. Then, a conservative number of warm-up and sampling iterations were taken into account.

3.4 Bayesian Fit Indices

In the second simulation study, model comparisons were used to evaluate the accuracy of recovering the number of latent classes and to compare the Mix2PL model with conventional 2PL model.

Different model selection methods, either under frequentist or the Bayesian framework, have been used in estimating conventional IRT models and MixIRT models. The most popular ones are the Bayesian information criterion (BIC) and Akaike's information coefficient (AIC). Li et al. (2009) examined the performances of BIC, AIC, deviance information coefficient (DIC), pseudo-Bayes factor (PsBF), and posterior predictive model checks (PPMC) in selecting the correct MixIRT model among three competing models (Mix1PL, Mix2PL, Mix3PL), fitted using Gibbs sampling algorithm. Results from a simulation study showed that the indices provided somewhat different recommendations. In particular, the results showed that BIC and PsBF are most effective, AIC and PPMC tend to choose more complex models in some simulating conditions, and DIC is the least effective method. Since no research to date has adopted NUTS to fit MixIRT models, the fully Bayesian selection methods including the widely applicable (or Watanabe-Akaike) information criterion (WAIC; Watanabe, 2010) and the leave-one-out cross-validation, which is computed through Pareto smoothed important sampling (PSIS-LOO; Vehtari, Gelman, & Gabry, 2017) and is incorporated in Stan, have not been used for model selection in the MixIRT literature. Such fully Bayesian methods that use the whole posterior distribution have various advantages over simpler estimates of predictive error such as AIC and DIC, although they are less used in practice due to the requirement of additional computational steps (Vehtari, et al., 2017). Luo and Al-Harbi (2017) compared the performances of WAIC and LOO with four popular methods: the likelihood ratio test (LRT), AIC, BIC, and DIC, in the context of dichotomous IRT model selection (1PL, 2PL, 3PL). The results showed that WAIC and LOO performed better than the other four methods, especially with the 3PL model. Also, AIC was inconsistent with different sample sizes and test lengths. This study focuses on selecting the best IRT model among three competing models, namely, the conventional 2PL, the two-class Mix2PL, and the three-class Mix2PL models, using the two fully Bayesian methods, namely, WAIC and LOO.

WAIC and LOO are two approximation measures that estimate the predictive accuracy of the fitted model using available data, without waiting for out-of-sample data (Gelman et al., 2014). WAIC estimates the out-of-sample expectation by first computing log pointwise posterior predictive density (LPPD) of the data, and then adding a correction (p_{WAIC}) for effective number of parameters to adjust for overfitting. The LPPD is defined as follows:

$$LPPD = \sum_{i=1}^{N} \log \int p(y_i \mid \theta) p_{post}(\theta) d\theta \, , \tag{3.2}$$

where $p_{post}(\theta) = p(\theta|y)$ is the posterior distribution of the parameters. Practically, to compute

LPPD, the expectation can be evaluated by sampling from $p_{post}(\theta)$ as follows:

$$LPPD = \sum_{i=1}^{N} \log(\frac{1}{S}\sum_{s=1}^{S} p(y_i \mid \theta^s) \text{ ,} \tag{3.3}$$

where s =1, 2,…,S denotes number of simulation samples from the posterior density. After computing the LPPD, WAIC can be computed as follows:

$$WAIC = -2LPPD + 2\,p_{WAIC} \text{ .} \tag{3.4}$$

The correction term, p_{WAIC} can be computed in the following two approaches:

$$p_{WAIC1} = 2\sum_{i=1}^{N} (\log(E_{post} p(y_i \mid \theta)) - E_{post}(\log p(y_i \mid \theta))) \text{ ,} \tag{3.5}$$

$$p_{WAIC2} = \sum_{i=1}^{N} \text{var}_{post}(\log p(y_i \mid \theta)) \text{ .} \tag{3.6}$$

As Gelman et al. (2014) noted, the second adjustment as expressed in equation (3.6) is more computationally stable since summing the variance for each data points produces stability. This adjustment, p_{WAIC2}, is implemented in the R package loo (Vehtari, et al., 2017), which is used for computation of both WAIC and LOO.

In Bayesian cross validation, a dataset is repeatedly partitioned into a training set and a validation set. The model of interest is fitted to the training set and a posterior distribution is obtained, with which the fit of the model to the validation set is evaluated. Leave-one-out cross validation (LOO) is a special case of cross validation in which one data point is left out each time and the LPPD is computed with N-1 data points as follows:

$$LPPD_{LOO} = \sum_{i=1}^{N} \log p_{post(-i)}(y_i \mid \theta) \text{ ,} \tag{3.7}$$

where $log\, p_{post(-i)}(y_i|\theta)$ is the log likelihood of the ith dataset without the ith data point, and is computed, according to Gelman, et al. (2014) as follows:

$$\sum_{i=1}^{N} \log p_{post(-i)}(y_i \mid \theta) = \sum_{i=1}^{N} \log(\frac{1}{S}\sum_{s=1}^{S} p(y_i \mid \theta^{is})) \,, \tag{3.8}$$

where θ^{is} is the *s*th simulated value in the posterior distribution conditioning on the *i*th dataset without the *i*th data point. In order to place LOO on the same scale as WAIC, the computed $LPPD_{\text{LOO}}$ is multiplied by -2. According to Gelamn, et al. (2014), WAIC is asymptotically equal to LOO.

3.5 Simulation Study I

As described previously, the design of the first simulation study includes two sample sizes per class (250, 500), two test lengths (20, 30), and two levels of latent classes (2-class, and 3-class) resulting in eight conditions (i.e., 2 sample sizes × 2 test lengths × 2 conditions of latent classes = 8 conditions). Due to the computational expenses and following Cho, Cohen, and Kim (2013), where the mixture Rasch model fitted using Gibbs sampling, ten replications were conducted for each of the eight conditions. Although twenty-five replications is the minimum number of replications recommended in IRT simulation studies using MCMC (Harwell, Stone, Hsu, & Kirisci, 1996), ten replications were adopted to keep the current study at a manageable level. Monte Carlo simulations were carried out to answer part (a) of research question one. Recovery of model parameters including mixing proportions, class mean ability, class item parameters, person abilities, and class memberships of individual persons were examined by fitting the Mix2PL model using NUTS algorithm implemented in the Stan program. Details of the simulation procedure are presented below.

3.5.1 Simulation Procedure

A recovery analysis was conducted to determine the extent to which the generating parameters could be recovered from the simulated data sets. The recovery analysis focused on five issues, the recovery of mixing proportions, the recovery of class-specific mean ability, the recovery of class-specific item parameters, the recovery of person ability parameters, and the recovery of class memberships of individual persons. First, the recovery of mixing proportions, the recovery of class-specific mean ability, and the recovery of class-specific item parameters were assessed using bias, mean square error (MSE), and root mean square errors (RMSE). Bias measures the mean difference between the simulated (i.e. true) parameter and the estimated one across *R* replications. If bias is close to zero, it indicates that the estimated parameter is close to the true parameter. On the other hand, a positive value of bias suggests the parameter is overestimated while a negative value suggests the parameter is underestimated. The bias in estimating each parameter is defined as follows:

$$bias_{\xi} = \frac{\sum_{r=1}^{R}(\hat{\xi}_r - \xi)}{R}, \tag{3.9}$$

where ξ is the true value of the parameter (e.g., π_g, μ_g, a_{jg}, or b_{jg}), and $\hat{\xi}$ is the estimated value of the parameter in the *r*th replication where $r = 1, \ldots, R$.

The RMSE measures the average squared difference between the true parameter and the estimated one across *M* replications. The smaller the value of RMSE, the more accurate the parameter estimate is. The RMSE in estimating each parameter can be expressed as follows:

$$RMSE_{\xi} = \sqrt{\frac{\sum_{r=1}^{R}(\hat{\xi}_r - \xi)^2}{R}}\ . \tag{3.10}$$

The MSE is simply the squared value of the RMSE. Similar to the RMSE, the smaller the value of MSE suggests more accurate the parameter estimate is. To summarize the recovery of item parameters, the bias, MSE, and the RMSE were averaged across items.

Second, to examine the recovery of the person ability parameters, Pearson product-moment correlations between the true and estimated ability parameters were computed, and averaged across the ten replications to obtain summary information.

Finally, the recovery of class memberships of individual persons was assessed by computing the proportion of persons that were correctly classified into the class from which they were simulated. To obtain summary information, these proportions were averaged across the ten replications.

3.6 Simulation Study II

Another set of Monte Carlo simulations was carried out to investigate the performance of NUTS in correctly identifying the number of latent classes for the Mix2PL model and the conventional 2PL IRT model using fully Bayesian fit indices, namely the widely applicable information criterion (WAIC) and the leave-one-out cross-validation (LOO). Also, The performance of the Mix2PL model in comparison to the 2PL model was compared in conditions where one or multiple latent classes existed. Details of the simulation procedure are described in the following section.

3.6.1 Simulation Procedure

Two test conditions were considered, with the first treating the two-class Mix2PL model as the true model whereas the second treating the conventional 2PL IRT model as true. With binary item response data generated from each condition for sample sizes of 500 and test lengths of 20, NUTS was implemented to fit the conventional 2PL model (equivalent to the one-class Mix2PL model), the two-class Mix2PL model, and the three-class Mix2PL model.

In order to assess the recovery of the number of latent classes for each data set, the three fitted models were compared using the fully Bayesian fit indices WAIC and LOO. The model with the smallest values of WAIC or LOO was selected as the best fitting model. With twenty-five replications, the proportion of the time the generating model was selected as the best fitting model indicates the accuracy of identifying the number of latent classes. The two fit indices were further averaged across replications to provide summary information.

CHAPTER 4

RESULTS

This chapter summarizes the simulation results for evaluating the performance of the non-random walk MCMC algorithm, namely NUTS, in fitting the two-parameter mixture (Mix2PL) IRT model and for comparing it to the conventional two-parameter (2PL) IRT model. The results are organized in two sections. The first section presents the results of parameter recovery of the Mix2PL model. Model comparison results are presented in Section two to compare the performance of the Mix2PL model with the conventional 2PL model under situations where one or more latent classes exist.

4.1 Parameter Recovery Results

In the first simulation study, convergence of the Markov chains was examined using the Gelman-Rubin R statistic (Gelman & Rubin, 1992). Different numbers of iterations were used to reach convergence. Table 1 summarizes the number of warm-up (or burn-in) and sampling iterations for the eight simulated conditions. For the conditions involving two latent classes, the warm-up stage of either 2000 or 3000 iterations followed by 3 chains with either 3000 or 5000 sampling iterations was sufficient for the chains to reach convergence when the sample size was 500 or 1000, respectively. For the conditions involving three latent classes, in order to reach convergence, the warm-up stage had to reach 3000, 5000 or 8000 iterations followed by 3 chains with 5000, 7000 or 10000 sampling iterations for $N = 750$ or $N = 1500$, respectively. Ten replications were conducted for each of the simulated condition. The Gelman-Rubin R statistic was less than the recommended threshold of 1.10 for each model parameters under all simulated conditions, indicating that convergence was achieved.

Moreover, trace plots of model parameters were examined to visually assess convergence and mixing across chains, where the sampled values of the parameter are plotted on the X-axis against the number of the sampling iterations on the Y-axis for each chain. For illustrative purposes, Figure 1 shows such plots for the two mixing proportion parameters under the situation with two latent classes, 500 person, and 20 items. Both trace plots appear as fat hairy caterpillars indicating that the three chains mixed well and converged to the posterior distribution.

Table 1
Number of warm-up and sampling iterations for the eight simulated conditions.

G = 2				*G* =3			
N	*J*	Warm-up	Sampling	*N*	*J*	Warm-up	Sampling
500	20	2000	3000	750	20	3000	5000
	30	2000	3000		30	3000	5000
1000	20	3000	5000	1500	20	5000	7000
	30	3000	5000		30	8000	10000

Note. *G* = number of latent classes; *N* = number of persons; *J* = number of items.

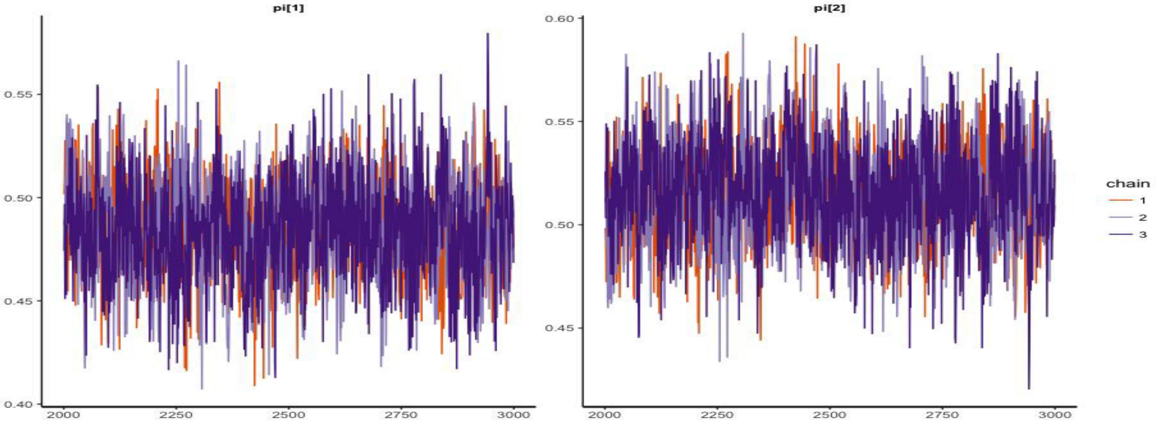

Figure 1. Trace plots of the mixing-proportion (π) parameter for the condition where $G = 2$, $N = 500$, and $J = 20$.

A recovery analysis was conducted to assess the extent to which the model parameters could be recovered from the simulated data sets. The following five subsections present the recovery results for the Mix2PL model parameters including the mixing proportions, the mean abilities, the item parameters, the person abilities, and the class memberships of individual persons, respectively.

4.1.1 Class Mixing Proportions Recovery

To evaluate the accuracy of recovering the mixing-proportion parameter for each latent class in the Mix2PL model, the bias, mean square error (MSE), and root mean square error (RMSE) based on ten replications were computed. The results are summarized in Tables 2 and 3 for the two- and the three- class conditions, respectively. The results suggest that NUTS accurately recovered the mixing-proportion parameters no matter whether the generated data sets consisted of two or three latent subpopulations. The values of bias and RMSE were close to zero, which indicate that the estimated mixing proportions were close to the simulated ones. The maximum absolute value of bias was 0.006 in the three-latent class condition where the sample size was 750 and the test length was 30. The maximum value of the RMSE was 0.019 in the two-class condition where the sample size was 500 and the test length was 20 items.

For the two-class scenarios, the RMSEs for estimating the mixing proportion parameters tended to decrease with the increase of either sample size or test length. For example, in the condition where there were 20 items, the RMSEs decreased from 0.019 to 0.012 when sample size increased from 500 to 1000. In addition, in the condition where the sample size was 500, the RMSEs decreased from 0.019 to 0.013 when the test length increased from 20 to 30 items. However, this pattern was not observed with the three-class scenarios. Specifically, for the three-class scenarios, the results show that the RMSEs tended to decrease with the increase in sample sizes except for one condition with 30 items where the RMSE for the mixing proportion for the second class (π_2) increased from 0.008 to 0.012. In addition, RMSEs tended to decrease with the increase in test length except for three conditions, and these conditions are for recovering π_3 when $N = 750$, and for recovering π_2 as well as π_3 when $N = 1500$. (In particular, the RMSEs for the third class (π_3) did not change when the sample size was 750 (i.e. RMSE $_{(\pi 3)} = 0.01$), whereas when the sample size was 1500, the RMSEs increased from 0.008 to 0.012 and from 0.007 to 0.008 for the second and the third classes, respectively).

Given that both two- and three-class conditions considered the same sample size per class ($n = 250$ or 500) and test length ($J = 20$ or 30) conditions, parameter recovery results can also be compared across the $G = 2$ versus $G =3$ scenarios. Hence, a comparison of Tables 2 and 3 reveals the following observation:

- The *RMSE*s for estimating the mixing-proportion parameters tended to decrease with the increase in the number of latent classes from two to three classes, except for one scenario (i.e., $N = 1000$, $J = 30$). Specifically, the RMSEs for the two-class condition (e.g., $n = 250$ so that $N = 500$, $J = 20$) were 0.019 and 0.019 for the first and second latent classes, respectively, while the RMSEs for the same test condition with three classes (i.e., $n = 250$ so that $N = 750$, $J = 20$) were 0.008, 0.010, and 0.010 for the first, second, and third latent classes, respectively. This pattern, however, was not observed for the scenario where $n = 500$ for each class and $J = 30$. Specifically, for the two-class condition, the RMSEs were 0.011 and 0.011 for the first and second latent classes, respectively, while those for the three-class condition were 0.007, 0.012, and 0.008, for the first, second, third latent classes respectively.
- For the two-class scenarios, π_1 tended to be underestimated while π_2 tended to be

overestimated, except in one scenario where $N = 1000$ and $J = 20$. For the three-class scenarios, π_1 tended to be underestimated while π_2 or π_3 tended to be overestimated.

Table 2
Bias, MSE, and RMSE for recovering mixing proportions with two latent classes.

N	*J*	Parameters	Bias	MSE	RMSE
500	20	π_1	-0.004	0.000	0.019
		π_2	0.004	0.000	0.019
	30	π_1	-0.003	0.000	0.013
		π_2	0.003	0.000	0.013
1000	20	π_1	0.005	0.000	0.012
		π_2	-0.005	0.000	0.012
	30	π_1	-0.001	0.000	0.011
		π_2	0.001	0.000	0.011

Table 3
Bias, MSE, and RMSE for recovering mixing proportions with three latent classes.

N	*J*	Parameters	Bias	MSE	RMSE
750	20	π_1	-0.002	0.000	0.012
		π_2	0.001	0.000	0.012
		π_3	0.001	0.000	0.010
	30	π_1	-0.002	0.000	0.008
		π_2	-0.002	0.000	0.010
		π_3	0.006	0.000	0.010
1500	20	π_1	-0.001	0.000	0.010
		π_2	-0.001	0.000	0.008
		π_3	0.002	0.000	0.007
	30	π_1	-0.005	0.000	0.007
		π_2	0.010	0.000	0.012
		π_3	-0.005	0.000	0.008

4.1.2 Recovery of Class Mean Ability

Similarly, the bias, MSE, and RMSE were obtained to evaluate the recovery of the mean ability for each latent class. The results are summarized in Tables 4 and 5 for the two- and the three- class conditions, respectively. From the tables, we can observe that NUTS performed well in recovering the mean ability for the latent classes, especially for the two-class scenarios. The maximum value of the RMSE equaled 0.205 in the two-class condition where $N = 500$ and $J = 20$ while the corresponding value in the three-class condition was 0.363 when $N = 1500$ and $J = 30$.

It is further noted that for the three-class scenarios, the accuracy of estimating the mean ability of the second latent class was better than that of the first or third latent class (see Figure 2). Moreover, the precision of the mean ability estimates for the second latent class improved with the increase in the sample size. For example, in the condition where $N = 750$ and $J = 20$, the RMSE for estimating μ_2 was 0.102 while the RMSEs for estimating μ_1 and μ_3 were 0.242 and 0.260, respectively. When the sample size increased to 1500, the RMSE for estimating μ_2 decreased to 0.085 while the RMSEs for estimating μ_1 and μ_3 changed to 0.260 and 0.189, respectively.

Table 4
Bias, MSE, and RMSE for recovering mean ability with two latent classes.

N	*J*	Parameters	Bias	MSE	RMSE
500	20	μ_1	-0.016	0.042	0.205
		μ_2	-0.085	0.039	0.196
	30	μ_1	-0.003	0.000	0.013
		μ_2	0.003	0.000	0.013
1000	20	μ_1	0.116	0.039	0.197
		μ_2	-0.039	0.021	0.146
	30	μ_1	0.111	0.023	0.152
		μ_2	-0.089	0.022	0.150

Table 5
Bias, MSE, and RMSE for recovering mean ability with three latent classes.

N	*J*	Parameters	Bias	MSE	RMSE
750	20	μ_1	0.074	0.058	0.242
		μ_2	-0.008	0.010	0.102
		μ_3	-0.026	0.067	0.260
	30	μ_1	-0.292	0.111	0.333
		μ_2	-0.034	0.018	0.133
		μ_3	0.190	0.086	0.293
1500	20	μ_1	0.032	0.067	0.260
		μ_2	-0.031	0.007	0.085
		μ_3	0.075	0.036	0.189
	30	μ_1	-0.282	0.126	0.355
		μ_2	-0.025	0.004	0.062
		μ_3	0.284	0.132	0.363

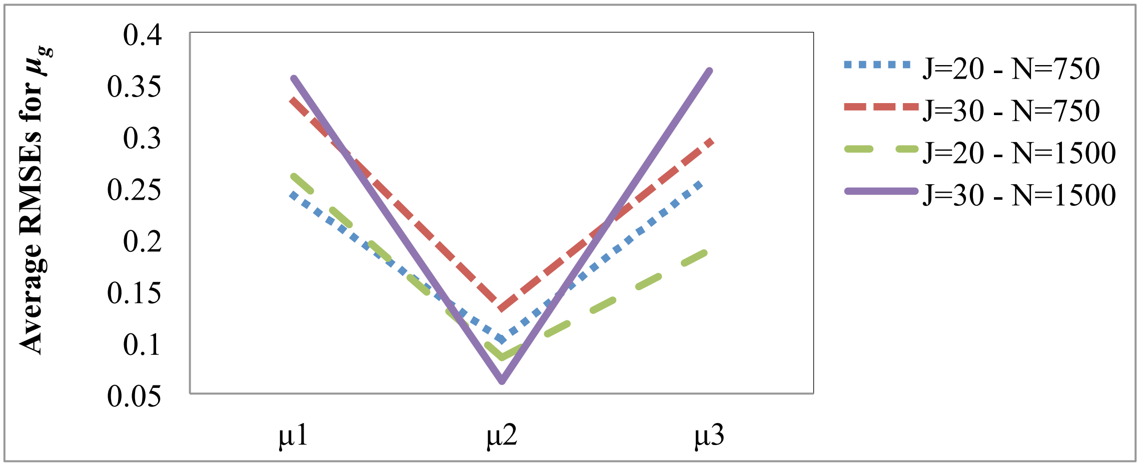

Figure 2. RMSE for recovering class mean ability (μ_g) for the each latent class under the four scenarios of the three-latent class condition.

Further, for the two-class scenarios, the results show that the RMSEs/MSEs tended to decrease with the increase in test lengths except for recovering μ_2 in the condition where N = 1000 and J = 30 (see Table 4). However, for the three-class scenarios, the RMSEs/MSEs tended to increase with the increase in test length except for recovering μ_2 in the condition where N = 1500 and J = 30 (see Table 5).

An investigation of Tables 4 and 5 concerning the recovery of the mean ability parameter for the two- versus the three-class situations reveals that without considering μ_2 for the three-class scenarios, the RMSEs/MSEs tended to increase with the increase in the number of latent classes from two to three classes. For example, the RMSEs for the two-class condition where n = 250 for each class and J = 20 were 0.205 and 0.196 for the first and second latent classes, respectively, while those for the three-class condition were 0.242 and 0.260 for the first and third latent classes, respectively.

4.1.3 Item Parameter Recovery

To evaluate the recovery of the discrimination (a_j) and the difficulty (b_j) parameters, values of the bias, MSEs, and RMSEs were averaged across items. The results are summarized in Tables 6 and 7 for the two- and the three-class conditions, respectively. For visual help, the average RMSEs for recovering the discrimination parameters are plotted in Figures 3 and 4 for the two- and the three-class conditions, respectively, while those for recovering the difficulty parameters are plotted in Figures 5 and 6 for the two- and the three-class conditions, respectively. These results indicate that NUTS had consistently smaller average bias, MSE, or RMSE values in recovering the discrimination parameter than the difficulty parameter of the Mix2PL model for both classes in the two-class condition and for the first and third classes in the three-class condition. However, the difficulty parameter had a smaller average bias, MSE, or RMSE values than the discrimination parameter for the second class in the three-class condition.

The small negative values of the average bias for estimating the discrimination parameters suggest that they were slightly underestimated across all the simulated conditions except for one condition (i.e., $N = 1500$ and $J = 20$) where the discrimination for the first class was overestimated (see Table 7). In addition, values of the averaged RMSEs/MSEs were relatively small with a maximum value of RMSE being 0.482 for estimating the discrimination parameter for the second class in two data size conditions (i.e., ($N =750$, $J = 30$) and ($N = 1500$, $J = 20$)). For the two-class condition, the recovery of the discrimination parameters improved with the increase in sample size or test length (see Figure 3). This pattern, however, was not observed with the three-class condition, which has mixed results (see Figure 4). In particular, for the three-class condition, when $N = 750$, the RMSEs/MSEs for the discrimination parameters a_1, a_2, and a_3 tended to decrease with the increase in test length. However, when $N = 15000$, the RMSEs/MSEs for the discrimination parameters a_2 and a_3 tended to increase with the increase in test length, but the RMSEs/MSEs for a_1 tended to increase with the decrease in test length (see Figure 4).

For the difficulty parameters, they were consistently underestimated for the last latent class while overestimated for the other classes, no matter whether there were two or three classes (see Tables 6 and 7). Also for the three-class condition, the recovery of the difficulty parameters in the second class, as indicated by the average values of *bias* and *RMSE*/MSE, was better than the recovery of those in the first or third class across the four data sizes. For example, in the condition where the sample size was 750 and test length was 20, the average bias for the second class was 0.057 while those for the first and third classes were 0.386 and -0.398, respectively. For this same condition, the average RMSE for the second class was 0.421 while those for the first and the third classes were 0.522 and 0.590, respectively. It is noteworthy that this same pattern occurred in the recovery of the class mean ability as illustrated in Figure 2.

Moreover, for the two-class condition, the recovery of the difficulty parameters for the first class (b_1) became less accurate with the increase in sample size, but it improved with the increase of test length. On the other hand, the recovery of the difficulty parameters for the second class (b_2) improved with the increase in sample size, yet it became less accurate with an increase of test length (see Table 6).

In addition, a comparison of Tables 6 and 7 for the recovery of the item parameters for the two- versus the three-class situations leads to the following observations:

- The RMSEs for estimating the discrimination parameters tended to increase with the increase in the number of latent classes from two to three classes. For example, the RMSEs for the two-class condition with n = 250 for each class and J = 30, were 0.356 and 0.359 for the first and second latent classes, respectively. Yet, the RMSEs for the three-class condition with n = 250 for each class and J = 30, were 0.413, 0.482, and 0.452 for the first, second, and third latent classes respectively.
- The RMSEs for estimating the difficulty parameters tended to decrease with the increase in the number of latent classes from two to three classes. For example, the RMSEs for the two-class condition with n = 250 for each class and J = 30, were 0.601 and 0.691 for the first and second latent classes, respectively. However, the RMSEs for the three-class condition with n = 250 for each class and J = 30, were 0.509, 0.396, and 0.545 for the first, second, and third latent classes respectively.

Table 6

Average Bias, MSE, and RMSE for recovering item parameters with two latent classes.

N	*J*	Parameters	Bias	MSE	RMSE
500	20	a_1	-0.074	0.158	0.397
		a_2	-0.063	0.160	0.400
		b_1	0.396	0.391	0.626
		b_2	-0.457	0.448	0.669
	30	a_1	-0.061	0.127	0.356
		a_2	-0.055	0.129	0.359
		b_1	0.419	0.361	0.601
		b_2	-0.493	0.478	0.691
1000	20	a_1	-0.014	0.089	0.298
		a_2	-0.076	0.115	0.339
		b_1	0.447	0.407	0.638
		b_2	-0.397	0.353	0.594
	30	a_1	-0.020	0.083	0.288
		a_2	-0.037	0.090	0.300
		b_1	0.436	0.380	0.616
		b_2	-0.382	0.370	0.609

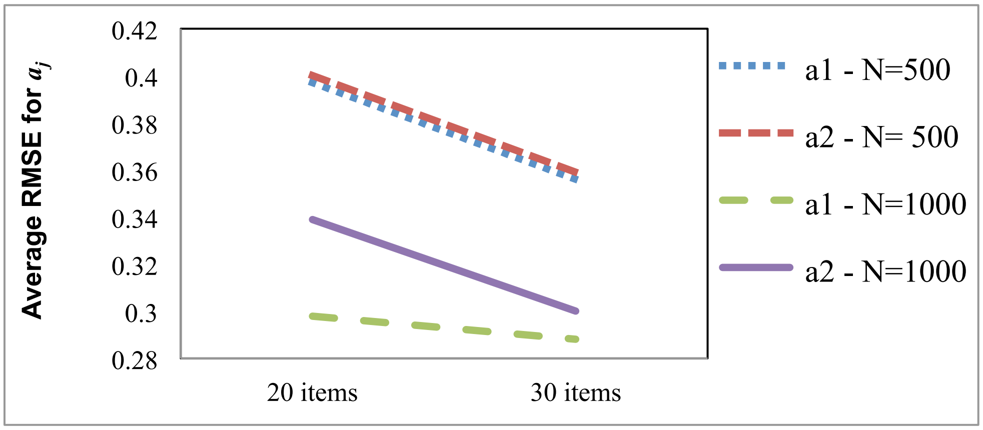

Figure 3. Average RMSEs for recovering the discrimination (a_j) with two latent classes.

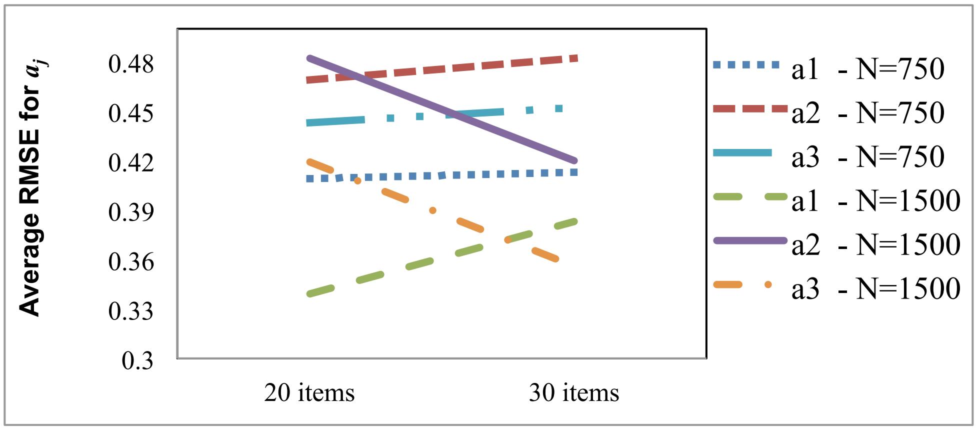

Figure 4. Average RMSEs for recovering the discrimination (a_j) with three latent classes.

Table 7

Average Bias, MSE, and MSE for recovering item parameters with three latent classes.

N	*J*	Parameters	Bias	MSE	RMSE
750	20	a_1	-0.054	0.167	0.409
		a_2	-0.049	0.220	0.469
		a_3	-0.053	0.197	0.443
		b_1	0.386	0.273	0.522
		b_2	0.057	0.177	0.421
		b_3	-0.398	0.348	0.590
	30	a_1	-0.108	0.171	0.413
		a_2	-0.078	0.232	0.482
		a_3	-0.085	0.204	0.452
		b_1	0.341	0.259	0.509
		b_2	0.017	0.156	0.396
		b_3	-0.375	0.297	0.545
1500	20	a_1	0.023	0.115	0.339
		a_2	-0.058	0.233	0.482
		a_3	-0.096	0.176	0.419
		b_1	0.352	0.267	0.517
		b_2	0.054	0.177	0.421
		b_3	-0.311	0.249	0.499
	30	a_1	-0.058	0.147	0.383
		a_2	-0.071	0.176	0.420
		a_3	-0.088	0.126	0.356
		b_1	0.377	0.311	0.558
		b_2	0.035	0.143	0.379
		b_3	-0.421	0.336	0.579

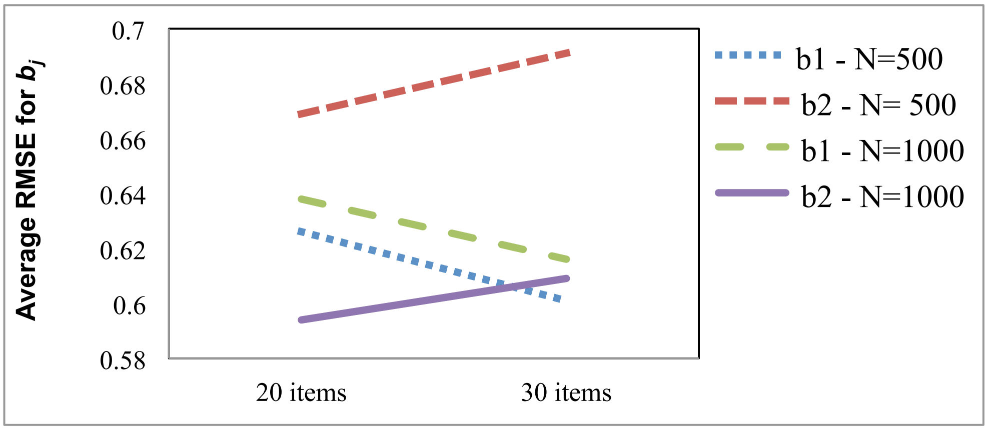

Figure 5. Average RMSEs for recovering the difficulty (b_j) with two latent classes.

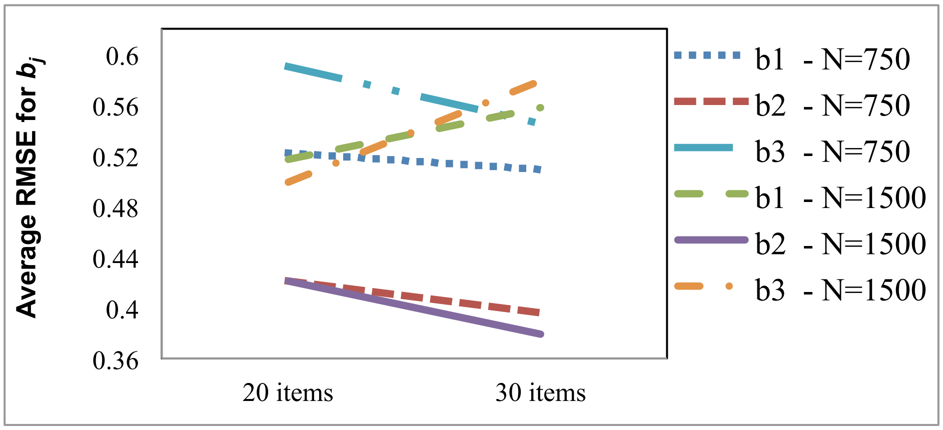

Figure 6. Average RMSEs for recovering the difficulty (b_j) with three latent classes.

4.1.4 Person Ability Parameter Recovery

Correlations between the true and the estimated person abilities were used to evaluate how well NUTS have recovered the person ability parameters under the different simulated conditions. Results are presented in Tables 8 and 9 for the two- and the three-class conditions, respectively. For visual help, the correlation values are summarized in Figures 7 and 8 for the two- and three-class conditions, respectively. The consistently large values of the correlations (i.e., $r(\theta,\hat{\theta})$ > 0.950) indicate that NUTS accurately recovered the person ability parameters no matter whether the population consisted of two or three latent subpopulations. In addition, the person ability parameters were estimated more accurately with an increased test length, for both the two- and the three-class conditions (see Figures 7 and 8). As an example, for the two-class condition, when the test length increased from 20 to 30 items, the average $r(\theta,\hat{\theta})$ increased from 0.953 and 0.954 to 0.966 for both sample sizes 500 and 1000.

Table 8
Correlations between the true and estimated person abilities with two latent classes.

N	*J*	$r(\theta,\hat{\theta})$	*SE*
500	20	0.953	0.001
	30	0.966	0.001
1000	20	0.954	0.002
	30	0.966	0.001

Table 9
Correlations between the true and estimated person abilities with three latent classes.

N	*J*	$r(\theta,\hat{\theta})$	*SE*
750	20	0.958	0.007
	30	0.973	0.001
1500	20	0.963	0.001
	30	0.972	0.001

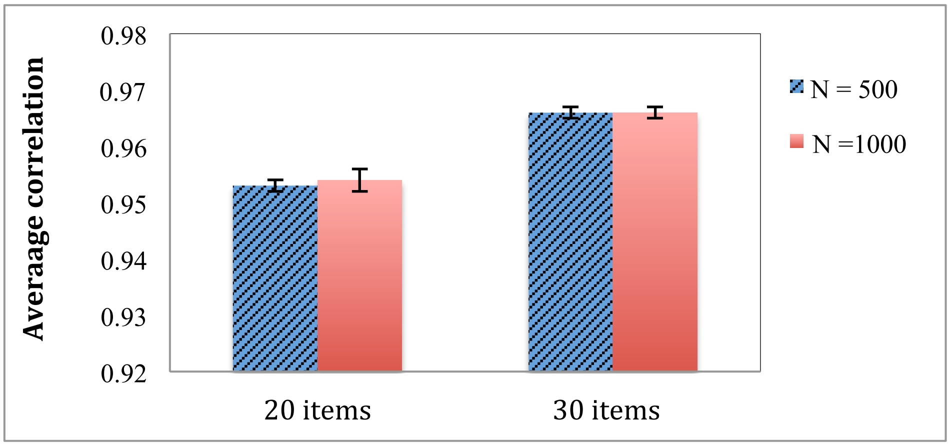

Figure 7. Average correlations ($r(\theta,\hat{\theta})$) between the true and estimated person abilities with two latent classes.

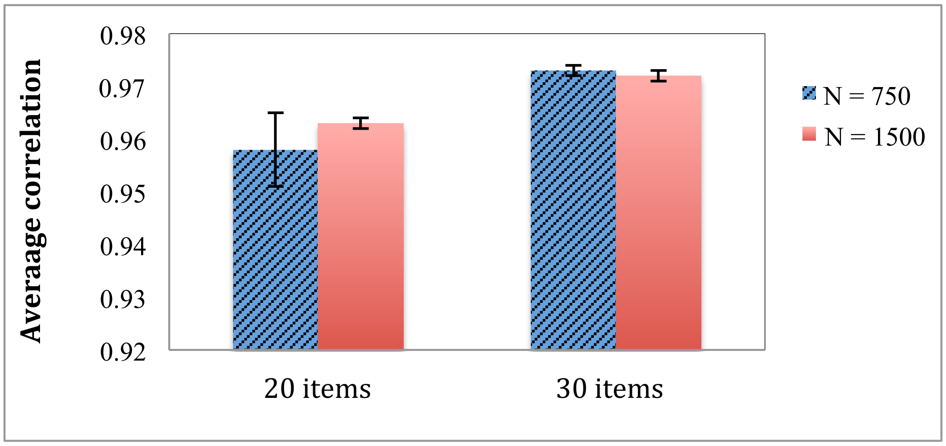

Figure 8. Average correlations ($r(\theta,\hat{\theta})$) between the true and estimated person abilities with three latent classes.

4.1.5 Class Membership Recovery

For the class membership, the percentages of correct classifications of individual persons were computed and displayed in Tables 10 and 11. The results suggest that NUTS was fairly accurate when the population consisted of two latent subpopulations. The average percentages of correct classifications, across the ten replications, for the four data sizes were 90.96, 92.38, 93.55, and 94.44 (see Table 10). However, in the conditions where the population consisted of three latent subpopulations, the recovery was less accurate, where the average percentages of correct classifications for the four data sizes were 69.65, 69.91, 71.59, and 75.13 (see Table 11). Moreover, the recovery of class memberships is apparently affected by sample size and test length. Specifically, the average percentage of correct classifications increased with an increase in sample size or test length, for both the two- and the three-class conditions.

Table 10
Percent of correct classifications of individual persons with two latent classes.

N	*J*	Average	Minimum	Maximum
500	20	90.96	74.40	97.20
	30	92.38	80.80	97.20
1000	20	93.55	82.80	96.10
	30	94.44	86.50	97.20

Table 11
Percent of correct classifications of individual persons with three latent classes.

N	*J*	Average	Minimum	Maximum
750	20	69.65	65.20	81.60
	30	69.91	66.53	87.60
1500	20	71.59	66.60	83.40
	30	75.13	64.20	90.73

4.2 Model Comparison Results

In the second simulation study, the convergence of Markov chains was also evaluated using the Gelman-Rubin R statistic, with a threshold of 1.10 as suggested by Gelman, et al. (2014). For the first condition where data conformed to the two-class Mix2PL model, Table 12 shows the number of warm-up and sampling iterations for the three candidate models. For two of the candidate models, namely the conventional 2PL IRT model and the three-class Mix2PL model, the warm-up stage was set to 6000 iterations followed by 6 chains with 9000 sampling iterations. For the two-class Mix2PL candidate model (i.e., the true model), the warm-up stage was set to 3000 iterations followed by 3 chains with 5000 sampling iterations. The Gelman-Rubin R statistic was less than the recommended threshold of 1.10 for each model parameters across the three candidate models indicating that convergence was achieved.

For the second condition where data conformed to the conventional 2PL IRT model, neither of the two MixIRT models reached convergence even with a warm-up stage of 60,000 iterations followed by 80,000 sampling iterations. It is possible to reach convergence by adding substantially more iterations. However, its computational expense causes problem and hence results for this condition are not reported. Although the three candidate models could not be compared given the non-convergence, it is noted that the conventional 2PL IRT model did converge with a warm-up of 3,000 iterations followed by 5,000 sampling iterations.

Table 12
Number of warm-up and sampling iterations where data conformed to the two-class Mix2PL model.

Model	Warm-up	Sampling	Chains
2PL (one-class)	6000	9000	6
Mix2PL (two-class)	3000	5000	3
Mix2PL (three-class)	6000	9000	6

For the first condition where data were generated from the two-class Mix2PL model, the three fitted models were compared using two fully Bayesian fit indices, namely the widely applicable information criterion (WAIC; Watanabe, 2010) and the leave-one-out cross-validation (LOO-PSIS; Vehtari, et al., 2017). The model with the smallest values of WAIC or LOO was selected as the best fitting model. With twenty-five replications, the proportion of the time the generating model was selected as the best fitting model was used to assess the precision of recovering the number of latent classes and is presented in Table 13. Values of the two fit indices were further averaged across replications to provide summary information as shown in Table 14.

The results suggest that WAIC performed better than LOO in recovering the number of latent classes. Specifically, LOO correctly detected the number of classes 44% of the time while WAIC was correct 80% of the time (see Table 13). In addition, the average WAIC value favored the correct model, whereas the average LOO favored the Mix2PL model with three classes; however, the difference between the two LOO values for the two- and three-class Mix2PL models is rather small (i.e., 0.352; see Table 14). As recommended by Gelman et al. (2014), when deciding on the best fitting model, the effective number of parameters associated with Bayesian fit indices should also be taken into account, especially when the differences between the values of these indices for the candidate models are small, such that the simpler model is preferred over the more complex one. The effective number of parameters presented in Table14 indicates that the two-class Mix2PL model was the least complex (p_{LOO} = 378.728, p_{WAIC} = 373.924) compared to the three-class Mix2PL model (p_{LOO} = 379.876, p_{WAIC} = 375.868) or the 2PL IRT model (p_{LOO} = 429.976, p_{WAIC} = 422.276). Hence, based on the average values of LOO and WAIC along with their associated average effective number of parameters, the results suggest that the two-class Mix2PL model (i.e., the correct model) is the best fitting model selected by both fully Bayesian fit indices. It is also noted that the conventional 2PL IRT model (i.e., the one-class Mix2PL model), with a substantially larger LOO or WAIC, was never selected as the best fitting model.

Table 13

Frequencies and relative frequencies for selecting three candidate models where the generating model is the two-class Mix2PL model.

Candidate model	Model selection method			
	LOO		WAIC	
	Frequency	Relative frequency	Frequency	Relative frequency
2PL (one-class)	0	0.00	0	0.00
Mix2PL (two-class)	11	0.44	20	0.80
Mix2PL (three-class)	14	0.56	5	0.20
Total	25	1.00	25	1.00

Note. The maximum frequency of selecting a model is 25.

Table 14

Average LOO and WAIC for recovering number of latent classes where the generating model is the two-class Mix2PL model.

Candidate model	Model selection method			
	LOO	p_{LOO}	WAIC	p_{WAIC}
2PL (one-class)	9935.384	429.976	9919.976	422.276
Mix2PL (two-class)	9877.708	378.728	9868.120	373.924
Mix2PL (three-class)	9877.356	379.876	9869.340	375.868

CHAPTER 5

DISCUSSION AND CONCLUSION

This chapter consists of two main sections. The first section summarizes the performance of the no-U-turn sampler (NUTS) in terms of parameter recovery of the two-parameter mixture (Mix2PL) IRT model as well as model comparison when one or more latent subpopulations exist. Section two discusses limitations of this dissertation and provides directions for future studies.

5.1 Performance of NUTS

Findings based on the two simulation studies related to parameter recovery and model comparison are summarized and discussed in the following two subsections.

5.1.1 Parameter Recovery

The first simulation study evaluates the performance of NUTS in terms of parameter recovery of the Mix2PL model by manipulating three factors: sample size (N), test length (J), and the number of latent classes (G). With Monte Carlo simulations, results of this study as presented in Section 4.1 suggest that overall, NUTS performs well in recovering parameters for the Mix2PL model, including the class parameters (π_g and μ_g), item parameters (a_{jg} and b_{jg}), and person parameters (θ_{ig}, g), although the recovery of the class membership of individual persons is not satisfactory for the three-class situation, which has a maximum of 75.13% of correct classification versus an average of 94.44% for the corresponding two-class condition.

With respect to the effects of sample size and/or test length, they play a role in recovering the class membership and person ability parameters no matter whether the generated data sets consisted of two or three latent subpopulations. Specifically, the proportion of correct classification of class membership increases with either sample size or test length, which is consistent with previous research (e.g., Cho, Cohen, & Kim, 2013). In addition, increased test lengths improve the precision in estimating person abilities. This finding is consistent with those from the IRT literature based on other models or estimation methods (i.e., Chang, 2017; Kuo, 2015; Sheng, 2005). Therefore, in order to obtain a better recovery of the person ability parameter, more items should be considered.

On the other hand, the sample size and/or test length effect on estimating other parameters in the Mix2PL model is not clear. Some patterns of recovery improvement with the increment of sample size and/or test length in the two-class condition are not observed in the three-class condition. For example, for the two-class condition, the accuracy of estimating the mixing-proportion parameters increases with the increase of either sample size or test length but this pattern is not observed with the three-class condition. In addition, for the two-class condition, the recovery of the discrimination parameter improves with the increase of either sample size or test length, which agrees with findings of Li et al. (2009); however, this pattern is not observed with the three-class condition. This is possibly due to the increased complexity of the mixture item response theory (MixIRT) model with the increased number of latent classes. Adding one subpopulation may seem trivial, but it would result in a substantial increase in the number of parameters to be estimated. For example, when fitting a two-class Mix2PL model to a data set with a sample size of 500 persons and a test length of 20 items, we need to estimate 1,584 parameters including: 2 mixing proportions, 2 class mean abilities, 40 difficulty parameters, 40 discrimination parameters, 500 person abilities, and a total of 1000 probabilities for all persons being on each of the two classes. On the other hand, when fitting a three-class Mix2PL to the same data set (i.e., $N = 500$, $J = 20$), 2,126 parameters are to be estimated, and they are: 3 mixing proportions, 3 class mean abilities, 60 difficulty parameters, 60 discrimination parameters, 500 person abilities, and a total of 1500 probabilities for all persons being on each of the three classes. This is already over a one-third increase for such a relatively small data size.

This complexity is further reflected in the estimation of person mean ability or item discrimination parameters, whose accuracy decreases with the increased number of classes, which agrees with previous research (Li et al., 2009) concerning the discrimination parameters when estimating the Mix2PL and Mix3PL models using Gibbs sampling. On the other hand, the recovery of the mixing proportions or individual item difficulties in the model is not seemingly affected by such added complexity, which is consistent with findings of Cho, Cohen, and Kim (2013) on the recovery of the difficulty parameters in the mixture Rasch model using Gibbs sampling. As a matter of fact, the RMSE values for the mixing proportions or individual item difficulties decrease when adding one more subpopulation. This reduction can be due to the fact that the magnitude of RMSE depends on the unit/scale of the parameter. For instance, the mixing proportion is larger for the two-class condition ($\pi_g = 0.50$) than the three-class condition ($\pi_g = 0.33$), and hence the RMSEs tend to be larger with the two-class condition. This is certainly a limitation of using RMSE for evaluating the accuracy in recovering model parameters in this study. Future studies shall consider other measures, such as the relative RMSE or normalized RMSE that are free from the scale of the parameters.

In terms of the precision of estimating item parameters (a_{jg} and b_{jg}), the results indicate that NUTS results in smaller RMSEs in recovering the item discrimination parameters than the item difficulty parameters for both classes in the two-class condition and for the first and third classes in the three-class condition but the opposite is true for the second class in the three-class condition. The seemingly better estimate of the difficult parameter in comparison to the discrimination parameter for the second class in the three-class condition is again due to the aforementioned limitation of RMSE, given that the scale of the difficulty parameter for this class (-0.5, 0.5) is much smaller than that for the discrimination parameter (0, 2). On the other hand, for the two-class condition, the scales for both item difficulty and discrimination parameters are the same, while for the first and third classes in three-class condition; the scale of the discrimination parameters (0, 2) is relatively larger than those for the difficulty parameters (-2, 0.5) and (0.5, 2). Therefore, the smaller RMSEs associated with estimating the discrimination parameters suggest that NUTS is clearly more accurate in estimating the discrimination parameters than the difficulty parameters for those scenarios.

Results based on the three-class situation suggest that the item difficulty (b_{jg}) or the class mean ability (μ_g) parameters are estimated more accurately for the second class than for the first or third class (see Sections 4.1.2 and 4.1.3). This is likely due to the choice of the simulated person ability and item difficulty parameters for each of the three latent classes. Specifically, the generated person abilities for the second class (i.e., $\theta_2 \sim N(0, 1)$) coincides with the generated item difficulty (i.e., $b_2 \sim U(-0.5, 0.5)$) for that class, such that the mean of person ability matches the mean of the item difficulty, which equals zero (see Figure 9 and the middle plot of Figure 10). However, the generated person abilities for the first class (i.e., $\theta_1 \sim N(-4, 1)$) is quite distant from the generated item difficulty (i.e., $b_2 \sim U(-2, -0.5)$) for that class, such that the average person ability (i.e., -4) is 2.75 standard deviations lower than the average item difficulty (i.e., -1.25) (see Figure 9 and the left plot of Figure 10). Similarly, the generated person ability for the third class (i.e., $\theta_3 \sim N(4, 1)$) is also quite distant from the generated item difficulty (i.e., $\mathrm{b}_2 \sim U(0.5, 2)$) for that class, such that the average person ability (i.e., 4) is 2.75 standard deviations higher than the average item difficulty (i.e., 1.5) (see Figure 9 and the right plot of Figure 10). In IRT models including the Mix2PL model, persons and items are placed on the same scale such that persons are scaled relative to items and vice-versa. In addition, the item information increases as the person ability and the item difficulty approach each other with the maximum information achieved at $\theta = b$ for the conventional one- and two-parameter IRT models (see Chapter 2 for more details on IRT models). This in turn leads to accurate estimation of both person ability and item difficulty parameters. Based on this, the estimation of the person mean ability and item difficulty are less accurate for the first and the third classes compared to the second class because of the lack of sufficient information considering the difficulty level of the items that persons from these classes are able to correctly answer based on their ability levels. Thus, in order to obtain more accurate estimates of the person mean ability and item difficulty parameters, more easy items should be added for the first class, whereas more difficult items should be added for the third class. This finding is consistent with Meyer (2010) in which RMSEs and bias were found to increase as the difference between item difficulty and mean ability for the "rapid-guessing" latent class increased in the condition where N =500.

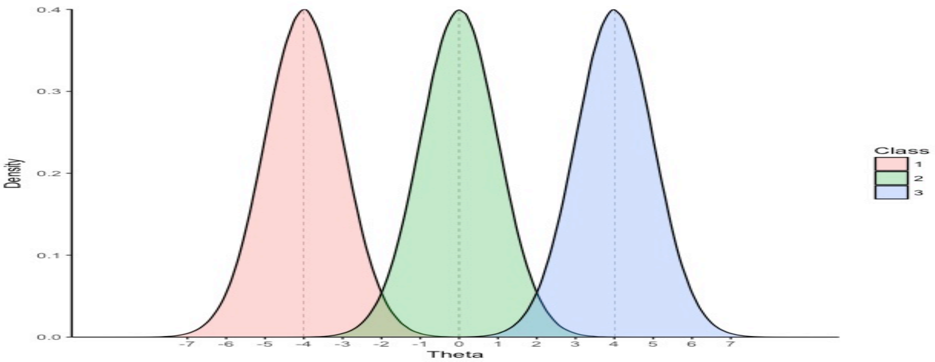

Figure 9. A probability density function of the ability parameter for the three-class population.

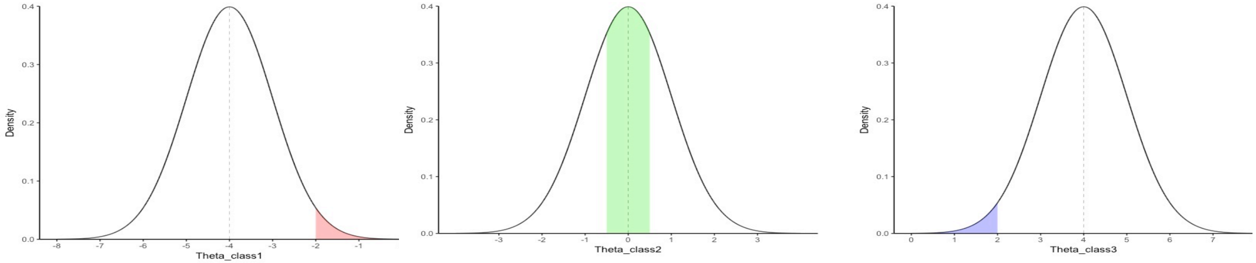

Figure 10. Probability density functions of the ability parameter for the first class (left), the second class (middle), and the third class (right) subpopulations.

5.1.2 Model Comparison

The second simulation study focuses on the performance of NUTS in terms of the accuracy of determining the number of latent classes of the Mix2PL model while comparing it to the conventional 2PL IRT model using fully Bayesian fit indices, namely the widely applicable information criterion (WAIC; Watanabe, 2010) and the leave-one-out cross-validation (LOO-PSIS; Vehtari, et al., 2017).

Nonconvergence issues associated with fitting MixIRT models to data that do not involve multiple subpopulations, as presented in Section 4.2, suggest that when data do not conform to MixIRT models, a substantially large number of iterations is required for the Markov chain to converge to the target posterior distribution, which is computationally expensive and sometimes impractical. For example, in the condition where the data were generated from the conventional 2PL IRT model, neither of the two Mix2PL models reached convergence with 60,000 of warm-up iterations followed by 80,000 of sampling iterations, even though it might be possible to reach convergence by adding substantially more iterations. Hence, researchers should use caution in real test situations where it is not clear about the structure of the latent groups of a certain population. Specifically, a large impractical number of iterations needed to reach convergence, especially when using efficient algorithms such as NUTS, may raise concerns regarding model-data conformity.

Regarding the accuracy in determining the number of latent classes, for the condition where data conformed to the two-class Mix2PL model, the results indicate that WAIC performs better than LOO in recovering the number of latent classes, in terms of the proportion of the time the correct model was selected as the best fitting model. It is noted in the results, although LOO favored the Mix2PL model with three classes, the three-class solution did not differ much from the two-class solution as presented in Table 14 of Chapter 4. In addition, when the effective number of parameters was also considered in selecting the best fitting model, as recommended by Gelman et al. (2014), the two fit indices perform equally well in determining the correct number of latent lasses. It is noteworthy that when both LOO and WAIC selected the three-class Mix2PL model as the best fitting model, in four replications among the 25 replications, the average proportion of persons (i.e., mixing proportion) for one of the three classes was 0.08 with a minimum of 0.04 and a maximum of 0.13. On the other hand, when only LOO selected the three-class Mix2PL model as the best fitting model, in six replications among the 25 replications, the average proportion of persons in one of the three classes was 0.17 with a minimum of 0.11 and a maximum of 0.20. This indicates that even when the three-class Mix2PL model (i.e., an incorrect model) was selected instead of the true two-class Mix2PL, especially by both fit indices, the proportion of persons in one of the classes was relatively low, which in turns suggest that the selected three-class Mix2PL model did not differ much from the true two-class Mix2PL. Different from the results of this study, Luo and Al-Harbi (2017) found that WAIC had slightly lower detection rate than LOO (although the difference is negligible) in the condition where the generating model was the conventional 1PL IRT model. Regarding the comparison of the Mix2PL model with the conventional 2PL IRT model, the simulation results suggest that when multiple latent classes exist, using either fully Bayesian fit indices (i.e., WAIC or LOO) would not select the conventional IRT model. On the other hand, when all persons came from a single unified population, fitting MixIRT models using NUTS causes problems in convergence.

5.2 Limitations and Directions for Future Studies

Through simulation studies, this dissertation provides empirical evidence on the performance of NUTS in fitting MixIRT models. It also shows that researchers and practitioners in educational and psychological measurement would benefit from using NUTS in estimating parameters of complex IRT models such as MixIRT models. The results of the present study suggest that NUTS generally performs well in recovering model parameters across all of the simulated conditions and hence offers advantages over conventional IRT models in fitting complex data sets that come from multiple subpopulations. However, conclusions that are made in the present study are based on the simulated conditions and cannot be generalized to other conditions. For example, the present study only considered two conditions of latent classes (i.e., 2-class and 3-class) with equal mixing proportions: π = (0.50 and 0.50) for the two-class condition and π = (0.33, 0.33, and 0.33) for the three-class condition, two test lengths (20, 30), and two sample sizes (250 and 500 for each class resulting in a total of 500 and 1000 for the two-class condition; and 750 and 1500 for the three-class condition). Therefore, for future studies, additional test conditions need to be explored such as unequal mixing proportions (i.e., 0.25 and 0.75), small sample size (i.e., 100, 200, and 300), as well as short test length (i.e., 10 and 15).

Furthermore, the two simulation studies were carried out using a Linux (CentOS-7) based computing cluster, which consists of 40 server nodes with at least 64 GB of memory each, 10-core chips, and 800 CPU cores in total. The running time to fit the Mix2PL models increased dramatically from an average of 22 minutes per replication for the simplest two-class condition where the sample size was 500 persons and the test length was 20 items to an average of 35 hours per replication for the most complicated three-class condition where the sample size was 1500 persons and the test length was 30 items. Given the computational expense of fitting NUTS to the complex Mix2PL model, this study only used 10 replications for parameter recovery and 25 replications for model comparison. However, as suggested by Harwell et al. (1996), a minimum of 25 replications is recommended for typical Monte Carlo studies in IRT modeling. Additional studies with similar experimental conditions are needed before one can conclude about the use of the algorithm with fitting the Mix2PL model and further the effects of sample size, test length, and number of classes on estimating the model.

In addition, this study focused on the dichotomous Mix2PL model. Future studies may consider evaluating the performance of NUTS using other dichotomous MixIRT models such as the Mix1PL or the Mix3PL models, or MixIRT models for polytomous responses such as a mixture version of Bock’s (1972) nominal response (mNR) model or a mixture version of Masters’s (1982) partial credit (mPC) model. Furthermore, findings from this study are based on simulated conditions where the true parameters are known. Future studies may adopt NUTS algorithms to fit the Mix2PL models to real data and examine how NUTS performs in real test situations. Also, findings from this study are limited to the choice of priors and hyperpriors for model parameters (i.e., $a \sim N_{(0, \infty)}(0, 1)$, $b \sim N(0, 1)$, $\theta \sim N(\mu_g, 1)$, and $\mu_g \sim N(0,1)$. Additional simulation studies are needed to consider other specifications of priors or hyperpriors for model parameters or hyperparameters.

Moreover, this study considered certain population distributions and difficulty ranges. Based on the results related to the accuracy of estimating the item difficulty and the class mean ability parameters, for the three-class scenario where the second class was estimated more accurately than the first or third class, this class focused on persons of medium ability such that their ability levels were drawn from a standard normal distribution and were administered a set of items that were sampled from a uniform distribution with the range (-0.5, 0.5), which can be a limitation. Additional studies are necessary to consider other person distributions and/or other ranges for item difficulty parameters to decide on the test condition that leads to more accurate estimates for all classes.

One of the concerns with estimating the Mix2PL model is that no constraint has been imposed on the item discrimination parameter, similar to what has been done with the item difficulty or the mean ability parameter to avoid the problem of label switching and hence identify the model. To further investigate it and to ensure such a constraint is necessary for the model considered in this dissertation, two simple Monte Carlo simulations were carried out: (1) The first simulation examines whether the ordered constraint for *b* has an effect on the accuracy of estimating the item difficulty parameter. For the two-class condition with a sample size of 500 persons and a test length of 20 items, the results indicate that removing the ordered constraint has a destructive effect on the estimation accuracy of the item difficulty parameter. Specifically, the average RMSEs based on ten replications for recovering this parameter in the first and the second latent classes are 1.132 and 1.078, respectively instead of 0.626 and 0.669 for the same condition but with the ordered constraint imposed (see Table 6). Clearly, the ordered constraint for the item difficulty parameter as adopted in this study is necessary to ensure the accuracy in estimating the item difficulty parameter in the Mix2PL model. (2) The second simulation was carried out to examine whether a positive-ordered constrained for *a* has an effect on recovering the item discrimination parameter. For the two-class condition with a sample size of 500 persons and a test length of 20 items, the average RMSEs based on ten replications for recovering the discrimination parameter in the first and the second latent classes are 0.363 and 0.361, respectively, instead of 0.397 and 0.400 for the same condition but without the positive-ordered constraint (see Table 6). This suggests that the positive-ordered constraint helps improve the precision of estimating the item discrimination. Although the effect may be trivial, future studies can consider imposing such a constraint on the item discrimination parameter in the Mix2PL model to help improve the precision in estimating it. Certainly, these two simulations are fairly simple as they only considered a specific test condition and only evaluated the recovery of the respective item parameter. More thorough investigations are needed in further studies to evaluate their effects, especially that of the positive constraint on *a*, on the accuracy of estimating MixIRT models in various test situations.

Since the results of this study suggest that the accuracy of recovering class membership decreases with an increase in the number of latent classes and that the recovery improves with the increase of either sample size or test length, future studies are needed to decide on the optimal number of persons and/or items for more accurate estimations of class membership in conditions where the population includes three or more subpopulations, for any given class size.

In addition, the recovery of class membership via proportions of correct classifications appears to be worse with this study than that from previous research with a Gibbs sampling approach (e.g., Li et al., 2009). Specifically, Li et al. (2009) found that the average proportions of correct classification, over sample sizes and test lengths, of class membership were 98.5 and 97.6 for the two- and three-class Mix2PL models, respectively whereas the corresponding proportions found in this study were 92.8 and 77.9. One possible reason is due to the inherent differences between the two MCMC algorithms in estimating a discrete parameter where such parameter (e.g., class membership) is not directly estimated via Stan program. Another possible reason can be the difference in the design of the two studies. Specifically, in Li et al. (2009), the simulated person abilities for all latent classes were from a standard normal distribution, the discrimination parameters were fixed to either 1 or 2, and the difficulty parameters were fixed within the range (-2.0, +2.0) with a 0.25 increment. In addition, the sample size (600 and 1200) and test length (6, 15, 30) conditions considered by Li et al (2009) are different from this study. It is hence not possible to directly compare the current study with the previous one. Consequently, future studies shall be directed to compare NUTS with Gibbs sampling in estimating class membership under different test conditions.

This study identified the Mix2PL model through imposing a zero-constraint on the difficulty parameter where the item difficulty values within each class sum to zero through soft centering (i.e., $b_g \sim N(0, 1)$; Stan Development Team, 2017; see Section 3.2 for more details). In the MixIRT literature, the usual approach to identify MixIRT models is to impose a constraint on the difficulty parameter such that the sum of item difficulties within each class equals to zero (i.e., $\sum_j b_j = 0$) in addition to the equal ability mean constraint for all classes (i.e., $\theta_g \sim N(0, \sigma^2)$). Some researchers (e.g., Wu & Paek, 2018), however, argue that both constraints might not be adequate to place parameters of the latent classes on a common scale, and hence, they suggest adding an anchor item constraint (i.e., invariant items across latent classes). Nevertheless, Wu and Paek (2018) found that the conventional constraint of the equal mean ability approach and the anchor item constraint approach showed high agreement in recovering the class membership. Thus, future research shall be directed to further investigate the role of different model identification methods on estimating MixIRT models.

In terms of model comparisons, only fully Bayesian fit indices, namely LOO and WAIC were used in this study. Future studies might consider comparing the performance of these full Bayesian fit indices with other partially Bayesian fit indices such as the deviance information criterion (DIC; Spiegelhalter, Best, Carlin, & van der Linde, 2002).

As discussed in Chapter 2, WAIC is an asymptotic approximation of LOO, which is computed through Pareto smoothed important sampling (PSIS-LOO; Vehtari, et al., 2017) approximation that is implemented in the R package “loo”. Although the two fit indices differ in performance in terms of the proportion of the time the correct model was selected as the best fitting model, it can be argued that the difference, especially in LOO values, between the two Mix2PL models are rather small. Moreover, when the effective number of parameters was taken into consideration in the selection process, LOO and WAIC perform equally well in determining the number of latent classes. Therefore, before making any conclusion regarding the performance of LOO and WAIC, future research shall be directed to investigate the performance of these fully Bayesian fit indices in selecting the true model using different MixIRT models such as the mixture one-parameter (Mix1PL) model and the mixture three-parameter (Mix3PL) model in addition to the Mix2PL model or generating data that have more than two classes.

Finally, the results of this study suggest that NUTS encounters problems in convergence when fitting MixIRT models to data from a single unified population. This result can raise a flag for researchers and practitioners concerning the latent structure of the population under investigation. This is also a potential advantage of NUTS if the same finding can be replicated. Certainly, additional studies are needed to further investigate this result and examine whether convergence issues also emerge using other MCMC algorithms such as Gibbs sampling or other MixIRT models such as the Mix1PL or Mix3PL models.

REFERENCES

Albert, J. H. (1992). Bayesian estimation of normal ogive item response curves using Gibbs sampling. *Journal of Educational Statistics, 17*, 251-269.

Andersen, E. B. (1970). Asymptotic properties of conditional maximum likelihood estimation. *Journal of the Royal Statistical Society. Series B, 3*2(2), 283-301.

Aryadoust, V. (2015). Fitting a mixture Rasch model to English as a foreign language listening tests: The role of cognitive and background variables in explaining latent differential item functioning. *International Journal Of Testing, 15*(3), 216-238. doi: 10.1080/15305058.2015.1004409

Baker, F. B., & Kim, S. H. (2004). *Item response theory: Parameter estimation techniques* (2nd ed.). New York, NY: Marcel Dekker.

Betancourt, M. (2017). A conceptual introduction to Hamiltonian Monte Carlo. Retrieved from arXiv:1701.02434

Bilir, M. K. (2009). *Mixture item response theory-MIMIC model: Simultaneous estimation of differential item functioning for manifest groups and latent classes* (Doctoral Dissertation). Retrieved from ProQuest Dissertations & Theses A&I. (Order No. 3399179)

Birnbaum, A. (1969). Statistical theory for logistic mental test models with a prior distribution of ability. *Journal of Mathematical Psychology, 6*(2), 258-276.

Bock, R. D. (1972). Estimating item parameters and latent ability when responses are scored in two or more nominal categories. *Psychometrika, 37(1)*, 29-51.

Bock, R. D. (1997). A brief history of item response theory. *Educational Measurement: Issues and Practice, 16*, 21-33.

Bock, R. D., & Aitkin, M. (1981). Marginal maximum likelihood estimation of item parameters: An application of an EM Algorithm. *Psychometrika, 46*(4), 443-459.

Bock, R. D., & Lieberman M. (1970). Fitting a response curve model for n dichotomously scored items. *Psychometrika, 35(2)*, 179-197. doi: 10.1007/BF02291262

Bolt, D. M., Cohen, A. S., & Wollack, J .A. (2001). A mixture item response model for multiple-choice data. *Journal of Educational and Behavioral Statistics, 26*(4), 381–409.

Bolt, D. M., Cohen, A. S., & Wollack, J .A. (2002). Item parameter estimation under conditions of test speededness: Application of a mixture Rasch model with ordinal constraints. *Journal Of Educational Measurement, 39*(4), 331-348.

Bolt, D. M., & Lall, V. F. (2003). Estimation of compensatory and noncompensatory multidimensional item response models using Markov chain Monte Carlo. *Applied Psychological Measurement, 27*(6), 395-414. doi: 10.1177/0146621603258350

Brooks, S., Gelman, A., Jones, G. L., & Meng, X. L. (2011). *Handbook of Markov chain Monte Carlo*. Boca Raton, FL: CRC Press.

Cai, L. (2010a). High-dimensional exploratory item factor analysis by a Metropolis-Hastings Robbins-Monro algorithm. *Psychometrika, 75*(1), 33-57.

Cai, L. (2010b). Metropolis-Hastings Robbins-Monro algorithm for confirmatory item factor analysis. *Journal of Educational and Behavioral Statistics, 35*(3), 307-335.

Caughey, D., & Warshaw, C. (2014). *Dynamic Representation in the American States, 1960-2012*. Paper presented at American Political Science Association 2014 Annual Meeting. Rettrieved from http://dx.doi.org/10.2139/ssrn.2455441

Chang, M. (2017). *A comparison of two MCMC algorithms for estimating the 2PL IRT models* (Doctoral Dissertation). Retrieved from ProQuest Dissertations & Theses A&I. (Order No. 10601766)

Chib, S., & Greenberg, E. (1995). Understanding the Metropolis-Hastings algorithm. *The American Statistician, 49*(4), 327-335.

Cho, S., & Cohen, A. S. (2010). A multilevel mixture IRT model with an application to DIF. *Journal Of Educational and Behavioral Statistics*, *35*(3), 336-370.

Cho, S., Cohen, A., & Kim, S. (2013). Markov chain Monte Carlo estimation of a mixture item response theory model. *Journal of Statistical Computation and Simulation, 83*(2), 278-306.

Choi, Y., Alexeev, N., & Cohen, A. S. (2015). Differential item functioning analysis using a mixture 3-parameter logistic model with a covariate on the TIMSS 2007 mathematics test. *International Journal Of Testing, 15*(3), 239-253. doi: 10.1080/15305058.2015.1007241

Cohen, A. S., & Bolt, D. M. (2005). A mixture model analysis of differential item functioning. *Journal of Educational Measurement, 42(*2), 133-14.

Cohen, A. S., Gregg, N., & Deng, M. (2005). The role of extended time and item content on a high-stakes mathematics test. *Learning Disabilities Research and Practice, 20*(4), 225–233.

Copelovitch, M., Gandrud, C., & Hallerberg, M. (2015). Financial regulatory transparency, international institutions, and borrowing costs. Retrieved from http://dx.doi.org/10.2139/ssrn.2701852.

De Ayala, R. J. (2009). *The theory and practice of item response theory*. New York, NY: Guilford Press.

De Ayala, R. J., Kim, S. H., Stapleton, L. M., & Dayton, C. M. (2002). Differential item functioning: A mixture distribution conceptualization. *International Journal of Testing, 2*(3&4). 243-276.

de la Torre, J., & Douglas, J.A. (2004). Higher-order latent trait models for cognitive diagnosis. *Psychometrika 69*(3), 333-353.

de la Torre, J., Stark, S., & Chernyshenko, O. S. (2006). Markov chain Monte Carlo estimation of item parameters for the generalized graded unfolding model. *Applied Psychological Measurement, 30*(3), 216-232. doi: 10.1177/0146621605282772

Duane, S., Kennedy, A., Pendleton, B. J., & Roweth, D. (1987). Hybrid Monte Carlo. *Physics Letters B, 195*, 216-222. doi: 10.1016/0370-2693(87)91197-X

Embreston, S. E., & Reise, S. P. (2000). *Item response theory for psychologist.* Mahwah, NJ: Lawrence Erlbaum Associates.

Fan, X. (1998). Item response theory and classical test theory: An empirical comparison of their item/person statistics. *Educational and Psychological Measurement, 58*(3), 357-381. doi: 10.1177/0013164498058003001

Finch, W. H., & Finch, M. E. (2013). Investigation of specific learning disability and testing accommodations based differential item functioning using a multilevel multidimensional mixture item response theory model. *Educational and Psychological Measurement, 73*(6), 973-993. doi: 10.1177/0013164413494776

Finch, W. H., & French, B. F. (2012). Parameter estimation with mixture item response theory models: A Monte Carlo comparison of maximum likelihood and Bayesian methods. *Journal of Modern Applied Statistical Methods, 11*(1), 167-178.

Fisher (1922). On the mathematical foundation of theoretical Statistics. *Philosophical Transactions of the Royal Society of London, 222,* 309-368.

Fox, J. (2010). *Bayesian item response modeling: Theory and applications*. New York, NY: Springer.

Gelfand, A. E. and Smith, A. F. M. 1990. Sampling-based approaches to calculating marginal densities. *Journal of the American Statistical Association, 85,* 398–409.

Gelman, A., & Rubin, D. B. (1992). Inference from iterative simulation using multiple sequences. *Statistical Science, 7*(4), 457-472.

Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., & Rubin, D. B. (2014). *Bayesian data analysis* (3rd ed.). Boca Raton, FL: CRC Press.

Geman, S., & Geman, D. (1984). Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 6*(6), 721-741. doi: 10.1109/TPAMI.1984.4767596

Geyer, C. J. (2011). Introduction to Markov chain Monte Carlo. In S. Brooks, A. Gelman, G. L. Jones, & X. Meng (Eds.), Handbook of markov chain monte carlo (pp. 3–48). CRC Press.

Grant, R. L., Furr, D. C., Carpenter, B., & Gelman, A. (2016). Fitting Bayesian item response models in Stata and Stan. Retrieved from arXiv:1601.03443

Griewank, A., & Walther, A. (2008). *Evaluating derivatives: Principles and techniques of algorithmic differentiation* (2nd ed.). Philadelphia, PA : Society for Industrial and Applied Mathematics.

Gulliksen, H. (1987). *Theory of mental test*s. Hillsdale, N.J. : L. Erlbaum Associates.

Hambleton, R. K., & Cook, L. L. (1977). Latent trait models and their use in the analysis of educational test data 1, 2, 3. *Journal of Educational Measurement, 14*(2), 75-96. doi: 10.1111/j.1745-3984.1977.tb00030.x

Hambleton, R. K., & Jones, R. W. (1993). Comparison of classical test theory and item response theory and their applications to test development. *Educational Measurement: Issues and Practice, 12*(3), 38-47. doi: 10.1111/j.1745-3992.1993.tb00543.x

Hambleton, R. K., Swaminathan, H, & Rogers, H. J. (1991). *Fundamentals of item response theory*. Newbury Park, CA: Sage.

Harwell, M., Stone, C. A., Hsu, T. C., & Kirisci, L. (1996). Monte Carlo studies in item response theory. *Applied Psychological Measurement, 20*(2), 101-125. doi: 10.1177/014662169602000201

Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications. *Biometrika, 57*(1), 97-109. doi: 10.1093/biomet/57.1.97

Hoffman, M. D., & Gelman, A. (2011). The no-U-turn sampler: Adaptively setting path lengths in Hamiltonian Monte Carlo. *Journal of machine learning research, 15*(2), 1593-1624.

Holland, P. W., & Wainer, H. (1993). *Differential item functioning*. New York, NY: Lawrence Erlbaum Associates.

Huang, H. (2016). Mixture random-effect IRT models for controlling extreme response style on rating scales. *Frontiers In Psychology*, *7*. doi: 10.3389/fpsyg.2016.01706

Hulin, C. L., Lissak, R. I., & Drasgow, F. (1982). Recovery of two and three-parameter logistic item characteristic curves: A Monte Carlo study. *Applied Psychological Measurement, 6*(3), 249–260.

Johnson, M. S. (2007). Marginal maximum likelihood estimation of item response models in R. *Journal of Statistical Software, 20*(10), 1-24.

Johnson, M. S., & Junker, B. W. (2003). Using data augmentation and Markov chain Monte Carlo for the estimation of unfolding item response models. *Journal of Educational and Behavioral Statistics, 28*(3), 195-230. doi: 10.3102/10769986028003195

Johnson, M. S., & Sinharay, S. (2005). Calibration of polytomous item families using Bayesian hierarchical modeling. *Applied Psychological Measurement, 29*(5), 369-400. doi: 10.1177/0146621605276675

Kang, T. & Cohen, A. S. (2007). IRT model selection methods for dichotomous itemss. *Applied Psychological Measurement, 31*, 331–358.

Kim, S-H. (2001). An evaluation of a Markov chain Monte Carlo method for the Rasch model. *Applied Psychological Measurement, 25*(2), 163-176.

Kim, S-H. (2007). Some posterior standard deviations in item response theory. *Educational and Psychological Measurement, 67*(2), 258-279. doi: 10.1177/00131644070670020501

Kruschke, J. K. (2011). *Doing Bayesian data analysis: A tutorial with R and BUGS*. Burlington, MA: Academic Press.

Kuo, T. (2015). *Bayesian estimation of multi-unidimensional graded response IRT models* (Doctoral Dissertation). Retreived from Dissertations & Theses @ Southern Illinois University at Carbondale. (Order No. 10012795)

Lamsal, S. (2015). *Comparing three estimation methods for the three-parameter logistic IRT model* (Doctoral Dissertation). Retrieved from Dissertations & Theses @ Southern Illinois University at Carbondale. (Order No. 10012825)

Lau, A. (2009). *Using a mixture IRT model to improve parameter estimates when some examinees are amotivated* (Doctoral Dissertation). Retrieved from ProQuest Dissertations & Theses A&I. (Order No. 3366561)

Levy, R. (2009). The rise of Markov chain Monte Carlo estimation for psychometric modeling. *Journal Of Probability and Statistics, 2009*. doi: 10.1155/2009/537139

Li, F., Cohen, A., Kim, S., & Cho, S. (2009). Model selection methods for mixture dichotomous IRT models. *Applied Psychological Measurement, 33*(5), 353-373. doi: 10.1177/0146621608326422

Lim, R. G., & Drasgow, F. (1990). Evaluation of two methods for stimating item response theory parameters when assessing differential item functioning. *Journal Of Applied Psychology, 75*(2), 164-174.

Linden, W. J. van der, & Glas, C. A. W. (2000). *Computerized adaptive testing: Theory and practice*. Norwell, MA: Kluwer Academic.

Linden, W. J. van der, & Hambleton, R. K. (1997). *Handbook of modern item response theory*. New York, NY: Springer.

Lord, F. M. (1980). *Applications of item response theory to practical testing problems* (2nd ed.). New Jersey, NJ: Hillsdale.

Lord, F. M. & Novick, M. R. (1968). *Statistical theories of mental test scores*. Maryland, MA: Addison-Wesley.

Lunn, D. J., Thomas, A., Best, N., & Spiegelhalter, D. (2000). WinBUGS - A Bayesian modeling framework: Concepts, structure, and extensibility. *Statistics and Computing, 10*(4), 325-337. doi: 10.1023/A:1008929526011

Luo, Y., & Al-Harbi, K. (2017). Performances of LOO and WAIC as IRT model selection methods. *Psychological Test and Assessment Modeling, 59*(2), 183-205.

Maij-de Meij, A.M., Kelderman, H., & van der Flier, H. (2008). Fitting a mixture item response theory model to personality questionnaire data: Characterizing latent classes and investigating possibilities for improving prediction. *Applied Psychological Measurment, 32*(8), 611-631. doi: 10.1177/0146621607312613

Maij-de Meij, A.M., Kelderman, H., & van der Flier, H. (2010). Improvement in detection of differential item functioning using a mixture item response theory model. *Multivariate Behavioral Research, 45*(6), 975-999. doi: 10.1080/00273171.2010.533047

Maraun, M. D. (1993). *Issues pertaining to the determinacy of item response models* (Doctoral Dissertation). Retrieved from ProQuest Dissertations & Theses A&I. (Order No. NN82692)

Martin-Fernandez, M., & Revuelta, J. (2017). Bayesian estimation of multidimensional item response models. A comparison of analytic and simulation algorithms. *Psicológica, 38*(1), 25-55.

Masters, G. N. (1982). A Rasch model for partial credit scoring. *Psychometrika, 47*, 149-174.

MathWorks, Inc. (1992). *MATLAB, version 4*. Natick, MA: Math Works Inc.

McLachlan, G. J., & Peel, D. (2000). *Finite mixture models*. New York, NY : Wiley.

Metropolis, N., & Ulam, S. (1949). The Monte Carlo method. *Journal of the American Statistical Association, 44*(247), 335-341.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., & Teller, E. (1953). Equation of state calculations by fast computing machines. *The Journal of Chemical Physics, 21*(6), 1087-1092.

Meyer, J. P. (2010). A Mixture Rasch model with Item response time components. *Applied Psychological Measurement, 34*(7), 521–538. doi: 10.1177/0146621609355451

Mislevy, R. J. (1986). Bayes modal estimation in item response models. *Psychometrika, 51*(2), 177-195. doi: 10.1007/BF02293979

Mislevy, R. J., & Verhelst, N. (1990). Modeling item responses when different subjects employ different solution strategies. *Psychometrika, 55*(2), 195-215.

Mroch, A. A., Bolt, D. M., & Wollack, J .A. (2005, April). *A new multi-class mixture Rasch model for test speededness*. Paper presented at the Annual Meeting of the National Council on Measurement in Education, Montreal, Quebe.

Muthen, B., & Asparouhov, T. (2006). Item response mixture modeling: Application to tobacco dependence criteria. *Addictive Behaviors, 31*(6), 1050-1066. doi: 10.1016/j.addbeh.2006.03.026

Muthén, L. K. & Muthén, B. O. (2011). *MPlus software, version 6.1*. Los Angeles, CA: MPlus.

Neal, R. M. (1992). An improved acceptance procedure for the hybrid Monte Carlo algorithm. Retrieved from arXiv preprint hep-lat/9208011.

Neal, R. M. (1996). *Bayesian learning for neural networks*. New York, NY: Springer.

Neal, R. M. (2003). Slice sampling. *The Annals of Statistics, 31*(3), 705–741.

Neal, R. M. (2011). MCMC using Hamiltonian dynamics. In S. Brooks, A. Gelman, G. Jones, & X. Meng (Eds.), *Handbook of Markov chain Monte Carlo* (pp. 113-162). Boca Raton, FL: CRC Press.

Novick, M. R. (1966). The axioms and principal results of classical test theory. *Journal of Mathematical Psychology, 3*(1), 1–18.

Novick, M. R. (1966). The axioms and principal results of classical test theory. *Journal of Mathematical Psychology, 3*(1), 1–18.

Oshima, T. C. (1994). The effect of speededness on parameter estimation in item response theory. *Journal Of Educational Measurement, 31*(3), 200-219.

Patz, R. J., & Junker, B. W. (1999a). A straightforward approach to Markov chain Monte Carlo methods for item response models. *Journal of Educational and Behavioral Statistics*, *24*(2), 146-178. doi: 10.3102/10769986024002146

Patz, R. J., & Junker, B. W. (1999b). Applications and extensions of MCMC in IRT: Multiple item types, missing data, and rated responses. *Journal of Educational and Behavioral Statistics*, *24*(4), 342-366. doi: 10.3102/10769986024004342

Plummer, M. (2003). JAGS: A program for analysis of Bayesian graphical models using Gibbs sampling. In *Proceedings of the 3rd international workshop on distributed statistical computing* (Vol. 124, p. 125).

Preinerstorfer, D., & Formann, A. K. (2012). Parameter recovery and model selection in mixed Rasch models. *British Journal of Mathematical and Statistical Psychology, 65*(2), 251-262. doi: 10.1111/j.2044-8317.2011.02020.x

Rasch, G. (1960). *Probabilistic models for some intelligence and attainment tests* (2nd ed.). Danmark: Danmarks Paedagogiske Institute.

Ravenzwaaij, D., Cassey, P., & . Brown. S. D. (2016). A simple introduction to Markov Chain Monte–Carlo sampling. *Psychonomic Bulletin & Review.* doi: 10.3758/s13423-016-1015-8

Rost, J. (1990). Rasch models in latent classes: An integration of two approaches to item analysis. *Applied Psychological Measurement, 14*(3), 271-282. doi: 10.1177/014662169001400305

Rost, J. (1997). Logistic mixture models. In W. J. van der Linden, & R. K. Hambleton (Eds.), *Handbook of modern item response theory* (pp. 449-463). New York, NY: Springer.

Sahin, A., & Anil, D. (2017). The effects of test length and sample size on item parameters in item response theory. *Educational Sciences: Theory and Practice, 17*, 321-335.

Samuelsen, K. (2005). *Examining differential item functioning from a latent class perspective* (Doctoral Dissertation). Retrieved from ProQuest Dissertations & Theses A&I. (Order No. 3175148)

Shea, C. A. (2013). *Using a mixture IRT model to understand English learner performance on large-scale assessments* (Doctoral Dissertation). Retrieved from ProQuest Dissertations & Theses A&I. (Order No. 3603151)

Sheng, Y. (2010). A sensitivity analysis of Gibbs sampling for 3PNO IRT models: Effects of prior specifications on parameter estimates. *Behaviormetrika, 37*(2), 87-110. doi: 10.2333/bhmk.37.87

Sheng, Y., & Wikle, C. K. (2007). Comparing multiunidimensional and unidimensional item response theory models. *Educational and Psychological Measurement, 67*(6), 899-919. doi: 10.1177/0013164406296977

Si, C. F. & Schumacker, R. E. (2004). Ability estimation under different item parameterization and scoring models. *International Journal of Testing, 4*(2), 137-181.

Skaggs, G., & Lissitz, R. W. (1986). IRT test equating: Relevant issues and a review of recent research. *Review of Educational Research, 56*(4), 495-529.

Spiegelhalter, D. J., Best, N. G., Carlin, B. P., & Van Der Linde, A. (2002). Bayesian measures of model complexity and fit. *Journal of the Royal Statistical Society: Series B (Statistical Methodology), 64*(4), 583-639.

Stan Development Team (2017). *Stan Modeling Language Users Guide and Reference Manual*, Version 2.17.0. Retrieved from http://mc-stan.org

StataCorp (2016). *Stata statistical software: release 14.1*. College Station, TX.

Stephens, M. (2000). Dealing with label switching in mixture models. *Journal Of The Royal Statistical Society. Series B (Statistical Methodology), 62*(4), 795-809.

Stone, C. A. (1992). Recovery of marginal maximum likelihood estimates in the two-parameter logistic response model: An evaluation of Multilog. *Applied Psychological Measurement, 16*(1), 1–16.

Thissen, D., & Wainer, H. (2001). *Test scoring.* Mahwah, N.J. : L. Erlbaum Associates, 2001.

Thomas, A., O'Hara, B., Ligges, U., & Sturtz, S. (2006). Making BUGS open. *R news, 6*(1), 12-17.

Toribio, S. G. (2006). *Bayesian model checking strategies for dichotomous item response theory model*s (Doctoral dissertation). Retrieved from ProQuest Dissertations & Theses A&I. (Order No. 3216849)

Vehtari, A., Gelman, A., & Gabry, J. (2017). Practical Bayesian model evaluation using leave-one-out cross-validation and WAIC. *Statistics and Computing, 27*(5), 1413-1432. doi: 10.1007/s11222-016-9696-4

Vermunt, J. K. (1997). *LEM: A general program for the analysis of categorical data [Computer software]*. Tilburg, Netherlands: Tilburg University.

Vermunt, J.K., & Magidson, J (2005). *Latent GOLD 4.0 [computer software]*. Statistical Innovations Inc., Belmont, MA.

von Davier, M. (2001). *WINMIRA [computer software]*. Groningen, The Netherlands: ASC Assessment Systems Corporation, USA and Science Plus Group.

Watanabe, S. (2010). Asymptotic equivalence of Bayes cross validation and widely applicable information criterion in singular learning theory. *Journal of Machine Learning Research, 11*(2), 3571-3594.

Wollack, J. A., Bolt, D. M., Cohen, A. S., & Lee, Y. S. (2002). Recovery of item parameters in the nominal response model: A comparison of marginal maximum likelihood estimation and Markov chain Monte Carlo estimation. *Applied Psychological Measurement, 26*(3), 339-352. doi: 10.1177/0146621602026003007

Wollack, J. A., Cohen, A. S., & Wells, C. S. (2003). A method for maintaining scale stability in the presence of test speededness. *Journal of Educational Measurement, 40*, 307-330.

Wu, X., Sawatzky, R., Hopman, W., Mayo, N., Sajobi, T. T., Liu, J., & ... Lix, L. M. (2017). Latent variable mixture models to test for differential item functioning: A population-based analysis. *Health and Quality of Life Outcomes*, 15. doi: 10.1186/s12955-017-0674-0

Wu, Y-J., & Paek, I. (2018). Agreement on the Classification of Latent Class Membership Between Different Identification Constraint Approaches in the Mixture Rasch Model. *Methodology-European Journal of Research Methods for the Behavioral and Social Sciences, 14*(2), 82–93. doi: 10.1027/1614-2241/a000148

VITA

Graduate School
Southern Illinois University

Rahab Al Hakmani

rehab.hekmani@gmail.com

Sultan Qaboos University, Oman
Bachelor of Science Education, Physics, November 1999

Sultan Qaboos University, Oman
Master of Education, Measurement and Evaluation, November 2007

Special Honors and Awards:
Patricia Borgsmiller Elmore and Donald E. Elmore Doctoral Scholar Award (2017)
Dissertation Research Assistantship Award (Fall 2018)
Psychometric Society Travel Award (2018)

DISSERTATION TITLE:
Bayesian Estimation of Mixture IRT Models using NUTS

MAJOR PROFESSOR: Yanyan Sheng