UiO : University of Oslo

Kondwani Kajera Mughogho

Subscale Score Estimation Methods in International Large-Scale Assessment

What is the subscale estimation method of choice?

A dissertation presented for the degree of Philosophiae Doctor

Centre for Educational Measurement at the University of Oslo
Faculty of Educational Sciences

2020

Series of dissertations submitted to the
Faculty of Educational Sciences, University of Oslo
No. 325

ISSN 1501-9862

Cover: Hanne Baadsgaard Utigard.
Print production: Reprosentralen, University of Oslo.

To my late parents Christina Nomathemba Munjoma-Mughogho and Spider Kajera Mughogho.

Abstract

In spite of a body of research into subscale score reporting at the individual level, there exists a paucity of research into subscale score estimation in international large-scale assessment (ILSA). This doctoral thesis aimed at evaluating the typically available methods for subscale score estimation in order to identify a model that was suitable for (a) item parameter estimation; (b) population score estimation; (c) reporting valuable subscale scores. This dissertation further examined the models in order to identify the better fitting model. Through investigating the accuracy and bias in estimating the model parameters given different test conditions (i.e., numbers of subdomains, subdomain lengths, and subdomain correlations), the key motivation of this dissertation was to provide practitioners with general guidelines when it comes to estimating subscale scores under different test specifications.

This thesis was based on two simulation studies and an empirical study. Simulation studies 1 and 2 were designed to resemble the Southern and Eastern Africa Consortium for Monitoring Educational Quality (SACMEQ) and Trends in International Mathematics and Science Study's (TIMSS') data. The difference between the two simulation studies was that Study 1 did not employ matrix sampled test booklets and latent regression methods in score estimation whilst Study 2 did. Within each of the simulation studies, data analysis were conducted assuming the data comprised of single- and multiple-groups. The empirical investigations were based on data from TIMSS 2015's eighth grade mathematics test.

Taken together, the findings presented in this doctoral thesis advance the existing knowledge about subscale score estimation by extending the conversation to an ILSA context. As subscale scores have become increasingly relevant for guiding educational policy and practice, this study can inform test practitioners as to the selection of the most appropriate subscale score estimation method. This thesis argues that different subscale score estimation methods may be more optimal under different test conditions (i.e., test length or subscale correlation) and sample composition (i.e., single or multiple groups). In addition, this thesis argues that the choice of model may depend on the practitioner's primary concern (i.e., item- or score-parameter estimates; subscale value; or model fit). This study also contributes to informing the choice of

model when the sample of participants becomes more diverse with regards to performance.

Acknowledgements

I Am Because We Are.

This project and PhD degree would not have been possible without my team of supervisors, who are also great mentors and friends: Leslie and David Rutkowski. I do thank you for your support and reliance, and for all you have done and still do for me. You have set aside time to review countless drafts of my work while protecting me within academia and giving me advice. You involve, mentor, encourage, inspire, share my enthusiasm from ideas to results, and most of all you do care. Meeting you was a life-changer. I have grown tremendously, both as a scientist and personally. You are all sources of inspiration from now, and the entire future that awaits me.

On any journey, who you travel with, is much more imperative than the last stop. The work in this thesis was done at the greatly motivating work place of the Centre for Educational Measurement at the University of Oslo (CEMO), and funded by the Norwegian Research Council (FINNUT). A very big thank you to Professor Sigrid Blömeke, Rolf Vegar, Johan Braeken, Ronny Scherer, Andreas Frey, Stephan Schauber, Bryan Maddox, and Anders Skrondal. A special thanks to for all the help Anne-Catherine (from my coming to CEMO) and Tara (as I prepared to leave CEMO); you made settling down, and navigating the Norwegian system easy. Thank you to the wonderful, supportive, and caring team of CEMO post-doctoral researchers; I look up to you. Alexandra, thank you. A hearty thank you to my fellow PhD candidates and friends: Saskia, Melaku, Fredrik, Henrik, Haakon, Jelena, Wangqiong, Maoxin, Jarl, Diego, Jose, and Ksenia. I am so happy to have known you all and I hope that we stay friends well into the future.

I am grateful for Tyler Matta, Yuan Ling-Linda Liaw, and Waldir Leoncio Netto who offered support throughout the project and gave me insights into my work whilst encouraging me. A special thank you to Yuanye Wu, Dubravka Svetina, Montserrat Valdivia Medinaceli, and everyone who was involved with the Embracing Heterogeneity in International Surveys: Optimal Test Design and Parameter Estimation project. Such an amazing group of researchers and friends.

Special thanks goes to Jesper Tjimstra and Björn Anderson who took time to review my work. Your feedback came when I needed it the most as it enabled me to improve my thesis.

I am forever grateful for the lifetime opportunity to do a PhD in a field that is so passionate to me. Many other people contributed in immeasurable ways by giving me much needed moral and spiritual support. A big thank you is in order to my siblings: Suzgo, Thasya, Tapuwa, and Tamara; as well as my friends (in Malawi, Zimbabwe, Norway, USA, Kenya, and beyond). Your blend of love, encouragement, and moral support is very much appreciated. You were there for me in the darkest of times, and I hope you are there for me and I hope you will continue being a part of my life.

Halleli Alemi.

Kondwani Mughogo
Oslo, October 2020

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 International Large-Scale Assessments

Since their inception, information gathered from ILSAs has been used to support evidence-based policy discussions within and beyond the participating countries (Baird et al., 2016; Lindblad et al., 2015). Further, findings from ILSAs have also been used as a tool to compare and make changes to curricula and instructional and learning strategies in participating educational systems (Chmielewski & Dhuey, 2017; Torney-Purta & Amadeo, 2013). Examples of these studies include the Trends in International Mathematics and Science Study (TIMSS) and the Programme for International Student Assessment (PISA). In addition to larger, more international studies, there are also regional assessments around the world including studies in Africa, South America, and Asia. One example, which is a focus of this dissertation, is the Southern and Eastern Africa Consortium for Monitoring Educational Quality (SACMEQ) study, described in detail in Chapter 2.

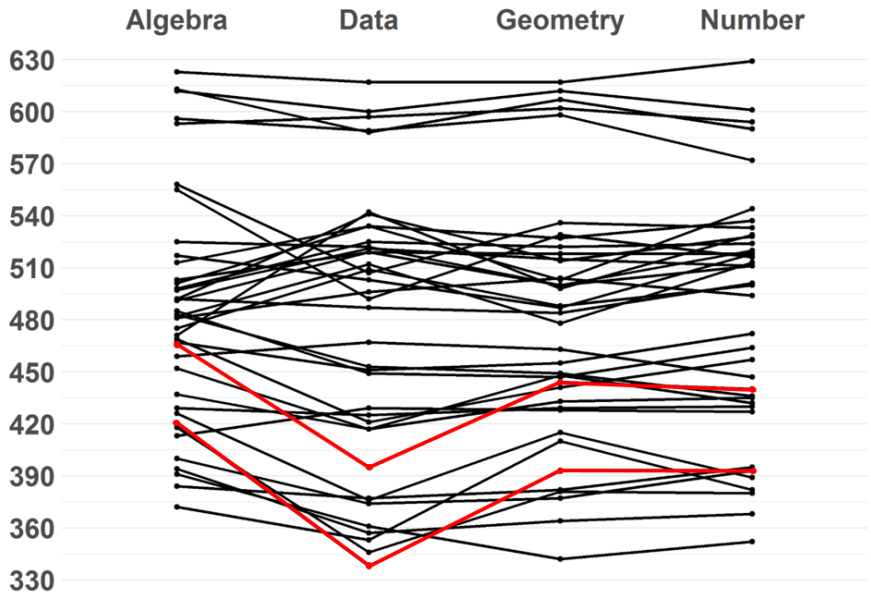
The spotlight on ILSAs has been placed on the achievement outcomes and rankings in broad content domains, which usually include math, science, and reading. However, the broad domains on these assessments, like many educational and psychological constructs, may be broken down into content areas, instructional objectives, or subscale categories (i.e., they are multidimensional; Liu, 2015). For example, the 8th grade TIMSS 2015 mathematics test framework was composed of four content subscales: Number, Algebra, Geometry, and Data and Chance; these subscores provide diagnostic information to differentiate aspects of mathematics achievement at the country level (Camilli & Dossey, 2019). Figure 1.1[1] is an illustration of the TIMSS 2015 eighth grade mathematics content subscale score profile for all participating countries. This figure highlights the variability in score performance across the four subscales. For example, Egypt scored 420, 338, 393, and 393 on the algebra, data and chance, geometry, and number subscales, respectively. Egypt's score difference between algebra and data and chance was close to one standard deviation[2]. The difference between the algebra and data and chance

[1] *Note.* The highlighted lines join the scores for Lebanon and Egypt, respectively.

[2] TIMSS arbitrarily sets an origin and unit size; i.e., mean of 500 and a standard deviation of 100, as was done originally for TIMSS in 1995 (Martin, Mullis, & Hooper, 2016).

subscales was also quite high in Lebanon, 466 and 395, respectively. Although subscale scores are reported for most ILSAs, they are not often reported for regional assessments such as SACMEQ.

Figure 1.1
TIMSS Empirical Mathematics Content Domain Scores



International assessments (a) often comprise of several hundred questions that are typically used to test the respective content domains, and (b) are administered to populations that differ substantially in achievement. As a result, ILSAs often use special test designs in data collection (these will be discussed in detail in Chapter 2). The intended level of inferences also means that population scores are calculated using special achievement estimation methods.

When scoring subscales, most ILSAs fit a unidimensional-IRT (UIRT) model to estimate item parameters. The resultant item parameters are subsequently used to calibrate population and subpopulation achievement distributions overall and for each subscale. TIMSS fits a multidimensional-IRT (MIRT) model which specifies two-dimensions, mathematics and science (Martin, Mullis, & Hooper, 2016). Nevertheless, all the items on these two dimensions are assumed to measure single (unidimensional) educational or

psychological constructs (i.e., mathematics which will, in part, be the focus of this thesis). However, these constructs are explicitly broken down to sub-constructs (i.e., mathematics content: algebra, data and display, geometry, number on TIMSS) in test specification. As such, other models may be fit that take into account the relationship between two or more latent constructs in the item calibration (amongst others, these include variants of multidimensional-IRT models; Reckase, 2009). MIRT models include consecutive-IRT (CIRT) and multidimensional-IRT (MIRT). The difference between CIRT and MIRT being that MIRT assumes the subdomains on a test are correlated whilst CIRT assumes that the subdomains are uncorrelated. Assuming a specific IRT model may have implications on subscale scoring in ILSA (see de la Torre & Song, 2009; Wang, 2017; Yao & Boughton, 2007). The choice between CIRT and MIRT may be driven by: (a) model simplicity; (b) time of score computation over large sample sizes; (c) computer power; and (d) to some extent, for small testing agencies, technical expertise. Although, MIRT models most closely adhere to the assumed actual factor structure of some educational constructs, these models are more complex and computationally cumbersome. Alternatively, UIRT and CIRT are easier to implement, though doing so has some consequences with respect to reported subscale scores. For instance, a study by Yao (2010) showed that MIRT povides less biased item parameter estimates when correlations between subdomains are low.

To complement the aforementioned IRT methods, subscale scores may be evaluated to examine whether they provide added value over the total test score. In other words, it is possible to evaluate subscale scores to determine whether they may be reported. A subscore is considered to be of added value if the correlation between the true subscore and the observed subscore is greater than the correlation between the true subscore and the observed total score (Sinharay et al., 2007). From a prediction perspective, a subscore is said to have value if it can predict the true subdomain better than the total score. Interest in determining if a subscore has value and deriving metrics for quantifying value has increased over the past decade (e.g., Brennan, 2012; Feinberg & Jurich, 2017; Haberman, 2008b). Haberman's proportional reduction in mean squared error (PRMSE) has received a considerable amount of attention in several studies (e.g., Meijer et al., 2017; Wang, 2017). These authors (ibid.) argue that the higher the PRMSE reported for an indicator of a score estimate, compared to other indicators for the score, the more valuable it is (see Section 2.5.1).

Research in subscale score reporting has received much attention in tests that report individual scores (e.g., de la Torre & Patz, 2005; DeMars, 2006; Edwards & Vevea, 2006; Haladyna & Kramer, 2004; Kahraman & Kamata, 2004; Wainer et al., 2001; Yao & Boughton, 2007; Yen, 1987). To that effect,

research has revealed three psychometric concerns. First, subscores often possess lower reliability than the overall score because subscores are drawn from a subset of the total test; consequentially, subscores may not precisely measure unique abilities (Edwards & Vevea, 2006; Goodman & Hambleton, 2004; Haberman, 2008b; Haberman et al., 2009; Monaghan, 2006; Shin, 2004; Wainer et al., 2000; Wainer et al., 2001; Yao & Boughton, 2007; Yen, 1987). Second, the use of biased item parameter estimates in [sub]score estimation can result in inaccurate person proficiency estimates (de la Torre & Hong, 2010; Hambleton et al., 1993). Third, we may not draw additional information from subscale scores over and above an overall score (Haberman, 2005; Haberman et al., 2009; Monaghan, 2006; Sinharay et al., 2007).

There exists a substantial body of research on subscale scores for individual reporting (e.g., de la Torre & Hong, 2010; Edwards & Vevea, 2006; Yao & Boughton, 2007). However, there is much less research on ILSAs and subscale score estimation (e.g., Camilli & Dossey, 2019; Erdemir & Atar, 2020). Regardless of the noted potential problems, subscale scores remain a prominent byproduct of ILSAs. Given the peculiarities of international assessments (the nature of which will be discussed in detail in Chapter 2), it is reasonable to investigate subscore estimation methods in this context. To that end, I focus my dissertation in subscale score estimation methods in the ILSA context.

1.2 Statement of the Problem

In spite of a body of research into subscale score reporting at the individual level, there is a paucity of research into the degree to which the above noted psychometric issues are present in an ILSA context. Importantly, there is little subscale score research in contexts where the emphasis is at the population level, where there are diverse populations (in terms of score performance), and where sophisticated booklet designs require specialized achievement methods. Each of these issues open the possibility of unanticipated impacts on estimated item parameters and subscale scores. To that end, this study aims at evaluating the quality of item parameters and subscore person parameters in different ILSA test designs. Furthermore, this study examines the potential for added subscore value by comparing the performance of different subscale score estimation methods under different conditions.

Therefore, the general purpose of this dissertation is to fill the void of research in subscore estimation methods in the ILSA context by systematically exploring when subscores have added value (using PRMSE) and by comparing the performance of different subscoring methods under various conditions.

1.3 Purpose of the Study

Given the gap in the literature about subscale score estimation in population models in a context where achievement is highly varied across dozens of populations and subscale score correlations vary across country, the purpose of the study is threefold. First, this study intends to evaluate which typically available methods are best suited for estimating item parameters that are to be used in subscale score estimation in an ILSA context. Second, the study intends to investigate which of the methods provide accurate (and/or meaningful) population and subpopulation subscale scores in the international context. Third, this study will investigate which model specific subscale scores provide more valuble subscale scores by examining the differences in PRMSE.

1.4 Research Questions

Given the peculiarities of ILSA, discussed briefly above and in detail subsequently, my study will address the following research questions:

1. Which of several typically available subscale score estimation models produce the best item parameters?
2. Which of several typically available subscale score estimation models produce the psychometrically-best population score estimates?
3. Which subscale score estimation method results in the most valuable subscale scores?

In this study, I will use the TIMSS and SACMEQ studies to motivate and inform my simulation studies. That is, the simulated test conditions (i.e., number of subscales, subscale length, correlation between subscales) will be influenced by the empirical structures of these two large-scale assessments. In addition, each of these research questions is considered from a single- and multiple-group contests in order to explore the effect of achievement heterogeneity. I will also test my findings on TIMSS 2015 eighth grade mathematics.

1.5 Summary

In summary, this study intends to identify which methods are most suitable for reporting subscale scores. First, this study evaluates item parameter estimates

to identify which model provides the least biased item parameters. Second, this study evaluates person parameters in order to observe which subscale scores are least biased. Third, the study evaluates which of the three IRT subscore estimation methods produces the most valuable subscores using the PRMSE index. The study's significance is in its potential ability to impact the accuracy and fairness of subscore reporting in ILSA. This issue is specifically important to international assessments which use subscale scores as a source of information that may be used to provide deeper understanding into various broad content domains.

In Chapter 2, I review literature pertinent to subscale score estimation under the IRT framework. Focus will largely be placed on subscale score estimation within ILSA. In Chapter 3, I will discuss the methods that I used in my simulation study. Chapter 4 discusses the methods that were used for my empirical study. Chapters 5 and 6 provide the results of the simulation- and empirical-studies, respectively. In Chapter 7, I will provide the discussion and conclusions.

Chapter 2

Literature Review

2.1 Overview of the Chapter

This chapter reviews literature on subscale score estimation methods and processes. The emphasis of this study is placed on subscale score estimation in the ILSA context. The literature review starts with an overview of item response theory. This is followed by a definition of subscale scores and a brief introduction to the notion of subscale score value. Then I proceed to take an in-depth look at how IRT models are being applied to estimate subscale scores at the individual student-level and the merits and demerits of each model. This is followed by a review of the scaling process in ILSA. In this portion of the literature review I highlight some key differences between ILSAs and other tests that emphasize individual inferences. Then, what follows is a review of the TIMSS and SACMEQ scaling procedures. The literature review will focus on how subscale scores are reported in ILSA and RLSA. This chapter closes by introducing the proportional reduction in mean squared error (PRMSE) which is used to quantify subscale score value. This literature review intends to reveal the issues raised with respect to subscale score estimation methods. The reviewed literature will also inform the methods that I apply in my study.

2.2 An Overview of the Item Response Theory Scoring Frameworks

Estimating examinee scores may arguably be one of the most important aspects of educational measurement (Hambleton et al., 1991). Test scores are a source of information from which inferences are made. These scores provide evidence reflecting an examinee's performance on a test and are used to support decisions about selection, diagnosis and placement (Kane, 2013; Liu, 2015). Test scores also serve as a major source of information that is used for educational policy analysis, program evaluation, research and accountability (Kane, 2013). As such, test scores are the basis for many population and individual based decisions in education and beyond.

There are two commonly used frameworks for reporting test scores. These are classical test theory (CTT) and item response theory (IRT). CTT assumes

that a test-taker has a true score on some *latent variable*[1] or construct (e.g., achievement, attitude, or behaviour), and that the observed score is the result of the true score measured with some unobservable measurement error (Crocker & Algina, 1986). Most often, assessments in CTT are scored by summing the responses to test items. The IRT framework, on the other hand, provides a set of probabilistic models that describe the relationship between an examinee's response to a test item and their underlying latent variable being measured by a scale (Fayers et al., 2005). (DeAyala, 2013, p. 20) writes that "IRT models assume that the response data are a manifestation of one or more person-oriented latent *dimensions*[2] or *factors*". IRT models locate examinees and items on the same continuum. This dissertation will focus on IRT scoring methods.

2.2.1 Unidimensional IRT

Early IRT models were applied to unidimensional assessments which measured a single educational or psychological construct (Embretson & Reise, 2000, p. 4). An example is the Southern and Eastern Africa Consortium for Monitoring Educational Quality (SACMEQ) study which, in part, measured sixth grade mathematics achievement in its four cycles. SACMEQ applied unidimensional item response models because it was presumed that a single proficiency (i.e., mathematics proficiency) was enough to account for examinee performance on the test. This class of IRT models that assumes a single underlying dimension can be referred to as unidimensional-IRT (UIRT-class of models) models.

UIRT is a group of statistical models that focus on the item level and are characterized by *item-* or *person-parameters*. These models employ a non-linear functional form that relates item and person properties to the probability of a correct answer. Typically, item parameters include: *difficulty parameter* (i.e., represents how easy or hard the item is with respect to examinees); *discrimination parameter* (i.e., represents how well the item differentiates examinees); *pseudo-guessing parameter* (i.e., represents the 'base probability' of answering or endorsing an item). Furthermore, these models assume that

[1] A latent variable is a hypothetical (unobserved) construct (e.g., knowledge) which may be inferred by an examinee's performance on manifest variables such as items on a test (DeAyala, 2013).

[2] Hypothetical constructs are usually modeled by common groupings or factors known as dimensions that underly the data (Skrondal & Rabe-Hesketh, 2004). For example, mathematics may be viewed as one dimension. Whereas, the *big-five theory* in personality psychology (e.g., Costa & McCrae, 1992) advocates that there are five dimensions of personality which have ontological status: extraversion, agreeableness, conscientiousness, neurotisism, and openness to experience.

the probability of an examinee endorsing an item is also conditional on that examinee's latent proficiency, θ (Lord et al., 1968). Commonly used UIRT models for dichotomously scored data (e.g., scored correct or incorrect) include the one-parameter logistic (1PL) model, the two-parameter logistic (2PL), and three-parameter logistic (3PL) model. The mathematical equation for a 3PL model is given as:

$$P(x_{in} = 1|\theta_n, a_i, b_i, c_i) = c_i + \frac{1 - c_i}{1 + e^{a_i(\theta_n - b_i)}} \tag{2.1}$$

where x_{in} is the n^{th} examinee's response to the i^{th} item; θ_n is the nth examinee's unidimensional proficiency; a_i is the item discrimination parameter; b_i is the item difficulty parameter; c_i is the pseudo-guessing parameter. The 2PL holds the pseudo guessing parameter constant at $c_i = 0$; whereas, the 1PL model holds a_i equal for all items and $c_i = 0$. The Rasch model may be considered a special type of 1PL model which holds $a_i = 1$ for all items.

There are other UIRT models that are applied to polytomous response data. Polytomous IRT models are suitable for items that have more than two scoring outcomes. Polytomous IRT include the nominal response model (NRM, Bock, 1972); partial credit model (PCM, Masters, 1982); the generalized partial credit model (GPCM, Muraki, 1992); the rating scale model (RSM, Andersen, 1997); and the graded response models (GRM, Samejima, 2006). The GPCM model was used in TIMSS 2015 (Martin, Mullis, & Hooper, 2016). GPCM is a polytomous-IRT extension of the 2PL model. GPCM stipulates that the probability of an examinee with proficiency θ_n on scale n will have, for the i^{th} item, a response x_i that is scored in the l^{th} of m_i ordered score categories as the function:

$$P(x_{in} = 1|\theta_n, a_i, b_i, d_{i,1}, \cdots, d_{i,k_i-1}) = \frac{e^{[\sum_{v=0}^{k} Da_i(\theta_n - b_i + d_{i,v})]}}{\sum_{g=0}^{m_i-1} e^{[\sum_{v=0}^{g} Da_i(\theta_n - b_i + d_{i,v})]}} \tag{2.2}$$

where x_{in} is the n^{th} examinee's response to the i^{th} item; m_i is the number of response categories for item i; θ_n is the proficiency of a student n on a scale; a_i is the item discrimination parameter; b_i is the location parameter, characterizing the item difficulty parameter; $d_{i,k}$ is the category k threshold parameter; and the scaling parameter, $D = 1.7$. Indeterminacy in the parameters of the GPCM model is resolved by setting $d_{i,0} = 0$ and $\sum_{k=1}^{m_i-1} d_{i,k} = 0$.

There are three main assumptions underlying UIRT models. These are: (a) *unidimensionality*, (b) *local independence*, and (c) *functional form*. First, the unidimensionality assumption affirms that a single construct underlies

the item responses on a test. Second, local independence states that when examinee proficiency is held constant, responses to any item are statistically independent. In other words, after accounting for the latent variable, there should be no shared residual variance among items. Under this assumption, an examinee's responses to test items is assumed to be conditional on an examinee's θ. Third, the functional form assumption states that data follow a function specified by the IRT model. In this case, the probability of endorsing an item increases monotonically as proficiency increases. However, it is recognized that the expectation to design a test that measures a single construct might be unrealistic in practice (e.g., Ackerman, 1994; de la Torre & Patz, 2005; Kahraman, 2013; McDonald, 2000).

2.2.2 Multidimensional IRT

Multidimensional IRT (MIRT) was developed to describe the relationship between two or more related latent constructs and the probability of endorsing an item (Reckase, 2009). As opposed to UIRT, MIRT assumes that an examinee's response to an item may be due to their location on more than one latent variable. In other words, MIRT models specify the location of examinee n in a multidimensional latent construct space, which then determines examinee n's probability of endorsing this item (Reckase, 2009).

MIRT models may be classified into either *compensatory-* or *non-compensatory*-models. To differentiate the two models, I use the following example: two dimensions may underly an examinee's response to a mathematics word problem; these are: (a) maths proficiency, and (b) reading proficiency. If the two interact to produce observed results, then they are classified as compensatory models (i.e., reading proficiency might influence a person's probability of a correct answer on a mathematics item). In contrast, non-compensatory MIRT models assume that an examinee's proficiency on one latent variable does not compensate for low levels of another latent variable required for correctly responding to an item. For example, the probability of a correct response on the math word item is the product of the probabilities for each construct (i.e., math- and reading-proficiency; Reckase, 2009).

Like UIRT, MIRT comprises a group of statistical models that focus on the item level and are characterized by item- or person-parameters. Therefore, a compensatory-MIRT 3PL model for a D-dimensional test may be defined as:

$$P(x_{in} = 1|\theta_n, \mathbf{a}_i, d_i, c_i) = c_i + \frac{1 - c_i}{1 + e^{(\mathbf{a_i}\theta_\mathbf{n} + d_i)}} \tag{2.3}$$

where θ_n is a D-dimensional vector of estimated latent proficiencies corresponding to each subscale, $\mathbf{a}_i$ is a D-dimensional vector of discrimination parameters

corresponding to each subscale, and d_i is the item location parameter that is related to item difficulty. In contrast, a non-compensatory 3PL MIRT model may be defined as:

$$P(x_{in} = 1|\theta_i, \mathbf{a}_i, b_i, c_i) = c_i + (1 - c_i) \prod_{k=1}^{K} \frac{1}{1 + e^{[\mathbf{a}_{id}(\theta_{id} - b_{id})]}} \tag{2.4}$$

where K indicates the total number of dimensions, b_n is a k-dimensional vector of item difficulties on all dimensions, b_{id} indicates the item difficulty on the d-th dimension of item i. The 2PL MIRT analogy holds $c_i = 0$; whereas the 1PL holds a_i equal to some constant and $c_i = 0$. It should be noted that MIRT models may also be extended to polytomous items.

MIRT models may be conceptualized at the item-level or at the test-level. Item-level MIRT assumes a single item measures multiple latent traits rather than just one (i.e., a case of the aforementioned mathematics word problem, the correct answer also depends on reading proficiency). Test-level MIRT (see Figure 2.1c) assumes each item is unidimensional but the test is made up of subtests that measure distinct latent traits. Both types of MIRT models assume that a test is a combination of latent traits that are often correlated (Ackerman, 1994; Luecht, 2003; Thissen & Edwards, 2005). Operationally, TIMSS fits a two-factor (mathematics and science) test-level MIRT, or they assume between-item multidimensionality in item calibration(Adams et al., 1997; Adams & Wu, 2007). The estimated item parameters are then fixed to estimate both the overall- and subscale-scores (for more details, see Section 2.4.2).

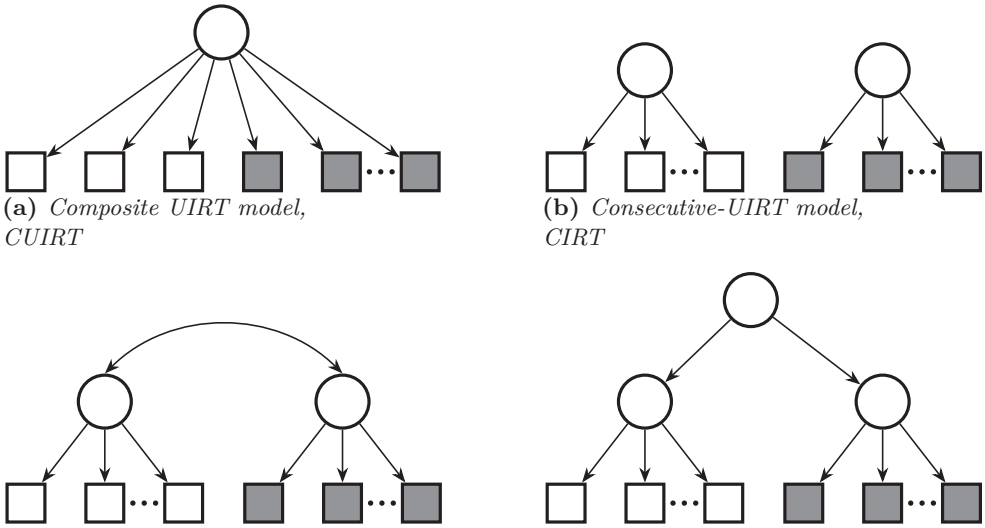
2.3 Methods to Estimate Subscale Scores

This section provides a literature review of the most commonly applied subscale score estimation methods. Section 2.3 reviews literature on score computation (i.e., person proficiency estimation techniques). Section 2.5 gives a review of subscale value with focus on the PRMSE index.

IRT-based subscale score estimation methods may be categorized into two general models based on the latent factor structure (a) UIRT, and (b) MIRT models. In practice, these models may be fit in relation to the assumed actual, possibly multidimensional factor structure of a test. The models may also be imposed because a stipulated factor structure fits the data the best. UIRT methods are broken down into two categories; composite-(CUIRT) and consecutive-UIRT (CIRT). The MIRT methods can be extended to include higher-order IRT and other more complex models (e.g., bi-factor model). Figure 2.1 summarises the factor structure of four hypothetical models.

Furthermore, each model category can be further classified into two groups: a) conventional methods that do not involve augmentation and b) methods that enhance subscale score reliability by augmenting a subscale score using collateral information which correlates with achievement such as demographic data or previous test scores (Skorupski & Carvajal, 2010). All methods have particular assumptions, strengths, and limitations. There has been a substantial amount of literature related to development and utilization of subscale score estimation methods for scores of individual student proficiency (e.g., de la Torre & Patz, 2005; DeMars, 2006; Edwards & Vevea, 2006; Haladyna & Kramer, 2004; Kahraman & Kamata, 2004; Wainer et al., 2001; Yao & Boughton, 2007; Yen, 1987). In the following sections, I will provide a review of each subscale estimation model within the IRT framework.

Figure 2.1
Selected Subscale Score Estimation Models



(a) *Composite UIRT model, CUIRT*

(b) *Consecutive-UIRT model, CIRT*

(c) *Multidimensional IRT model, MIRT*

(d) *Higher-Order IRT model, HO-IRT*

2.3.1 Unidimensional IRT Models

Two classifications of UIRT models are typically used for subscale score computation (Luecht, 2003). The first approach, referred to as *composite-UIRT*

(CUIRT), uses unidimensional item calibration to estimate item parameters for all items on the test. In CUIRT all items $j = 1, 2, \ldots, J$ are assumed to measure one latent variable. However, in CUIRT (Figure 2.1a), subsets of the total test item parameters are then used to score each subdomain. For example, the item parameter estimates for the item parameter estimates for the items in the grey blocks in Figure 2.1a are used to seperately score those items. The second approach —*consecutive-IRT (CIRT)*— uses unidimensional item calibration to estimate item parameters for items corresponding to each subdomain which are used to estimate subscale scores. In the case of CIRT, each scale has an underlying unidimensional latent structure (CIRT, Figure 2.1b). In other words, each item j measures one, and only one, of the D subscales. The main difference between the two is that, like MIRT models (to be discussed in Section 2.3.2), CIRT models assume distinct factors (Figure 2.1b) whereas CUIRT (Figure 2.1a) does not. In other words, CUIRT does not assume distinct factors across subdomains.

There are a few problems and issues with CUIRT when used on a test that is multidimensional in nature. First, in developing a test with multiple dimensions, we violate the unidimensionality assumption of UIRT. Second, in a comparison of CUIRT and CIRT, it was found that both CUIRT and CIRT lose some unique variance associated with subdomains and induces an upward correlation bias between subscale scores (Luecht, 2003). Further, correlations between latent traits cannot be directly estimated with both UIRT models. It is possible to estimate correlations in a two-step procedure that involves estimation of correlations that are then adjusted for attenuation caused by unreliability; such estimation of the correlation matrix has been found to be biased (Longabach, 2015). Third, de la Torre and Song (2009) found that, depending on the extent to which the unidimensional assumption is violated, overall proficiency estimates in UIRT may not be valid and the model provides unreliable subscale score estimates, especially when the number of items in each subdomain is small. Unreliable subscale scores may enhance error of measurement and foster unfairness of a test and its interpretation (ibid). de la Torre and Patz (2005) propose that UIRT scores may be reported at the subdomain level (e.g., CUIRT, CIRT), however they should be complemented by multidimensional scores which may be used to inform a finer grained reporting. In other words, UIRT may be used to report scale scores and their associated norm-referenced and or criterion-referenced scores, and MIRT could be used to inform skills profiles and objective level scores where subscores are estimated from multiple short subdomains that are highly correlated.

2.3.2 Multidimensional IRT Models

MIRT models may be more appealing for subscale score estimation as the subdomains are explicitly specified and modified. A key difference between CIRT and MIRT is that for CIRT, the factors are assumed to have no covariance[3]. Notice that the MIRT model of Figure 2.1c was conceptualized to estimate an examinee's proficiency on two or more correlated factors. However, the model does not directly estimate an overall score. As one possible solution, overall scores may be obtained from MIRT models by using a two-step procedure. After estimating subscale scores, they may be averaged in order to obtain a composite score (Sheng & Wikle, 2007; van der Linden, 1999). For example, the overall TIMSS mathematics score can be calculated by computing the average of the number, geometric shapes and measures, and data display subscale scores[4]. However, a two-step averaging method may produce biased estimates and ignores the relationship between latent traits (Sheng & Wikle, 2007, p. 414). This method also ignores the fact that: (a) the subdomains have different maximum score points; (b) subscores are related, and (c) at different score points, the relationship between composite scores and subscores may be different (Yao, 2012). Finally, two-step approaches treat observed scores as true scores, ignoring the inherent measurement error.

The *Higher-Order IRT* (HO-IRT) model may be considered as one such extension of the MIRT model. In HO-IRT (Figure 2.1d), items within a subdomain are assumed to measure a single latent variable. However, these latent variables are analogous to a second order latent variable. That is, HO-IRT extends the MIRT model by simultaneously estimating an overall score by assuming that subscores are a linear function for the total score and that the subdimensions are regressed onto the higher-order latent variable (Yao, 2010). One advantage of HO-IRT is its ability to obtain overall and subscores from a single model (de la Torre & Song, 2009). Note that subdomain proficiencies are independent to each other and conditional on the overall proficiency (de la Torre & Song, 2009). In other words, a second order factor accounts for the correlations among first order factors (Rijmen et al., 2014). There are several IRT models that may be considered extensions of the MIRT model. Although I include or mention some of the models in my literature review for completeness, my focus is on CUIRT, CIRT and MIRT because they are those methods that are used in practice.

[3] CIRT specifies a MIRT model, but then the factor covariances are fixed to 0.

[4] This is not done in practice. TIMSS 2015 fit a multidimensional model that specified the two broad content domains: mathematics and science. They did not assume multidimensionality within each of these (Martin et al., 2016).

Although a multidimensional structure is often a better representation of complex constructs like math and reading, there are a few problems and issues with MIRT subscale estimation methods. In addition to the previously noted problem around estimating an overall score in a standard MIRT, the complexity of MIRT makes subscale estimation computationally cumbersome. As a result, Tao (2009) argues that MIRT subscore estimation methods may be less feasible in the context of some assessments (e.g., state assessments which require fast reporting of scores). Luecht (2003) and Skorupski (2008) also argue that the complexity of MIRT models means that their application requires higher technical expertise to deal with issues such as: (a) choosing estimators, (b) performing confirmatory factor analysis to identify the number of factors, (c) sorting rotational indeterminacies, and (d) dealing with non-convergence of an estimation solution (the last of which Haberman and Sinharay (2010) argue may not be an average task for a typical psychometrician working for a testing organisation). We note, however, that in the case of TIMSS (as we discuss subsequently), MIRT models are used to a limited degree for the main domains (math and science).

2.3.3 Augmented Scores

It is often the case that even though the total test has adequate reliability, individual subscales might suffer from poor reliability (Sha & McCoy, 2014). For instance, an algebra score may be obtained from a relatively short 12 item sub-test on an 80-item mathematics test (from which an overall mathematics score is reported). Because of being estimated from a subset of the entire test, subscales will generally possess less reliability than the overall test. However, it is possible to increase subscale reliability by incorporating information from other observed data in a procedure known as *augmentation*. In other words, subscale reliability may be improved by exploiting *collateral-* or *ancillary-information* in the subscale score estimation process. Collateral information may be viewed as collected data that correlates with the examinee's proficiency in order to reduce error (Mislevy & Sheehan, 1989). Wang et al. (2004) classify collateral information as: (a) item information such as format, content, or cognitive processes or, (b) examinee information such as educational background, demographic information, or item response information and/or overall performance on other tests. For this reason, the methods applied in ILSA, discussed in Section 2.4.1.2, population modeling or latent regression, are essentially augmented methods.

Wainer et al. (2001) describe subscale score augmentation for IRT-based scores as a multi-stage procedure for subdomain proficiency estimation. In the

first stage, CUIRT examinee proficiency estimates are obtained using maximum likelihood estimation (MLE), maximum a posteriori probability (MAP), or expected a posteriori (EAP) methods. In the second stage, CUIRT reliability estimates in conjunction with the observed covariance matrix among IRT proficiency are computed. In the third, and final stage, the IRT subscale score estimates are regressed on all subscores and weighted using the IRT-based reliability estimate. Therefore, mathematically, the MLE subscale score estimation may be represented as:

$$MLE(\hat{\theta}) = \overline{\mathbf{MLE}(\theta)} + \mathbf{B}(MLE(\theta) - \overline{MLE(\theta)}) \tag{2.5}$$

where $MLE(\hat{\theta})$ is an estimate of the true subscale score, $\overline{MLE(\theta)}$ is a mean vector of each MLE subscale score, $\mathbf{MLE}(\theta)$ is a vector of an examinee's observed subscale scores, and $\mathbf{B}$ is "a matrix that is the multivariate analog for the estimated reliability" (see Wainer et al., 2001). MAP and EAP subscale score estimates may be obtained in the same way. Note that the reliability value ($\mathbf{B}$) is not a CTT based-reliability index; rather, it is a marginal reliability of θ (for computation, see Green et al., 1984).

Several studies have pointed out that the purpose of the reported scores should guide the type of auxiliary information that may be used in the augmentation process (de la Torre et al., 2011; Skorupski, 2008; Stone et al., 2009; Wang, 2017). For instance, de la Torre and Patz (2005) argue that using auxiliary information pertaining to school or student characteristics may be suitable for reporting scores at population or subpopulation level since the observed examinee scores will all be closer to the overall group score. On the other hand, using auxiliary information that is directly obtained from an examinee's test information (e.g., performance in other subscales) may be more appropriate when it comes to reporting subscale scores at student-level (de la Torre & Patz, 2005). As opposed to using other test information, ILSAs, such as TIMSS, use the plethora of information collected in the student background questionnaire and other demographic information to augment their scores in what is known as the *conditioning model* (see Section 2.4.1.2).

2.3.4 A Comparison of Subscale Score Estimation Models

Several simulation and real-data studies have compared the different subscale score estimation methods. Studies have been conducted to compare IRT and MIRT methods, as well as augmented and non-augmented methods of reporting subscale scores. Most of the studies compared different combinations of subscore estimation methods by using varying statistical methods as a basis

for comparisons. Table 2.1 and 2.2 present a summary of simulation and empirical studies, respectively, that compared the performance of IRT-based subscale score estimation models. Table 2.1 summarizes the (a) data generation model (DGM); (b) test characteristics (i.e., number of domains, items per domain, subdomain correlation); (c) item parameters used in the studies, (d) sample sizes; (e) subscore methods studied; and (f) the number of replications. Table 2.2 outlines the (a) test characteristics (i.e., number of domains, items per domain); (b) sample sizes; and (e) subscore methods studied. The studies presented in the tables describe some of the research that has been conducted in subscale score estimation. In what follows, I will compare the UIRT (CUIRT and CIRT) and MIRT models (Section 2.3.4.1). This section will be finalized by a comparision of the augmented and non-augmented methods (Section 2.3.4.2). All of the studies reviewed in Section 2.3.4.1 and 2.3.4.2 were studies whose inferences were aimed at individual inferences.

Table 2.1

IRT-Based Subscore Estimation Methods Comparison Studies that Used Simulated Data

Author	DGM	Test				Examinees		Subscoring Method						Rep
		D	I	r	Item Par.	J	θ	CUIRT	CIRT	MIRT	HO-IRT	Aug	Other	
de la Torre & Hong (2010)	HO-IRT, 3PL	2, 4	10, 20	.5, .7, .9.	Emp.	500	SN		✓		✓			25
de la Torre & Song (2009)	HO-IRT, 3PL	2, 5	10, 20, 30	.0, .4, .7, .9.	Emp.	1,000, 2,000, 4,000	SN	✓			✓			20
de la Torre, Song, & Hong (2011)	HO-IRT	2, 5	10, 20, 30	.0,.4, .7, .9	Emp.	1,000	SN			✓	✓	✓	OPI	1
Edwards & Vevea (2006)	MIRT	2, 4	5, 10, 20, 40	.3, .6, .9	Emp.	2000	MSN	✓					OPI	100
Wang (2018)	MIRT, 3PL	4	5 to 30	.0, .4, .7, .8, .9	b	2,000	NR	✓		✓			OPI, WA	100
Yao & Boughton (2007)	MIRT, PCM	4	12 to 18	.0, .1, .3, .5, .7, .9	Emp.	1000, 3000, 6000	MSN		✓	✓			PC, OPI	20
Yao (2010)	MIRT, HO-IRT	4	12 to 18	.0, .3, .5, .7, .9	Emp.	500, 1,000, 2,000	MSN for MIRT, SN for HO-IRT		✓	✓	✓		Bi-fac.	20

Note. "DGM" = data generation model; "Item Par." = item parameters; "Rep" = number of replications; "D" = number of subscales; "I" = subscale length; "r" = correlation between subcales; "J" = sample size; "θ" = true latent ability; "CUIRT" = composite-UIRT; "CIRT" = consecutive-UIRT; "MIRT" = multidimensional item response theory; "HO-IRT" = higher-order-IRT; "Aug" = non-augmentation subscoring method; "PC" = percent correct; "OPI" = objective performance index; "WA" = Wainer, et. al.'s augmented scoring; "MSN" = multivariate standard normal; "SN" = standard normal distribution; "HO-IRT" = higher order item response theory; "NR" = not reported; "PCM" = partial credit model; "3PL" = three-parameter logistic model, "b" = item difficulty, $b = N(0, -1)$.

Table 2.2
Subscore Estimation Methods Comparison Studies that Used Empirical Data

Author	Test			Methods					
	D	I	J	CUIRT	CIRT	MIRT	HO-IRT	Aug	Other
de la Torre & Hong (2010)	4	15 to 20	2255	✓			✓		
de la Torre & Song (2010)	4	15 to 20	2255	✓			✓		
de la Torre, Song & Hong (2010)	4	15 to 20	2255	✓		✓	✓	✓	OPI
DeMars (2005)	2	15 to 20	2552	✓		✓			Bi-factor, WA
Longabach (2015)	4	9 to 29	3,649 to 10,363		✓	✓		✓	
Stone, Ye, Zhu, & Lane (2009)	4	10 to 17	10545	✓	✓	✓		✓	
Wang, Chen, & Cheng (2004)	5	5 to 18	1716			✓		✓	

Note. "D" = number of subscales; "I" = subscale length; "J" = sample size; "CUIRT" = composite-UIRT; "CIRT" = consecutive-UIRT; "MIRT" = multidimensional item response theory; "HO-IRT" = higher-order-IRT; "Aug" = non-augmentation subscoring method; "OPI" = objective performance index; "WA" = Wainer, et. al.'s augmented scoring; "Bi-fac." = bi-factor; "HO-IRT" = higher order item response theory.

According to Table 2.1, the most commonly manipulated factors in these model comparison studies were related to test characteristics (e.g., number of examinees, number of subdomains, subdomain correlation, subdomain length,item discrimination patterns). Other conditions for comparison included normality and non-normality of subdomain ability distribution. Some of the simulation studies were conducted to evaluate a single condition or a combination of several. However, all of the studies presented in Table 2.1 were conducted to provide inferences relating to tests that measured individual proficiency. These simulation studies did not consider situations where complex test design methods (i.e., the booklet designs) and associated population modeling techniques were used.

2.3.4.1 UIRT vs. MIRT

From the studies presented in Table 2.1 and 2.2, I compared the performance of UIRT and MIRT. Several studies showed that MIRT –and variants thereof– generally have an advantage in precision over UIRT methods of subscore reporting across manipulated testing conditions (Reckase, 2009; Wang et al., 2004; Yao, 2010). In essence, there is often a non-zero correlation between subscales, which may mean that, in theory, Figure 2.1c is more appropriate for subscale score estimation and will result in less biased item- and person-parameter estimates because it takes into account subscale correlations (Wang et al., 2004). It is only if there is no subscale correlation that UIRT and MIRT are expected to produce similar estimates. Studies have confirmed that as subscale correlation increased, MIRT significantly outperformed CUIRT at improving subscale proficiency estimation and classification (Wang et al., 2004; Yao & Boughton, 2007).

Evaluations of different test characteristics and conditions have also shown that MIRT-methods have the advantage of producing better item- and person-parameter estimates for assessments that reports subscale scores. For example, a study by Yao and Boughton (2007) found that MIRT estimates using the Markov chain Monte Carlo (MCMC) method produced significantly better item- and person-parameter estimates than UIRT when subscale correlation was high. In their work, Yao and Boughton (2007) added that UIRT- and MIRT-based methods perform similarly when subscale correlation was low (Yao & Boughton, 2007). That is to say that, on tests where inferences are made at the individual level, item- and person-parameter estimates were found to be fundamentally similar when there was little to no subscale correlation.

de la Torre and Song (2009) showed that total score estimates calculated from CIRT and MIRT were particularly similar when subscales were highly

correlated. The same study also found that the two methods produced similar item parameter estimates when there were fewer subscales (i.e., a two subscale test compared to a four subscale test). It was also observed that MIRT item parameter estimates were more precise when the number of subscales increased. Improvements in the precision of item parameter estimates in MIRT-based methods were larger in shorter tests. Research conducted by Yao (2010) pointed out that in situations of high subscale correlation (i.e., at subscale correlation greater than .8), most IRT methods perform similarly. Wang (2017) wrote that subscale correlation did not affect the performance of UIRT; in other words, though MIRT seemed to perform well across all conditions, UIRT's performance was relatively the same (no gains nor losses in precision). This makes sense since the different subscales are assumed to be different parts of the test. As such, the subscales are modeled separately, and the relationship between subscales may not matter much for the results.

This subsection shows that most literature highlights the advantages of using MIRT-based methods for subscale score estimation over UIRT-based methods. The advantages of MIRT stem from these models' inherent property that they often incorporate information from other subscales (i.e., subscale correlation) in subscale score estimation. However, UIRT and MIRT methods perform the same when subscale correlation is low. Several studies have also shown that though MIRT-based methods (e.g., MIRT, HO-IRT) may generally perform the same; a simple MIRT that implies correlated factors is significantly better than the other MIRT-based methods like HO-IRT (i.e., de la Torre & Song, 2009; Yao & Boughton, 2007). de la Torre and Song (2009) showed that UIRT-based models perform similar to HO-IRT when there are fewer subscales being assessed (i.e., two instead of 5). Nonetheless, the inferences that were drawn from all of these cited studies were not aimed towards ILSA but rather for tests aimed at providing individual learners achievement scores. Taken together, these studies provide evidence that under some conditions, MIRT (and its variants thereof) has an advantage over the UIRT based methods and that the models perform the same in other situations.

2.3.4.2 Augmented vs. Non-augmented

Ostensibly, the methods that are applied in ILSA are essentially augmented in that population level estimates are obtained from a model that incorporates background variables to students test responses (ILSA score estimation is discussed in-depth in Section 2.4.1.2). With further reference to Table 2.1 and 2.2, I compared the augmented and non-augmented methods in studies whose inferences were for the individual student. Several studies conducted on tests

that are for individual inference have shown that augmentation dramatically increased subscale reliability across all simulated conditions (Longabach, 2015; Skorupski, 2008; Wang et al., 2004). In addition to augmentation improving reliability, Skorupski's (2008) study revealed that augmentation resulted in a relatively stable decrease in subscale score variability and an increase in subscale correlation. As such, studies have shown that augmented subscale scores are estimated with more precision over non-augmented subscale scores (de la Torre et al., 2011; Edwards & Vevea, 2006; Kahraman & Kamata, 2004).

Research conducted by Edwards and Vevea (2006) revealed that subscale scores obtained using an empirical Bayes IRT augmentation procedure provided overall improvement in subscale score estimation over non-augmented IRT scores. The empirical Bayes IRT procedure increased the reliability of subscores. However, the magnitude of the gains observed via augmentation were found to be a function of the manipulated test characteristics. For example, augmented IRT-summed scale scores showed the greatest improvement in situations of high subscale correlation, low numbers of items in each subscale and high reliability in the subscale providing ancillary information (Edwards & Vevea, 2006). For tests that used within-test information as auxilliary information, de la Torre et al. (2011) added that correlation based subscore estimation methods – MIRT, augmented scoring, and HO-IRT – provide sufficiently better subscore estimates than non-augmented subscale scores in tests comprised of short subtests and highly correlated abilities. In addition, Wang (2017) observed that the advantage of the two augmentation subscoring methods (MIRT and Wainer's augmentation method) over UIRT was more prominent when the (a) subscale correlation was higher, (b) subscale being estimated was shorter, and (c) length of the other subscales in the tests was longer.

Though Skorupski's (2008) study showed that augmentation improved reliability, it was observed that there was a relatively stable decrease in variability of the subscales and an increase in subscale correlation. Longabach (2015) showed that observed subscale scores may become less distinctive to the overall group's pattern of subscores or to the examinee's other subscale scores as a result of the decrease in subscale variability and increased correlation. To that effect, Skorupski (2008) argued that the observed high inter-correlation of subscales resulting from augmentation may be an indicator that the resulting subscale scores are not useful in providing individual diagnostic information since the scores will be too similar.

In conclusion, literature indicates that augmented methods have some advantages over non-augmented methods. This literature has found that augmented subscale scores are more reliable and precise than non-augmented subscale scores. Nonetheless, though augmented methods were found to be

advantageous, it is worthy to note that their improvements were greater over some specific test conditions (i.e., high subscale correlation, short subscales, subscale length). However, it has also been shown that choice of auxilliary information greatly impacts on the utility of subscale scores for different situations; that is to say, out of test information (e.g., demographic data) may not be suitable for individual based scoring whereas within-test information (e.g., other subscale performance) may enhance student level scores (see Section 2.3.3).

2.4 Subscale Score Reporting in ILSA

This section takes the reader from the widely researched area of subscale score estimation models for tests of individual inferences to those tests that emphasize population and subpopulation level scores. To do so, this section will inform the reader that though there are a lot of borrowed elements in score estimation, the tests are designed, administered, scored and reported differently. Additionally, this section will take the reader through the TIMSS overall and subscale score estimation methods, which are grounded in the IRT framework. The little information that is available for SACMEQ methods will also be discussed.

2.4.1 From the Individual to the Population

As established in the introduction, literature on subscale score estimation for tests of individual inferences is well developed. However, there are several fairly significant differences between tests that report achievement estimates for individuals (e.g., the Norwegian national exams) and assessments that have been designed to provide population-level achievement estimates (e.g., ILSAs). These tests sometimes differ in the nature of their administration and overall methods applied. For example, individual-level scores in a Norwegian national mathematics test are obtained by (a) administering a single test to all the candidates sitting for the exams that year, and then (b) using conventional IRT methods to estimate examinee scores. In contrast, ILSA's population-level scores may be estimated from assessments that: (a) employ complex booklet designs, and (b) whose scores are estimated through population modeling techniques. Section 2.4.1.1 and 2.4.1.2 describe some prominent features of ILSAs that make them different from tests that are intended to report individual student scores. These methods —or variants thereof— are applied in TIMSS, PISA, PIRLS and many other ILSAs.

2.4.1.1 Multiple matrix sampling

ILSAs often assess broad content domains (e.g, mathematics and science in TIMSS; and reading, mathematics, and science in PISA), their main aim is to collect as much information as necessary on a construct to warrant the reporting of population level estimates. As a result, large scale educational assessments often contain a large number of items that assess broad content and cognitive domains. For example, the TIMSS 2015 fourth grade assessment of student achievement included a total of 169 mathematics and 176 science items, respectively. All these items were intended to provide enough information that would warrant the reporting of 14 scales[5]. In total, TIMSS prepared over 10 hours of test material.

As a response to the increased demand of content coverage without increasing the testing time, most ILSA's have adopted complex test designs that rely on experimental designs (Carstens & Hastedt, 2010). These sampling designs are also known as *multiple–matrix sampling* or *rotated booklet designs* (Gonzalez & Rutkowski, 2010). The items on the test are distributed into non-overlapping blocks according to the test specification. In other words, these sampling designs assign items into blocks which are then systematically placed into booklets; each examinee responds to a single booklet ((Gonzalez & Rutkowski, 2010). Items on the instrument are administered to some portion of the sample, and each examinee sees some proportion of the total test. For instance, fourth grade TIMSS 2015 distributed all mathematics and science items into a total of 28 non-overlapping blocks; that is, 14 mathematics blocks (M01-M14) and 14 science blocks (S01-S14) (see Table 2.3, copied from Martin, Mullis, & Foy, 2016).

Each block was composed of approximately 10 to 14 items. These items were distributed to each block to reflect the content and cognitive distribution of the total test item pool. The blocks were then distributed into 14 student booklets (refer to Table 2.3 for block distribution). Each sampled student responded to a single booklet that contained two mathematics blocks and two science blocks; in other words, each participant responded to items from both assessed content domains.

Literature outlines many other booklet designs that may be used in ILSAs (Frey et al., 2009; Gonzalez & Rutkowski, 2010). Some examples include:

[5] According to Martin, Mullis, and Hooper (2016), TIMSS 2015 grade four reported seven mathematics scales (i.e., three-content, three cognitive and an overall scale score) and seven science scales (i.e., three-content, three cognitive and an overall scale score). The mathematics content domains were: number, algebra, geometry, and data and chance. Science domains included: life science, physical science, and earth science. Both the mathematics- and science-domains each reported three cognitive domains: knowing, applying, and reasoning.

Table 2.3
TIMSS 2015 Student Achievement Booklet Design — Fourth and Eighth Grades

Student Achievement Booklet	Part 1		Part 2	
Booklet 1	M01	M02	S01	S02
Booklet 2	S02	S03	M02	M03
Booklet 3	M03	M04	S03	S04
Booklet 4	S04	S05	M04	M05
Booklet 5	M05	M06	S05	S06
Booklet 6	S06	S07	M06	M07
Booklet 7	M07	M08	S07	S08
Booklet 8	S08	S09	M08	M09
Booklet 9	M09	M10	S09	S10
Booklet 10	S10	S11	M10	M11
Booklet 11	M11	M12	S11	S12
Booklet 12	S12	S13	M12	M02
Booklet 13	M13	M14	S13	S14
Booklet 14	S14	S01	M14	M01

Note. Note. "M" = Mathematics; and "S" = Science. Adapted from TIMSS 2015 Assessment Design (p. 91), by M. Martin, I. Mullis, & P. Foy, 2016, Boston, TIMSS. Copyright 2016 by TIMSS.

complete permutation designs, Youden squares designs and others. For a thorough descriptions of each of these booklet designs see, for example, Frey et al. (2009); and Gonzalez and Rutkowski (2010). Note that methods for TIMSS 2015 that were previously described may be classified as a partially balanced incomplete block design.

Booklet designs have several advantages and disadvantages. Multiple-matrix sampling allows assessments to measure broad content domains in relatively reasonable time for examinees (Rutkowski et al., 2014). Booklet designs may also be a solution to cluster position- and item carryover-effects, whilst supporting increased item security and facilitating linking (see Frey et al., 2009, for a full description). However, Mislevy (1991) acknowledges that booklet designs result in two psychometric challenges for item calibration and estimating

of individual examinee proficiency. First, only a few examinees respond to a given item. Second, each examinee is exposed to a subset of the entire test. One consequence is that these challenges may affect the accuracy of item parameter estimates that may be used for scoring the individual examinee (Gonzalez & Rutkowski, 2010; Mislevy, 1991). However, booklet designs facilitate reporting population or sub-population scores (Frey et al., 2009; Gonzalez & Rutkowski, 2010; Mislevy, 1991).

To address issues and complexities relating to multiple-matrix sampling designs, TIMSS 2015 employed latent regression methodology. These methods will be described in Section 2.4.1.2.

2.4.1.2 Population modeling using latent regression

In addition to responding to achievement tests, examinees respond to a background questionnaire that collects information about the students' demographic data, academic and non-academic information, as well as attitudes and motivation. Some ILSAs, like TIMSS, PISA and PIRLS, fit a statistical model that incorporates a variety of information about the examinee to obtain achievement estimates[6]. These background variables serve as covariates of students' achievement that are used in the scoring model.

ILSAs often use a single administered test to assess multiple domains. For example, TIMSS assesses examinees in mathematics and science and PISA assesses mathematics, reading and science. For these tests, examinee proficiency, $\boldsymbol{\theta}_n$, on the entire assessment may be represented by a k-dimensional vector $\boldsymbol{\theta}_n = \theta_{n1}, \ldots, \theta_{nk}$, where k is the total number of assessed domains in each ILSA. It is assumed that these subscales are assessed by different items for each scale and that $\boldsymbol{x}_n = (x_{n11}, \ldots, x_{nI_11}), \ldots, (x_{n1k}, \ldots, x_{nI_kk})$ represents k sets of I_1 to I_k responses. Therefore, $\boldsymbol{x}_n$ may be a vector of responses for examinee n from the k-dimensions; in practice, a vector of responses includes 169 and 172 mathematics and science items for TIMSS 2015 fourth grade. But because both TIMSS and PISA use rotated booklet designs, $\boldsymbol{x}_n$ is not a complete vector since each student will be exposed to a subset of the items. For example, TIMSS uses a multidimensional IRT model for overall mathematics and science. This multidimensional-IRT model assumes test-level or between-item multidimensionality (Adams et al., 1997; Adams & Wu, 2007).

Since IRT-models are latent variable models it is reasonable to think of $\boldsymbol{\theta}$ as a missing value and to approximate a statistic involving θ (e.g., population mean, a percentile point, or a sample regression coefficient) by its expectation

[6] These techniques are similar to the augmented methods described in Section 2.3.3.

given a matrix of item responses, $\boldsymbol{x}_n$, and a matrix of all examinees responses to the administered background variables, $\boldsymbol{y}$ (Mislevy, Johnson, et al., 1992). Mislevy considered θ as missing data, an approximate of $t(\theta, y)$ (i.e., sample mean or sample percentile point) by its expectation given (x, y). That is to say:

$$\begin{aligned} \hat{t}(\boldsymbol{x}, \boldsymbol{y}) &= E[t(\theta, \boldsymbol{y})|\boldsymbol{x}, \boldsymbol{y}] \\ &= \int t(\theta, \boldsymbol{y}) P(\theta|\boldsymbol{x}, \boldsymbol{y}) d\theta \end{aligned} \tag{2.6}$$

where $\hat{t}(\boldsymbol{x}, \boldsymbol{y})$ is an estimate of the statistic $t(\boldsymbol{\theta}, \boldsymbol{y})$ (i.e., mean or sample percentile point to estimate a corresponding population quantity T) by its expectation given $(\boldsymbol{x}, \boldsymbol{y})$. Since closed-form solutions are not forthcoming in IRT models, the integration in Equation (2.6) uses random draws from the conditional distributions $(P(\theta_i|\boldsymbol{x}_i, \boldsymbol{y}_i))$ for each examinee, i where, $i = 1, \ldots, n$ (Mislevy, Johnson, et al., 1992). Most often, these values are drawn multiple times, and the values are known as *plausible values* in ILSA or *multiple imputations* in missing data analysis (Rubin, 1987).

The conditional distribution of θ may be derived in the following way. Using Bayes' theorem and then the IRT assumption of local independence (i.e., $P(\boldsymbol{x}_n|\theta, \boldsymbol{y}_n) = P(\boldsymbol{x}_n|\theta)$),

$$\begin{aligned} P(\theta|\boldsymbol{x}_n, \boldsymbol{y}_n) &\propto P(\boldsymbol{x}_n|\theta, \boldsymbol{y}_n) P(\theta|\boldsymbol{y}_n) \\ &= P(\boldsymbol{x}_n|\theta) P(\theta|\boldsymbol{x}_n) \end{aligned} \tag{2.7}$$

where $P(\boldsymbol{x}_n|\theta)$ is the likelihood function for θ induced by observing $\boldsymbol{x}_n$ and $P(\theta|\boldsymbol{y}_n)$ is the distribution of θ for the observed background variables, $\boldsymbol{y}_n$. Equation (2.7) stipulates that the posterior distribution of a student with observed responses, $\boldsymbol{x}_n$, vector of background variables, $\boldsymbol{y}_n$ is proportional to the product of the likelihood of θ induced by $\boldsymbol{x}_n$ through the response model and the population density (Mislevy, Beaton, et al., 1992).

The distribution of θ is assumed multivariate normal with a mean given by a linear model (also called the *conditioning model*):

$$\theta = \Gamma' y + \epsilon \tag{2.8}$$

where $\boldsymbol{y}$ is a vector of background- or conditioning-variables; $\epsilon \sim N(0, \boldsymbol{\Sigma})$; Γ is a matrix each of whose columns is the effects for each conditioning variable; and $\boldsymbol{\Sigma}$ is a residual covariance matrix for $\boldsymbol{\theta}$[7], θ. As a means to estimating proficiency

[7]The method for estimating Γ and $\boldsymbol{\Sigma}$ with the Expectation and Maximization (EM) has been thoroughly described in Mislevy (1985)

using plausible values, all student background questionaire variables and some student demographic information are considered as conditional variables to form $\boldsymbol{y}$. Operationally, all student background variables in TIMSS are subject to a principal component analysis in order to reduce the number of variables used in the model used to estimate Γ. It is common practice that only those components accounting for 90% of the variance in the data are selected (see Martin, Mullis, & Hooper, 2016, p. 13.16). The resulting principal components are used as the predictors, $\boldsymbol{y}$, of the conditioning model.

2.4.2 Scaling for TIMSS and SACMEQ

The following section describes an overview of the TIMSS and SACMEQ scaling methodology. This section will provide an overview on the IRT-models that are used to obtain overall and subscale scores.

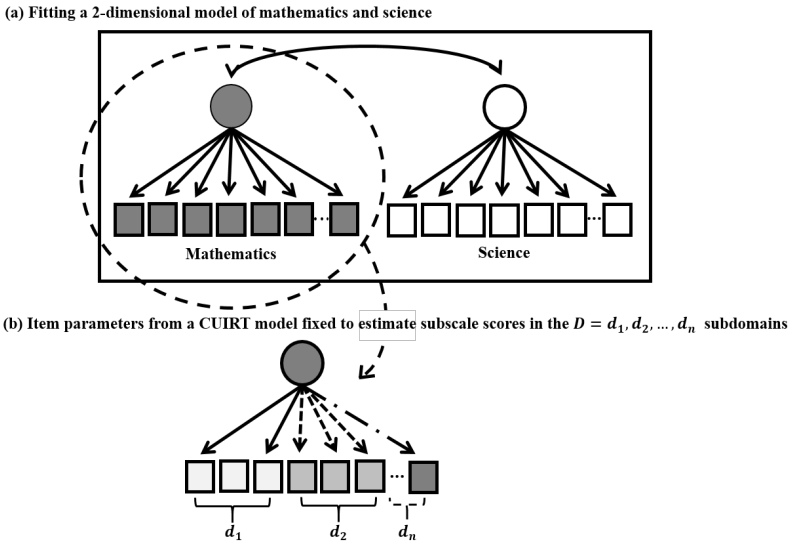
2.4.2.1 TIMSS

Three distinct IRT models, depending on item type and scoring procedure, were used in the analysis of the TIMSS 2015 assessment data (Martin, Mullis, & Hooper, 2016). These latent variable models were used to describe the probability of a student's pattern of responses given their proficiency, unobserved latent trait, and various characteristics of the item (Martin, Mullis, & Hooper, 2016). TIMSS 2015 fit a two-dimensional model of mathematics and science. A 3PL model (see Equation (2.1)) was used in the calibration of multiple-choice items (scored correct or incorrect). TIMSS 2015 also employed a 2PL model (i.e., similar to Equation (2.1) with $c_i = 0$) for constructed response items that were scored correct or incorrect. A GPCM (Equation (2.2)) was used for the polytomous constructed items. The constructed response items for which the GPCM model was fit had three possible score levels: 0, 1, and 2.

The overall procedure for obtaining population-level scores may be summarized into a three-step process (von Davier & Sinharay, 2010). First, item parameters are estimated from the multidimensional-IRT (where the two dimensions are math and science). Second, the estimated item parameters are fixed and used to get estimates of Γ and Σ. Third, using estimates of Γ and Σ, plausible values are drawn. Though TIMSS fits a multidimensional model of mathematics and science, these constructs are treated as unidimensional even though subscale scores are reported. In other words, after obtaining item parameters for the overall scale, subscale scores are estimated assuming an augmented UIRT model which considers the conditioning variables. Yamamoto and Kulick (2000) argue that one major disadvantage to this calibration

approach is that the differences in content domains are often attenuated as the scores tend to be regressed towards the mean. The researchers further argued that "each content area, mathematics or science, is treated separately when estimating item parameters, differential profiles of content area proficiency can be examined, both across countries and across subpopulations within a country" (p. 265). In other words, TIMSS fixes the item parameters from the mathematics domain in Figure 2.2 (a). These models do not assume further dimensionality in the mathematics construct and specific subscale correlation in item calibration. In TIMSS 2015 eighth-grade mathematics, a four-dimensional MIRT model which fixes the unidimensional mathematics item parameters is fit to estimate subscale scores (see (b) in Figure 2.2).

Figure 2.2
Example TIMSS Scaling Process



Educational Testing Service's MGROUP program (Sheehan, 1985; Thomas, 1993) was used to generate the IRT proficiency scores. One advantage of MGROUP is that it can be used to perform multidimensional scaling using the responses on the test. In other words, the multidimensional scaling feature makes it possible to estimate content and cognitive domain proficiency scores. However, in practice, TIMSS fits a two-dimensional model of mathematics and science that does not assume any multidimensionality within each broad content domain wherefrom subscale scores may be estimated (see Figure 2.2).

2.4.2.2 SACMEQ

Students and teachers scores on SACMEQ's achievement domains are reported as a mean achievement score that is scaled using the Rasch model in IRT (Sandefur, 2018; Spaull, 2011). All of the items on SACMEQ's achievement domains are dichotomously scored. After item calibration has produced item parameter estimates using all items, $\widehat{\boldsymbol{b}} = (\widehat{b_1}, \cdots, \widehat{b_n})$, those item parameters are used to estimate an overall score. For every cycle, the Rasch-scaled scores on reading, mathematics and HIV/AIDS knowledge test (HAKT) test are transformed to a Grade 6 pupil average of 500 and standard deviation of 100 (Spaull, 2011).

However, some key components of SACMEQ's scaling process are not publicly available (i.e., SACMEQ did not release the list of item parameters that were used for scoring in any of its cycles). Beyond test construction information and sampling, among other pieces of relevant information, not much information about the actual scoring process (i.e., what estimator they use) is disclosed. A review of the technical documentation simply provides a listing of the IRT-model used to generate scores (the Rasch model in this case); a specification of the linear transformation of the scores; and a thorough definition of the proficiency levels.

2.5 Subscale Score Value

Because subscale scores are: (a) estimated from less information relative to the total score, and (b) correlated with the total score, it may be the case that the subscore does not provide unique information about the subdomain than can be garnered from the total score. This issue is often characterized in terms of value (Haberman, 2008a, 2008b; Puhan et al., 2010). A subscale score is considered to be of added value if the correlation between the true subscale score and the observed subscale score is greater than the correlation between the true subscore and the observed total score (Sinharay et al., 2007). From a prediction perspective, a subscore is said to have value if it can predict the true subdomain better than the total score (or any other score). Interest in determining if a subscore has value and deriving metrics for quantifying value has increased over the past decade (e.g., Brennan, 2012; Feinberg & Jurich, 2017; Haberman, 2008b; Tate, 2004). Of particular interest to the study, I will discuss the proportional reduction in mean squared error (PRMSE)

2.5.1 Haberman's PRMSE

Haberman (2008b) proposed the proportional reduction in mean squared error (PRMSE) index that could be used to evaluate the value of a subscale score. The original PRMSE was developed based on Kelley's regressed-score estimates (Kelley, 1947), which are rooted in CTT. The general idea was to compare PRMSEs of several indicators of a true subscale score based on multiple scores. These were: the subscale score itself, the total score estimate of the subscale score, and the augmented subscale score (a linear combination of the subscale score and the total test score).

Haberman and Sinharay (2010) also proposed a MIRT-based PRMSE ($PRMSE(\theta_d|M_d)$)[8] and its CUIRT equivalent ($PRMSE(\theta_d|U_d)$)[9]. The IRT-based PRMSEs measure "how much the mean squared error in estimating the score is reduced by any observed score (relative to using the mean)" (Thissen, 2013, p. 30). Mathematically, the model implied $PRMSE$ of a subdomain, d, may be presented as:

$$PRMSE_{d\theta} = 1 - \frac{\hat{\tau}^2_{d\theta}}{\hat{\tau}^2_{d0\theta}} \tag{2.9}$$

where $\hat{\tau}^2_{d\theta}$ is the error variance associated with the observed IRT score's approximation of the true subscale score, and $\hat{\tau}^2_{d0\theta}$ is the variance of the true IRT subscale score. By definition, $PRMSE$ is analogous to the reliability of a model-specific subscale score serving as an estimate of a true score (see Green et al., 1984; Rosa et al., 2001). Therefore, PRMSE is equivalent to:

$$\rho_{d\theta} = \frac{{\sigma_{\theta_d}}^2 - {\sigma_{e_d}}^2}{{\sigma_{e_d}}^2} \tag{2.10}$$

where $\rho_{d\theta}$ is the subdomain reliability, ${\sigma_{\theta_d}}^2$ is the variance of the observed subdomain proficiency, and ${\sigma_{e_d}}^2$ is the average subdomain score error variance. It should be noted that the larger the PRMSE, the smaller the mean squared error to estimate the true mean squared error (Meijer et al., 2017).

Studies have illustrated how the PRMSE may be used to quantify the added value of subscale scores (Haberman, 2008b; Haberman & Sinharay, 2010). Operationally, the PRMSE may reveal which subscale score from several competing models contains more information and may be more useful in providing diagnostic information (Wedman & Lyrén, 2015). Therefore, the PRMSEs

[8] $PRMSE(\theta_d|M_d)$ is the PRMSE using the EAP estimate for θ for subscale d computed from a multidimensional IRT model fitted to subscale d as an estimate of θ_d.

[9] $PRMSE(\theta_d|U_d)$ is the PRMSE using the EAP estimate for θ for subscale d computed from a unidimensional-IRT model fitted to subscale d as an estimate of θ_d.

obtained from CUIRT ($PRMSE(\theta_d|CUIRT)$), CIRT ($PRMSE(\theta_d|CIRT)$)[10], and MIRT($PRMSE(\theta_d|MIRT)$) subscale scores will be compared in order to evaluate which subscale scores contain the most information, and thus reveal which subscale score would be better to report. Extending Haberman's example, it should be seen that if:

1. $PRMSE(\theta_d|CUIRT) > PRMSE(\theta_d|CIRT)$ and $PRMSE(\theta_d|MIRT)$, then subscale score $\theta_d|CUIRT$ has value over $\theta_d|CIRT$ and $\theta_d|MIRT$.
2. $PRMSE(\theta_d|CIRT) > PRMSE(\theta_d|CUIRT)$ and $PRMSE(\theta_d|MIRT)$, then subscale score $\theta_d|CIRT$ has value over $\theta_d|CUIRT$ and $\theta_d|MIRT$.
3. $PRMSE(\theta_d|MIRT) > PRMSE(\theta_d|CUIRT)$ and $PRMSE(\theta_d|CIRT)$, then subscale score $\theta_d|MIRT$ has value over $\theta_d|CUIRT$ and $\theta_d|CIRT$.

2.6 Summary

This literature review discussed subscale score estimation within the IRT framework for tests that are designed to report individual- and population-level scores. Major emphasis was placed on UIRT- and some MIRT-based models (i.e., MIRT and HO-IRT), as well as augmented and non-augmented scoring procedures. Much of the literature that was reviewed comprised of application and comparison studies of these IRT-models that were conducted on tests that were designed to report individual-level scores. It was widely suggested through literature that MIRT-models perform better than UIRT; though in some conditions they performed the same. Despite the model's common use for subscale score estimation in tests that report individual scores, little has been done to compare how well they would perform when applied to tests that are designed for reporting population-level scores.

The literature review also provided a brief review of the ILSA scaling process with emphasis on TIMSS and SACMEQ. As much as TIMSS, PISA, and PIRLS —to name but a few— employ similar methods, they are different from those applied in the SACMEQ scaling process. For instance, TIMSS, PISA and PIRLS employ complex booklet designs, and use population modeling and latent regression to estimate population parameters. In contrast, SACMEQ does not use these complex sampling designs and it is not stated in any of their technical documentation that they draw on the use of plausible value methodology.

[10] $PRMSE(\theta_d|CIRT)$ is the PRMSE using the EAP estimate for θ for subscale d computed from a CIRT model fitted to subscale d as an estimate of θ_d.

ILSAs generally tend to apply a UIRT model to estimate population-level subscale scores. TIMSS uses a multidimensional model to obtain item parameter estimates for mathematics and science. However, the item parameters for each of these domains are used to score each subdomain. Yet, UIRT models, of which CUIRT and CIRT are classified, tend to ignore that each domain being tested is comprised of several content-based subdomains whose scores are also reported.

One observation from the literature review is that ILSA score estimation methods are a departure from the methods developed for scoring tests geared towards reporting scores of individual proficiencies. For example, IRT model specification which specifies the factor structure of the test. The assumed model is used to obtain item parameter estimates. As another example, ILSAs employ the conditioning model to obtain scores. This may be viewed as a form of augmentation which is geared towards improving the quality of reported scores. However, research into how issues present in subscale score estimation for individual inference may manifest in ILSA is limited.

Literature also pointed out conditions that may result in more valuable subscale scores. Ideally, subscale scores are estimated from less information, and the more correlated subscale scores are with the overall score, the less likely the subscale score would be valuable. From a model perspective, the nature and structure of the CUIRT model may result in lower PRMSE (an index of subscale value) compared to CIRT and MIRT. Conceptually, CUIRT does not model subscales. Since CUIRT assumes that a test is unidimensional, reported subscale scores may not result in larger PRMSE values because the subscales may inherently be highly correlated with one another and the overall score. A subscore is considered to be of added value if the correlation between the true subscore and the observed subscore is greater than the correlation between the true subscore and the observed total score (Sinharay et al., 2007). Therefore if the subscores are highly correlated among themselves, as well as with the overall score, then the likelihood of estimating valuable subscales would be low from CUIRT than the models that assume multidimensionality.

Based on the literature review, Chapters 3 and 4 describe the simulations- and empirical-methods. The two chapters will describe how I intend to evaluate the performance of several subscale score estimation methods: CUIRT (and it's extension, CUIRT-Op), CIRT, MIRT. Though there exist many subscale score estimation models, the three studied models are commonly used in practice.

Chapter 3

Simulation Methods

3.1 Introduction

In an effort to answer the research questions in Section 1.4, several simulation studies were conducted. This study used the TIMSS and SACMEQ studies to motivate and inform my simulation studies. That is, the simulated test conditions (i.e., number of subscales, subscale length, correlation between subscales) were influenced by the empirical structures of these two large-scale assessments. The first simulation (Study 1) was designed to resemble the SACMEQ III's HAKT test, and the second (Study 2), the TIMSS 8th grade mathematics test (for descriptions of their test specifications, see Chapter 2). The difference between the two simulation studies is that Study 1 does not employ matrix sampling or booklet designs whilst Study 2 does.

Each of the simulation studies were conducted separately assuming single- and multiple-groups samples (see Table 3.1). Since ILSAs are cross-cutural studies, considering these two samples made it possible to understand how competing IRT subscale score estimation models perform when you move from single- to multiple-populations. For the single group case, I chose one middle performing country whilst the multiple groups case sampled nine populations. Section 3.2.2 and 3.3.2 describe how the samples were selected. In what follows, I describe methods for implementing these two simulations.

Table 3.1
Simulation Studies

	Sample	
Study	A	B
Study 1	Single group	Multiple groups
Study 2	Single group	Multiple groups

Sections 3.2 and 3.3 describe the (a) study conditions; (b) sample sizes; (c) data generation processes (DGP); and (d) scoring techniques for each simulation study. To simulate data, the following steps were repeated for each studied condition; generate: (a) item parameters, (b) subscale specific person

proficiency parameters, and (c) item response patterns. To ensure stability, these steps were repeated across 100 replications. This chapter concludes by outlining simulation study analysis. Section 3.4 discusses the criteria for evaluation of the accuracy of parameter estimation from the CUIRT, CIRT, and MIRT models. To evaluate model performance, I examined: bias, absolute bias, root mean squared error of the parameter estimates, PRMSE, and model fit.

3.2 Simulation Study 1

Study 1 simulates data to reflect the SACMEQ III's HAKT test design. In its current design, the HAKT test assesses five subdomains. These are: (a) definitions and terminology, (b) transmission mechanisms, (c) avoidance behaviors, (d) diagnosis and treatment, and (e) myths and misconceptions (Maughan-Brown & Spaull, 2014). The test was comprised of 86 multiple choice (MC) items which were not equally distributed across the domains (i.e., a distribution of 10-28-24-16-8). The items on the HAKT test are all dichotomously scored.

3.2.1 Study Conditions

In this section, I will describe the conditions that were studied in my simulations. To conduct the study, tests were simulated to mimic empirical conditions observed in an ILSA setting. The conditions studied in this simulation study manipulated three factors that are directly related to general test characteristics. These are: number of subscales, the correlation between subscales, and the number of items in each subscale. For each choice, both empirical and theoretical justification will be provided.

3.2.1.1 Number of Subscales

A study by Sinharay (2010) provided a summary of the test characteristics of 25 empirical tests that report subscale scores for person inferences. The summary outlined the number of subscales in each test, average subscale length, subscale correlation and reliabilities. A total of 20 of the 25 tests had between two and four subdomains. Most of the scales in ILSA (e.g., TIMSS, PISA, and SACMEQ) are comprised of between three and five content subscales (Martin, Mullis, & Hooper, 2016; Moloi & Chetty, 2014; OECD, 2017). SACMEQ's HAKT test had 5 content subscales. With that in consideration, in simulation Study 1, I evaluated tests of three and five subscales.

3.2.1.2 Subscale Correlation

Previous research has shown that subscale scores have better psychometric properties (i.e., added value) when there are lower correlations between subscales (Haberman, 2008b; Sinharay, 2010). In other studies, upper bounds of the correlations have been specified at .9 (e.g., de la Torre & Patz, 2005; Wang, 2017; Yao, 2010) and anything higher than that makes the test essentially unidimensional, thus making the scores less distinctive from one another. In contrast, low subscale correlation may render the subscales more distinct. Sinharay's (2010) summary revealed that correlations between subscales may range from .41 to .77. However, some ILSAs exhibit large correlations between subscales within the same content scale (i.e., .90 and above). Appendix A shows the correlations of each subscale on the TIMSS 2015 assessment by country. These correlations were calculated using IDB Analyzer (IEA, 2020). The IDB Analyzer is a windows-based tool that appropriately treats the ILSA study designs and creates SAS code or SPSS syntax to perform analysis. Tables A.1 and A.2 show each country's correlation between the mathematics and science domains, respectively. Table A.3 to A.6 show the correlations between each of the mathematics subdomains and the science subdomains. Based on Appendix A, it is evident that the subscale score correlations may differ between countries. For example, the correlation between the algebra and geometry subscales were .94 and .74 in the BSJG districts in China, and Saudi Arabia, respectively. The lowest observed subscale correlation was .45 between geometry and earth sciences for Morocco. To summarise, Appendix A shows that subscale correlations may be moderate between subscales from different content domains (i.e., .45 and above) and higher between subscales from the same content scales (i.e., between .71 and .95).

Therefore, to explore the subscale score estimation process in ILSA, I consider several between-subdomain correlations that represent a realistic, empirically observed range: ρ, where $\rho = .45, .75, .95$[1]. Note that $\rho = 0$ was not specified in the study because it is highly unlikely that such correlations would occur in a practical testing situation (Wang, 2017). The corresponding

[1]The magnitude of the correlations is larger between .45 and .75 than it is between .75 and .95.

correlation matrix, $\boldsymbol{\Sigma}$, for a $d = 1, 2, 3, \ldots, D$ domain test may be expressed as:

$$\boldsymbol{\Sigma} = \begin{array}{c} \\ \theta_1 \\ \theta_2 \\ \theta_3 \\ \vdots \\ \theta_D \end{array} \begin{array}{c} \begin{array}{ccccc} \theta_1 & \theta_2 & \theta_3 & \ldots & \theta_D \end{array} \\ \begin{pmatrix} 1 & & & & \\ \rho & 1 & & & \\ \rho & \rho & 1 & & \\ \vdots & \vdots & \ddots & \ddots & \\ \rho & \rho & \rho & \ddots & 1 \end{pmatrix} \end{array}$$

As a limitation, subscale correlations in all simulated conditions were fixed equal (i.e., I did not consider different correlations between subdomains). However, because I consider several bivariate correlations, it is possible to examine how different correlations can impact the subscale score estimation process in ILSA. That is, what are the expected results when subscale correlation is low-, moderate, or high?

3.2.1.3 Number of Items per Subdomain

One important consideration when developing a test is the test length. Researchers have shown that test length may impact on the accuracy of IRT item- and person-parameter estimates (de la Torre et al., 2011; Kahraman & Kamata, 2004; Shin, 2007). Test length may be defined in terms of the number of items and the number of score points. Sinharay's (2010) summary showed that most of the tests possess an average of 10 to 30 items in each subscale, which were distributed into tests whose total length ranged from 40 to 204 items.

SACMEQ's HAKT test comprised of between 8 and 28 items per subdomain. As a result, Study 1 (no booklet designs) specifies 5, 10, and 15 (short, moderate and long, respectively) item subtests. That is, Study 1 evaluated several tests of equal subdomain length: (1) 5-5-5, (2) 10-10-10, (3) 15-15-15, (4) 5-5-5-5-5, (5) 10-10-10-10-10, and (6) 15-15-15-15-15. As a result, the total test lengths were between 15 and 75 items[2]. Simulated examinees were administered all of the items (i.e., each sampled student responded to all of the items on the test). These simulations were conducted over single- and multiple-group samples, the details of which are described next.

[2]Since I only considered subscales of equal length (i.e., 5-, 10- and 15-items per domain), I did not get to a test condition that had 28 items-per-subdomain because it would have resulted in an unrealistically long test. The three and five-subdomain tests would have resulted in tests of 84 and 140 items. Such test lengths are uncharacteristic of the SACMEQ assessment and may need more time allocated to test administration compared to the actual time allocated for assessing each content area.

3.2.2 Sample

Edwards and Vevea (2006) argued that a sample size of 2,000 was large enough to produce stable item parameter estimates whilst being small enough to converge relatively fast for UIRT models. Yao and Boughton (2007), on the other hand, showed that a sample size of 3,000 was sufficient enough for estimating item parameter models in MIRT models that consider factor level multidimensionality. Sample size sufficiency is typically not an issue in ILSA since these studies are usually conducted in multiple countries, resulting in large sample sizes. For example, a total of 61,396 students participated in SACMEQ III. It was from this representation that 9 populations in each study were drawn: three top-, middle- and low-performing countries. For the single group case, I chose one middle performing country whilst the multiple groups case I sampled nine populations. The reasons for selecting nine countries out of all the participants on the surveys were twofold. First, this combination of groups captured the spread of the populations across the two assessments. Second, limiting the study to nine groups would keep the simulation manageable. As a result, the empirically observed sample sizes came up to 34,847 in SACMEQ III.

To facilitate an examination of how different subscale score estimation methods perform over diverse populations, all the simulations were conducted over two different types of samples that resulted in the pre-specified total sample sizes. The first component of each study assumed that the tests were administered to a single group of simulated examinees. In other words, the entire sample was generated from a sample that had the same mean (μ) and standard deviation (σ), respectively. For the single group case in Study 1, I simulated a sample of 6,000 examinees, which is typical, if large for individual country sample sizes. The second sample assumed that the tests were administered to diverse populations. A simulated total sample size of 30,000 for Study 1 was comparable to empirically observed sample sizes. Each of the group sizes were drawn similar to the empirically observed country samples on SACMEQ III. Table 3.2 shows the sample distribution for the multiple group's simulation that was drawn from the countries that participated in SACMEQ III. The table also shows the each group's proficiency distribution and standard deviation that were used in the data generation process. It should be noted that Study 1's domain proficiencies and standard deviation were equal across all subdomains (see Section 3.2.3.2). Since Study 1 was based on SACMEQ III's HAKT test and I was not able to obtain empirical subdomain scores[3], I used the overall

[3]I was not granted item-level information by SACMEQ because of concerns related to test-security and validity.

Table 3.2
Sampling- and Proficiency-Distribution Used in the Data Generation Process for Simulation Study 1: Multiple Groups

G	N	θ_D	sd_D
1	3656	1.76	.82
2	3491	1.12	.85
3	2242	.44	.97
4	5859	.07	.75
5	2253	.04	.88
6	3329	-.04	.88
7	2482	-.83	.80
8	3701	-1.27	.87
9	2987	-1.71	.86

Note. N = Sample size; G = group; θ_D = True population subdomain proficiency; sd_D = Subdomain proficiency standard deviation.

score for each country to generate each subscale proficiency[4].

Table 3.2 shows the distributions for each population. The population means and standard deviations were different for each group and these were drawn from the empirical data sets.

3.2.3 Data Generation Process

In this section, I will describe the data generation process (DGP) for Study 1. Data were generated for a single group and multiple groups assuming sample sizes of 6,000 and 30,000, respectively.

3.2.3.1 Item Parameter Generation

The unique item set that was used in Study 1 comprised of multiple choice (MC) items that were generated using the Rasch model (see the description of the model in Chapter 2). Rasch was specifically chosen because that is the model that SACMEQ uses (Sandefur, 2018; Spaull, 2011). The difficulty parameters were randomly generated from a uniform distribution:

[4]The overall scores are publicly reported in the technical reports and executive summaries.

$b_{di} \sim \mathcal{U}(-2.35, 2.35)$. Researchers have argued that the distribution of difficulty parameters, $(-2.35, 2.35)$, helps to ensure that low and high proficiency levels are modeled (Kim & Lee, 2006; Wolf, 2014). The generated item parameters were then randomly assigned to each subscale. These generated item parameters were fixed for all 100 replications on each test condition. An example of the distribution of difficulty parameters for a three-subdomain, 5 items per subdomain test are illustrated in Table 3.3.

Table 3.3

An Example of Item Parameters for a Three-Subscale Test.

Subdomain	Item	b_{1i}	b_{2i}	b_{3i}
1	1	-.34		
1	2	.52		
1	3	-.07		
1	4	1.58		
1	5	-1.7		
2	6		-1.54	
2	7		.87	
2	8		1.01	
2	9		-.99	
2	10		1.51	
3	11			-1.62
3	12			.13
3	13			1.59
3	14			-1.32
3	15			1.22

Table 3.4 shows some descriptive statistics for all of Study 1's generating difficulty parameters. The average item difficulties are around 0 for all simulated test conditions. Subsequent standard deviations ranged between between .858 and 1.491. The median of the item parameters, across all of the simulated conditions, were all between $-.610$ and .430. The table also shows the minimum and maximum values of the generating item difficulty parameters. Table 3.4 also presents the subsequent ranges of the item parameters (i.e., the difference between the maximum and minimum item difficulty by condition). For example, Domain 1 on the simulated test with 3 subdomains, 5 items per domain had a mean item difficulty of $-.002$, standard deviation (SD) of 1.202, median of $-.070$. The respective minimum and maximum values of -1.700 and

Table 3.4
Descriptive Statistics for Study 1's Generating Difficulty Parameters

I	J	D	Mean	SD	Median	Min	Max	Range
3	5	1	-.002	1.202	-.070	-1.700	1.580	3.280
		2	-.002	1.249	.280	-1.540	1.510	3.050
		3	.000	1.449	.130	-1.620	1.590	3.210
	10	1	.001	1.129	.295	-1.760	1.630	3.390
		2	-.001	1.171	.095	-1.820	1.550	3.370
		3	-.001	1.491	.405	-2.330	1.570	3.900
	15	1	.000	1.245	-.600	-1.650	1.910	3.560
		2	.001	1.291	-.230	-1.880	1.820	3.700
		3	-.001	1.172	.390	-1.640	1.580	3.220
5	5	1	-.002	.780	.390	-1.250	.690	1.940
		2	.000	1.370	-.380	-1.250	1.810	3.060
		3	.000	1.415	-.610	-1.250	2.320	3.570
		4	.000	.858	-.280	-.870	1.140	2.010
		5	.002	1.475	-.420	-1.230	2.310	3.540
	10	1	-.001	1.326	-.050	-1.530	2.160	3.690
		2	.000	1.484	.065	-2.270	1.620	3.890
		3	.001	1.110	.310	-1.510	1.580	3.090
		4	-.001	1.202	-.165	-1.510	1.950	3.460
		5	.000	1.373	-.105	-1.930	1.600	3.530
	15	1	.000	1.340	.430	-2.210	1.350	3.560
		2	.001	1.246	.280	-1.950	1.860	3.810
		3	.000	1.075	.010	-2.010	1.600	3.610
		4	.000	1.201	-.480	-1.270	2.350	3.620
		5	-.001	1.232	.150	-1.920	1.650	3.570

Note. I = number of domains; J = items per domain; D = Domain; SD = standard deviation; Min = minimum; Max = maximum.

1.580 resulted in a range of 3.280. It should be noted the the generating item parameters were the same for the single and the multiple-groups simulations.

3.2.3.2 Person parameter generation

The person parameters used in Study 1 were resampled across conditions, and across replications. Subscale proficiency was estimated from a multivariate normal (MVN) distribution for both the single and multiple group's simulations. To obtain subscale proficiency estimates for the single group's simulations, a vector of each examinee's true subscale scores, $\boldsymbol{\theta}_j$, were simulated from a distribution, $\boldsymbol{\theta}_j \sim \mathcal{N}_D(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, where $\boldsymbol{\mu}$ is a $1 \times D$ vector of sample means, and $\boldsymbol{\Sigma}$ is a $D \times D$ correlation matrix of the true subscale scores (refer to Section 3.2.1.2 for details). Based on the subscale correlation matrix ($\boldsymbol{\Sigma}$) with 1s on the diagonal and correlations of .45, .75 and .95; respective domain means were set to 0.

The $d = 1, 2, \ldots, D$ subscale proficiency scores for the $p = 1, 2, \ldots, P$ multiple groups were drawn from a MVN distribution: $\boldsymbol{\theta}_{pj} \sim \mathcal{N}_{D_p}(\boldsymbol{\mu_{pd}}, \boldsymbol{\Sigma})$ where $\boldsymbol{\theta}_{pj}$ and $\boldsymbol{\mu_{pd}}$ are the country specific subscale proficiency estimates and subscale mean vector's. Therefore, subscale proficiency estimates were simulated from $P \times D$ country mean and standard deviation matrices, $\boldsymbol{\Sigma}$. The estimated means and variances of nine countries on SACMEQ III's HAKT test were used as the generating subscale proficiency values for the populations that comprised the multiple group's sample. Each reported country score was converted to a $Z-score$ (assuming the SACMEQ mean and standard deviation of 500 and 100, respecively) and that was used as the true mean. In this study, $\boldsymbol{\mu_{pd}} = \left[\boldsymbol{\mu_{p1}}, \quad \boldsymbol{\mu_{p2}}, \quad \ldots, \quad \boldsymbol{\mu_{pD}}\right]$ and $\boldsymbol{\sigma_{pd}} = \left[\boldsymbol{\sigma_{p1}}, \quad \boldsymbol{\sigma_{p2}}, \quad \ldots, \quad \boldsymbol{\sigma_{PD}}\right]$. However, each group's subscale score was set to be equal on all subscales, and these values were drawn from SACMEQ III's population means (i.e., $\boldsymbol{\mu_{p1}} = \boldsymbol{\mu_{p2}} = \ldots = \boldsymbol{\mu_{PD}}$). Based on the means and correlations between subscales ($\boldsymbol{\Sigma}$), mean and standard deviation matrices for the multiple group simulations were as follows: $\boldsymbol{\mu_{PD}} = 1.76, 1.12, .44, .07, .04, -.04, -.83, -1.27, -1.71$ and $\boldsymbol{\sigma_{PD}} = .82, .85, .97, .75, .88, .88, .80, .87, .86$. $\boldsymbol{\mu_{PD}}$ and $\boldsymbol{\sigma_{PD}}$ were the same over all domains for the multiple populations.

3.2.3.3 Response Pattern Generation

Test responses were generated from a test-level multidimensional Rasch model (the components of the Rasch model were explained in Section 2.2.1 in Chapter 2). The simulation assumes the underlying factor structure is a multidimensional one where the number of subdomains set according to the

simulation condition, and that the total score is a linear composite of possibly correlated sub-domains. `R` package `lsasim` (Matta et al., 2018) was used to generate item responses from a multidimensional item response model of the form,

$$P(x_{dij} = 1|\theta_{dj}, b_{di}) = \frac{1}{1 + \exp\left(-(\theta_{dj} - b_{di})\right)}, \tag{3.1}$$

where b_{di} is the item difficulty parameter associated with item $i = 1, 2, \ldots, I_d$ and sub-domain $d = 1, 2, \ldots, D$, and θ_{dj} is the proficiency for sub-domain d of examinee $j = 1, 2, \ldots, J$. Test responses were generated seperately for 6,000 and 30,000 examinees used in the single- and multiple-groups studies, respectively.

3.2.4 Item Calibration and Scoring

Based on the generated item responses from all simulations, the next step was to estimate population proficiency distributions. SACMEQ administers all the items to every sampled examinee; each student is exposed to all the items on the entire test. SACMEQ estimates item parameters using the Rasch model. The estimated item parameters are then fixed and used to estimate proficiency scores.

Scores in Studies 1 were estimated from three IRT models (based on assumed factor structure). These models were: composite-UIRT (CUIRT), consecutive-UIRT (CIRT), and multidimensional (MIRT)[5]. First, the items were calibrated assuming each specified model. Second, estimated and assumed fixed item parameters were used for proficiency estimation.

Scores in Study 1 were estimated using EAP estimation. Since SACMEQ does not report all the details about proficiency estimation, Study 1 estimated individual examinee's proficiency using Expected-a-posteriori method (EAP, Bock & Mislevy, 1982). The choice of the EAP method was strengthened by findings from a series of simulation studies that have been conducted (e.g., Kim & Nicewander, 1993; Lu et al., 2005; von Davier et al., 2009). The researchers showed that, though scores tend to regress towards the mean, EAP provide better group-level estimates than marginal maximum likelihood (MML) or Warm's mean weighted likelihood estimates (WLE). von Davier et al. (2009) also argued that mean EAP estimates obtained on ILSAs were not biased.

To complement the proficiency estimates, I also estimated the PRMSE, and obtained model fit indices. The PRMSE was used to compare which IRT subscale score estimation method resulted in the most valuable subscale

[5] All of the models have been thoroughly described in the literature review.

scores. This index was estimated and reported for each subscale score that was calculated based on an IRT method. PRMSE was estimated for each simulated condition. As previously discussed, the PRMSE was estimated equivalent to the subdomain marginal reliability (Haberman, 2008b). In addition, the model fit indices were used to compare model fit. The fit indices I used in my study were: -2 log likelihood, AIC and BIC indices. These model fit indices will be discussed in Section 3.4.3.

3.2.5 Summary of Simulation Study 1

Table 3.5 provides a summary of Study 1's simulation conditions. The three simulation conditions (number of subscales, number of items per subscale and subscale correlation) yielded 18 conditions in total. One hundred replications were carried out under each condition. All analysis were conducted for the the single and multiple groups studies.

Table 3.5
Summary of Study 1 Simulation Conditions

Groups	N	D	J	ρ		
Single	6,000	3	5	.45	.75	.95
			10	.45	.75	.95
			15	.45	.75	.95
		5	5	.45	.75	.95
			10	.45	.75	.95
			15	.45	.75	.95
Multiple	30,000	3	5	.45	.75	.95
			10	.45	.75	.95
			15	.45	.75	.95
		5	5	.45	.75	.95
			10	.45	.75	.95
			15	.45	.75	.95

Note. N = Sample size; D = Number of subscales; J = Subscale length; ρ = Subscale correlation.

3.3 Simulation Study 2

Study 2 simulated data to reflect an assessment that employs multiple matrix sampling and latent regression techniques to estimate population proficiency

(a clear description is provided in the literature review). To achieve this, data were simulated to mimic the TIMSS 2015 eighth grade mathematics test design. The assessment was composed of 209 dichotomous and polytomous scored MC and constructed response (CR) items, which were drawn from four content subdomains.

3.3.1 Study Conditions

In this section, I will describe the conditions that were studied in simulation study 2. To conduct the study, tests were simulated to mimic empirical conditions observed in an ILSA setting. The conditions studied in this simulation manipulated three factors that are directly related to general test characteristics. These are: number of subscales, the correlation between subscales, and the number of items in each subscale. The details and the rationale behind their choice was empirically and theoretically justified.

3.3.1.1 Number of Subscales

Most of the scales in ILSA (e.g., TIMSS, PISA, and SACMEQ) are comprised of between three and five content subscales (Martin, Mullis, & Hooper, 2016; Moloi & Chetty, 2014; OECD, 2017). More specifically, TIMSS 2015 had 3 and 4 subscales in its fourth and eighth mathematics assessment, respectively (Martin, Mullis, & Hooper, 2016). With that in consideration, simulation Study 2 specifies 3 and 4 subscale tests since the study resembles TIMSS 2015.

3.3.1.2 Subscale Correlation

To explore the subscale score estimation process in Study 2, I consider several between-subdomain correlations that represent a realistic, empirically observed range: ρ, where $\rho = .45, .75, .95$. The decisions for selecting these examined item parameter estimates and how I used them was specified in Section 3.2.1.2. Note that, since I consider several bivariate correlations, it is possible to examine how different correlations can impact the subscale score estimation process in ILSA.

3.3.1.3 Number of Items per Subdomain

Beyond stable item- and person-parameters, an important consideration in test development includes test taking time, where considerations of classroom periods, fatigue, and other factors are important. As ILSAs often employ complex booklet designs to minimize test length while optimizing parameters

of interest, the total number of items can be substantially higher than on a test that does not use such designs. For example, TIMSS eighth grade mathematics subscale lengths ranged between 41 and 64 items per domain (making the total test 209 items long).

In its current design, items on the mathematics test were distributed across 14 booklets using balanced incomplete booklet designs. Each of these booklets was composed of four blocks: 2 science and 2 mathematics with approximately 12-18 items in each block at eighth grade. In total, the assessment had a total of 28 blocks: 14 containing mathematics items and 14 containing science items.

In this study, the number of items for each subdomain was 40 and 60, representing short and long subdomains. The numbers were similar to the minimum and maximum number of items on the TIMSS subscales (Mullis, et al., 2016). These resulted in tests of different total length: 120, 160, 180, and 240 items. Table 3.6 describes how the items were distributed into the different booklets.

Table 3.6
Study 2 Booklet Design

Booklet	Block 1	Block 2
1	M01	M02
2	M02	M03
3	M03	M04
4	M04	M05
5	M05	M06
6	M06	M07
7	M07	M08
8	M08	M09
9	M09	M10
10	M10	M11
11	M11	M12
12	M12	M13
13	M13	M14
14	M14	M01

3.3.2 Sample

The use of booklet designs, in Study 2, reduces the number of examinees responding to some items on the test. Typically, countries participating in

TIMSS aim for sample sizes of about 4,500 students in order to ensure population coverage and that there are enough students responding to each item[6] (Martin & Mullis, 2019). On TIMSS 2015 fourth grade, the item calibration samples for each country ranged between 2,397 (Kuwait) and 21,177 (United Arab Emirates). The item calibration sample at eighth grade were between 2,933 (Lithuania) and 18,012 (United Arab Emirates).

To draw a total sample size for my simulation studies, the groups of participating countries in TIMSS were arranged according to performance. It was from this representation that 9 populations in each study were drawn: three top-, middle- and low-performing countries in mathematics[7]. For the single group case, I chose a sample of 6,000 examinees. This sample was deemed sufficient to ensure adequate exposure of the items in the single-groups conditions. The multiple groups case included nine sampled populations from the TIMSS 2015 dataset for reasons similar to those presented in Table 3.7.

Table 3.7

Sampling- and Proficiency-Distribution Used in the Data Generation Process for Simulation Study 2

G	N	θ_1	sd_1	θ_2	sd_2	θ_3	sd_3	θ_4	sd_4
1	3656	1.40	.88	1.23	.95	1.33	.88	1.53	.84
2	3491	1.19	.94	1.17	.86	1.35	.83	1.16	.87
3	2242	1.09	1.04	.89	.99	1.07	1.00	.90	1.00
4	5859	-.23	.82	-.04	.93	.05	.83	-.08	.73
5	2253	-.24	.74	.16	.77	-.24	.92	.17	.77
6	3329	-.27	.92	.09	.99	-.14	.86	.00	.98
7	2482	-1.36	.94	-1.86	.79	-1.17	.77	-1.42	.83
8	3701	-1.30	.82	-1.61	.89	-1.68	.81	-1.53	.86
9	2987	-1.31	.83	-1.57	.88	-1.64	.96	-1.63	.91

Note. N = Sample size; G = group; DGP = data generation process; θ_D = Study 1 true population subdomain proficiency; sd_D = Study 1 subdomain proficiency standard deviation; θ_1 - θ_4 = Study 2 true population subdomain proficiency; sd_1 - sd_4 = Study 2 subdomain proficiency standard deviation.

To facilitate an examination of how different subscale score estimation methods perform over diverse populations, all the simulations were conducted over two different types of samples that resulted in the pre-specified total

[6]The first three cycles of PISA used a random sample of 500 examinees from each OECD member country as the item calibration sample (OECD, 2002, 2005, 2009).

[7]As a result, the empirically observed sample size came up to 62,884 for TIMSS 2015.

sample sizes. The first component of each study assumed that the tests were administered to a single group of simulated examinees. In other words, the entire sample was generated from a sample that had the same mean (μ_d) and standard deviation (σ_d), respectively. Recall that for the single group case in both Study 2, I simulated a sample of 6,000 examinees. Table 3.7 shows the distribution of the samples considered in simulation study 2. Country 4[8] provided the proficiency and standard deviation of each domain for the single groups study. That is: $\boldsymbol{\mu_{pd}} = [-.23, -.04, .05, -.08]$ and $\boldsymbol{\sigma_{pd}} = [.82, .93, .83, .73]$. The second sample assumed that the tests were administered to diverse populations. Study 2's sample size was 30,000[9]. The table also shows the each group's proficiency distribution and standard deviation that were used in the data generation process.

3.3.3 Data Generation Process

In this section, I will describe the DGP for Study 2. There are some significant differences with Study 1. Similar to Study 1, data were generated for a single group and multiple groups with sample sizes of 6,000 and 30,000, respectively.

3.3.3.1 Item Parameter Generation/Specification

Study 2 simulated several tests of different subdomain length: 40 and 60 irems per domain. These resulted in tests of different total length: 120, 160, 180, and 240 items. The item parameters that were used to simulate data in the study were empirically drawn. These parameters were obtained from the TIMSS 2015 international report (Martin, Mullis, & Hooper, 2016). An item parameter bank was compiled containing item parameters that were used for proficiency estimation in TIMSS 2015 eighth grade mathematics. The unique item set comprised of 209 MC and CR items that were estimated using 2PL-, 3PL-, and GPCM-models. For illustrative purposes, in this study I only assumed that the items were 2PL and GPCM. To do so, I collapsed the 3PL items to 2PL by dropping the pseudo-guessing parameter. Each subdomain had a separate set of item parameters.

Based on the 209 empirically observed items that were used to scale the TIMSS 2015 mathematics test, only 12 were estimated using the GPCM. In other words, about 6% of the total test items were polytomously scored. I took

[8]Country 4 was the middle performing country on TIMSS 2015's overall mathematics achievement test at eighth grade.

[9]I did not simulate the data to the empirical sample sizes in order to save computation time.

that into account when selecting the items in my simulation. Table 3.8 shows the distribution of 2PL and GPCM items on the simulated tests. Table 3.8, I illustrate how many of each were in the single and multiple groups simulations. For each, I specified the total number of (a) subscales, (b) subscale length (total number of items in each subscale); (c) 2PL and GPCM items in each domain; (d) 2PL and GPCM items on the test; and (e) the total number of items on each simulated test. For each simulated domain, I assumed that the number of GPCM (polytomously scored) items were the same proportion as the total test. As a result the 40 and 60 subdomain lengths were comprised of two and four GPCM. However, the total number of GPCM items in the tests depended on the total number of domains. According to Table 3.8, the tests comprised of 6, 12, 8, and 16 GPCM items on all of the conditions for the single- and multiple-groups simulations.

Table 3.8
Simulation Study 2 Distribution of Items

Groups	Number of Subscales	Subscale Length	Items in domain		Items on Test		Total Number of Items
			2PL	GPCM	2PL	GPCM	
Single	3	40	38	2	114	6	120
		60	56	4	168	12	180
	4	40	38	2	152	8	160
		60	56	4	224	16	240
Multiple	3	40	38	2	114	6	120
		60	56	4	168	12	180
	4	40	38	2	152	8	160
		60	56	4	224	16	240

To conduct my experiment, item parameters were randomly drawn from the pool of 209 item parameters to equal the total number of items on the test. For example, 40 item parameters were randomly selected, with replacement, for the 40-item per subdomain test. Item parameters were selected in the same way for the 60 items per subdomain tests. This was done because one of the simulated test designs had a total of 240 items (4 domains $\times$ 60 items per domain). That presented a 31-item disparity from the 209 item parameters in the item bank. As a result, I resampled the items to make up for the difference. The selected item parameters were fixed across all conditions with the same subscale length, and 100 replications.

Tables 3.9 to 3.11 present summaries of the descriptive statistics for the generating item parameters. These descriptive statistics are reported across all of the studied conditions in Study 2. It should be noted that each condition has a different range of item parameters. For example, Table 3.9 shows that though the mean difficulties, β, are all over .5, the ranges of the item parameters are different. The values ranged from as low as 1.543 for domain 1 in the three subdomain, 40 item per subdomain test to as high as 2.996 in the second domain of the same test. It should be noted that the second domain has items which have the highest generating item difficulties, 2.163. Tables 3.10 and 3.11 show the descriptive statistics of the location parameters for the GPCM items; $d1$ and $d2$, respectively. The values of the threshold parameters ranged from 1.435 to 1.435 for domain 1 in the three subdomain, 60 item per subdomain test. However, some of the ranges were lower (see Tables 3.10 and 3.11).

Table 3.9

Descriptive Statistics of the Generating Item Difficulty and Discrimination Parameters

			a						*b*					
Domains	*j*	*d*	Mean	*SD*	Median	Min	Max	Range	Mean	SD	Median	Min	Max	Range
3	40	1	1.322	.345	1.398	.513	1.926	1.413	.722	.411	.682	.039	1.582	1.543
		2	.995	.339	.965	.474	1.886	1.412	.422	.709	.553	-.834	2.163	2.996
		3	1.171	.380	1.159	.524	2.166	1.641	.691	.492	.783	-.231	1.498	1.730
	60	1	1.236	.313	1.203	.513	1.926	1.413	.774	.467	.739	.039	1.727	1.689
		2	1.116	.400	1.142	.474	1.976	1.502	.505	.647	.607	-.834	2.163	2.996
		3	1.121	.297	1.109	.524	1.847	1.322	.674	.527	.710	-.308	1.517	1.824
4	40	1	1.281	.339	1.230	.683	1.861	1.178	.766	.466	.694	.048	1.727	1.679
		2	1.067	.335	1.110	.580	1.800	1.220	.511	.607	.569	-.831	2.163	2.994
		3	1.168	.278	1.147	.674	1.847	1.173	.644	.522	.597	-.308	1.517	1.824
		4	1.198	.297	1.136	.681	1.660	.979	.365	.454	.498	-.615	1.055	1.670
	60	1	1.385	.339	1.321	.513	1.926	1.413	.717	.406	.635	.039	1.727	1.689
		2	1.127	.386	1.142	.504	1.886	1.381	.484	.608	.569	-.831	2.163	2.994
		3	1.206	.381	1.150	.659	2.351	1.692	.680	.494	.819	-.308	1.517	1.824
		4	1.202	.293	1.167	.527	2.076	1.549	.445	.436	.465	-.441	1.639	2.080

Note. j = subscale length; d = specific domain; SD = standard deviation; "Min" = minimum; "Max" = maximum.

Table 3.10
*Descriptive Statistics of the Generating Threshold Parameters: d*1

D	*j*	*d*	Mean	*SD*	Median	Min	Max	Range
3	40	1	-.300	.000	-.300	-.300	-.300	.000
		2	.297	.000	.297	.297	.297	.000
		3	.645	.000	.645	.645	.645	.000
	60	1	-1.435	.000	-1.435	-1.435	-1.435	.000
		2	.297	.000	.297	.297	.297	.000
		3	.372	.473	.645	-.174	.645	.820
4	40	1	-.867	.802	-.867	-1.435	-.300	1.134
		2	.297	.000	.297	.297	.297	.000
		3	-.508	.472	-.508	-.842	-.174	.668
		4	-.268	.091	-.268	-.332	-.203	.129
	60	1	-1.056	.655	-1.435	-1.435	-.300	1.134
		2	.297	.000	.297	.297	.297	.000
		3	-.500	.000	-.500	-.500	-.500	.000
		4	-.667	.353	-.633	-1.036	-.332	.704

Note. D = number of subdomains; j = subscale length; d = specific subdomain; *SD* = Standard Deviation; "Min" = Minimum; "Max" = Maximum.

3.3.3.2 Person Parameter Generation

The person parameters used in Study 2 were resampled across conditions, and across replications. Subscale proficiency was estimated from a multivariate normal (MVN) distribution for both the single and multiple group's simulations. To obtain subscale proficiency estimates for the single groups simulations, a vector of each examinee's true subscale scores were simulated from a distribution, $\boldsymbol{\theta}_j \sim \mathcal{N}_D(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, where $\boldsymbol{\mu}$ is a $1 \times D$ vector of sample means, and $\boldsymbol{\Sigma}$ is a $D \times D$ correlation matrix of the true subscale scores. In Study 2, D for each population was drawn from empirically observed subscale score means. The $d = 1, 2, \ldots, D$ subscale proficiency scores for the $p = 1, 2, \ldots, P$ multiple groups were drawn from a MVN distribution: $\boldsymbol{\theta}_{pj} \sim \mathcal{N}_{D_p}(\boldsymbol{\mu_{pd}}, \boldsymbol{\Sigma})$ where $\boldsymbol{\theta}_{pj}$ and $\boldsymbol{\mu_{pd}}$ are the respective country specific subscale proficiency estimates and subscale mean vector's. Based on the correlations between subscales, $\boldsymbol{\Sigma}$, mean- and standard deviation-vectors for the three- and four-subdomain single group simulations were drawn from one middle performing country. The $d = 1, 2, \ldots, D$ subscale proficiency scores for the $p = 1, 2, \ldots, P$ multiple

Table 3.11
Descriptive Statistics of the Generating Threshold Parameters: d2

D	*j*	*d*	Mean	*SD*	Median	Min	Max	Range
3	40	1	.300	.000	.300	.300	.300	.000
		2	-.297	.000	-.297	-.297	-.297	.000
		3	-.645	.000	-.645	-.645	-.645	.000
	60	1	1.435	.000	1.435	1.435	1.435	.000
		2	-.297	.000	-.297	-.297	-.297	.000
		3	-.372	.473	-.645	-.645	.174	.820
4	40	1	.867	.802	.867	.300	1.435	1.134
		2	-.297	.000	-.297	-.297	-.297	.000
		3	.508	.472	.508	.174	.842	.668
		4	.268	.091	.268	.203	.332	.129
	60	1	1.056	.655	1.435	.300	1.435	1.134
		2	-.297	.000	-.297	-.297	-.297	.000
		3	.500	.000	.500	.500	.500	.000
		4	.667	.353	.633	.332	1.036	.704

Note. D = number of subdomains; j = subscale length; d = specific subdomain; *SD* = Standard Deviation; "Min" = Minimum; "Max" = Maximum.

groups were drawn from a MVN distribution: $\boldsymbol{\theta}_{pj} \sim \mathcal{N}_{D_p}(\boldsymbol{\mu_{pd}}, \boldsymbol{\Sigma})$ where $\boldsymbol{\theta}_{pj}$ and $\boldsymbol{\mu_{pd}}$ are the respective country specific subscale proficiency estimates and subscale mean vector's. As such, subscale proficiency estimates were estimated from $P \times D$ country mean ($\boldsymbol{\mu_{pd}}$) and standard deviation ($\boldsymbol{\sigma_{pd}}$) matrices, and $\boldsymbol{\Sigma}$. To simulate the multiple groups' data, nine observed subscale scores were obtained from the reported country subscale scores on TIMSS 8th grade mathematics. Each score was then converted to a z-score (assuming the TIMSS mean and standard deviation of 500 and 100, respectively). In this study, $\boldsymbol{\mu_{pd}} = \left[\boldsymbol{\mu_{p1}}, \quad \boldsymbol{\mu_{p2}}, \quad \ldots, \quad \boldsymbol{\mu_{pd}}\right]$, $\boldsymbol{\sigma_{pd}} = \left[\boldsymbol{\sigma_{p1}}, \quad \boldsymbol{\sigma_{p2}}, \quad \ldots, \quad \boldsymbol{\sigma_{pd}}\right]$, $\boldsymbol{\mu_{p1}} \neq \boldsymbol{\mu_{p2}} \neq \ldots \neq \boldsymbol{\mu_{pd}}$, and $\boldsymbol{\sigma_{p1}} = \boldsymbol{\sigma_{p2}} = \ldots = \boldsymbol{\sigma_{pd}}$. Based on the correlations between subscales, $\boldsymbol{\Sigma}$; a mean- and standard deviation-matrix for the multiple group simulations was specified as follows[10]:

[10]Since one of the simulation conditions assumed a three subscale test, one subscale mean and deviation column was dropped from $\boldsymbol{\mu_{pd}}$ and $\boldsymbol{\sigma_{pd}}$ in Study 2's multiple groups case.

$$\boldsymbol{\mu_{pd}} = \begin{array}{c} \\ p_1 \\ p_2 \\ p_3 \\ p_4 \\ p_5 \\ p_6 \\ p_7 \\ p_8 \\ p_9 \end{array} \begin{array}{c} \begin{array}{cccc} \theta_{p1} & \theta_{p2} & \theta_{p3} & \theta_{p4} \end{array} \\ \begin{bmatrix} 1.29 & 1.23 & 1.17 & 1.76 \\ 1.01 & 1.12 & 1.12 & 1.00 \\ .90 & 1.13 & 1.07 & .88 \\ -.06 & -.09 & .04 & -.04 \\ .01 & -.08 & -.16 & -.13 \\ .00 & -.25 & -.12 & .09 \\ -1.18 & -1.28 & -.90 & -1.47 \\ -1.32 & -1.06 & -1.36 & -1.43 \\ -1.48 & -1.09 & -1.58 & -1.39 \end{bmatrix} \end{array}, \text{ and } \boldsymbol{\sigma_{pd}} = \begin{array}{c} \\ p_1 \\ p_2 \\ p_3 \\ p_4 \\ p_5 \\ p_6 \\ p_7 \\ p_8 \\ p_9 \end{array} \begin{array}{c} \begin{array}{cccc} \theta_{p1} & \theta_{p2} & \theta_{p3} & \theta_{p4} \end{array} \\ \begin{bmatrix} .82 & .82 & \dots & .82 \\ .85 & .85 & \dots & .85 \\ .97 & .97 & \dots & .97 \\ .75 & .75 & \dots & .75 \\ .88 & .88 & \dots & .88 \\ .88 & .88 & \dots & .88 \\ .80 & .80 & \dots & .80 \\ .87 & .87 & \dots & .87 \\ .86 & .86 & \dots & .86 \end{bmatrix} \end{array}.$$

Because Study 2 emphasizes complex booklet designs and latent regression achievement estimates, I also simulated background data with specified correlations with each sub-dimension of theta. These background variables served as covariates in the latent regression. However, my model was not exactly the same as the TIMSS model. First, TIMSS uses all student background variables which number in the hundreds. To keep the simulation manageable, I only selected 10 background variables for each country's conditioning model. Second, to summarize the largest number of student background variables, TIMSS uses principal components in their model. I did not use principal components in my simulation.

Each domain specific proficiency, θ_d, took into account the influence background variables have on them. From a statistical perspective, the linear relationship was modeled as:

$$\theta_d = \boldsymbol{\beta}_{Q^T}\boldsymbol{Y} + \boldsymbol{\epsilon} \tag{3.2}$$

where $\boldsymbol{Y}$ is the vector of Q background variables; $\boldsymbol{\beta} = \{\beta_1, \dots, \beta_Q\}$ are the regression coefficients that relate the vector of Q^T predictor variables to the latent response; $\boldsymbol{\epsilon} \sim N(0, \boldsymbol{\sigma})$ and $\boldsymbol{\beta}$ and $\boldsymbol{\sigma}$ are estimated from the empirical data set. However, since such data is generated from a known true covariance matrix Σ_Q, sample-generated regression coefficients can also be calculated (Liaw & Netto, 2019 in Rutkowski & Rutkowski, 2019).

First, for each country, I selected 10-categorical background variables from the TIMSS 2015 8th grade dataset. Table 3.12 provides a brief description of the questionnaire items that were extracted from the students questionnaire. The selected variables: (a) were of the nominal and ordinal scales of measurement, (b) had 2, 5, 4, and 8 response options, and (c) were amongst those variables that were used in TIMSS 2015 scaling. Second, I summarized the empirical response proportions on $\mathbf{Y} = 1, 2, \dots, Y$ for P. Third, I then estimated the empirical correlations between $\mathbf{Y}$ (a) amongst

Table 3.12
Selected TIMSS 2015 8th Grade Background Questionnaire Items

Variable code	Scale	Labels	Description
BSBG01	Nominal	2	Sex of student
BSBG04	Ordinal	5	Amount of books in your home
BSBG05	Ordinal	5	Digital information devices
BSBG06E	Nominal	2	Home possession (internet connection)
BSBG06H	Nominal	2	Home possession (country specific)
BSBG06I	Nominal	2	Home possession (country specific)
BSBG06J	Nominal	2	Home possession (country specific)
BSBG07A	Ordinal	8	Highest level of education of mother
BSBG07B	Ordinal	8	Highest level of education of father
BSBG10B	Ordinal	4	Age came to country
BSBG15A	Ordinal	4	Being in school
BSBG16D	Ordinal	4	Stole something from me
BSBM17A	Ordinal	4	Enjoy learning mathematics
BSBM17B	Ordinal	4	Wish may not have to study math
BSBM20B	Ordinal	4	Need math to learn other things
BSBM20D	Ordinal	4	Need math to get the job I want

themselves, and (b) θ_{pd} as observed on the TIMSS 2015 dataset which were calculated using the IDB Analyzer (IEA, 2020) and transformed back to the IRT scale of $N(0,1)$. Table B.1 to B.9 in Appendix B show each country's empirical correlations. Some of the variables selected for Korea and Saudi Arabia were different from those selected for the other seven countries (see Table B.3 and B.6 in Appendix B) because there was some multicollinearity[11]. Upon estimating these response proportions and relationships, each of the 10 selected variables were consistent across the domains. In other words, $\boldsymbol{Y}$ for all group's $\theta_d = \theta_1 + \theta_2 + \cdots + \theta_D$ in Equation (3.2) comprised of the same (country specific) variables that had unique relationships described in Appendix B.

3.3.3.3 Response Pattern Generation

Test responses were generated from a test-level multidimensional IRT model. The components of each model were explained in Chapter 2. The simulation

[11] Two or more explanatory variables in the latent regression model were highly linearly related.

assumes the underlying factor structure is a multidimensional one where the number of subdomains are set according to the simulation condition and that the total score is a linear composite of correlated sub-domains. Based on the generating item- and person-parameters, R package `lsasim` (Matta et al., 2018) was used to generate item responses from a multidimensional item response model (see Chapter 2 for a description of all the model components). Separate test responses were generated for the single and multiple groups simulations. In other words, I simulated separate datasets with 6,000 and 30,000 examinee responses in both Studies 1 and 2 representing the single and-multiple groups. The multiple groups' simulation took into account each groups subdomain proficiency (and latent regression for the the multiple groups simulation in Study 2).

3.3.4 Item Calibration and Scoring

Based on the generated item responses from all simulations, the next step was to estimate population proficiency distributions. In Study 2, I applied both the 2PL- and GPCM-models. Scores in Study 2 were estimated from three IRT models (based on assumed factor structure). These models were: CUIRT, CIRT, and MIRT[12]. I also estimated scores assuming one extra model that resembled the one operationalized on TIMSS 2015[13]. In this case, depending on the number of domains, item parameters estimated from CUIRT were fixed to a three- or four-dimensional MIRT model to obtain subdomain scores. For purposes of simplicity, I will refer to the operationalized model as CUIRT-Op which serves as an operational baseline model for which I compare the other model results. As a reminder, CUIRT-Op differs from MIRT in that CUIRT-Op fixes CUIRT item parameters to the MIRT model when scoring.

In order to estimate scores, I followed two general steps. First, the items were calibrated assuming the CUIRT, CIRT, and MIRT models. Second, estimated and assumed fixed item parameters were used for proficiency estimation. The specified models corresponded with the underlying factor structure that was used in calibration. However, since it may not be possible to directly estimate scores from CUIRT, I fit a (a) CIRT and (b) MIRT model to estimate the subscale scores. Fixing the CUIRT item parameter estimates to a MIRT model to score the test is what I refer to as CUIRT-op. Scores in Study 2 were estimated assuming latent regression, respectively.

For Study 2, proficiency estimates were obtained using a population model to obtain five plausible values which were aggregated to obtain overall country

[12] All of the models have been thoroughly described in the literature review.
[13] For a description of the TIMSS scoring methods, see Section 2.4.2 in Chapter 2.

scores. For more details see Chapter 2. In Study 2, the conditional model was given by Equation (3.2). Population parameter estimates were obtained by averaging the plausible values using Rubin's (1987) methodology. Proficiency estimation was conducted using R package `mirt` (Chalmers, 2012). To complement the proficiency estimates, I also estimated the PRMSE. The PRMSE was used to compare which IRT subscale score estimation method resulted in the most valuable subscale scores. This index was estimated and reported for each subscale score that was calculated based on an IRT method. PRMSE was estimated for each simulated condition. As previously discussed, the PRMSE was estimated equivalent to the subdomain marginal reliability (Haberman, 2008b).

3.3.5 Summary of Simulation Study 2

Table 3.13 provides a summary of Study 2's simulation conditions. The three factors considered in this study (number of subscales, number of items per subscale and subscale correlation[14]) yielded 12 conditions in total. One hundred replications were carried out under each condition.

Table 3.13
Summary of Study 2 Simulation Conditions

Groups	N	D	J	ρ		
Single	6,000	3	40	.45	.75	.95
			60	.45	.75	.95
		4	40	.45	.75	.95
			60	.45	.75	.95
Multiple	30,000	3	40	.45	.75	.95
			60	.45	.75	.95
		4	40	.45	.75	.95
			60	.45	.75	.95

Note. N = Sample size; D = Number of subscales; J = Subscale length; ρ = Subscale correlation.

[14] I did not include a subscale correlation $\rho = 0$, because it would not make sense to have mathematics items on the assessment that are uncorrelated.

3.4 Analysis

The primary concern of the simulation studies was to evaluate the performance of three IRT models in estimating item parameters, subscale proficiency distribution, and subscale value for studies that resemble ILSA, which comprise of single- and multiple-groups samples. In Study 1, the DGP assumed item parameters from a uniform distribution because I did not have access to the SACMEQ III's item parameters. The Study 2 DGP was slightly different in that I: (a) used empirically observed item parameters, (b) generated person proficiency and background questionnaire items, (c) generated item responses based on the item and person parameters (a) and (b). Proficiency was estimated assuming the data fell into four IRT factor structures: CUIRT, CUIRT-Op[15], CIRT, and MIRT. All the aforementioned steps were conducted over 100 replications.

Kolen and Brennan (2014) state that parameter estimates that result from IRT parameter estimation procedures are on different IRT scales. To overcome this, when the IRT model holds, the IRT parameter estimates from different computer runs are linearly related θ scales. In other words, a linear equation can be used to transform IRT parameter estimates to the same scale. For example, Scale J and Scale I as the generating and estimated IRT scales, respectively, that differ by a linear transformation. Then the θ values for the two scales are related as follows:

$$\theta_{Ji} = A\theta_{Ii} + B \tag{3.3}$$

where A and B are constants in the linear equation, θ_{Ji} and θ_{Ii} are values of θ for the individual item- and population-parameter i on Scale G and Scale E. As a result, before conducting the analysis of my simulation studies, I transformed the estimated item- and person-parameters to the scales of their respective generating parameters. To do so, I used the Mean/Mean scaling method (Loyd & Hoover, 1980). This scaling method uses the means of the a-parameter estimates of the parameter estimates in Equation (3.3). According to the Mean/Mean method:

$$A = \frac{\mu(\boldsymbol{a}_I)}{\mu(\boldsymbol{a}_J)}, \text{ or} \tag{3.4}$$

$$= \frac{\sigma(\boldsymbol{\theta}_J)}{\sigma(\boldsymbol{\theta}_I)} \tag{3.5}$$

[15] CUIRT-Op was included only in Study 2 because it resembled the method TIMSS employs when estimating subscale scores

$$B = \mu(\boldsymbol{b}_J) - A\mu(\boldsymbol{b}_I), \text{ or} \tag{3.6}$$
$$= \mu(\boldsymbol{\theta}_J) - A\mu(\boldsymbol{\theta}_I) \tag{3.7}$$

where a_I and a_j are the item descrimination parameters for Scale I and Scale J; b_I and b_j are the item difficulty parameters for Scale I and Scale J; and θ_I and θ_j are the item descrimination parameters for Scale I and Scale J.

The analyses that were done are presented in the section that follows. Section 3.4.1 discusses how the models were evaluated for precision in item- and person-parameter estimation. In the section, I discuss three statistical indices that were used to evaluate parameter recovery.

3.4.1 Evaluation Criteria

The simulation studies were conducted in order to examine which IRT model resulted in the least biased item parameters and population scores. These simulations also aimed to examine which of the models produces the most valuable subscale scores. The studies were conducted over several conditions that represented different ILSA test specifications. As such, I outline how I evaluated my simulation studies. The indices that were used in the evaluation were ideal to examine the accuracy of item parameters and subscale scores over the 100 replications for each condition. In other words, I intended to identify which model (under specific condition) resulted in the psychometrically best (least biased in this study) item- and population-score-parameters. I also intended to identify which of the methods produces the most valuable subscale scores.

This subsection discusses the indices that were used to evaluate item- and population-parameter recoveries from each condition (i.e., compare item- and person-parameter recovery). Three statistics were calculated to examine the performance of the score estimation models in estimating item parameters and country proficiency scores: bias, absolute bias (AB), root mean squared error (RMSE). Researchers have shown that these indices may be used to quantify the accuracy of estimated item- and person-parameters across replications (de la Torre et al., 2011; Dwyer et al., 2006, April; Shin, 2007; Stone et al., 2009; Yao & Boughton, 2007).

Bias, AB, and RMSE can be used examine the difference between estimated (θ_{est}) and true (θ_{true}) item- and person-parameter estimates (θ_{true} are those values specified in the DGP; Debanne, 2000; Kotz & Johnson, 1982; West, 1999). Specifically, (a) bias reveals whether θ_{est} under- or over-estimates θ_{true}; (b) ABS (also known as the error in estimation) highlights the numerical difference between θ_{est} and θ_{true}; and (c) RMSE quantifies the spread of θ_{est} and θ_{true}.

In my studies, bias, AB, and RMSE for each item- or person-parameter were averaged across 100 replications within each condition by the following formulae:

$$Bias_\theta = \frac{1}{r}\sum_{r=1}^{R}(\theta_{est} - \theta_{true})$$

$$AB_\theta = |Bias_\theta| = \frac{1}{r}\sum_{r=1}^{R}|\theta_{est} - \theta_{true}|$$

$$RMSE_\theta = \frac{1}{r}\sqrt{\sum_{r=1}^{R}(\theta_{est} - \theta_{true})^2}$$

3.4.2 Proportional Reduction in Mean Square Error

For each studied condition, I compared the PRMSEs of several indicators of a true subscale score; subscale scores estimated using CUIRT, CIRT and MIRT (as well as CUIRT-op for Simulation study 2). In other words, the PRMSEs obtained from CUIRT ($PRMSE(\theta_d|CU_d)$), CUIRT-op ($PRMSE(\theta_d|CU - op_d)$), CIRT ($PRMSE(\theta_d|C_d)$), and MIRT($PRMSE(\theta_d|M_d)$) subscale scores were compared in order to evaluate which subscale scores contain the most information, and thus reveal which subscale score would be better to report. Extending Haberman's example, it should be seen that if:

1. $PRMSE(\theta_d|CUIRT) > PRMSE(\theta_d|CIRT)$ and $PRMSE(\theta_d|MIRT)$, then subscale score $\theta_d|CUIRT$ has value over $\theta_d|CIRT$ and $\theta_d|MIRT$.
2. $PRMSE(\theta_d|CIRT) > PRMSE(\theta_d|CUIRT)$ and $\theta_d|MIRT$, then subscale score $\theta_d|CIRT$ has value over $\theta_d|CUIRT$ and $\theta_d|MIRT$.
3. $PRMSE(\theta_d|MIRT) > PRMSE(\theta_d|CUIRT)$ and $PRMSE(\theta_d|CIRT)$, then subscale score $\theta_d|MIRT$ has value over $\theta_d|CUIRT$ and $\theta_d|CIRT$.

Values of the PRMSE often lie between 0 and 1. However, Sinharay (2010) noted that the PRMSE can exceed 1 when the disattenuated correlations among the subscores exceed 1. A subscale score estimate with the highest PRMSE provides a more valuable subscale score. I provided a table which allows for the values to be compared.

3.4.3 Model Fit

In both simulation Study 1 and 2, I compared IRT model fit estimates across the studied models using three indices. These were: (a) -2logLikelihood ($-2ll$); (b) Akaike's Information Criterion (AIC, Akaike, 1974); and (c) the Bayesian Information Criterion (BIC, Schwarz, 1978). AIC is defined as:

$$AIC = -2ll + 2k + 2k(k+1)/(n-k-1) \tag{3.8}$$

where k is the number of estimated parameters in the model and n is the number of observations used in the models. BIC is defined as:

$$BIC = -2ll + kln(n) \tag{3.9}$$

where k is the number of estimated parameters in the model and n is the number of observations used in the models. Both the AIC and BIC are information-based criteria that are based on $-2ll$. The AIC and BIC indices differ in that the BIC penalizes model complexity (i.e., having a large number of parameters) more than AIC with a term that depends on the sample size (Oliveri & von Davier, 2011). Smaller values of AIC and BIC (or the negative log-likelihood) indicate better relative model fit (Singer & Willett, 2003).

AIC and BIC appealed to the studies because they may be used to compare fit for non-nested models as long as the models are fit to the same dataset (Singer & Willett, 2003). In addition, I did not use the likelihood ratio tests for the same reason that the models were not nested.

3.5 Summary

This chapter reviewed research methods that were used in the simulation studies, including a description of the DGP and how subscale proficiency was estimated, and how IRT subscale score estimation models were compared. The results of these comparisons helped me answer the research questions posed in Section 1.4.

Chapter 4

Empirical Methods

4.1 Introduction

The simulation studies described in Chapter 3 examined the estimation of item parameters, population scores and subscale score value in various ILSA test designs. This chapter continues by describing methods to provide an illustration of how these IRT models perform using an empirical dataset: the TIMSS eighth grade mathematics test. The TIMSS 2015 dataset was used to demonstrate how the models perform in estimating population and subpopulation subscale score achievement where items were sampled using the matrix sampling booklet designs. This empirical study was conducted in order to validate the findings from the simulation studies. That is, are the simulation findings consistent in the empirical setting. To achieve this, I emulated all TIMSS score estimation methods to the best of my ability with some amendments that I describe subsequently. I did not conduct empirical analysis on the SACMEQ III dataset because item level data and scoring keys are not publicly available. I was also not able to identify specific items that belonged to the HAKT domains on the test, similarly, the test specification is not publicly available.

This chapter provides a description of how the research was carried out. The chapter starts by describing the data that was used in the study. This is followed by a description of the measures that were used. The chapter also specifies the data analysis plan that highlights the steps taken to estimate scores. The chapter finally outlines how the estimated subscale scores were evaluated.

4.2 Data

The study was conducted using achievement data collected by TIMSS 2015. TIMSS ambitiously assessed students in two broad achievement domains: mathematics and science. Altogether, the assessment comprised a total of 494 items (225 and 269 mathematics and science items respectively) that were distributed using rotated booklet designs to make up an examinee's test. The items from both the mathematics- and science-domains were sampled into 28 blocks (14-mathematics and 14-science blocks), with each block containing 12-18 items. These blocks were then distributed into 14 student achievement

booklets (see Table 2.3 in Chapter 2). Each sampled student completed one booklet that was composed of two mathematics- and science-blocks.

To examine model performance in estimating overall- and subscale-scores, I purposefully sampled nine countries whose performance was spread across the proficiency scale. To build the sample, overall country performance in mathematics was used as a determinant of high, medium, and low performing countries. The distribution to which these countries were selected was three of each: high, medium, and low performing countries. This decision was made to keep the analysis manageable.

The specific sample sizes and overall mathematics achievement scale scores of the sampled countries are listed in Table 4.1. All the examinees from the nine sampled countries were included in the empirical example. The resulting total sample size was 64,112 examinees; a sample that adequately resembles (or exceeds) the total sample size of some ILSAs (e.g., SACMEQ, TERCE). This sample size was more than sufficient to conduct most IRT analysis (de la Torre & Hong, 2010).

Table 4.1
Empirical Sample from TIMSS 2015

Country	N	Overall Mathematics Scale Score
Australia	10280	505 (3.1)
Chinese Taipei	5711	599 (2.4)
Italy	4481	494 (2.5)
Korea, Republic of	5309	606 (2.6)
Jordan	7863	386 (2.3)
New Zealand	8142	493 (3.4)
Saudi Arabia	3759	368 (4.6)
Singapore	6116	621 (3.2)
South Africa	12514	372 (4.5)

Note. N = Sample size; Standard errors appear in parentheses

4.3 Measures

The main instrument that was used in the current study was the TIMSS 2015 eighth grade mathematics test. The test specification of the assessment outlined four content domains (algebra, data and chance, geometry, and number) of unequal subscale length. Table 4.2 illustrates the number of domains, total

number of items, and item types for the items that comprised the test. Of the 209 items on the test, 111 were multiple-choice (MC) and 98 were constructed response (CR) items. The MC items on the test were scored correct-incorrect (1, 0 respectively), and the CR items were worth one to two points to allow for partial as well as full credit[1] (Martin, Mullis, & Hooper, 2016). These items were scored using three IRT models: 2PL, 3PL, and GPCM (for full details of the models, please see Chapter 2).

Table 4.2
Assessment Structure: Number of Items and Possible Item Type for Each Domain

Domain	Item type		Total number of items
	MC	CR	
Algebra	34	27	61
Data and Chance	27	14	41
Geometry	22	21	43
Number	28	36	64
Total	111	98	209

Note. MC = multiple choice; CR = Constructed response.

The student background questionnaire used in this study comprises a plethora of information. On it, TIMSS collected demographic data as well as information about students' home environment and school climate for learning. The background questionnaire also collects auxiliary information regarding the contexts of teaching and learning. In addition, TIMSS collects data pertaining to students' self-perception and attitudes towards learning mathematics and science. Due to the abundance of information, TIMSS uses principal components as a means of variable reduction (see Chapter 2 for details).

The conditioning variables that were used were taken from the student background data, and were dummy coded. Students who participated in TIMSS 2015 were administered a context questionnaire with questions related to their home background, school experiences, and attitudes towards mathematics and science (Foy, 2017; p. 59). In this dissertation, I only selected students responses on the general contextual items as well as those directly associated

[1]On TIMSS 2015, the 1-point CR items were scored as correct (1 point) or incorrect (0 points); the 2-point CR items were scored fully correct (2 points), partially correct (1 point) and incorrect (0 points) (Martin, Mullis, & Hooper, 2016). This shows that not all CR items allowed for partial credit.

with mathematics. In other words, I did not include questionnaire responses pertaining to students attitudes towards science and its different subdomains. As a result, a total of 84 conditioning variables were used for each country. These variables were used in the latent regression model in order to optimize the precision of overall achievement and subpopulation differences. The student responses to the background questionnaire that were omitted or not administered were given all zeros on the dummy codes. All of the variables for use were dummy coded using the `dummy_cols` function in the R package `fastDummies` (Kaplan, 2019). After dummy coding the variables, I then used the function `prcomp` in R (R Core Team, 2013) to obtain a dataset of principal components. Since the analyses were conducted separately for each country, the number of conditioning variables were not equal for all the populations[2]. In the end, after dummy coding, at least 279 variables were used[3]. Table 4.3 provides information about the number of primary conditioning variables used for each country as well as the number of other principal components used, and the percentage of variance explained by each country's model. Notable primary conditioning variables, similar to TIMSS 2015, included: gender, language of the test and an optional 'country specific variable'.

Table 4.3
Conditioning Models for Proficiency Estimation

Country	# of PC	%age var.
Australia	340	.62
Chinese Taipei	320	.60
Italy	301	.53
Jordan	353	.54
Korea, Republic of	300	.62
New Zealand	320	.58
Saudi Arabia	279	.47
Singapore	323	.56
South Africa	308	.77

Note. %age var. = Number of principal components; CR = Percentage of variance explained.

[2]On TIMSS 2015, different numbers of principle components were required to account for the recommended percentage of common variance in each country (Martin, Mullis, & Hooper, 2016).

[3]Some response options on several items were not selected in some countries. This resulted in fewer variables after dummy coding.

4.4 Analysis/Models

To estimate population and subpopulation scores, I analyzed my data using the following steps. First, I specified an IRT model; either CUIRT, CIRT, or MIRT. These models outlined the dimensional factor structure of the test (see Chapter 2). Models were specified in R using the `mirt.model` function within mirt: A Multidimensional Item Response Theory Package for the R Environment (Chalmers, 2012). Second, I proceeded on to calibrating the items assuming the factor structure specified in the first step. This was done three times in order to obtain model-specific item parameters that could be used for scoring. Like TIMSS, in this step, non-reached items were treated as not administered. Item parameters were estimated for the entire dataset using the `mirt` function on the mirt package. This function made it possible to fit a variety of IRT models, specify covariates, and specify an implied latent regression. I also used the Metropolis-Hastings Robbins-Monro (MH-RM) estimation algorithm for speed (Cai, 2010a, 2010b). Third, after fixing the estimated item parameters, I proceed to estimate five model-specific[4] plausible values for each country using the `fscores` function in mirt. Consistent with the operational procedures in TIMSS, non-reached items were treated as incorrect. Fourth, using latent regression, I obtained five plausible values for each subscale. Rubin's (1987) multiple imputation average in `mirt` was used to obtain the population scores. The plausible values on the θ metric were linearly transformed to the reporting metric assuming the linear relationship specified by TIMSS 2015. According to Martin, Mullis, and Hooper, 2016, the linear transformation for student i and draw p was:

$$PV_{ip} = A_p + B_p \times \theta_{ip}. \tag{4.1}$$

Similar to TIMSS 2015 (Martin, Mullis, & Hooper, 2016), a different set of transformation constants was used for each of the five plausible values (see Table 4.4). The transformation was implemented in order to place the results from this study on the same scale as the results from the previous TIMSS assessments.

The data that were used in the study were analyzed following steps similar to those undertaken in TIMSS 2015. However, there were several differences with the broader TIMSS study that may result in slightly different estimates.

[4] Two models were fit to obtain scores after calibrating the test using CUIRT. These were CIRT and CUIRT-op. On one hand, CUIRT item parameters were fixed to a CIRT model that had three- or four-uncorrelated dimensions when scoring. On the other hand, CUIRT item parameters were fixed to a three- or four-dimensional MIRT model, CUIRT-op. For CIRT and MIRT, the calibration model was also used to score the test.

Table 4.4
Linear Transformation Constants for the TIMSS 2015 Eighth-Grade Mathematics Assessment

Draw (p)	A_p	B_p
1	507.00	103.10
2	506.97	103.63
3	507.29	102.32
4	506.76	103.14
5	506.56	103.20

Source: Martin et. al (2016).

First, I sampled countries from the entire TIMSS dataset and used that as my overall study sample. Second, I was unable to perfectly mimic the development of conditioning variables, as detailed methods from TIMSS are not publicly available. Third, I did not include all background variables into my conditioning model because I wanted to keep the analysis manageable. This resulted in lower percentage of variance explained by the specified latent regression model. Fourth, I used different software for analysis of my data that may have applied different underlying algorithms to TIMSS' estimation techniques.

4.5 Evaluation Criteria

The empirical study was conducted to examine how the empirical results compare to the simulation findings. To do so, the study was conducted in order to examine how each of the studied models performed in the subscale score estimation. The empirical study also aimed to examine which of the models produces the most valuable subscale scores. Since simulation Study 2 best resembles the empirical study, I compared the results. That is, I examined whether the psychometrically best performing models at (a) score estimation, (b) subscale value, and (c) model fit were the same between the two studies.

This subsection discusses the evaluation criteria that were used to to compare the models. I proceed by describing how I compared the resulting population and subpopulation scores. Then, I describe how I evaluated the estimated subscale score value. Lastly, I describe how I compared IRT model fit estimates across the studied models.

4.5.1 Comparison of Score Estimation Methods

4.5.1.1 Achievement by Population and Subpopulation

Achievement results, overall- and subscale-scores for the content domains, for each population and subpopulation were estimated using IRT scaling techniques similar to those of TIMSS 2015. Scores were estimated from the four subscale score models under comparison—CUIRT, CUIRT-op, CIRT, MIRT. The average scale score for each content domain was examined, together with the difference between overall mathematics achievement and achievement in each subscale. These analyses were done for every population (i.e., country) as well as subpopulation (boys vs girls, number of books at home). These analyses were only done for the empirical study and were intended to give a picture of whether we would expect to empirically observe any differences in score magnitude.

4.5.1.2 Analysis of Variance (ANOVA)

To explore whether there were differences in the model specific subpopulation score standard errors, I conducted analysis of variance (ANOVA) analyses. Standard error (SE) was treated as a dependent variable (DV) and the model as the independent variable (IV). In a their study that compared the performance of three IRT models, Erdemir and Atar (2020) conducted an ANOVA on repeated-measures data to examine whether there was a significant difference among the mean errors calculated by the estimation models. Post-hoc tests using the Bonferroni correction were then conducted in order to make pairwise comparisons. Since the SE's were also specific to a domain and gender, I run a cluster robust ANOVA using the `anova_test` function in the R package `rstatix` (Kassambara, 2020).

Statistically, the SE's (from the empirical study) and the biases/ABS/RMSE (from simulation Study 2) may provide different information. That is, SE is a measure of precision or efficiency of the estimator that does not require knowledge of the true value θ (Morris et al., 2019). In this study, a model that produced scores with smaller SEs was considered better than models with larger SE. Bias/ABS/RMSE[5] may be considered to be measures that quantify whether reported estimates target θ. I then observed whether the same methods that showed the least Bias/ABS/RMSE in the simulation studies resulted in lower SE's on the empirical study[6].

[5]For the equations used in the computation of Bias, ABS or RMSE, see Section 3.4.1 in Chapter 3.

[6]Recall that the SE's were not estimated from the simulation Studies.

4.5.2 Subscale Score Value

The PRMSE was used to evaluate the performance of the subscale scores estimated from each model. Values of the PRMSE often lie between 0 and 1. However, Sinharay, 2010 noted that the PRMSE can exceed 1 when the disattenuated correlations among the subscores exceed 1. A subscale score estimate with the highest PRMSE provides a more valuable subscale score. I provided a table that allowed for the values to be compared. In this study, a subscale score estimate with the highest PRMSE provides valuable subscale score (see Section Section 3.4.2 in Chapter 3 for comparisons). In Chapter 7, I concluded by observing whether the same model produced more valuable subscale scores in simulation Study 2 and the empirical study. This enabled me to determine whether the model performs consistently.

4.5.3 Model Fit

In the empirical study, I evaluated IRT model fit using the three indices. These were: (a) $-2ll$; (b) Akaike's Information Criterion (AIC, Akaike, 1974); and (c) the Bayesian Information Criterion (BIC, Schwarz, 1978). Both the AIC and BIC are information-based criteria that are based on $-2ll$. Smaller values of AIC and BIC (or the negative log-likelihood) indicate better relative model fit (Singer & Willett, 2003).

AIC and BIC appealed to the studies because they may be used to compare fit for non-nested models as long as the models are fit to the same dataset (Singer & Willett, 2003). In addition, I did not use the likelihood ratio tests for the same reason that the models were not nested. In Chapter 7, I conclude by observing whether the same model fits the data better than the other studied models in simulation Study 2 and the empirical study. This enabled me to determine whether the models performed consistently.

4.6 Summary

This empirical study was conducted in order to examine how well the three IRT models performed in estimating item parameters, population score estimates, and ultimately produced valuable subscale scores. As such, the chapter reviewed the empirical research methods that were used in this study. This chapter included a description of the participants and the instruments that were used in the study. Furthermore, Chapter 4 outlined the procedures that were taken in data analysis and how scores were compared. The results of these comparisons helped me answer the research question regarding how the three IRT methods

of subscale score estimation compare in providing score estimates given an evaluation of their precision (as shown in the simulation study).

Chapter 5

Simulation Results

5.1 Introduction

Two simulation studies were designed to answer research questions presented in Section 1.4. Simulation studies 1 and 2 were designed to resemble SACMEQ and TIMSS data, respectively. The difference between the two simulation studies is that Study 1 does not employ matrix sampled test booklets whilst Study 2 does. Three design characteristics (i.e., number of subscales, correlation between subscales, subscale length) were manipulated to create conditions of various characteristics. Results were observed over 100 replications.

In this chapter, results from the two simulation studies are presented. For both Study 1 and Study 2, the results presented in Section 5.2 answer the first research question presented in Chapter 1 of this dissertation. Section 5.2 reports which item parameter estimation method produces the psychometrically best item parameter estimates. Furthermore, Section 5.3 focuses on score recovery that answers the second research question. In other words, I report which of the typically available IRT methods provides the psychometrically-best population subscale scores. Section 5.4 responds to the third research question. I use the section to identify which of the subscale score estimation methods produce the most valuable subscale scores. Finally, Section 5.5 presents the model fit statistics to itentify the model that fit the data the best.

5.2 Item Parameter Recovery

Tables 3.5 and 3.13 show the simulation conditions for Studies 1 and 2. The conditions that I studied were: (a) number of subdomains; (b) number of items per factor; and (c) subscale correlation. The performance of three subscale score estimation models were compared across the simulated test conditions. In what follows, I will describe the item parameter recovery for all the simulation studies. To investigate all the studied methods' subscale score recovery (RQ1), I computed bias, ABS, and RMSE. Sections 5.2.1 and 5.2.2 present Study 1's single- and multiple-groups' results, respectively. Sections 5.2.3 and 5.2.4 present Study 2's single- and multiple-groups' results, respectively.

To illustrate the patterns of results, I present separate plots that show the different evaluation criteria. In other words, I present the bias, ABS and RMSE

of the item parameters that were estimated in all of the studied conditions. Each plot presents the results by (a) subdomain length and (b) number of items per subdomain. The plots are arranged in such a way that the each panel in one row shows the results from the three separate models that were studied (i.e., CUIRT, CIRT, MIRT). Each of the three rows in the plots represents a different correlation (i.e., .45, .75, .95).

The x-axis on each panel shows the number of items, whereas the y-axis shows the evaluation criteria (e.g., bias, ABS or RMSE). The points in each panel represent a specific item. The points also include error bars that show the standard deviation of each estimated parameter across replications. Ideally, the best estimates are those with estimates that are closest to 0 on all figures.

The results for the average bias, ABS and RMSE of the item difficulty parameter are presented in Tables 5.1 and 5.2 for all of Study 1's conditions described in Table 3.5. The tables summarise the evaluation criteria for both, the single- and multiple groups studies, respectively, for the Rasch calibration model. Tables 5.3 and 5.5 report the the average bias, ABS and RMSE of the item discrimination for all of Study 2's single- and multiple-groups conditions, respectively. Tables 5.4 and 5.6 report the the average bias, ABS and RMSE of the item difficulty for all of Study 2's single- and multiple-groups conditions, respectively. These averages enabled me to quantify each model's item parameter recovery by (a) subscale correlation, (b) subscale length, and (c) number of subscales. The lower the value, the better the model.

5.2.1 Item Parameter Recovery for Study 1: Single Groups

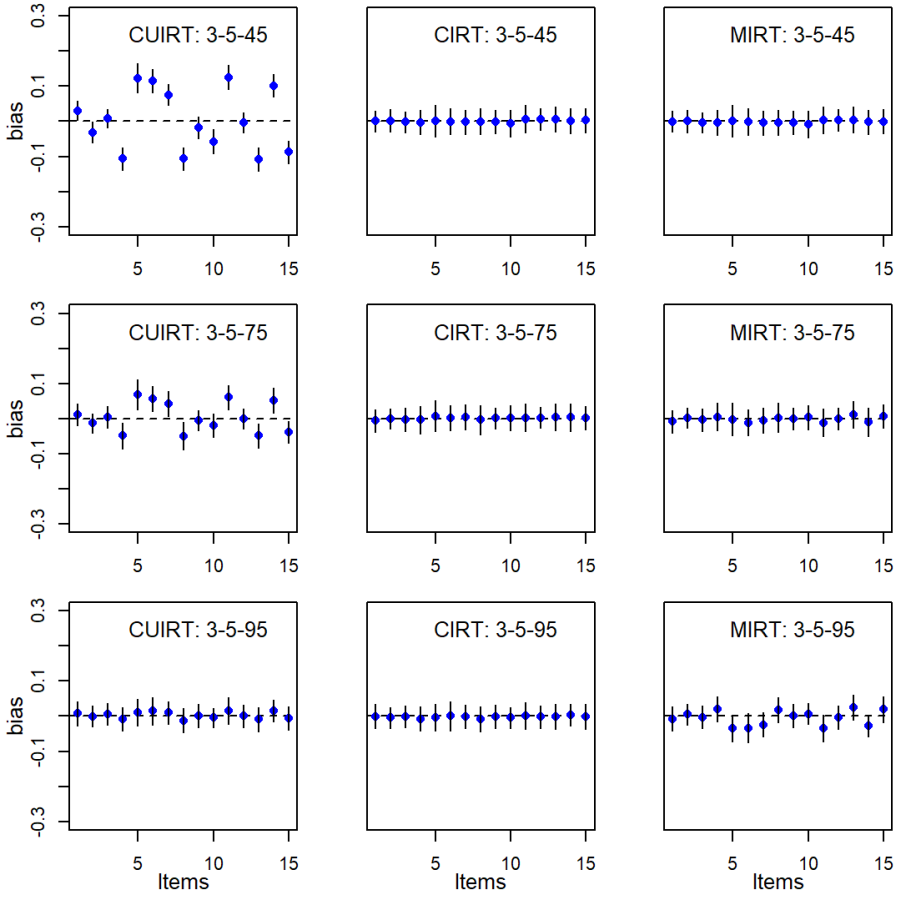
Figures 5.1 and 5.6 plot the bias of the item difficulty, b for all the items on a test over different subtest length and subscale correlation. Each point in the figures corresponds to an item. The whiskers on the points are $\pm 1SD$ over the replications.

Sections 5.2.1.1 and 5.2.1.2 provide the results of the item parameter recovery in Study 1's single group test conditions. Under these sections, I will provide the results for the single- and multiple-groups simulation conditions. An emphasis is placed on the simulation design factors. Sections 5.2.1.1 and 5.2.1.2 outline how each of the studied models perform at different subscale correlations and lengths, respectively.

5.2.1.1 Subscale Correlation

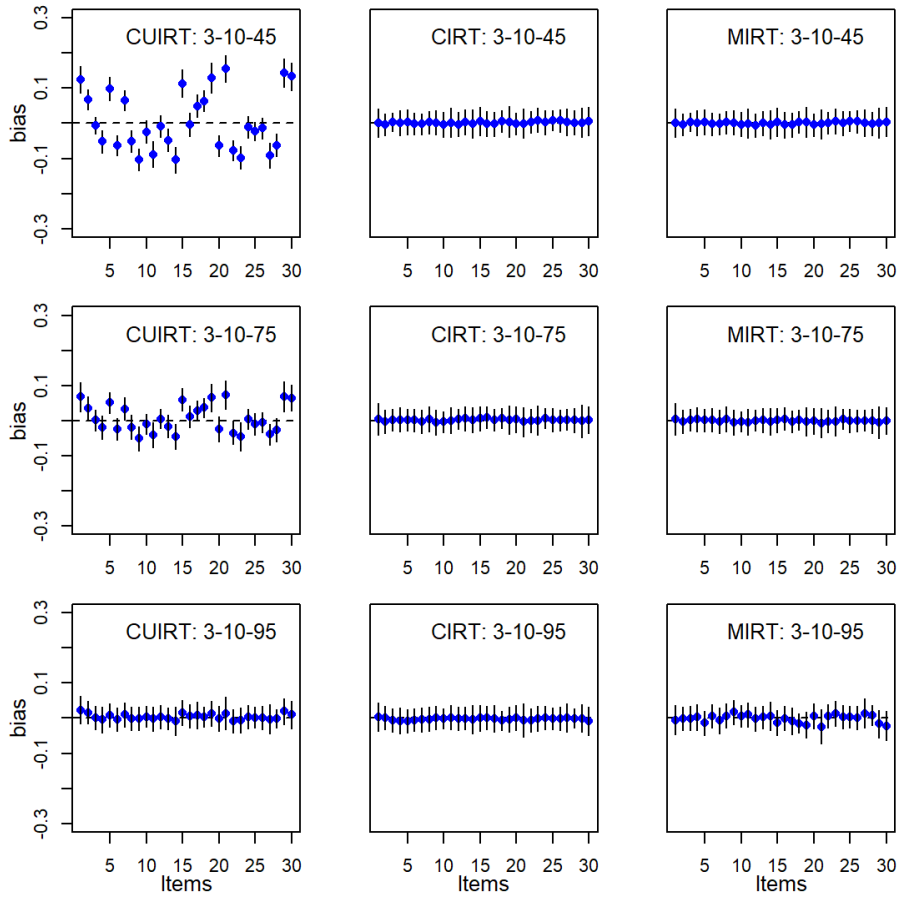
Three-Subdomain Tests.
Figure 5.1 to 5.3 show that CUIRT produced the most biased item parameters regardless of test length where correlations were .45 and .75. In contrast,

Figure 5.1
Item Difficulty Bias for the 3 Domain, 5 Items per Domain Tests: Single Groups



CIRT produced the least biased results across conditions. Although MIRT resulted in little bias when correlations were .45 and .75, correlations of .95 produced increased bias. In addition, the figures showed that as subscale correlation increased to .95, CUIRT produced biases that were comparable to CIRT. These results were expected since, at high correlations, the test is close to being unidimensional. The ABS and RMSE reported in Table 5.1 supported these findings. That is, at respective test lengths, (a) CIRT results

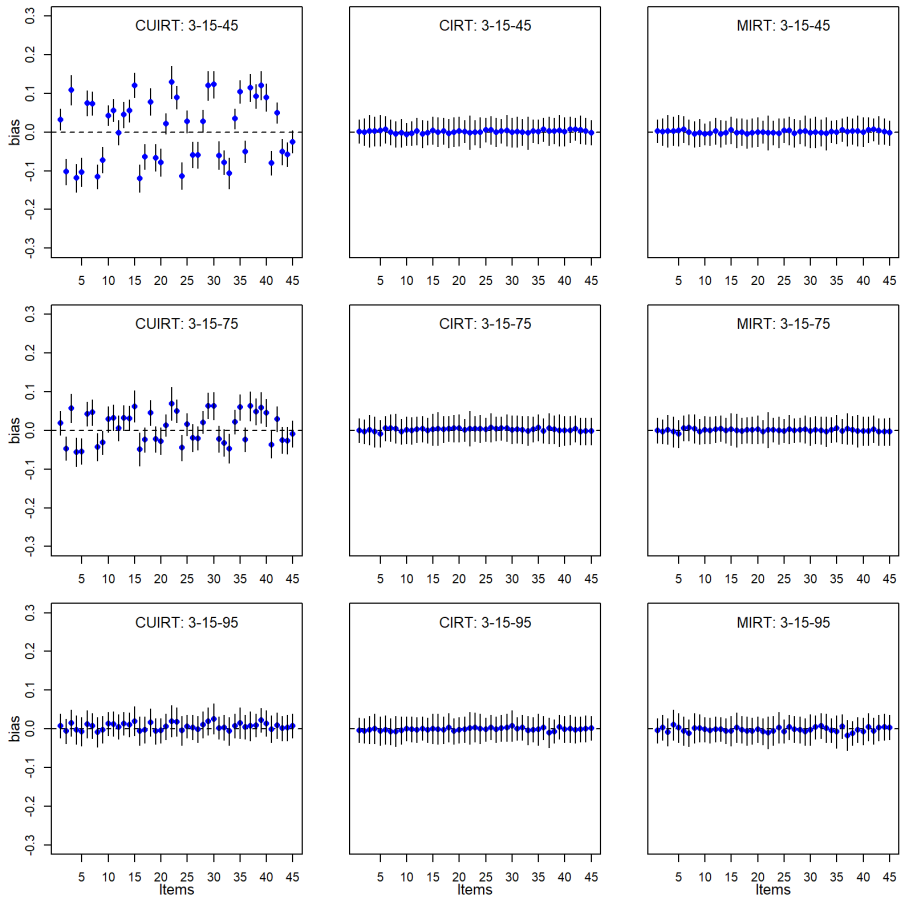
Figure 5.2
Item Difficulty Bias for the 3 Domain, 10 Items per Domain Tests: Single Groups



were comparable across correlations, (b) CUIRT produced better results when subscale correlation was .95, and (c) MIRT reported larger ABS and RMSE where subscale correlation was .95. However, Figure 5.1 to 5.2 show that CUIRT (at low correlations) and MIRT (at high correlations) are slightly mirrored. That is, the positive bias on an item when using CUIRT matched with a negative bias on MIRT, and vice-versa.

Figure 5.3

Item Difficulty Bias for the 3 Domain, 15 Items per Domain Tests: Single Groups



Five-Subdomain Tests.

Figure 5.4 to 5.6[1] show that CUIRT produced the most biased item parameters regardless of test length where correlations were .45 and .75. In contrast, CIRT produced the least biased results across conditions. Although MIRT resulted in little bias when correlations were .45 and .75, correlations of .95 produced

[1]The results presented by the ABS (see Figure E.1 to E.6), and RMSE (see Figure E.7 to E.12) plots in Appendix E plots supported the results presented by the bias plots.

increased bias. In addition, the figures showed that as subscale correlation increased to .95, CUIRT produced biases that were comparable to CIRT. These results were expected since, at high correlations, the test is close to being unidimensional.

Figure 5.4

Item Difficulty Bias for the 5 Domain, 5 Items per Domain Tests: Single Groups

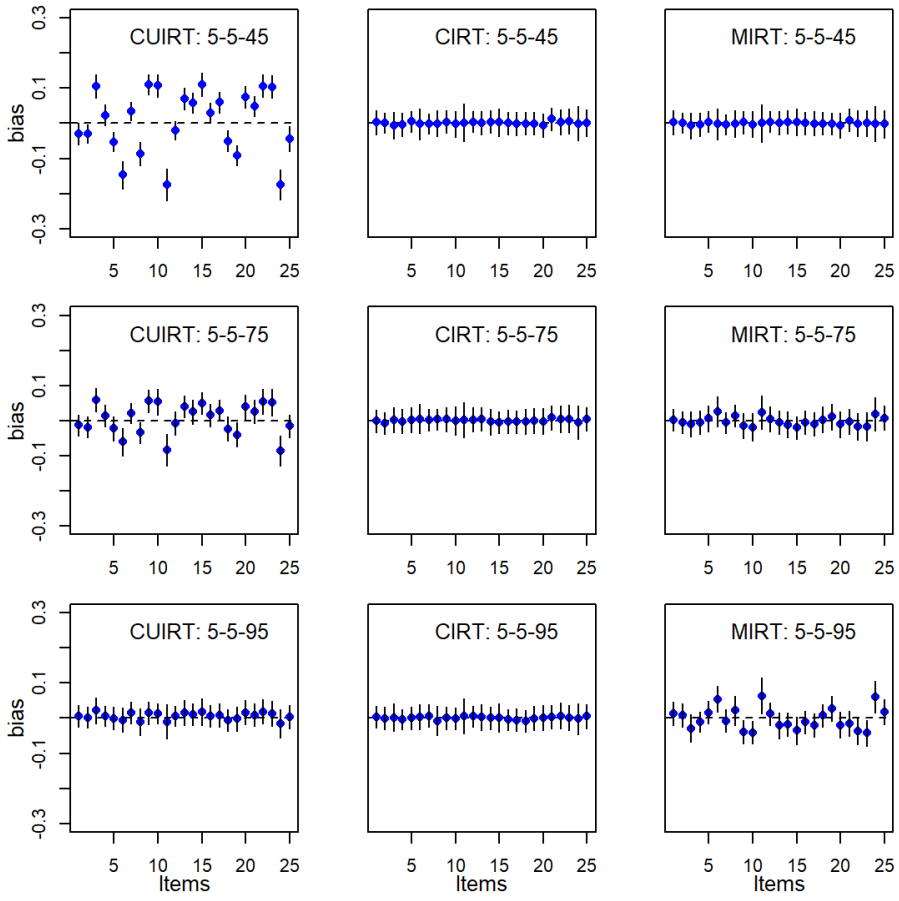
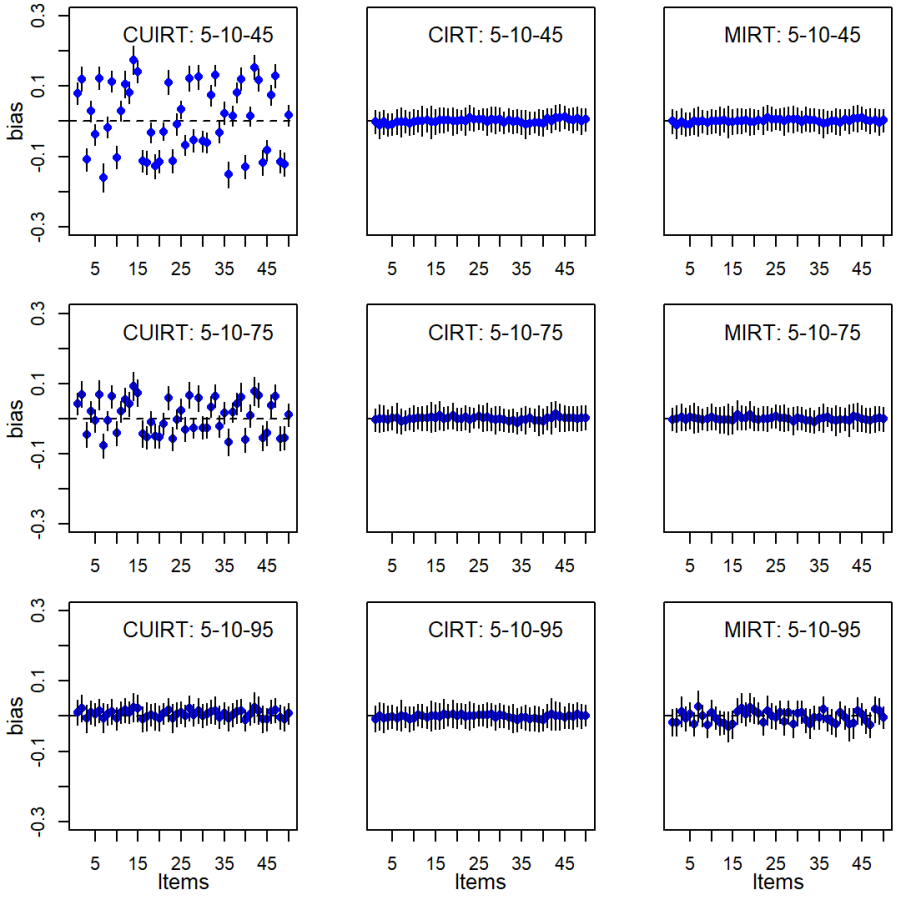


Figure 5.5
Item Difficulty Bias for the 5 Domain, 10 Items per Domain Tests: Single Groups

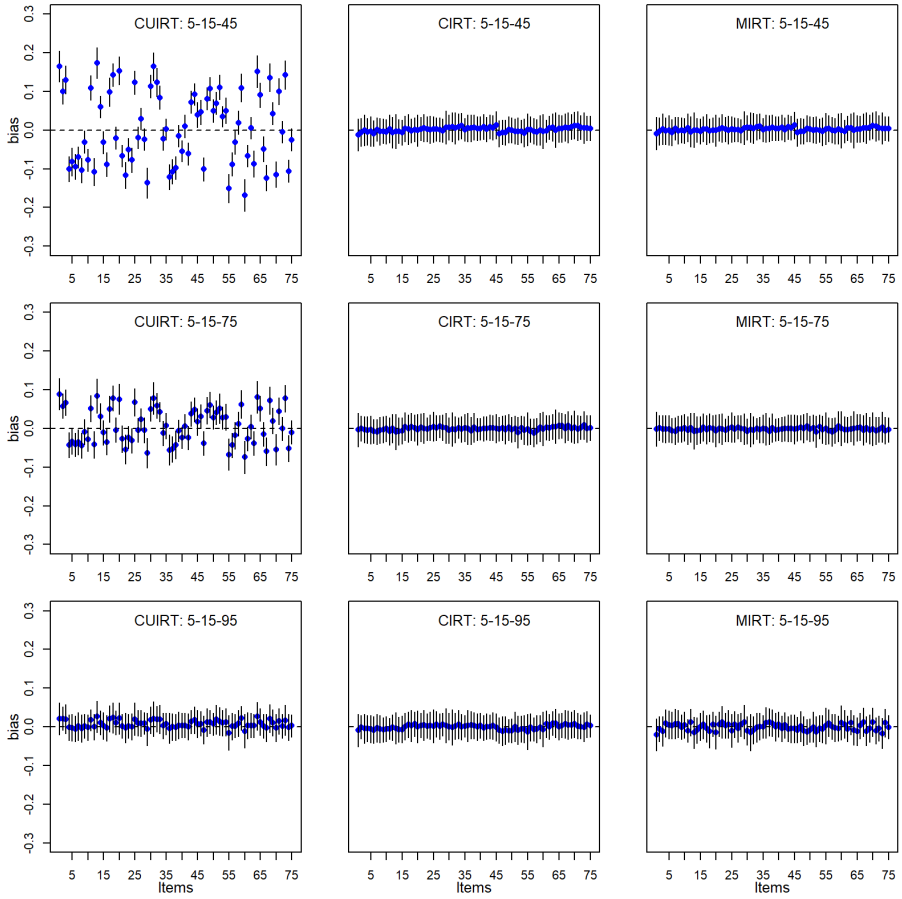


5.2.1.2 Subscale Length

Three-Subdomain Tests.
Figure 5.1 to 5.3 showed that MIRT item bias reduced as the number otems per subdomain increased from 5 to 15 items. Inspection of Figure 5.1 to 5.3 showed that the performance of CUIRT and CIRT was comparable as subscale length increased. Table 5.1 supported these findings. In contrast, Table 5.1 showed

Figure 5.6

Item Difficulty Bias for the 5 Domain, 15 Items per Domain Tests: Single Groups



that CUIRT had more bias for respective subscale correlations as the number of items per subdomain increased. The noted differences were to the third decimal place. For example, on the three subdomain tests where $J = 5, 10, 15$ and subscale correlation was .45, the corresponding biases were .004, .005, and .006. In this case, an increase in subscale length corresponded to an increase in the number of biased items. However, Table 5.1 ABS and RMSE for CUIRT were constant as subscale length increased. MIRT showed a decrease in ABS and

RMSE on the .95 correlation test conditions as subscale length increased (see Table 5.1). According to the results presented in Table 5.1, CUIRT generally showed biases further from 0 regardless of subscale correlation compared to CIRT and MIRT. However, MIRT had the highest bias and RMSE compared to CUIRT and CIRT on 5 subdomain item tests where subscale correlation was .95.

Five-Subdomain Tests.
Inspection of Figure 5.4 to 5.6 showed that the performance of CIRT was comparable as subscale length increased. Table 5.1 supported these findings by also showing that CIRT had comparable ABS and RMSE as the number of items-per-subscale increased. In contrast, Table 5.1 showed that as the number of items increased, CUIRT and MIRT resulted in higher average bias. However, when tests had subscale correlations of .95, MIRT bias was the lowest on the test conditions with 15 items-per-subdomain. In addition, Table 5.1 ABS and RMSE for CIRT were comparable as subscale length increased. MIRT showed a decrease in ABS and RMSE on the .95 correlation test conditions as subscale length increased (see Table 5.1). According to the results presented in Table 5.1, CUIRT generally showed biases further from 0 regardless of subscale correlation compared to CIRT and MIRT. However, MIRT had the highest bias and RMSE compared to CUIRT and CIRT on 5 subdomain item tests where subscale correlation was .95.

Table 5.1
Study 1 Item Difficulty Bias, ABS and RMSE: Single Group

				Bias			ABS			RMSE		
N	D	J	ρ	CUIRT	CIRT	MIRT	CUIRT	CIRT	MIRT	CUIRT	CIRT	MIRT
6,000	3	5	.45	.004	.001	−.001	.077	.028	.028	.083	.035	.035
			.75	.005	.001	−.001	.043	.029	.029	.050	.036	.036
			.95	.002	−.002	−.005	.028	.028	.032	.034	.034	.040
		10	.45	.005	.002	.000	.075	.028	.028	.081	.035	.035
			.75	.007	.002	.000	.042	.028	.028	.049	.035	.035
			.95	.004	−.002	−.001	.028	.028	.029	.035	.035	.037
		15	.45	.006	.001	.000	.077	.028	.028	.084	.035	.035
			.75	.008	.002	.000	.043	.028	.028	.051	.035	.035
			.95	.006	−.001	−.002	.028	.028	.028	.035	.034	.035
	5	5	.45	.005	.001	.000	.079	.028	.028	.085	.035	.035
			.75	.005	.001	−.002	.044	.029	.030	.052	.036	.038
			.95	.006	.001	−.002	.028	.028	.037	.035	.035	.046
		10	.45	.005	.001	.001	.090	.028	.028	.096	.035	.035
			.75	.007	.000	.000	.049	.028	.028	.056	.035	.036
			.95	.006	−.001	−.003	.029	.028	.031	.036	.035	.039
		15	.45	.006	.001	.002	.085	.028	.028	.091	.035	.035
			.75	.007	−.001	−.002	.046	.028	.028	.054	.035	.036
			.95	.007	−.001	−.001	.029	.028	.029	.036	.035	.037

Note. N = sample size; D = number of subscales; J = subscale length; ρ = subscale correlation.

5.2.2 Item Parameter Recovery for Study 1: Multiple Groups

Sections 5.2.2.1 and 5.2.2.2 provide the results of the item parameter recovery in Study 1's multiple group test conditions. An emphasis is placed on the simulation design factors. Under these sections, I will provide the results for the single- and multiple-groups simulation conditions. Sections 5.2.2.1 and 5.2.2.2 outline how each of the studied models perform at different subscale correlations and lengths, respectively.

5.2.2.1 Subscale Correlation

Figure 5.7 to 5.12 plot the bias of the item difficulty, b for all the items on a test over different subtest length and subscale correlation. Each point in the figures corresponds to an item. The whiskers on the points are $\pm 1SD$ over the replications

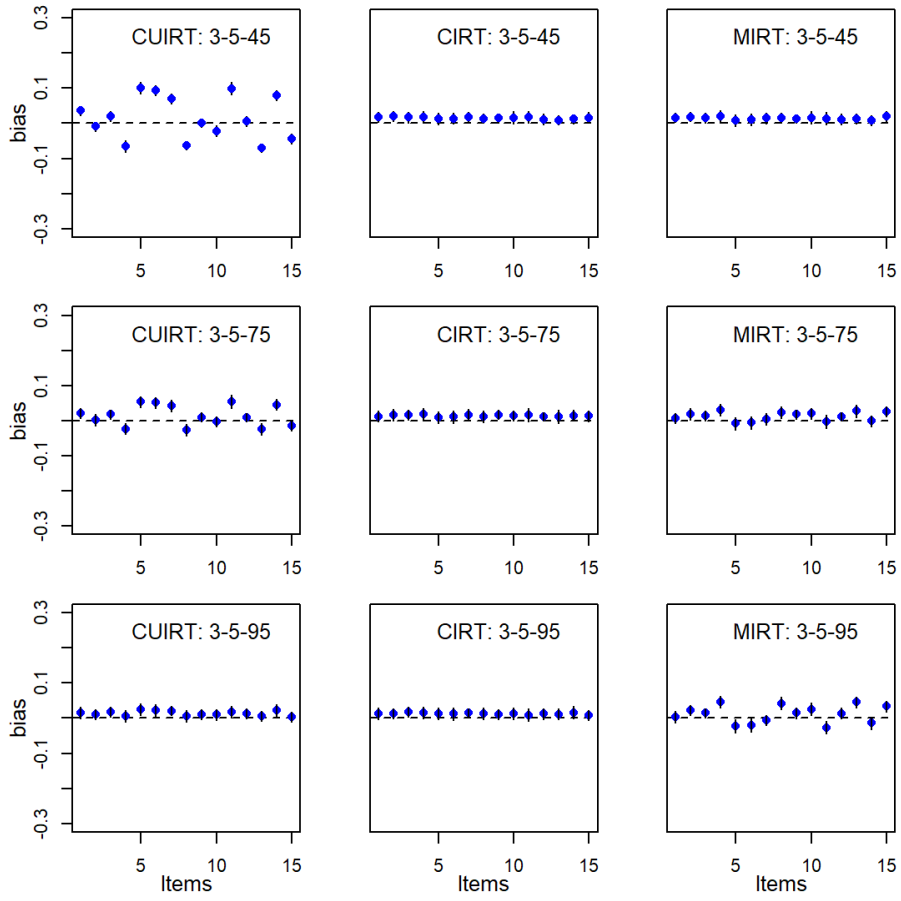
Figure 5.7 to 5.12[2] plot the bias of the item difficulty, b for all the items on a test over different subtest length and subscale correlation. In general, the patterns observed for the multiple groups example were similar to those of the single groups case, regardless of the number of subscales. Of the three models, the figures show that CUIRT produced the most biased item parameters regardless of test length where correlations were .45 and .75. In contrast, CIRT produced the least biased results across conditions. Although MIRT resulted in little bias when correlations were .45 and .75, correlations of .95 produced increased bias. In addition, the figures showed that as subscale correlation increased to .95, CUIRT produced biases that were comparable to CIRT.

5.2.2.2 Subscale Length

Three-Subdomain Tests.
Inspection of Figure 5.7 to 5.9 showed that the performance of all of the models was generally comparable (to the third decimal) as subscale length increased. However, Table 5.2 showed that MIRT had the lowest average bias on all of the 5 items-per-subdomain test conditions regardless of subscale correlation compared to CUIRT and CIRT. As the subdomain length increased to 10- and 15-item subdomain tests, the studied models were generally comparable across all of the specified correlations. The ABS and RMSE reported in Table 5.2 showed that CUIRT had the highest ABS and RMSE compared to CIRT and MIRT on the 5-item-per-subdomain tests where subscale correlation was .45.

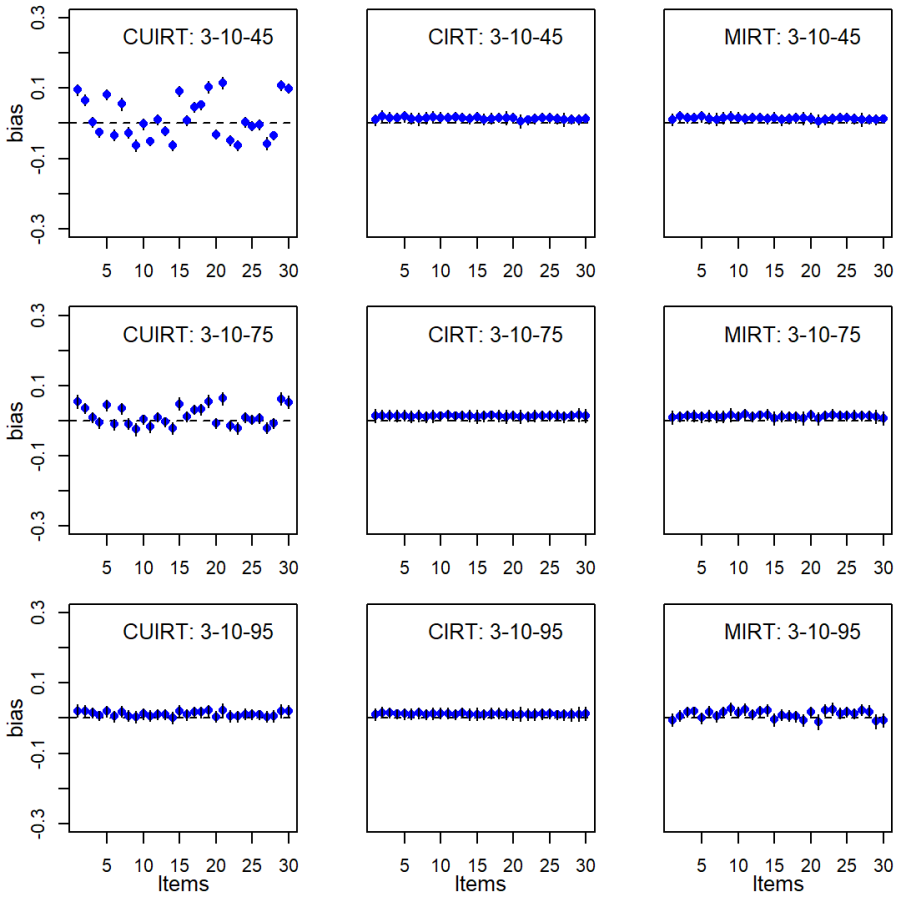
[2]The results presented by the ABS (see Figure F.1 to F.6) and RMSE (see Figure F.7 to F.12) plots in Appendix F confirmed the results presented by the bias plots.

Figure 5.7
Item Difficulty Bias for the 3 Domain, 5 Items per Domain Tests: Multiple Groups



In addition, MIRT had the highest ABS and RMSE compared to CUIRT and CIRT on the 5-item-per-subdomain tests where subscale correlation was .95. But then, as subscale length increased (to 10 and 15 items), all of the models were comparable. The ABS and RMSE reported in Table 5.2 generally showed that CIRT was not sensitive to an increase in subscale length.

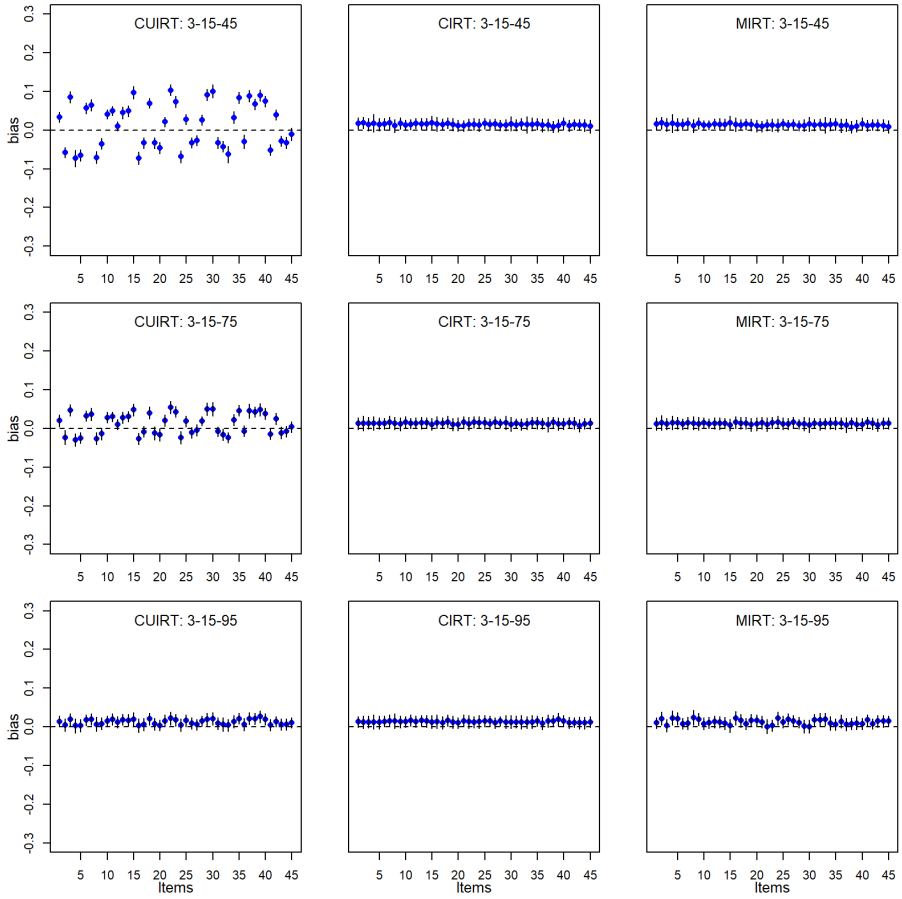
Figure 5.8
Item Difficulty Bias for the 3 Domain, 10 Items per Domain Tests: Multiple Groups



Five-Subdomain Tests.
Inspection of Figure 5.7 to 5.9 showed that the performance of all of the models was generally comparable (to the third decimal) as subscale length increased. However, Table 5.2 showed that CUIRT generally had the lowest average bias (to the third decimal) regardless of test length. CIRT and MIRT showed comparable average biases and these were higher than the values reported by CUIRT. The ABS and RMSE reported in Table 5.2 showed that CUIRT had the highest ABS and RMSE compared to CIRT and MIRT where

Figure 5.9

Item Difficulty Bias for the 3 Domain, 15 Items per Domain Tests: Multiple Groups



subscale correlation was .45. CUIRT average ABS and RMSE increased as subscale length increased. In addition, MIRT had the highest ABS and RMSE compared to CUIRT and CIRT on the 5-item-per-subdomain tests where subscale correlation was .95. However, as subscale length increased to 15 items, all of the models were comparable. The ABS and RMSE reported in Table 5.2 generally showed that CIRT was not sensitive to an increase in subscale length.

Figure 5.10

Item Difficulty Bias for the 5 Domain, 5 Items per Domain Tests: Multiple Groups

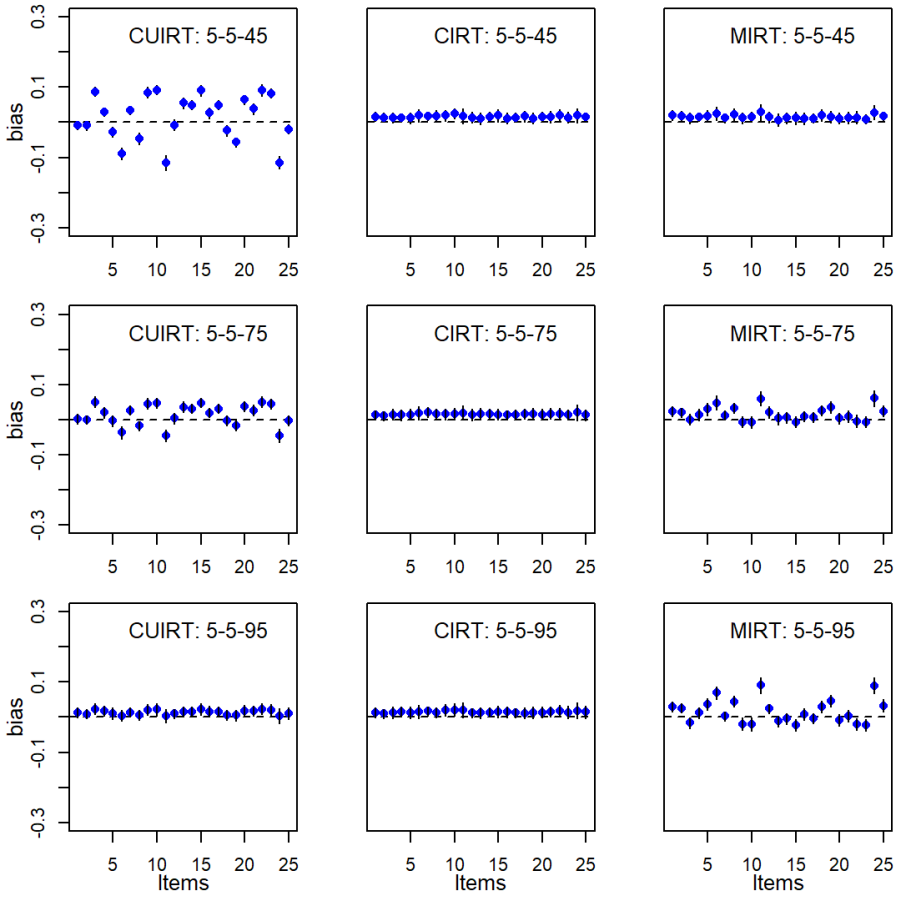


Figure 5.11
Item Difficulty Bias for the 5 Domain, 10 Items per Domain Tests: Multiple Groups

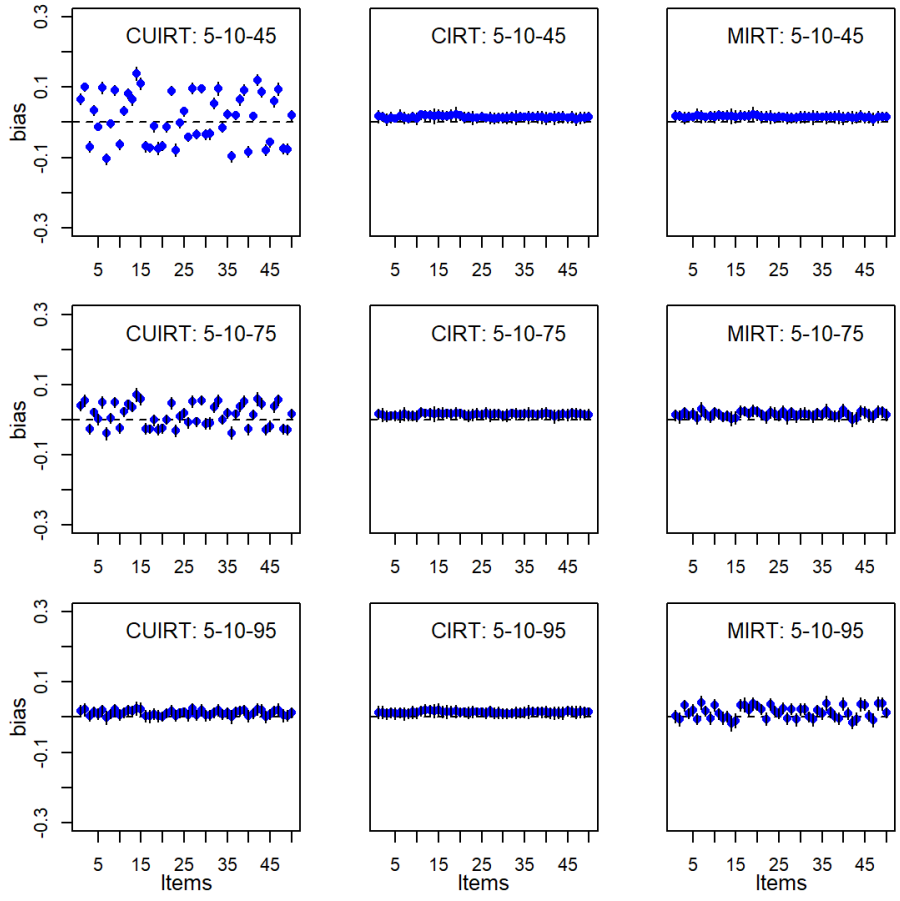


Figure 5.12

Item Difficulty Bias for the 5 Domain, 15 Items per Domain Tests: Multiple Groups

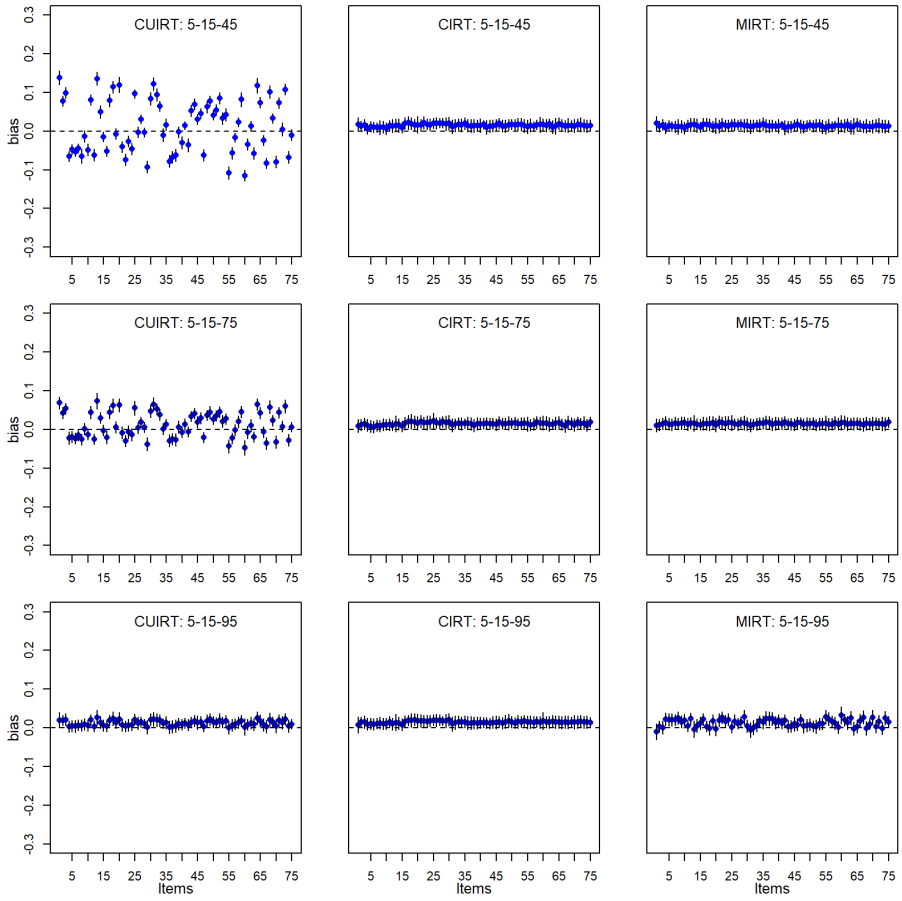


Table 5.2

Study 1 Item Difficulty Bias, ABS and RMSE: Multiple Groups

				Bias			ABS			RMSE		
N	D	J	ρ	CUIRT	CIRT	MIRT	CUIRT	CIRT	MIRT	CUIRT	CIRT	MIRT
30,000	3	5	.45	.015	.015	.014	.053	.018	.017	.055	.021	.021
			.75	.015	.014	.012	.029	.017	.019	.032	.021	.023
			.95	.014	.013	.011	.018	.017	.026	.021	.020	.029
		10	.45	.014	.014	.014	.051	.017	.017	.053	.021	.021
			.75	.013	.014	.013	.027	.017	.017	.030	.021	.020
			.95	.012	.012	.011	.017	.016	.018	.020	.020	.022
		15	.45	.014	.014	.014	.054	.018	.017	.056	.021	.021
			.75	.013	.013	.013	.028	.017	.016	.031	.020	.020
			.95	.013	.013	.012	.017	.017	.017	.021	.020	.021
	5	5	.45	.014	.016	.016	.056	.018	.019	.058	.022	.023
			.75	.014	.016	.017	.030	.018	.023	.033	.022	.027
			.95	.014	.015	.016	.018	.018	.030	.021	.022	.034
		10	.45	.012	.015	.016	.063	.018	.018	.065	.022	.022
			.75	.013	.015	.015	.032	.018	.019	.035	.022	.023
			.95	.012	.014	.014	.017	.017	.023	.020	.021	.026
		15	.45	.012	.015	.014	.059	.018	.017	.062	.022	.021
			.75	.012	.015	.015	.031	.018	.018	.034	.022	.022
			.95	.012	.015	.013	.017	.018	.018	.020	.022	.022

Note. N = sample size; D = number of subscales; J = subscale length; ρ = subscale correlation.

5.2.3 Item Parameter Recovery for Study 2: Single groups

Figure 5.13 to 5.19 plot the bias of the item- (a) discrimination, a, and (b) difficulty, b, parameters for all the items on a test over different subtest length and subscale correlation. Each point in the figures corresponds to an item. The whiskers on the points are $\pm 1SD$ over the replications.

5.2.3.1 Subscale Correlation

Item Discrimination.

Three-Subdomain Tests.

Figures 5.13 and 5.14 plot the biases of each items discrimination parameter, a, across all of Study 2's 3-subdomain, single groups conditions. Of the three models, the figures show that CUIRT produced the most biased item parameters regardless of test length where correlations were .45. In contrast, CIRT and MIRT produced the least biased results across conditions. Although CUIRT resulted in larger bias when correlations were .45, correlations of .95 produced decreased bias. Based on Figures 5.13 and 5.14[3], CUIRT showed less bias of a estimates over 100 replications regardless of test length where correlations were .75 and .95.

Four-Subdomain Tests.

Figures 5.15 and 5.16 show the biases of each items discrimination parameter, a, across all of Study 2's 4-subdomain, single groups conditions. The figures showed the same pattern of results that was reported on the 3-subdomain tests[4].

Item Difficulty.

Three-Subdomain Tests.

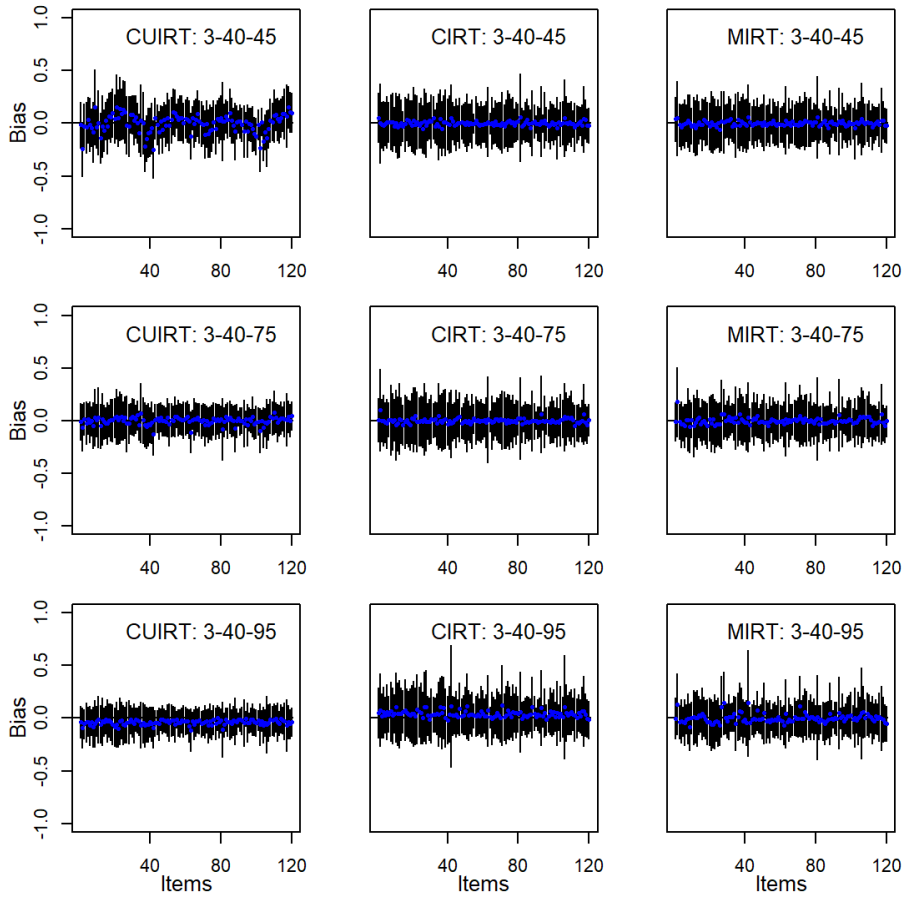
Figures 5.17 and 5.18[5] show the biases of the item difficulty, b, parameter across all conditions for Study 2's single groups simulations. Of the three models, the figures show that CUIRT produced the most biased item parameters regardless of test length where correlations were .45. In contrast, CIRT and MIRT produced the least biased results across conditions. That is, the average

[3]The ABS and RMSE plots (see Figure G.1 to G.4, and Figure G.17 to G.20 in Appendix G) generally showed the same inclination as the bias plots.

[4]The ABS and RMSE plots (see Figures G.3 and G.4, and Figures G.19 and G.20 in Appendix G) generally showed the same inclination as the bias plots.

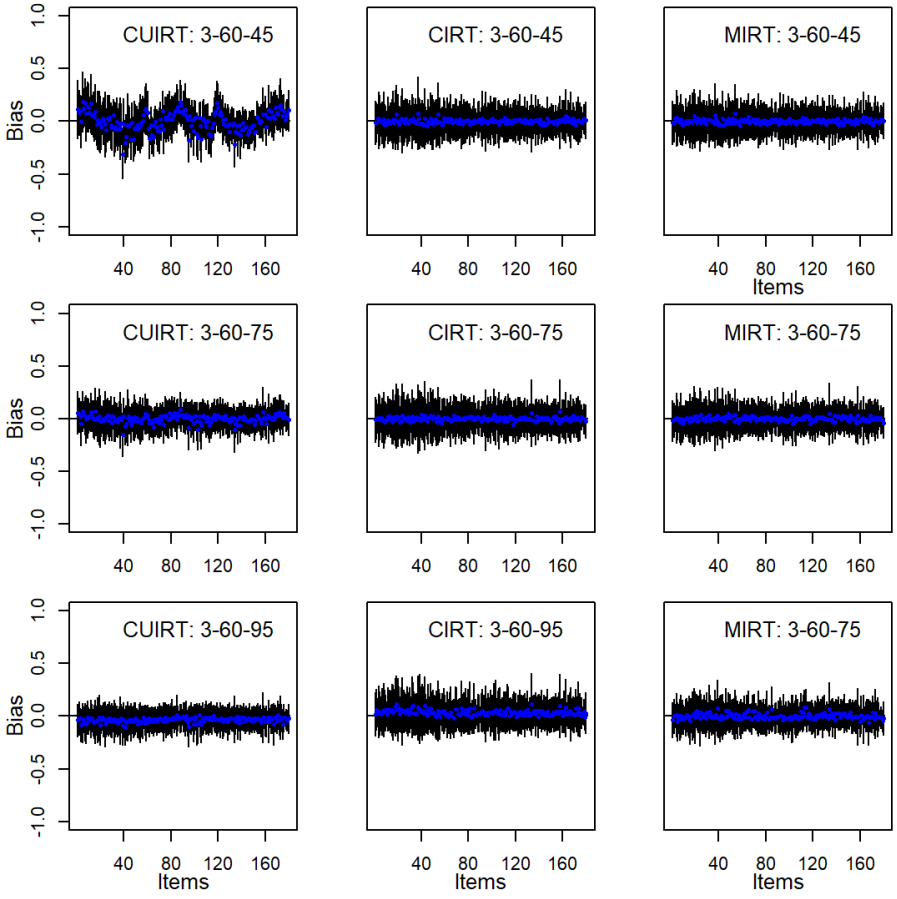
[5]The ABS and RMSE plots (see Figure G.5 to G.6, and Figure G.21 to G.22 in Appendix G) echo the results displayed by the b-parameter bias plots shown in Figure 5.17 to 5.18.

Figure 5.13
Bias of a-Parameter for the 3 Domain, 40 Items per Domain Tests: Single Groups



biases for the items were closer to 0; as opposed to CUIRT whose estimates exhibited more instances where item biases were further from 0. Similar to the results presented in Sections 5.2.1 and 5.2.2, CUIRT resulted in less bias when correlations were .95 than in conditions with weaker correlation. Although MIRT resulted in less bias when correlations were .45 and .75, correlations of .95 showed slightly more biased *b*-parameters. What sticks out in the single groups is that several items have some outliers. That is, few items seemed to

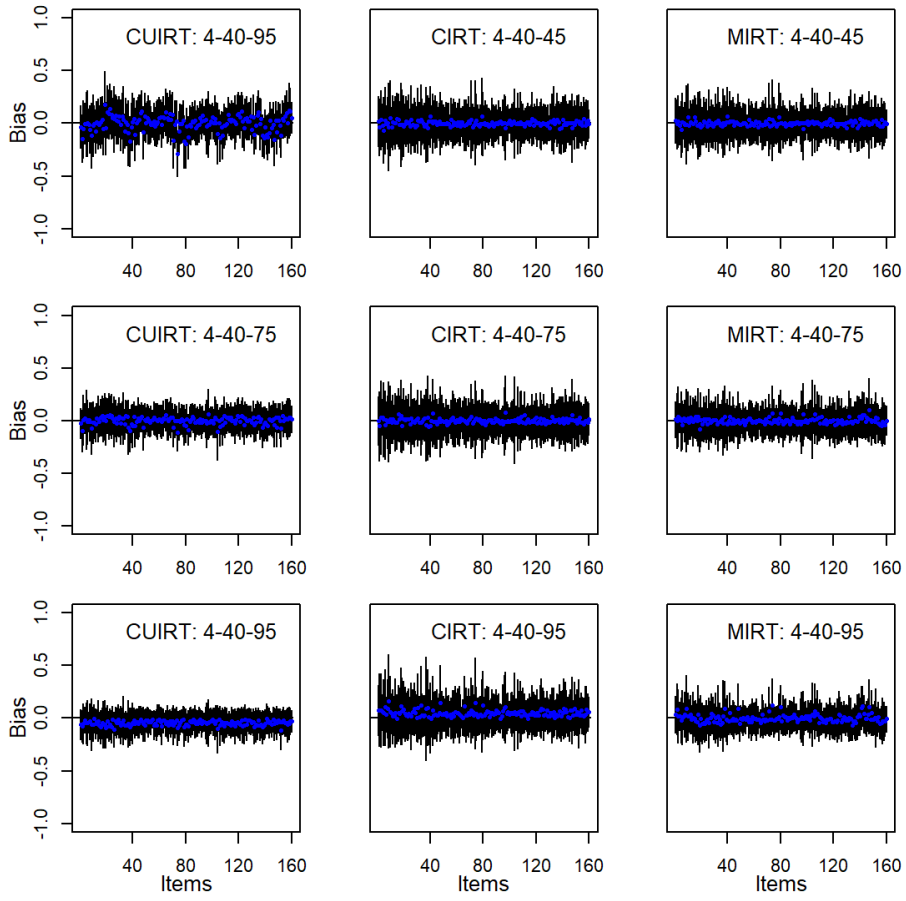
Figure 5.14
Bias of a-Parameter for the 3 Domain, 60 Items per Domain Tests: Single Groups



show larger variability in the item difficulty estimates than the other items. In the case of these items, it is probable that there were fewer candidates that were exposed to the items; a problem which may have been compounded if there was less variation in item responses because the items were too difficult or easy in some replications.

Figure 5.15

Bias of a-Parameter for the 4 Domain, 40 Items per Domain Tests: Single Groups

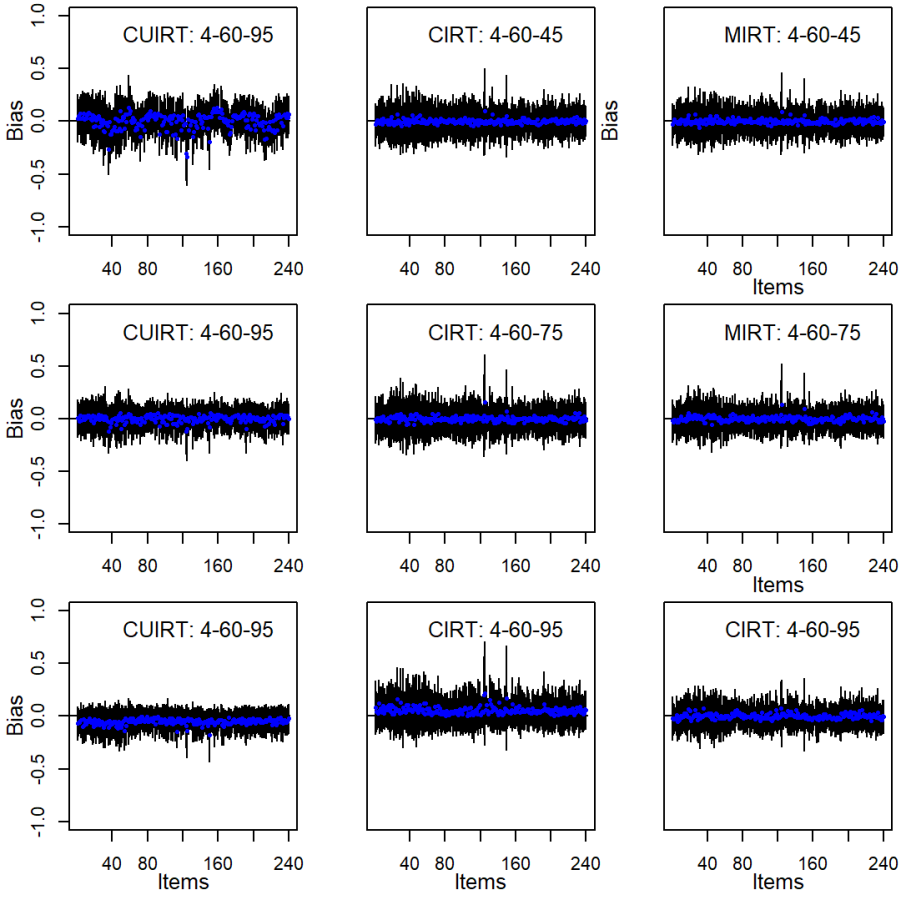


Four-Subdomain Tests.

Figures 5.17 and 5.20[6] show the biases of the item difficulty, b, parameter across all conditions for Study 2's single groups simulations. The figures showed the same pattern of results that was reported on the 3-subdomain tests.

[6]The ABS and RMSE plots (see Figure G.7 to G.8, and Figure G.23 to G.24 in Appendix G) echo the results displayed by the b-parameter bias plots shown in Figure 5.19 to 5.20.

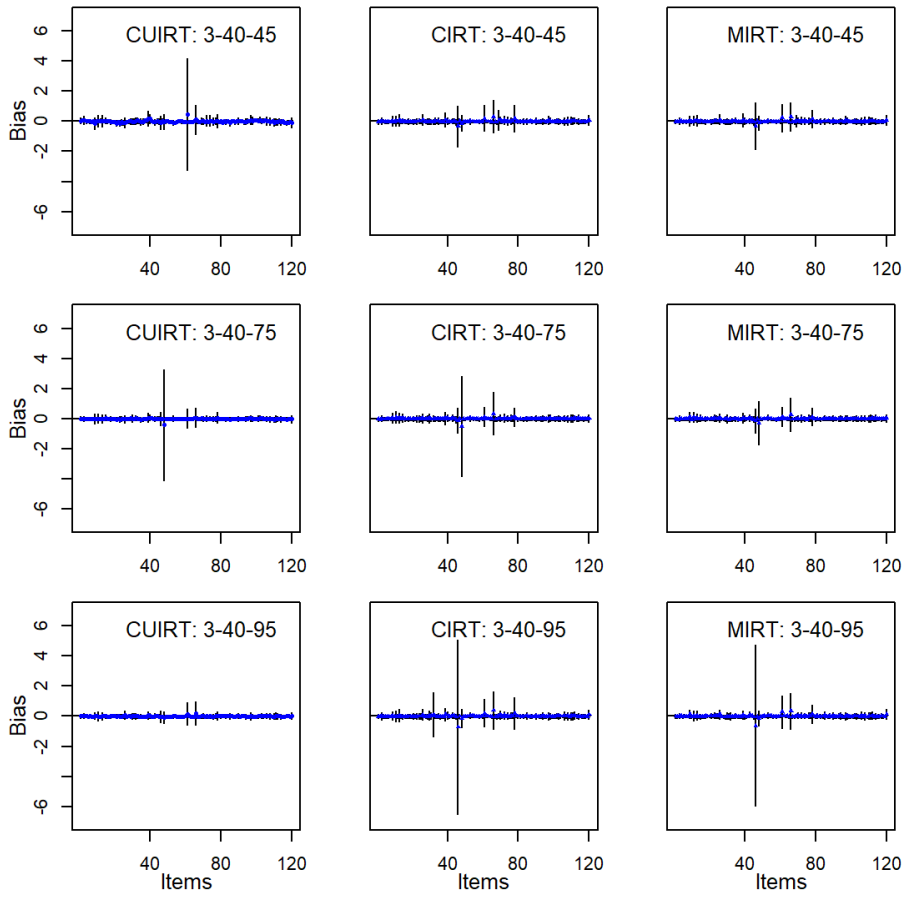
Figure 5.16
Bias of a-Parameter for the 4 Domain, 60 Items per Domain Tests: Single Groups



5.2.3.2 Subscale Length

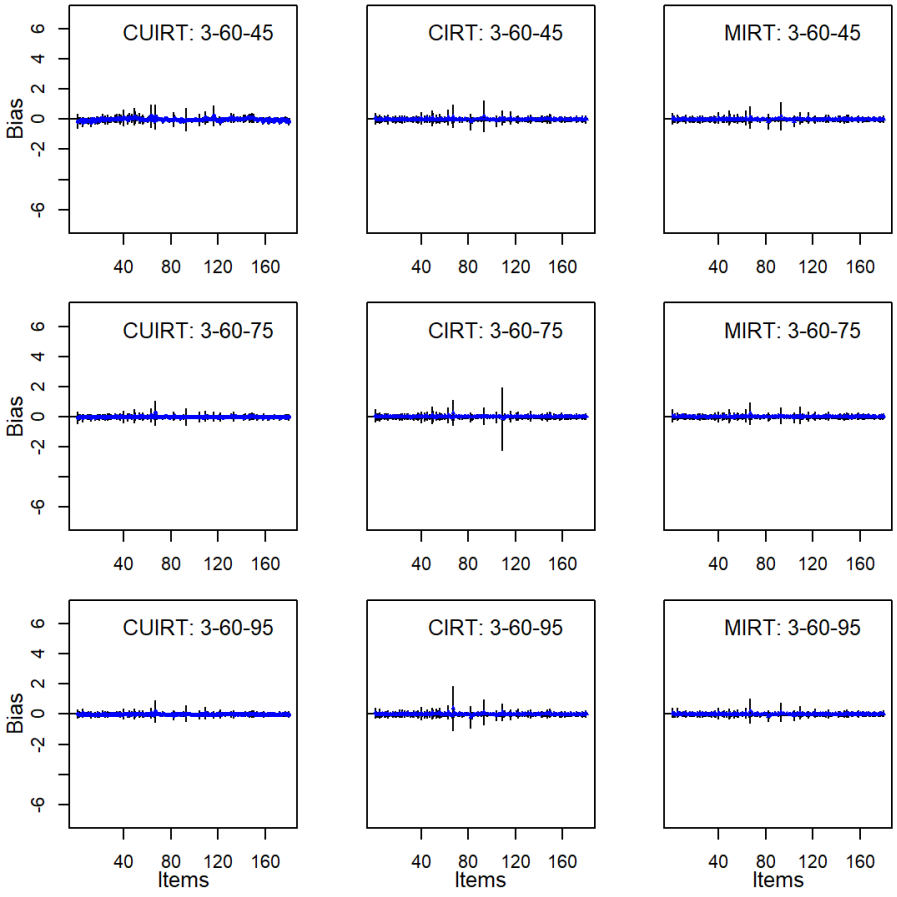
Item Discrimination and Difficulty.
The results for the item (a) discriminations, and (b) difficulty parameters will be presented together. This is because the results generally followed the same patterns and trends.

Figure 5.17
Bias of b-Parameter for the 3 Domain, 40 Items per Domain Tests: Single Groups



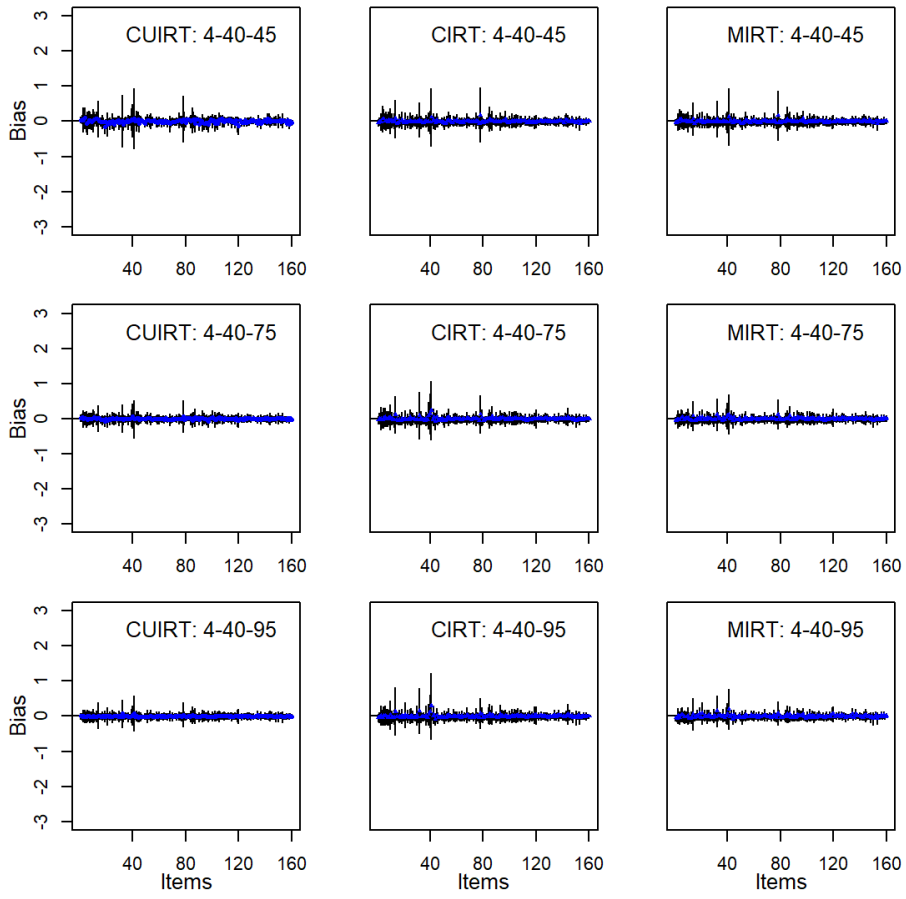
Three-Subdomain Tests.
The results presented in Figure 5.13 to 5.14 (item discrimination) and Figure 5.17 to 5.18 (item difficulty) showed that subscale length did not impact on the performance of the studied models. However, the average bias/ABS/RMSE of the item discrimination and difficulty parameters presented in Tables 5.3 and 5.4 suggested some trends. That is, the average biases for all of the studied models decreased as subscale length increased. In addition, Tables 5.3 and 5.4 showed that the average ABS and RMSE of CUIRT and

Figure 5.18
Bias of b-Parameter for the 3 Domain, 60 Items per Domain Tests: Single Groups



CIRT reduced as subscale length increased. MIRT reported average ABS' that were not sensitive to an increase in subscale length (see Tables 5.3 and 5.4). In contrast, MIRT RMSE's were lower on the 60-item subscale tests than the tests that had 40-item subscales. CIRT generally reported the lowest bias and ABS.

Figure 5.19
Bias of b-Parameter for the 4 Domain, 40 Items per Domain Tests: Single Groups



Four-Subdomain Tests.
Figures 5.15 and 5.16, and Figure 5.19 to 5.20 did not show differences with respect to the sensitivity of the models to test length. However, the trends that were reported on Tables 5.3 and 5.4 for the 4 subdomain tests were similar to those presented for the 3 subdomain tests. One key difference was that the 4-subdomain test conditions reported lower bias, ABS, and RMSE compared to the 3-subdomain test condition results.

Figure 5.20

Bias of b-Parameter for the 4 Domain, 60 Items per Domain Tests: Single Groups

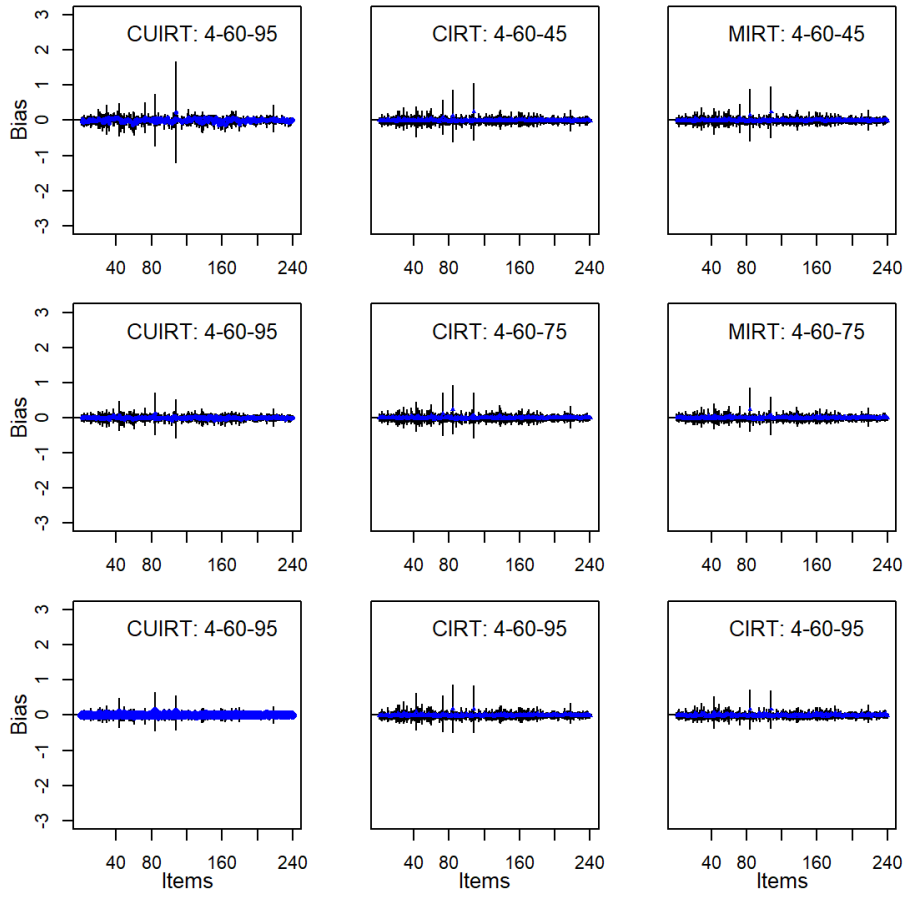


Table 5.3
Study 2 Item Discrimination Summary: Single Groups

				Bias			ABS			RMSE		
Par	D	J	ρ	CUIRT	CIRT	MIRT	CUIRT	CIRT	MIRT	CUIRT	CIRT	MIRT
a	3	40	.45	−.47 (.04)	−.21 (.02)	−.20 (.03)	.47 (.04)	.25 (.01)	.26 (.02)	.49 (.04)	.29 (.01)	.29 (.02)
			.75	−.33 (.02)	−.21 (.01)	−.19 (.04)	.33 (.02)	.25 (.01)	.26 (.02)	.36 (.02)	.28 (.01)	.30 (.02)
			.95	−.24 (.02)	−.21 (.01)	−.16 (.07)	.25 (.01)	.25 (.01)	.28 (.03)	.28 (.01)	.29 (.01)	.32 (.02)
		60	.45	−.47 (.04)	−.21 (.01)	−.21 (.03)	.47 (.04)	.24 (.01)	.25 (.02)	.48 (.04)	.27 (.01)	.28 (.02)
			.75	−.33 (.02)	−.21 (.01)	−.18 (.07)	.33 (.02)	.24 (.01)	.30 (.02)	.35 (.02)	.27 (.01)	.32 (.02)
			.95	−.24 (.02)	−.21 (.01)	−.16 (.10)	.25 (.01)	.24 (.01)	.33 (.02)	.27 (.01)	.27 (.01)	.36 (.02)
	4	40	.45	−.52 (.03)	−.22 (.01)	−.22 (.03)	.52 (.03)	.26 (.01)	.26 (.02)	.53 (.03)	.29 (.01)	.29 (.02)
			.75	−.36 (.02)	−.22 (.01)	−.18 (.05)	.36 (.02)	.26 (.01)	.26 (.02)	.38 (.02)	.29 (.01)	.29 (.02)
			.95	−.26 (.01)	−.22 (.01)	−.14 (.07)	.27 (.01)	.25 (.01)	.27 (.02)	.29 (.01)	.29 (.01)	.31 (.02)
		60	.45	−.54 (.04)	−.24 (.01)	−.24 (.04)	.54 (.04)	.26 (.01)	.27 (.03)	.55 (.04)	.29 (.01)	.30 (.03)
			.75	−.38 (.02)	−.23 (.01)	−.21 (.08)	.38 (.02)	.25 (.01)	.29 (.04)	.39 (.02)	.29 (.01)	.32 (.04)
			.95	−.27 (.02)	−.23 (.01)	−.19 (.10)	.28 (.01)	.25 (.01)	.32 (.04)	.30 (.01)	.28 (.01)	.34 (.04)

Note. Par = item parameter; D = number of subscales; J = subscale length; ρ = subscale correlation; a = item discrimination; b = item difficulty; the values in parentheses represent the standard deviations across replications.

Table 5.4
Study 2 Item Difficulty Summary: Single Groups

				Bias			ABS			RMSE		
Par	D	J	ρ	CUIRT	CIRT	MIRT	CUIRT	CIRT	MIRT	CUIRT	CIRT	MIRT
b	3	40	.45	.54 (.37)	.30 (.12)	.32 (.15)	.70 (.17)	.41 (.05)	.43 (.08)	.82 (.59)	.49 (.09)	.49 (.11)
			.75	.37 (.17)	.29 (.12)	.32 (.17)	.49 (.07)	.41 (.05)	.43 (.10)	.58 (.31)	.50 (.21)	.49 (.13)
			.95	.30 (.12)	.29 (.13)	.31 (.21)	.41 (.05)	.42 (.05)	.43 (.13)	.45 (.06)	.55 (.53)	.53 (.43)
		60	.45	.58 (.34)	.31 (.11)	.34 (.17)	.69 (.21)	.39 (.05)	.43 (.11)	.75 (.25)	.44 (.07)	.47 (.12)
			.75	.40 (.15)	.30 (.10)	.34 (.22)	.48 (.08)	.39 (.05)	.43 (.16)	.52 (.10)	.45 (.09)	.47 (.17)
			.95	.32 (.11)	.31 (.11)	.34 (.26)	.40 (.06)	.39 (.05)	.44 (.19)	.43 (.06)	.44 (.08)	.47 (.20)
	4	40	.45	.46 (.37)	.20 (.11)	.24 (.18)	.62 (.19)	.34 (.05)	.37 (.10)	.69 (.24)	.39 (.06)	.41 (.11)
			.75	.29 (.17)	.21 (.12)	.25 (.22)	.42 (.08)	.34 (.05)	.39 (.14)	.46 (.09)	.39 (.07)	.42 (.15)
			.95	.21 (.12)	.20 (.12)	.25 (.24)	.34 (.05)	.34 (.05)	.40 (.14)	.37 (.05)	.39 (.07)	.43 (.15)
		60	.45	.44 (.30)	.20 (.10)	.24 (.18)	.59 (.15)	.32 (.04)	.37 (.10)	.64 (.19)	.36 (.05)	.40 (.11)
			.75	.28 (.15)	.19 (.10)	.26 (.24)	.40 (.07)	.31 (.04)	.40 (.15)	.43 (.07)	.35 (.04)	.43 (.15)
			.95	.21 (.11)	.18 (.10)	.27 (.28)	.33 (.05)	.31 (.04)	.44 (.16)	.36 (.05)	.35 (.04)	.46 (.17)

Note. Par = item parameter; D = number of subscales; J = subscale length; ρ = subscale correlation; a = item discrimination; b = item difficulty; the values in parentheses represent the standard deviations across replications.

5.2.4 Item Parameter Recovery for Study 2: Multiple Groups

Figure 5.21 to 5.27 plot the bias of the item- (a) discrimination, a, and (b) difficulty, b, parameters for all the items on a test over different subtest length and subscale correlation. Each point in the figures corresponds to an item. The whiskers on the points are $\pm 1SD$ over the replications.

5.2.4.1 Subscale Correlation

Item Discrimination.
Three-Subdomain Tests.
Figures 5.21 and 5.22[7] show the biases of each item's discrimination parameter, a, across all of Study 2's multiple groups conditions. The panels in Figures 5.21 and 5.22 showed the same trends as the results presented in the single groups simulation study (presented in Section 5.2.3.1). Of the three models, the figures show that CUIRT produced the most biased item parameters regardless of test length where correlations were .45. In contrast, CIRT and MIRT produced the least biased results across conditions. Although CUIRT resulted in larger bias when correlations were .45, correlations of .95 produced lower bias. Based on Figures 5.13 and 5.14, CUIRT showed less bias of a estimates over 100 replications regardless of test length where correlations were .75 and .95. However, Figure 5.21 to 5.22 show that of the three models, CIRT produced more biased item discrimination parameters regardless of test length where correlations were .95.

Four-Subdomain Tests.
Figures 5.23 and 5.24[8] show the biases of each items discrimination parameter, a, across all of Study 2's 4-subdomain, single groups conditions. The figures showed the same pattern of results that was reported on the 3-subdomain tests[9].

Item Difficulty.
Three-Subdomain Tests.
Figure 5.25 to 5.26[10] show the biases of each items difficulty parameter, b, across all of Study 2's multiple groups conditions. The results show

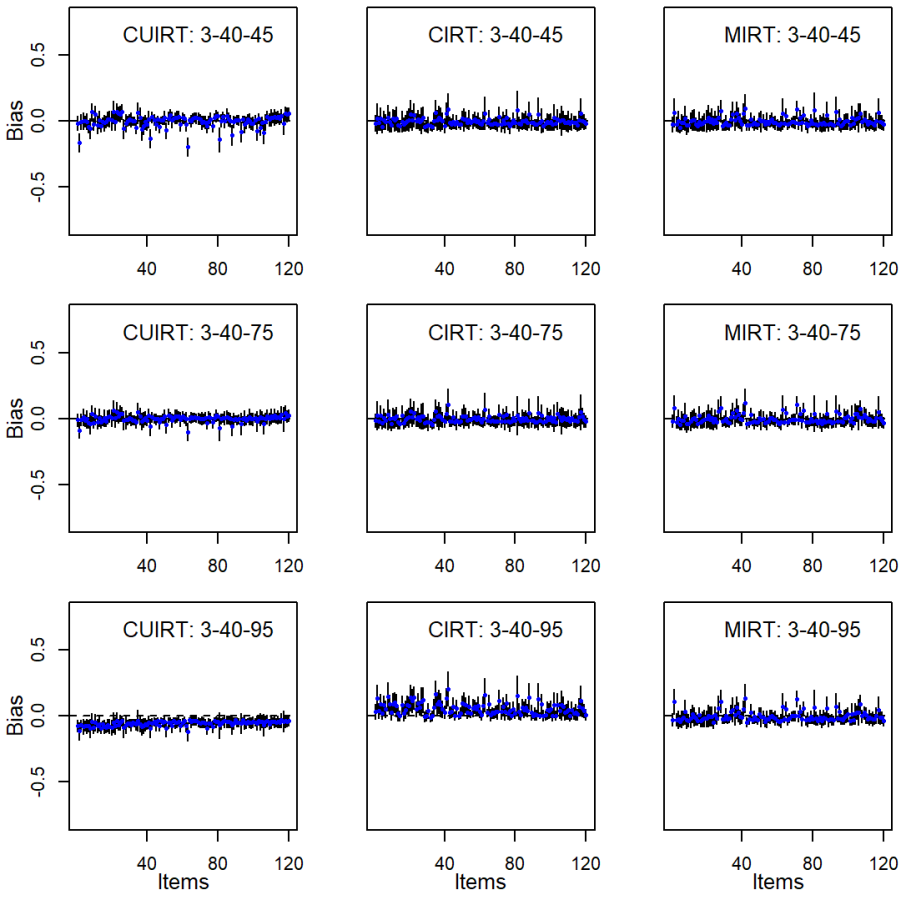
[7] The ABS and RMSE plots (see Figures H.1 and H.2, and Figures H.17 and H.18 in Appendix H) also showed the same patterns.

[8] The ABS and RMSE plots (see Figures H.3 and H.4, and Figures H.19 and H.20 in Appendix H) also showed the same patterns.

[9] The ABS and RMSE plots (see Figures H.3 and H.4, and Figures H.19 and H.20 in Appendix H) generally showed the same inclination as the bias plots.

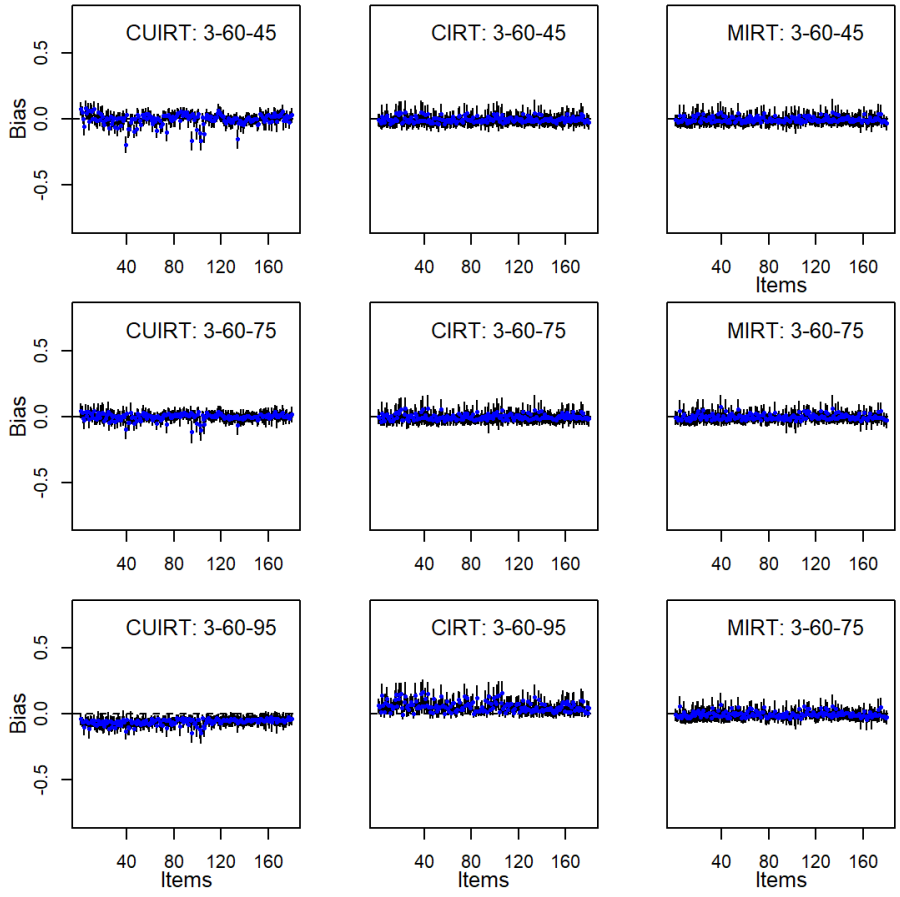
[10] The ABS and RMSE plots (see Figures H.5 and H.6, and Figures H.21 and H.22 in Appendix H) generally showed the same inclination as the bias plots.

Figure 5.21
Bias of a-Parameter for the 3 Domain, 40 Items per Domain Tests: Multiple Groups



that CUIRT produced more biased item parameter estimates regardless of test length where subscale score correlations were .45 and .75. In contrast CIRT produced less biased item parameter estimates regardless of test length and subscale correlation. In addition, the bias of the MIRT item difficulty parameter estimates were generally comparable across the studied test conditions. However, MIRT resulted in slightly higher bias regardless of test length where subscale correlation was .95.

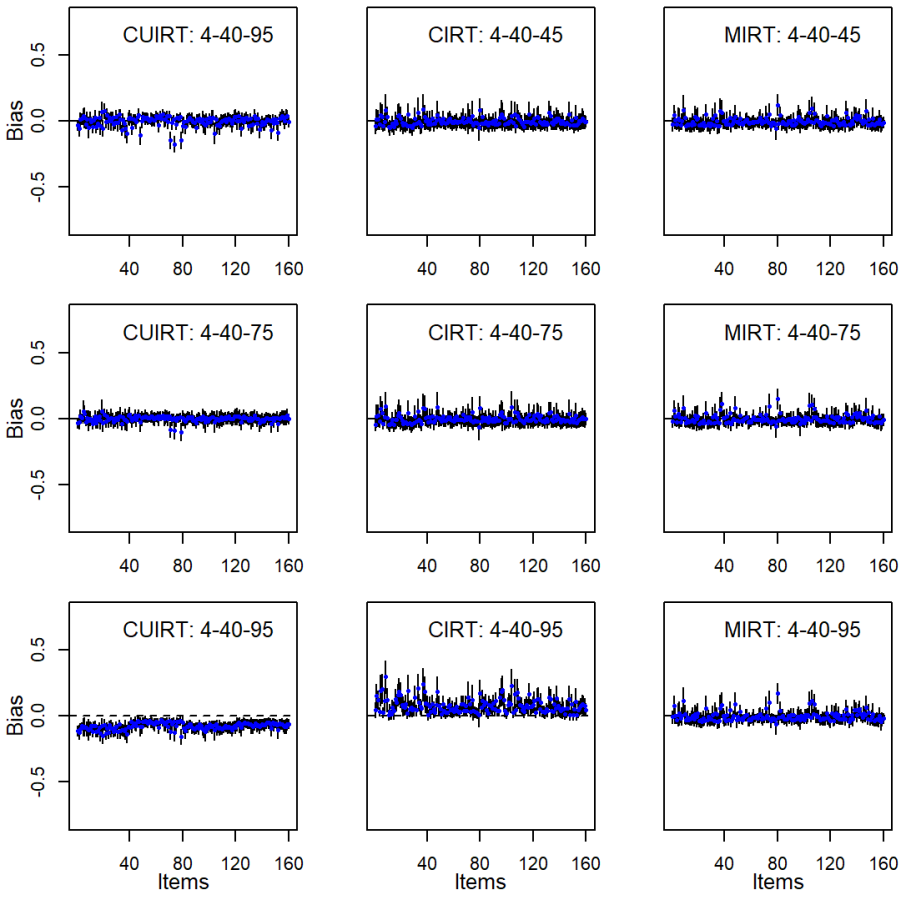
Figure 5.22
Bias of a-Parameter for the 3 Domain, 60 Items per Domain Tests: Multiple Groups



Four-Subdomain Tests.
Figures 5.27 and 5.28[11] show the biases of each items discrimination parameter, a, across all of Study 2's 4-subdomain, multiple groups conditions. The figures showed the same pattern of results that was reported on the 3-subdomain tests.

[11]The ABS and RMSE plots (see Figures H.7 and H.8, and Figures H.23 and H.24 in Appendix H) generally showed the same inclination as the bias plots.

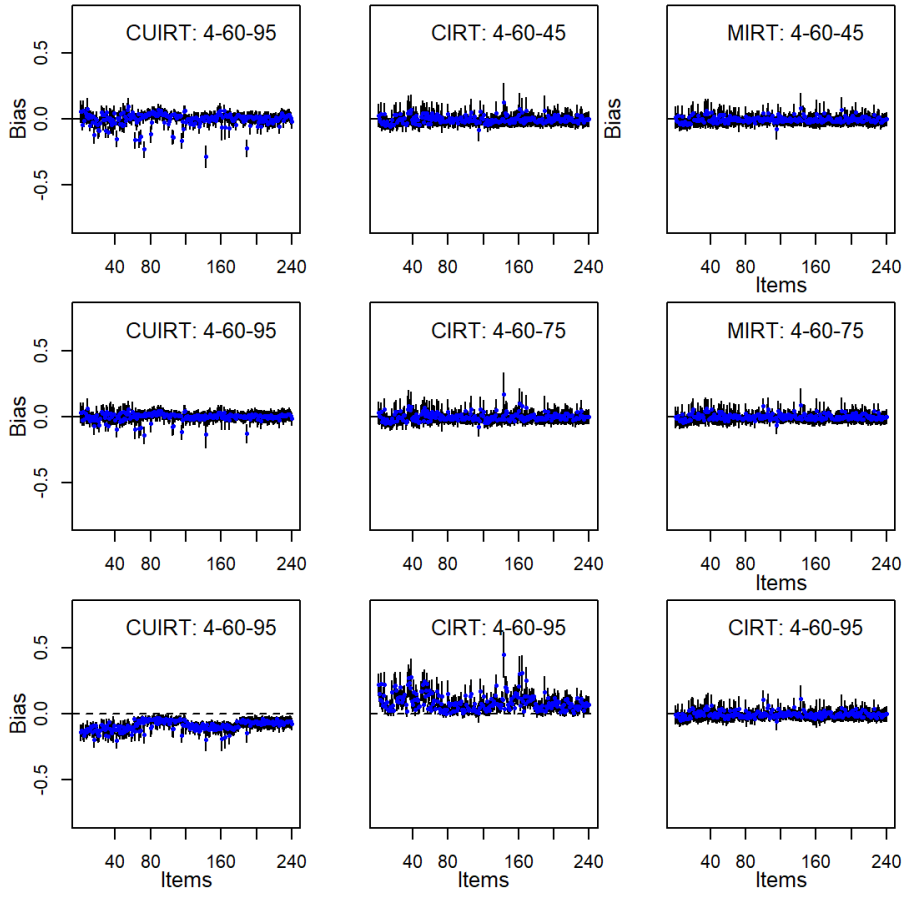
Figure 5.23
Bias of a-Parameter for the 4 Domain, 40 Items per Domain Tests: Multiple Groups



5.2.4.2 Subscale Length

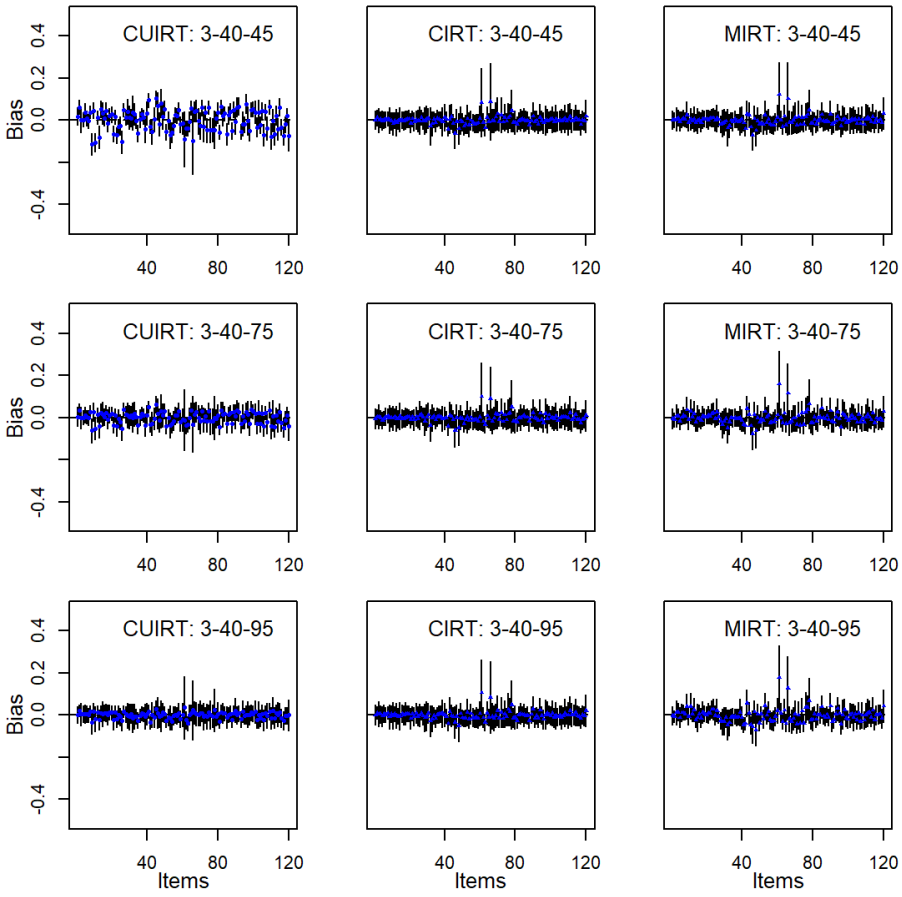
Item Discrimination and Difficulty.
The results for the item (a) discriminations, and (b) difficulty parameters will be presented together. This is because the results generally followed the same patterns and trends.

Figure 5.24
Bias of a-Parameter for the 4 Domain, 60 Items per Domain Tests: Multiple Groups



Three-Subdomain Tests.
The results presented in Figures 5.21 and 5.22 (item discrimination), and Figure 5.25 to 5.26 (item difficulty) showed that subscale length did not impact on the performance of the studied models. However, the average bias/ABS/RMSE presented in Tables 5.5 and 5.6 reported some trends. That is, the average biases for all of the studied models decreased as subscale length increased. In addition, Tables 5.5 and 5.6 showed that the average ABS and RMSE of CUIRT and CIRT reduced as subscale length increased. MIRT

Figure 5.25
Bias of b for the 3 Domain, 40 Items per Domain Tests: Multiple Groups

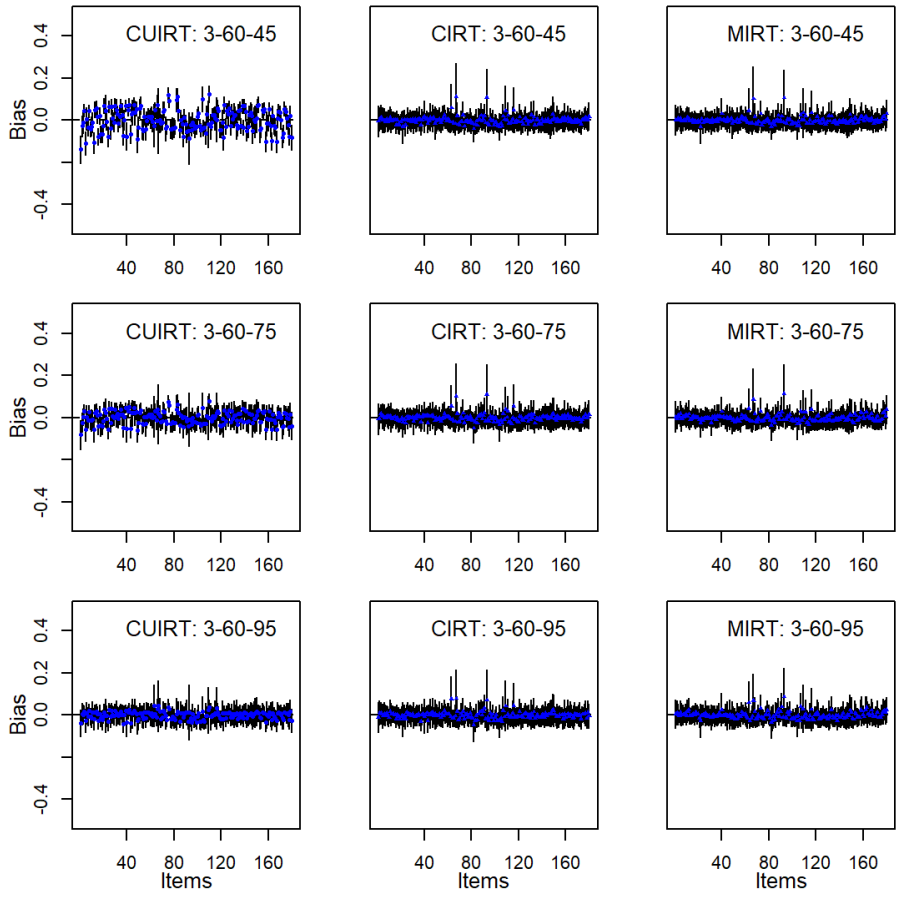


reported average ABS' that were not sensitive to an increase in subscale length (see Tables 5.5 and 5.6). In contrast, MIRT RMSE's were lower on the 60-item subscale tests than the tests that had 40-item subscales. CIRT generally reported the lowest bias and ABS.

Four-Subdomain Tests.
Figures 5.23 and 5.24, and Figure 5.27 to 5.28 did not show differences with respect to the sensitivity of the models to test length. However, the trends that were reported on Tables 5.5 and 5.6 for the 4 subdomain tests were similar to

Figure 5.26

Bias of b for the 3 Domain, 60 Items per Domain Tests: Multiple Groups



those presented for the 3 subdomain tests. One key difference was that the 4-subdomain test conditions reported lower bias, ABS, and RMSE compared to the 3-subdomain test condition results.

Figure 5.27

Bias of b-Parameter for the 4 Domain, 40 Items per Domain Tests: Multiple Groups

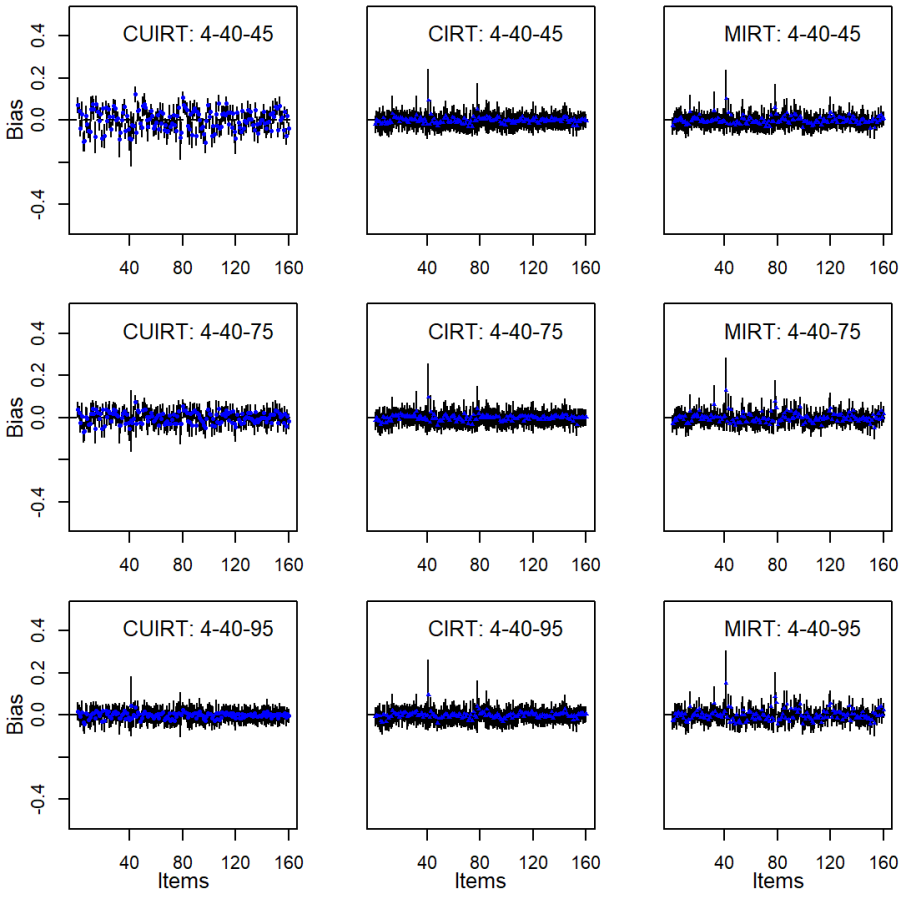


Figure 5.28
Bias of b-Parameter for the 4 Domain, 60 Items per Domain Tests: Multiple Groups

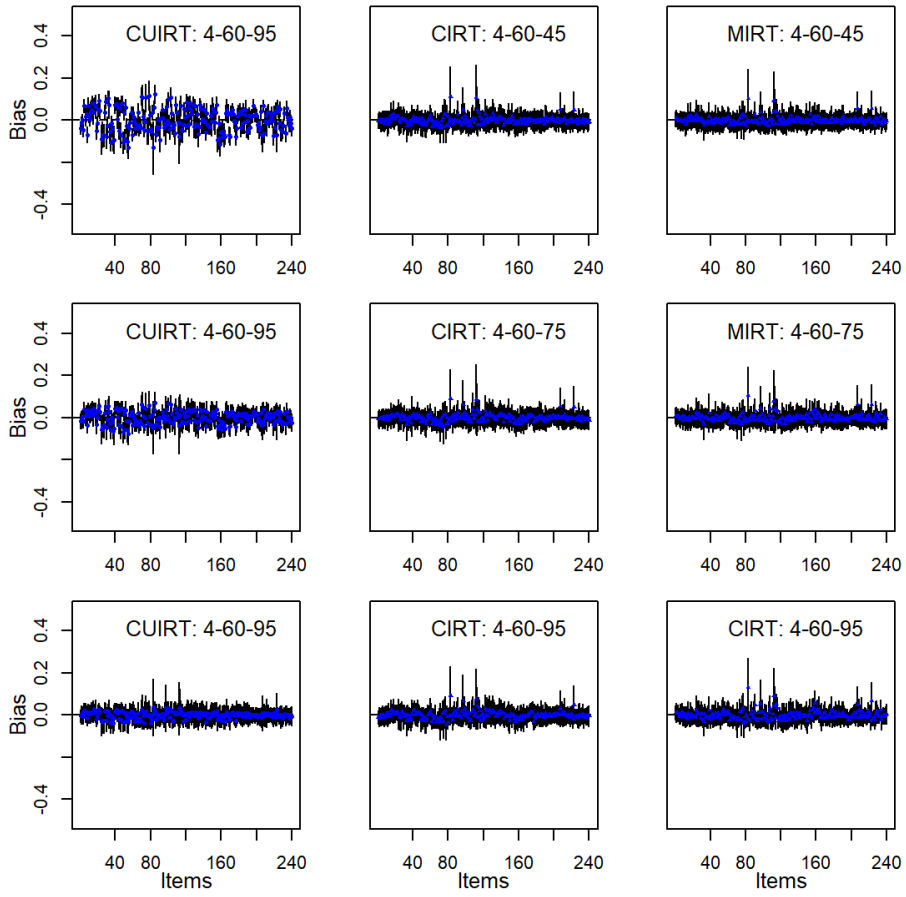


Table 5.5
Study 2 Item Discrimination Summary: Multiple Groups

				Bias			ABS			RMSE		
Par	D	J	ρ	CUIRT	CIRT	MIRT	CUIRT	CIRT	MIRT	CUIRT	CIRT	MIRT
a	3	40	.45	.36 (.01)	.66 (.06)	.66 (.23)	.36 (.01)	.66 (.06)	.66 (.23)	.37 (.01)	.67 (.06)	.67 (.23)
			.75	.48 (.02)	.66 (.06)	.67 (.28)	.48 (.02)	.66 (.06)	.67 (.28)	.49 (.02)	.67 (.06)	.67 (.28)
			.95	.58 (.04)	.66 (.06)	.67 (.32)	.58 (.04)	.66 (.06)	.67 (.32)	.58 (.04)	.67 (.06)	.68 (.32)
		60	.45	.35 (.01)	.64 (.05)	.62 (.20)	.35 (.01)	.64 (.05)	.62 (.20)	.35 (.01)	.65 (.05)	.62 (.20)
			.75	.47 (.02)	.64 (.05)	.61 (.25)	.47 (.02)	.64 (.05)	.61 (.25)	.47 (.02)	.65 (.05)	.62 (.25)
			.95	.55 (.03)	.65 (.05)	.60 (.28)	.55 (.03)	.65 (.05)	.61 (.28)	.56 (.03)	.65 (.05)	.61 (.28)
	4	40	.45	.33 (.01)	.68 (.05)	.68 (.29)	.33 (.01)	.68 (.05)	.68 (.29)	.34 (.01)	.69 (.05)	.69 (.28)
			.75	.47 (.01)	.69 (.05)	.69 (.36)	.47 (.01)	.69 (.05)	.69 (.36)	.47 (.01)	.70 (.05)	.70 (.36)
			.95	.57 (.03)	.69 (.05)	.69 (.42)	.57 (.03)	.69 (.05)	.70 (.41)	.57 (.03)	.70 (.05)	.70 (.41)
		60	.45	.32 (.01)	.68 (.06)	.64 (.30)	.32 (.01)	.68 (.06)	.64 (.30)	.33 (.01)	.69 (.06)	.65 (.29)
			.75	.46 (.02)	.69 (.06)	.63 (.38)	.46 (.02)	.69 (.06)	.64 (.37)	.46 (.02)	.70 (.06)	.65 (.37)
			.95	.56 (.03)	.70 (.06)	.63 (.45)	.56 (.03)	.70 (.06)	.64 (.43)	.56 (.03)	.70 (.06)	.65 (.43)

Note. Par = item parameter; D = number of subscales; J = subscale length; ρ = subscale correlation; a = item discrimination; b = item difficulty; the values in parentheses represent the standard deviations across replications.

Table 5.6
Study 2 Item Difficulty Summary: Multiple Groups

				Bias			ABS			RMSE		
Par	D	J	ρ	CUIRT	CIRT	MIRT	CUIRT	CIRT	MIRT	CUIRT	CIRT	MIRT
b	3	40	.45	.13 (.03)	.07 (.04)	.10 (.07)	.18 (.02)	.18 (.02)	.24 (.02)	.19 (.02)	.18 (.02)	.25 (.02)
			.75	.10 (.04)	.07 (.04)	.11 (.07)	.18 (.02)	.18 (.02)	.25 (.02)	.18 (.02)	.18 (.02)	.26 (.02)
			.95	.08 (.04)	.07 (.04)	.11 (.08)	.18 (.02)	.18 (.02)	.26 (.02)	.18 (.02)	.18 (.02)	.26 (.02)
		60	.45	.12 (.03)	.06 (.04)	.10 (.07)	.17 (.02)	.17 (.02)	.24 (.02)	.17 (.02)	.17 (.02)	.24 (.02)
			.75	.09 (.03)	.06 (.04)	.12 (.07)	.16 (.02)	.17 (.02)	.26 (.02)	.17 (.02)	.17 (.02)	.26 (.02)
			.95	.07 (.04)	.06 (.04)	.13 (.08)	.16 (.02)	.17 (.02)	.27 (.02)	.17 (.02)	.17 (.02)	.27 (.02)
	4	40	.45	.14 (.02)	.07 (.04)	.13 (.05)	.17 (.01)	.17 (.01)	.23 (.01)	.18 (.01)	.17 (.01)	.24 (.01)
			.75	.11 (.03)	.07 (.04)	.14 (.05)	.17 (.01)	.17 (.01)	.25 (.01)	.17 (.01)	.17 (.01)	.25 (.01)
			.95	.09 (.03)	.07 (.04)	.15 (.06)	.17 (.01)	.17 (.01)	.26 (.02)	.17 (.01)	.18 (.01)	.26 (.01)
		60	.45	.15 (.03)	.07 (.04)	.15 (.05)	.18 (.02)	.18 (.02)	.25 (.02)	.19 (.02)	.19 (.02)	.25 (.01)
			.75	.12 (.03)	.07 (.05)	.17 (.06)	.18 (.02)	.18 (.02)	.26 (.02)	.18 (.02)	.19 (.02)	.27 (.02)
			.95	.10 (.04)	.07 (.05)	.19 (.06)	.18 (.02)	.19 (.02)	.28 (.02)	.18 (.02)	.19 (.02)	.28 (.02)

Note. Par = item parameter; D = number of subscales; J = subscale length; ρ = subscale correlation; a = item discrimination; b = item difficulty; the values in parentheses represent the standard deviations across replications.

5.2.5 Synthesis of Item Parameter Recovery

The results presented in Section 5.2 suggested that the studied subscale score estimation models were sensitive to subscale correlation regardless of the number of subscales. That is, some models were likely to report better item parameter estimates than others under certain subscale correlations. For example, MIRT and CIRT performed better when subscale correlations were low and moderate. In contrast, CUIRT reported better item parameter estimates when subscale correlation was high. In general, the models performed comparatively at different subscale lengths regardless of the number of subscales.

It would have been expected that since MIRT was the generating model, the model would have resulted in better item parameter estimates over all simulated test conditions. However, the results presented in Section 5.2 showed that this was not always the case. That is, MIRT was not the best performing model across all simulated test conditions. For example, CUIRT and CIRT performed better than MIRT where subscale correlation was .95. One likely reason that this was the case is that the underlying "true" data generating mechanism of the chosen data where subscale correlations were high represented a more unidimensional model. That is, since the subscales were highly correlated, they may inherently be measuring the same construct thus rendering the data unidimensional. Indeed, at such high correlations, the data that were generated in the DGP may have essentially been unidimensional as opposed to being multidimensional.

5.2.5.1 Study 1

Of the three subscale score estimation models, CIRT was least sensitive to the studied conditions (number of subscales, subscale length and subscale correlation). In other words, CIRT performed consistently better than CUIRT and MIRT. Additionally, the performance of CUIRT was better when subscale correlation was high (.95). Given the structure of the model, it was not surprising that CUIRT performed better when subscale correlation was high. However, when $\rho = .95$, MIRT had larger bias, ABS and RMSE.

The results presented in Sections 5.2.1 and 5.2.2 showed that CIRT and MIRT had a structural positive bias in the multiple groups simulations. Tables 5.1 and 5.2 further showed that the single groups simulations showed less average bias. These results suggest that as the number of groups increase, from single- to multiple-groups, there may be an added complexity that results in more biased item parameters.

5.2.5.2 Study 2

Figure 5.25 to 5.28[12] showed that multiple items exhibited larger biases. The magnitudes of the reported biases suggested that some replications reported under- or over-estimates of the item difficulty. The under- or over-estimation was observed where the tests had fewer subdomains. Some under- or over-estimation was also reported by CUIRT regardless of test length where subscale correlation was $\rho = .45$. CIRT and MIRT suggested some under- and over-estimated on the test that had three-subdomains, each subdomain had 40 items, and were subscale correlation was $\rho = .95$. The item discrimination parameters reported in Figure 5.25 to 5.24 also showed that though subscale correlations were irrelevant in CIRT, the item parameters showed a positive bias when subscale correlation was .95. Conceptually, it would be argued that at high correlations, the subscales are highly correlated to an overall proficiency (ideally unidimensional), and each other as a result. Therefore, CIRT would present a misspecification. However, these findings were mostly observed for the discrimination parameters in Study 2.

A closer review of Table 5.3 to 5.6 showed that CUIRT item discrimination had larger RMSE than CIRT and MIRT in the single groups simulations. These results were in line with findings from other studies. For example, Yao and Boughton (2007) pointed out that CUIRT produce worse item parameter estimates than MIRT. However, results presented in Table 5.3 to 5.6 also showed lower RMSE's on the CUIRT-based item discrimination parameter in the multiple groups simulations. In providing some rough guidelines as to the required calibration size, DeAyala (2013) stated that practitioners may need to consider several factors including: (a) the variability and distribution of respondents, and (b) the amount of missing data on the test. As such, the reduction in RMSE between the single- and multiple-groups studies may in part be attributed to the increase in sample sizes (i.e., from 3,000 to 6,000). The increase in sample sizes translated into an increased exposure to each of the items on the test translating into more information about a specific item being collected. As a result, better item parameter estimates would be expected. The benefits of an increase in sample size would be prevalent in Study 2 where multiple matrix sampling introduces missing data on the test.

In general, CUIRT, CIRT and MIRT resulted in comparable item location bias for the polytomous items. That is, the results were similar for $d1$ and $d2$.

[12] An examination of the ABS (see Figure H.5 to H.8), and RMSE plots (see Figure H.21 to H.24) presented in Appendix H show that for some items, the difference between the estimated and the true item difficulty parameters was indeed quite large in some replications. This occurred mostly on the three-subdomain tests (see Figures H.5, H.6 and H.21).

When the models were compared in a condition, the observed magnitude of bias were similar. It was also observed that all of the three studied models showed larger bias in the last three GPCM items (see Figures D.4 and D.8 in Appendix D) regardless of subscale correlations on tests which comprised of 240 items.

5.3 Score Recovery

To investigate all the methods' subscale score recovery (RQ2), I report bias, ABS, and RMSE of the estimated population scores. I used the three evaluation criteria to compare different methods across various simulation conditions. In the sections that follow, I present the results for Study 1 (Sections 5.3.1 and 5.3.2), and Study 2 (Sections 5.3.3 and 5.3.4). Each of the sections present the single- and multiple-groups´ results for the respective studies.

Three and five population-domain-proficiency estimates from CUIRT, CIRT, and MIRT were compared with true values. All of the figures presented in the results for Study 1 follow the same outline. The rows represent the 5, 10, and 15 subdomain test conditions, whereas each plot in a row represents a specific domain. Each plot shows the average bias by each estimated correlation.

5.3.1 Score Recovery for Study 1: Single Groups

Figures 5.29 and 5.30 show the bias plots for all of the 3- and 5-subdomain conditions, respectively. The findings are presented by number of domains.

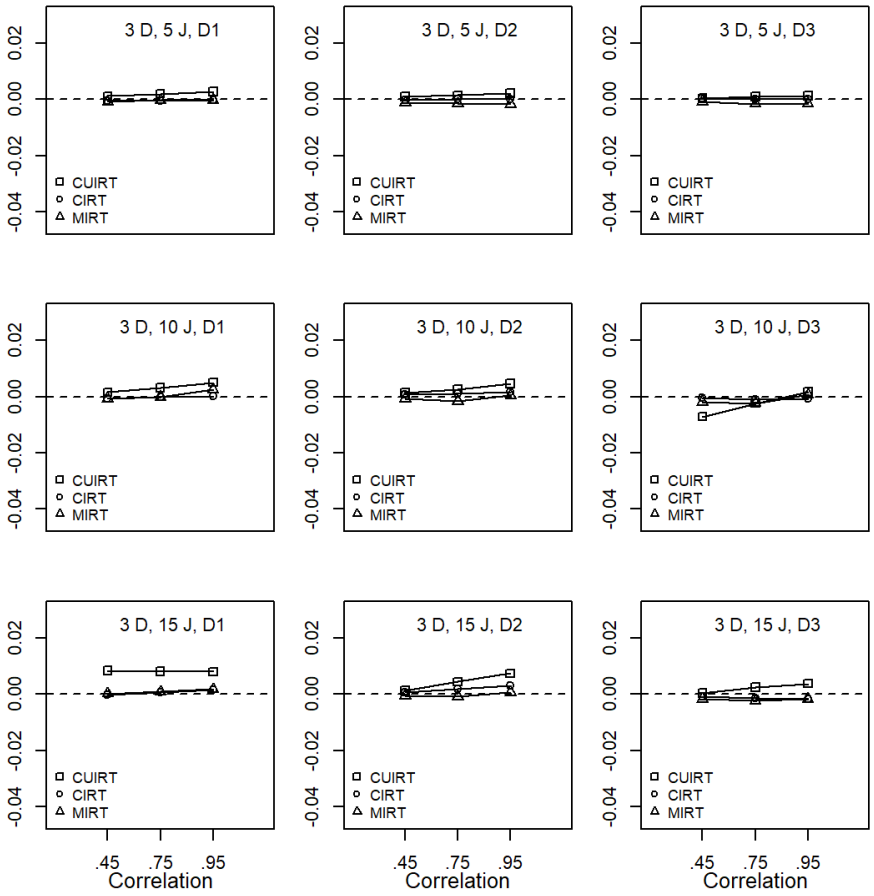
5.3.1.1 Subscale Correlation

Three-Subdomain Tests.
The first plot in the grid presented in Figure 5.29 shows that the bias of the scores on the three domain tests over all correlations (i.e., .45, .75, .95) were close to 0[13]. This pattern was observed over a majority of the conditions. CUIRT consistently showed more bias, ABS and RMSE in domain 1 on the 3 domain, 15 item tests (across all correlations; see Figures I.1 and I.3 in Appendix I). CUIRT also showed higher ABS in domains 3 and 2 on the (a) $D = 3$, $J = 10$, $\rho = .45$ and (b) $D = 3$, $J = 15$, $\rho = .95$ conditions, respectively.

[13]Figures I.1 and I.3 in Appendix I also reveal that ABS and RMSE, respectively, on the three subdomain tests were also close to 0.

Figure 5.29

Subscale Score Bias for the Three-Subomain Tests: Single Groups



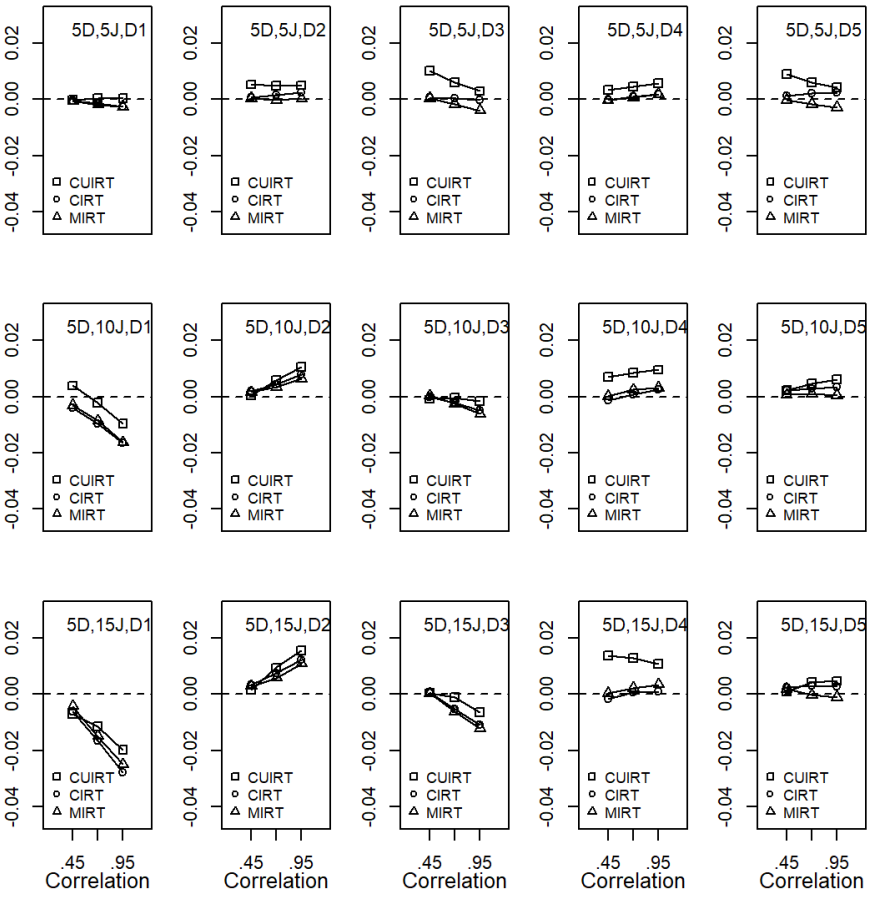
Note. D = number of domains; J = number of items; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

Five-Subdomain Tests.

Figure 5.30 shows that bias were similar on some of the 5 subdomain test conditions. Though this was the case, CUIRT showed higher bias, ABS and RMSE in domain 4 across all test lengths and specified correlations (see Figures I.2 and I.4 in Appendix I). CIRT and MIRT showed the same bias

Figure 5.30

Subscale Score Bias for the Five-Submain Tests: Single Groups



Note. D = number of domains; J = number of items; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4; $D5$ = domain 5.

across a majority of the conditions except in the fifth subdomain on the tests with five subdomains that comprised of 5 items per domain tests where CIRT had the lowest bias. Though this was the case in that condition, CUIRT had the highest ABS and RMSE. CIRT and MIRT consistently showed slightly

higher bias, ABS and RMSE in domain 1. CUIRT ABS and RMSE were also lowest in domain 3 on the 10 and 15 subdomain tests whilst being higher for correlations of .45 and .75 on the same domain in the 5 subdomain test.

5.3.1.2 Subscale Length

Three-Subdomain Tests.
The trends that were reported on Figure 5.29 do not show a consistent trend across subdomains. The results showed that models performed differently depending on the subdomain. This was probably because the item parameters that were used in scoring each domain were different and had different distributions (see the item parameters specified in data generation, Table 3.4). However, CUIRT bias on domain 1 was largest on the 15-items-per-subscale test compared to the test conditions with subscale lengths of 5 and 10. CUIRT also showed larger biases compared to CIRT and MIRT on domains 2 and 3 where subscale correlation was .95.

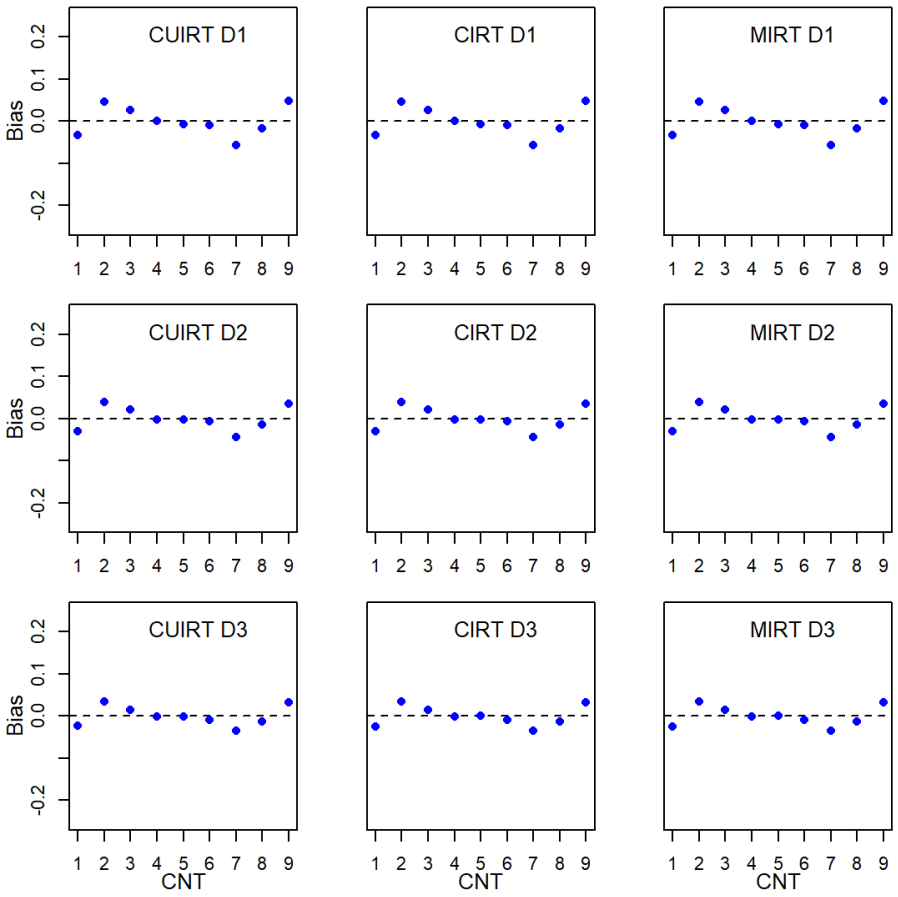
Five-Subdomain Tests.
Similar to the 3-subdomain results, the trends that were reported in the 5-subdomain rest conditions did not show a consistent trends, and the models performed differently depending on the subdomain. Figure 5.30 showed that all of the models reported larger bias on domains 1 and 2 on the longer subscales where subscale correlation was .95. CUIRT bias on domain 4 was grew larger as the subscale length increased from 5- to 15-items-per-subdomain.

5.3.2 Score Recovery for Study 1: Multiple Groups

Figure 5.31 to 5.48 show the bias plots for all of the 3- and 5-subdomain conditions, respectively. For illustrative purposes, each figure shows a different test condition. The findings are presented by number of domains. As an example, Figure 5.31 shows the bias of a test where $D = 3$, $J = 5$, and $\rho = .45$). Within the figure, each row represents a different domain and each column presents a different model (i.e., CUIRT, CIRT, MIRT). To save space, ABS and RMSE plots are presented in Appendix J. Each plot within the grids presents either bias, ABS, or RMSE of the estimated scores from a specific subscale score estimation method for each subdomain on a reported test. For example, in Figure 5.31, the top left plot shows the bias of CUIRT population scores on a test where $D = 3$, $J = 5$, and $\rho = .45$.

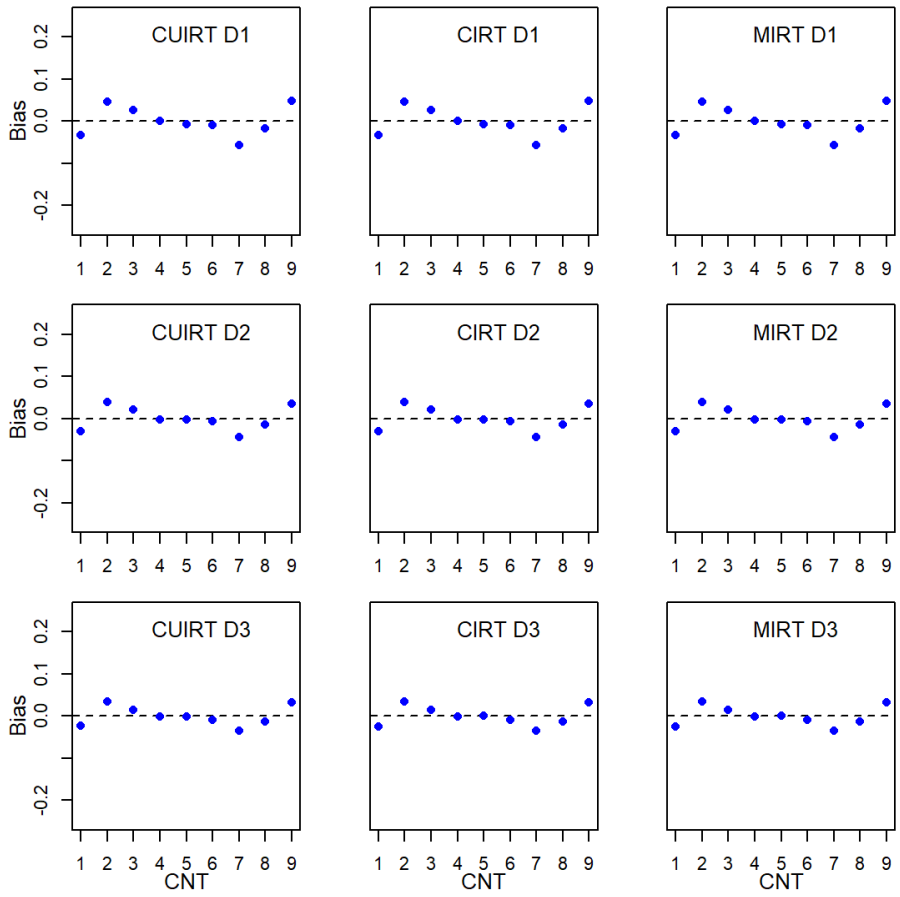
Figure 5.31

Subscale Score Bias for the 3-Domain, 5-Item, .45 Correlation Subdomain Tests: Multiple Groups



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

Figure 5.32
Subscale Score Bias for the 3-Domain, 5-Item, .75 Correlation Subdomain Tests: Multiple Groups



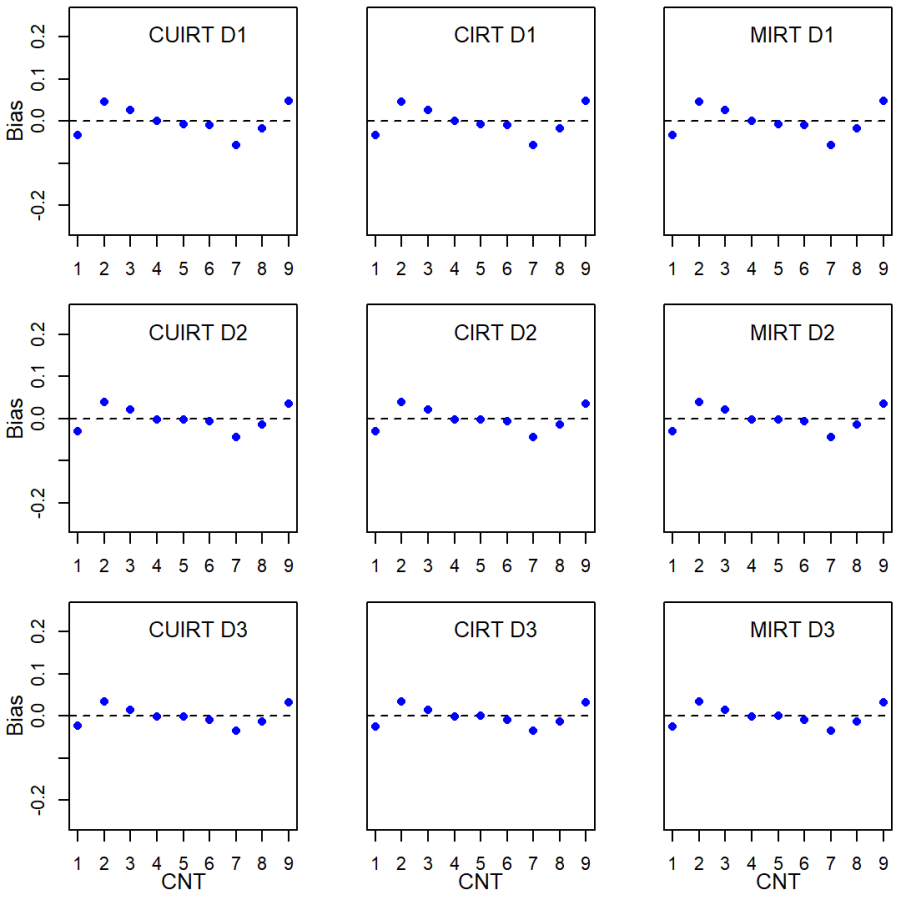
Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

5.3.2.1 Subscale Correlation

Three-Subdomain Tests.
According to the plots shown in Figure 5.31 to 5.39 the estimated scores did not show much sensitivity to the specified model. That is, the estimated scores

Figure 5.33

Subscale Score Bias for the 3-Domain, 5-Item, .95 Correlation Subdomain Tests: Multiple Groups

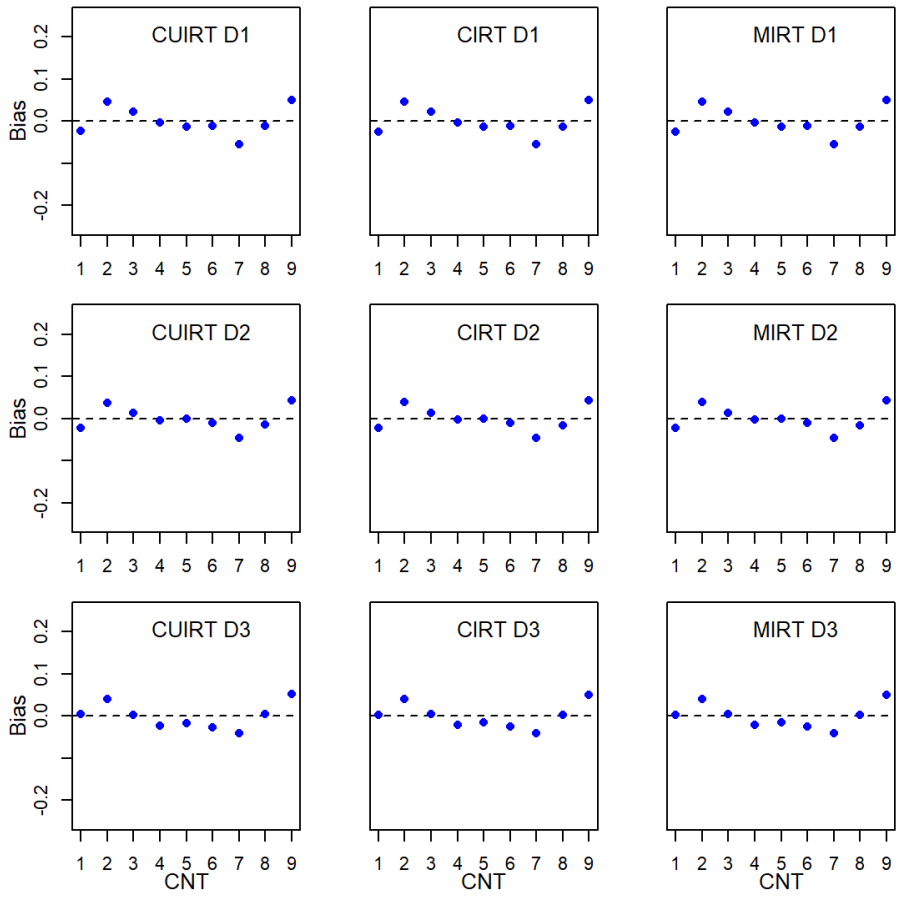


Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

that came from CUIRT, CIRT, and MIRT were very similar. However, there were a few findings that were observed. The biases across all three-subdomain test conditions ranged between -.06 and .05. In general, middle performing

Figure 5.34

Subscale Score Bias for the 3-Domain, 10-Item, .45 Correlation Subdomain Tests: Multiple Groups

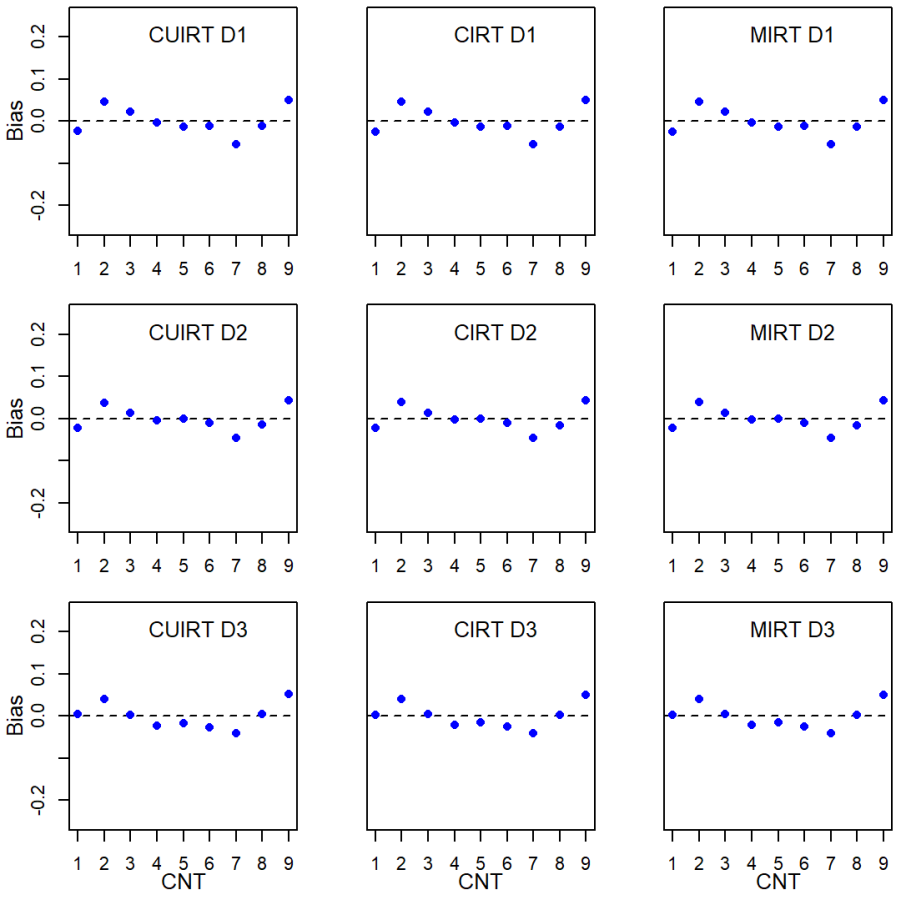


Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

countries consistently showed biases closest to 0 (i.e., between 0 and ±.01). The plots showed the largest biases were prevalent for the low and high performing populations. Countries 1, 2, 7, and 9 consistently showed some large bias with

Figure 5.35

Subscale Score Bias for the 3-Domain, 10-Item, .75 Correlation Subdomain Tests: Multiple Groups



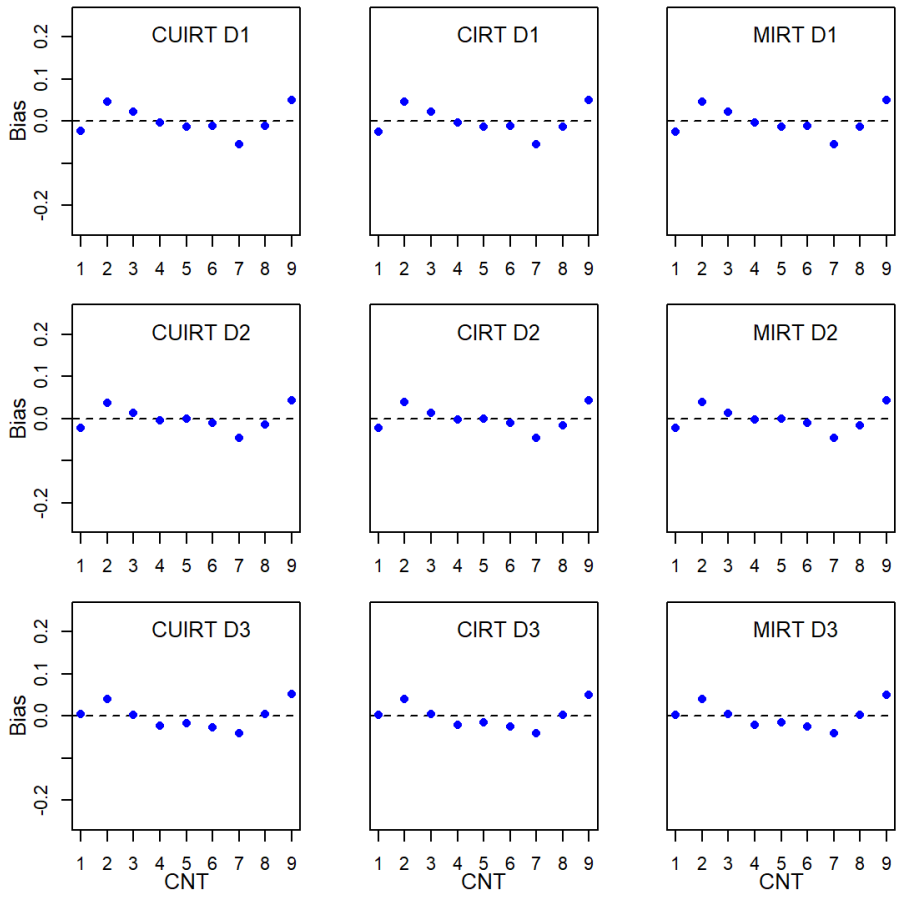
Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

the values being the largest for countries 2 and 7.

The ABS (see Figure J.1 to J.9), and RMSE plots (see Figure J.19 to J.27) in Appendix J showed the same patterns. The ABS was particularly high for countries 2, 7 and 9. Nevertheless, the RMSE slightly reduced across

Figure 5.36

Subscale Score Bias for the 3-Domain, 10-Item, .95 Correlation Subdomain Tests: Multiple Groups

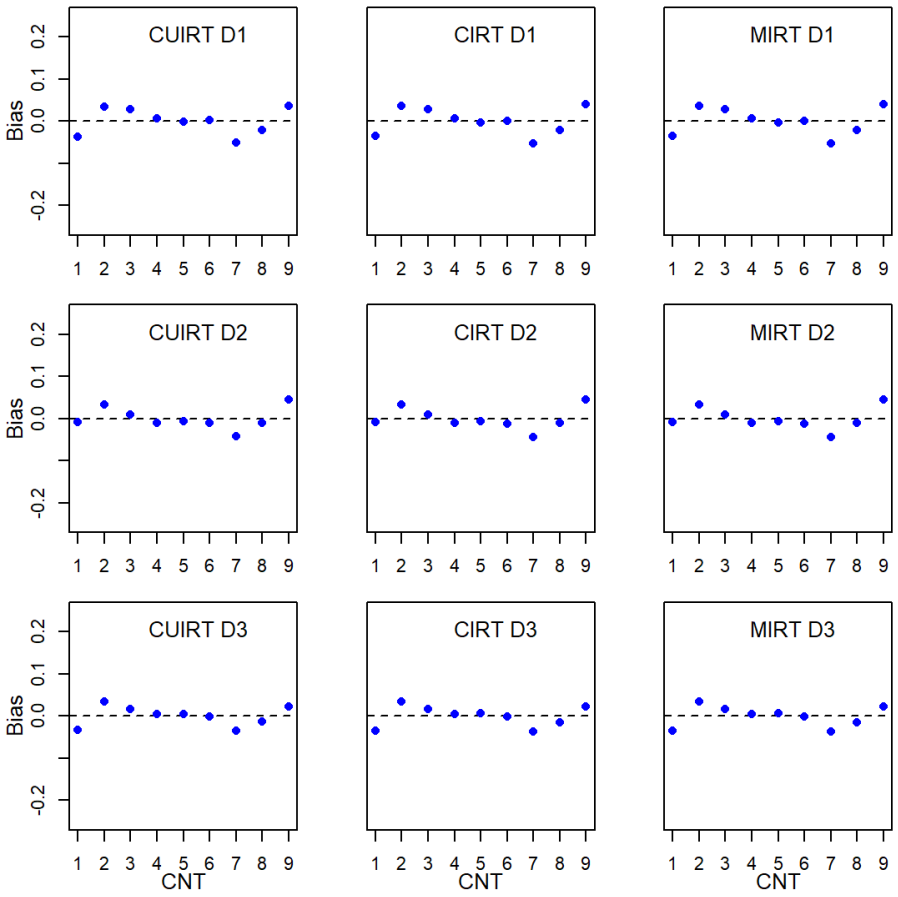


Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

all countries (were closer to 0) as the studied number of items per domain increased from $J = 5$ to $J = 15$. Though each model performed better as the number of items per subdomain increased, ABS was particularly high for

Figure 5.37

Subscale Score Bias for the 3-Domain, 15-Item, .45 Correlation Subdomain Tests: Multiple Groups

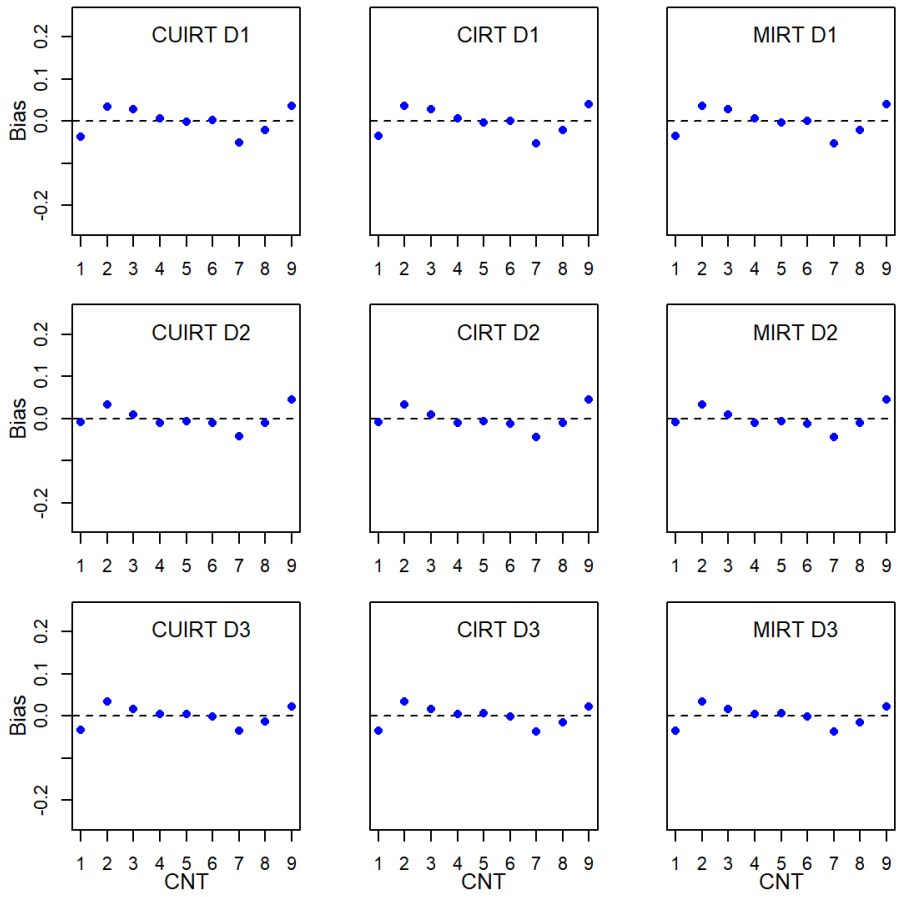


Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

countries 7 and 9. This result contradicted the results that were reported in Figure 5.29 which showed slight increases in bias as the number of items in each subdomain increased from 5 to 15. In addition, the RMSEs (see Figure J.19 to

Figure 5.38

Subscale Score Bias for the 3-Domain, 15-Item, .75 Correlation Subdomain Tests: Multiple Groups



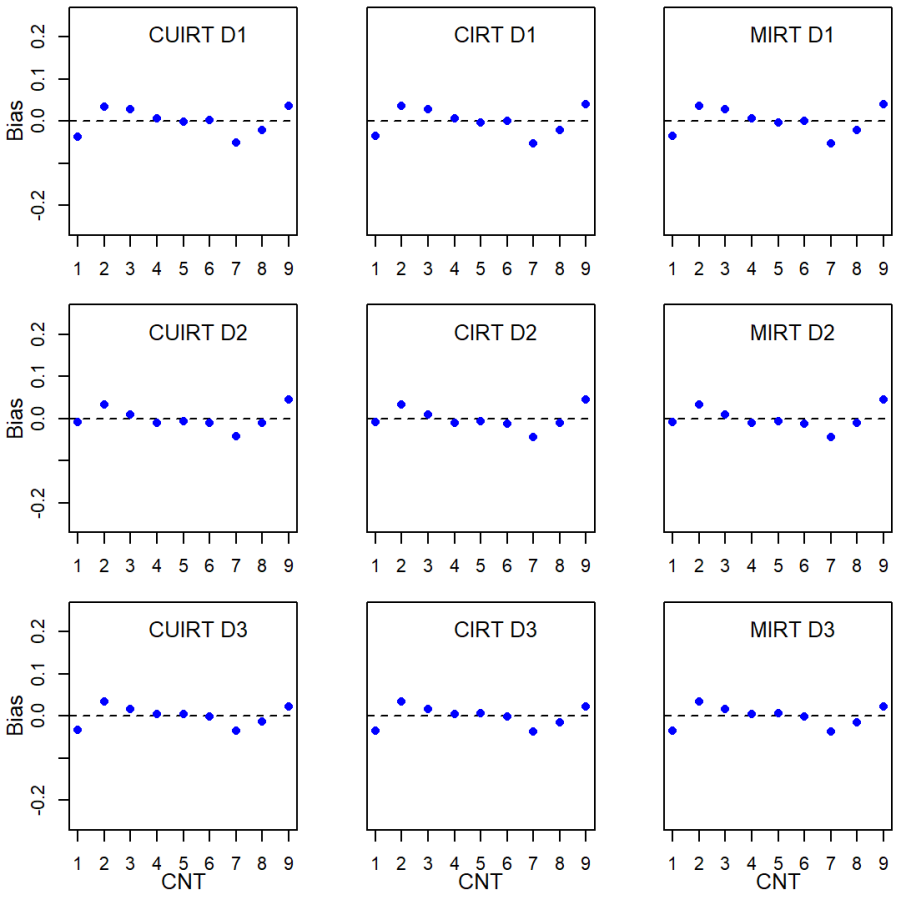
Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

J.27 in Appendix J) seemed to be equal across country.

Despite the middle-performers, and Country 8 showing average biases closest to 0 across all simulated conditions, thorough investigation revealed that all models reported larger average absolute biases on Subdomains 1 for the other

Figure 5.39

Subscale Score Bias for the 3-Domain, 15-Item, .95 Correlation Subdomain Tests: Multiple Groups



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

countries. On this domain, the reported average ABS for countries 2, 7, and 9 that were .04 or larger, regardless of subscale correlation, test length, and subscale score estimation model. For example, inspection showed that all of

the studied models reported an average absolute bias of .06 for Country 7 on the first subdomain for all five-items-per-subdomain conditions. Upon digging into the results, Table 5.7 showed that, though average item difficulties were close to 0, low standard deviations were reported on all of the test conditions that comprised of tests with 5- and 10-items-per-subdomain. This meant that the spread of the item difficulties were smaller, and closer to 0, than those observed on the other two subdomains. However, this was not the case on the three subdomain tests that had 15 items-per-subdomain, regardless of subscale correlation; in this case subdomain 3 reported the lowest SD.

It should also be noted that all of the models showed the least bias, RMSE and ABS on subdomain 3 across all 3-subdomain test conditions, regardless of subscale correlations, for tests whose subscale lengths were five- and 10-items (Figure 5.31 to 5.36, respectively). The average absolute biases did not exceed .05 on the 10- and 15-items-per-subdomain test conditions. In contrast, Figure 5.37 to 5.39 show that the 15 subdomain item tests did not show much difference in bias, ABS and RMSE. Further scrutiny of the average item-difficulty for Study 1's multiple groups simulation conditions (see Table 5.7) showed that lower bias, RMSE and ABS were observed where the standard deviation of the difficulty parameters were higher. For example, Table 5.7 reported larger standard deviations where subscale lengths were 5 and 10 on domains two and three which also coincided with lower average absolute biases. In contrast, the item difficulty of the first subdomain in the five- and 10-item-per-subdomain test conditions had lower standard deviations and showed the largest bias for countries 2, 7, and 9, in particular. In addition, the reported biases where $J = 15$ were lower across all subdomains, and the standard deviations of the item difficulty parameters on these conditions were larger (than those reported in subdomain 1 on the 5 and 10 item per subdomain conditions). These findings suggest that a larger spread of the item-difficulties makes it possible to collect more information from the entire score distribution (DeAyala, 2013; Embretson & Reise, 2000).

Five-Subdomain Tests.
Similar patterns were observed on the 5 subdomain tests (see Figure 5.40 to 5.48[14]). However, it was observed that Country 1 had the highest bias and ABS in subdomain 1 across all estimation models where subscale lengths were 10- and 15-items-per-subdomain. The observed biases, ABS and RMSE in country 1 were over .15 on these specific 5-subdomain test conditions. The corresponding ABS' for country 1 also exceeded .15. Country

[14]The corresponding ABS and RMSE plots are presented in Appendix J.

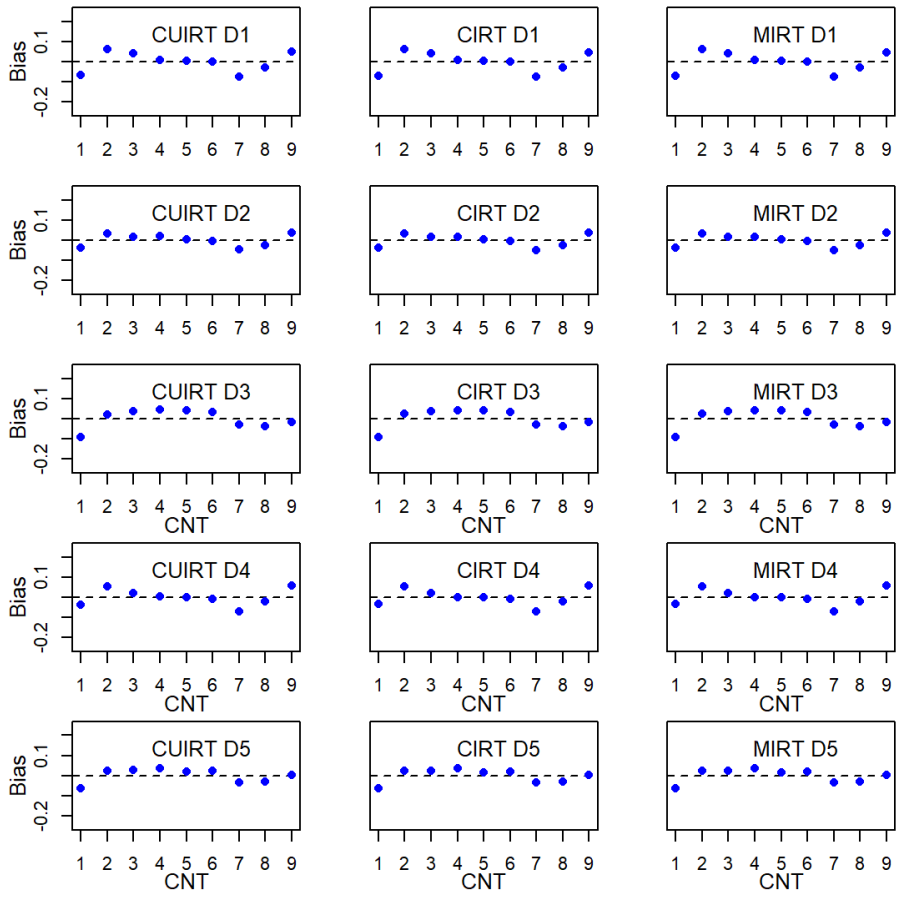
Table 5.7
Study 1 Average Item Difficulty of 3-Subdomain Tests : Multiple Groups

			Domain		
J	ρ	Model	1	2	3
5	.45	UIRT	.01 (1.02)	.01 (1.06)	.01 (1.23)
		CIRT	.01 (1.08)	.01 (1.12)	.01 (1.30)
		MIRT	.01 (1.08)	.01 (1.12)	.01 (1.30)
	.75	UIRT	.01 (1.05)	.01 (1.09)	.01 (1.27)
		CIRT	.01 (1.08)	.01 (1.12)	.01 (1.30)
		MIRT	.01 (1.09)	.01 (1.13)	.01 (1.31)
	.95	UIRT	.01 (1.07)	.01 (1.11)	.01 (1.29)
		CIRT	.01 (1.08)	.01 (1.12)	.01 (1.30)
		MIRT	.01 (1.10)	.01 (1.14)	.01 (1.32)
10	.45	UIRT	.02 (1.02)	.01 (1.06)	.01 (1.35)
		CIRT	.02 (1.07)	.01 (1.11)	.01 (1.42)
		MIRT	.02 (1.07)	.01 (1.11)	.01 (1.42)
	.75	UIRT	.01 (1.05)	.01 (1.09)	.01 (1.38)
		CIRT	.01 (1.07)	.01 (1.11)	.01 (1.41)
		MIRT	.01 (1.07)	.01 (1.11)	.01 (1.42)
	.95	UIRT	.01 (1.07)	.01 (1.11)	.01 (1.41)
		CIRT	.01 (1.07)	.01 (1.11)	.01 (1.41)
		MIRT	.01 (1.08)	.01 (1.12)	.01 (1.43)
15	.45	UIRT	.02 (1.15)	.01 (1.19)	.01 (1.08)
		CIRT	.02 (1.20)	.01 (1.25)	.01 (1.13)
		MIRT	.02 (1.20)	.01 (1.25)	.01 (1.13)
	.75	UIRT	.01 (1.18)	.01 (1.22)	.01 (1.11)
		CIRT	.01 (1.20)	.01 (1.25)	.01 (1.13)
		MIRT	.01 (1.20)	.01 (1.25)	.01 (1.13)
	.95	UIRT	.01 (1.20)	.01 (1.24)	.01 (1.13)
		CIRT	.01 (1.20)	.01 (1.25)	.01 (1.13)
		MIRT	.01 (1.21)	.01 (1.25)	.01 (1.14)

Note. J = subscale length; ρ = subscale correlation; the values in parentheses represent the standard deviations across replications.

Figure 5.40

Subscale Score Bias for the 5-Domain, 5-Item, .45 Correlation Subdomain Tests: Multiple Groups



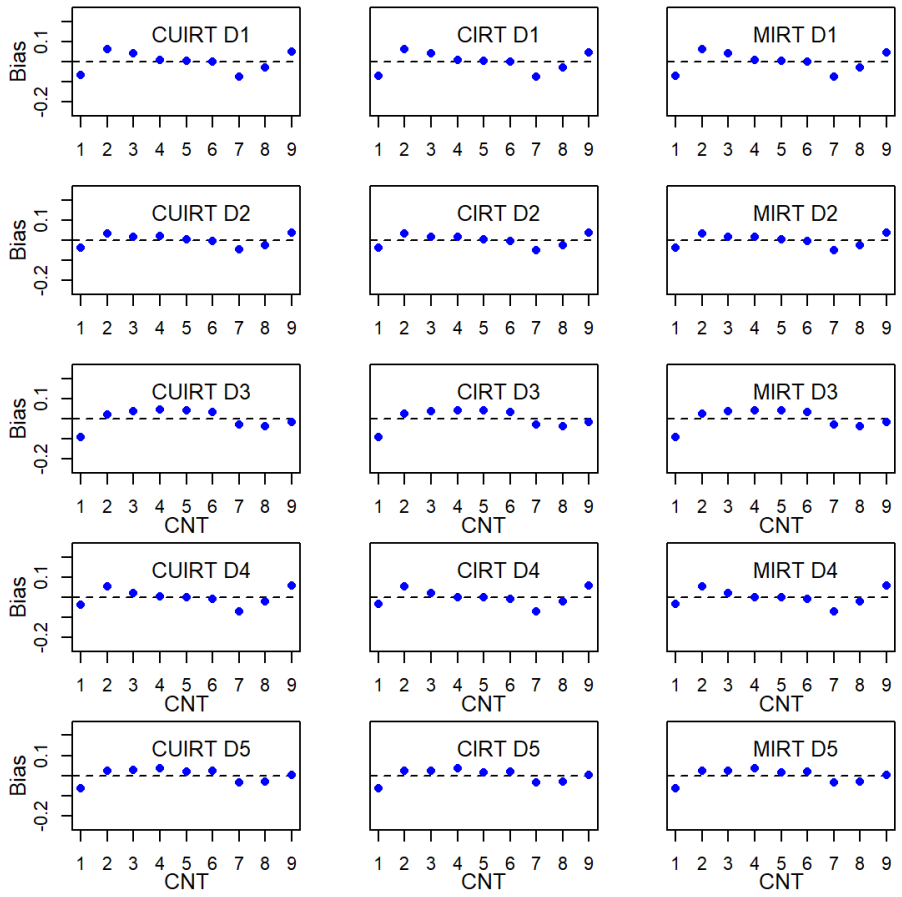
Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4; $D5$ = domain 5.

1 also showed slightly higher ABS compared to other countries in subdomain 3 (across all conditions) despite the reported ABS' being less variable in that domain.

Nonetheless, most of the countries showed biases and ABS closer to 0 on

Figure 5.41

Subscale Score Bias for the 5-Domain, 5-Item, .75 Correlation Subdomain Tests: Multiple Groups

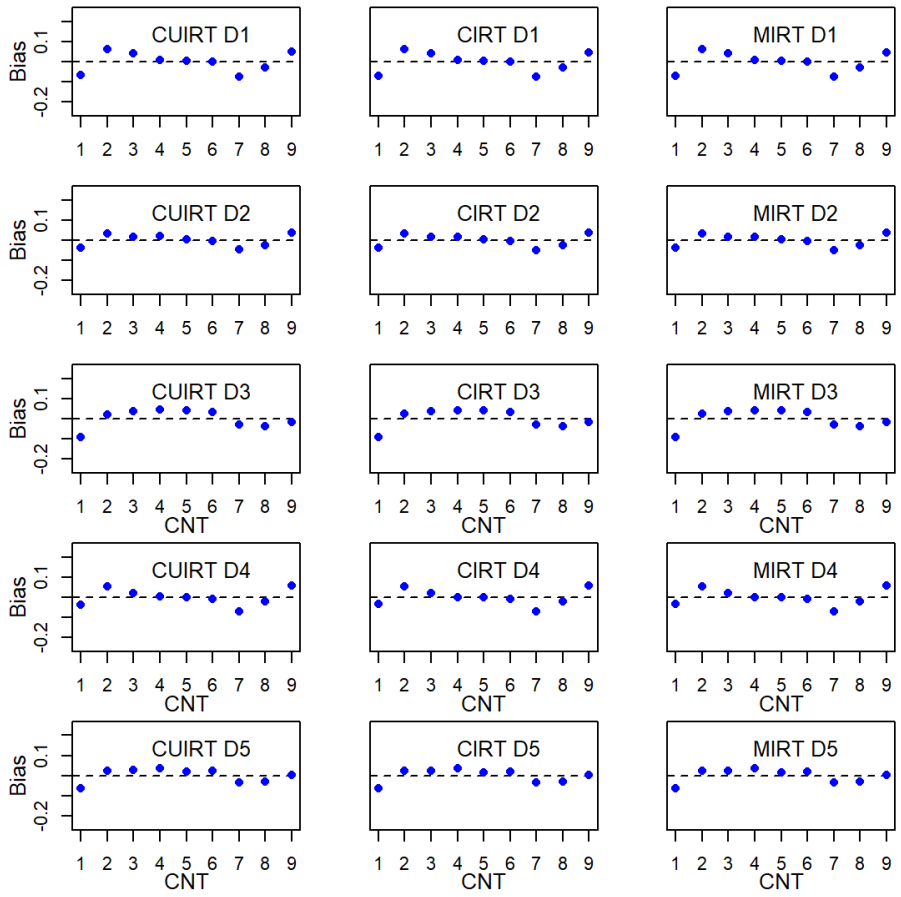


Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4; $D5$ = domain 5.

all other subdomains; with the lowest ABS across all countries being observed in domain 2. The results also suggested that the low performing countries had higher bias and ABS on domains 2 and 4 where $J = 5$ and $J = 15$. Over all

Figure 5.42

Subscale Score Bias for the 5-Domain, 5-Item, .95 Correlation Subdomain Tests: Multiple Groups

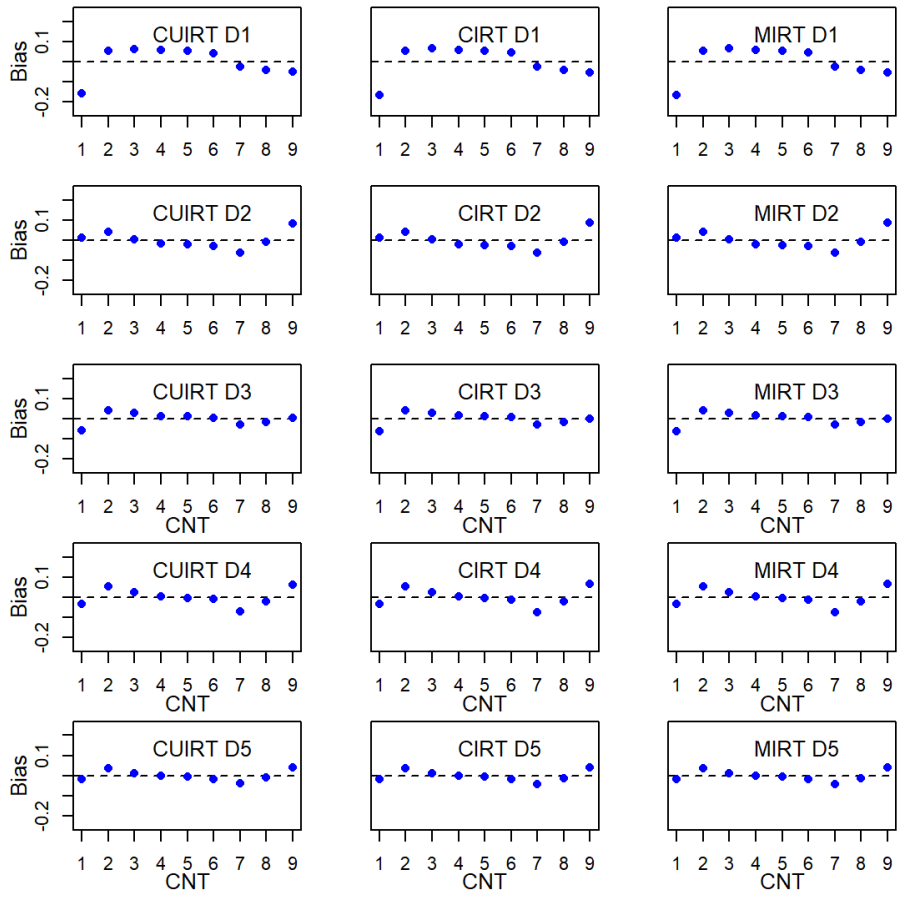


Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4; $D5$ = domain 5.

conditions, RMSEs slightly increased as the number of items per subdomain increased. However, it would be expected that longer tests have lower RMSE because with more items comes more information. As such, better population

Figure 5.43

Subscale Score Bias for the 5-Domain, 10-Item, .45 Correlation Subdomain Tests: Multiple Groups



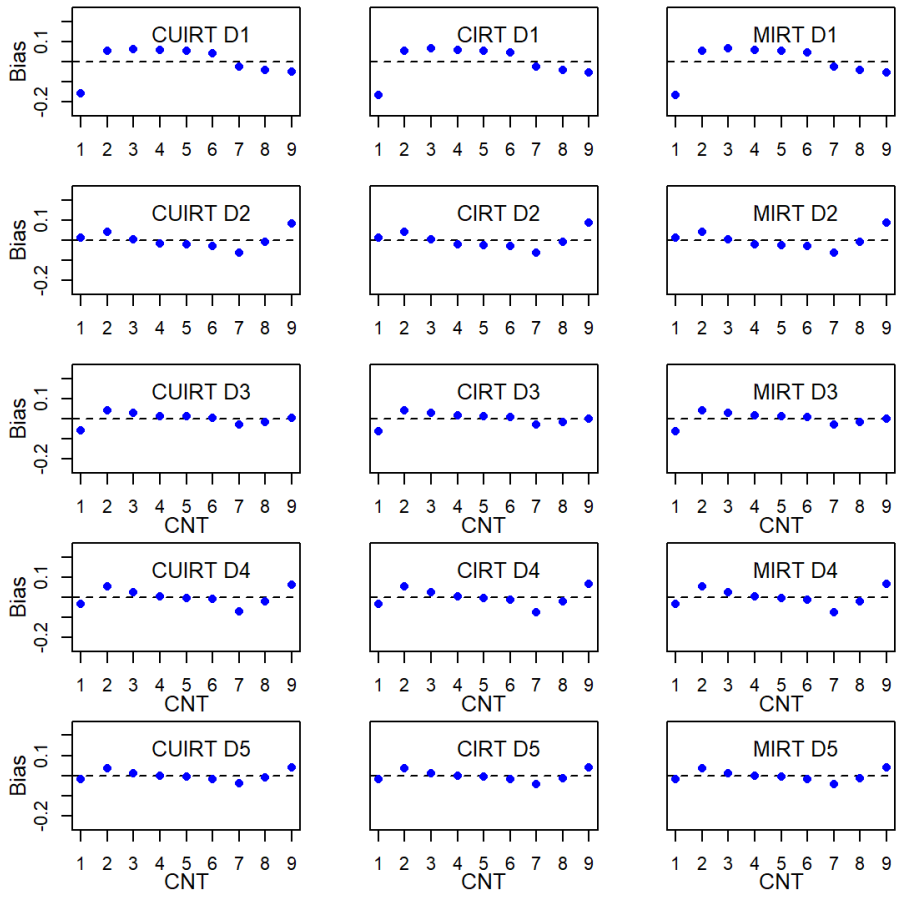
Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4; $D5$ = domain 5.

means would be recovered. But then, these high RMSEs may be confounded to specific conditions. That is, since different item parameters were used for the different subscale lengths, these results may be specific to condition.

All of the models reported the least biased score estimates on subdomains

Figure 5.44

Subscale Score Bias for the 5-Domain, 10-Item, .75 Correlation Subdomain Tests: Multiple Groups

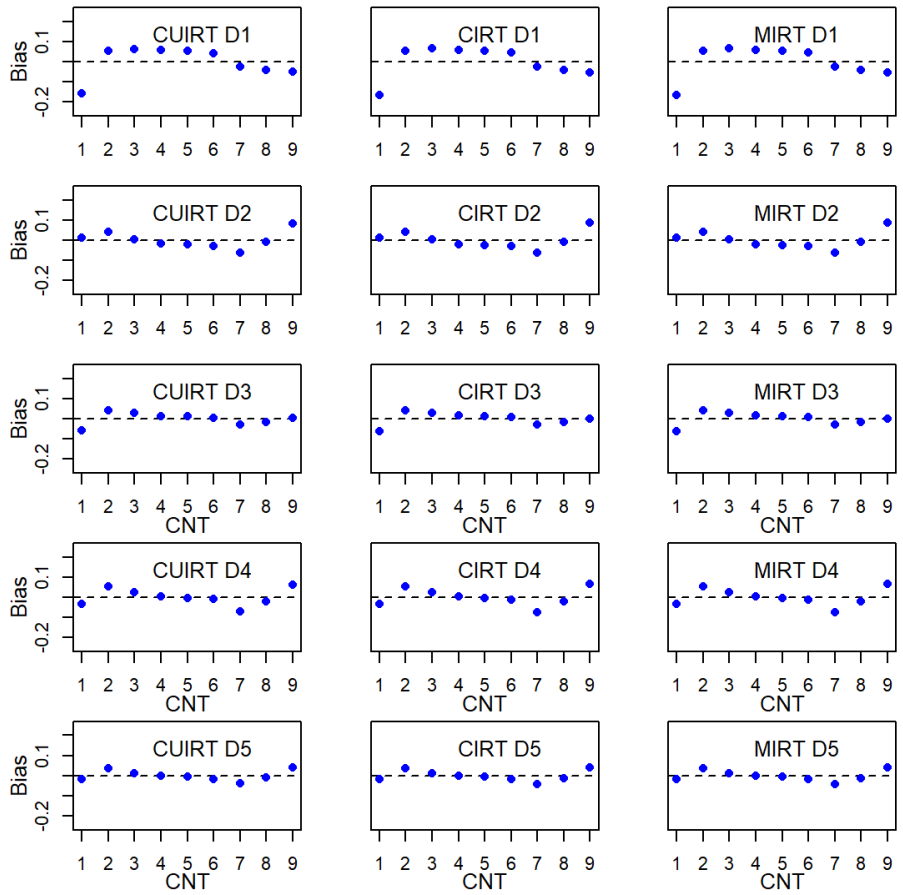


Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4; $D5$ = domain 5.

3 and 5 across the 5-subdomain test conditions for tests where subscale lengths were five- and 10-items per domain (see Figure 5.40 to 5.45). Further scrutiny of the average item-difficulty for Study 1's multiple groups simulation conditions

Figure 5.45

Subscale Score Bias for the 5-Domain, 10-Item, .95 Correlation Subdomain Tests: Multiple Groups

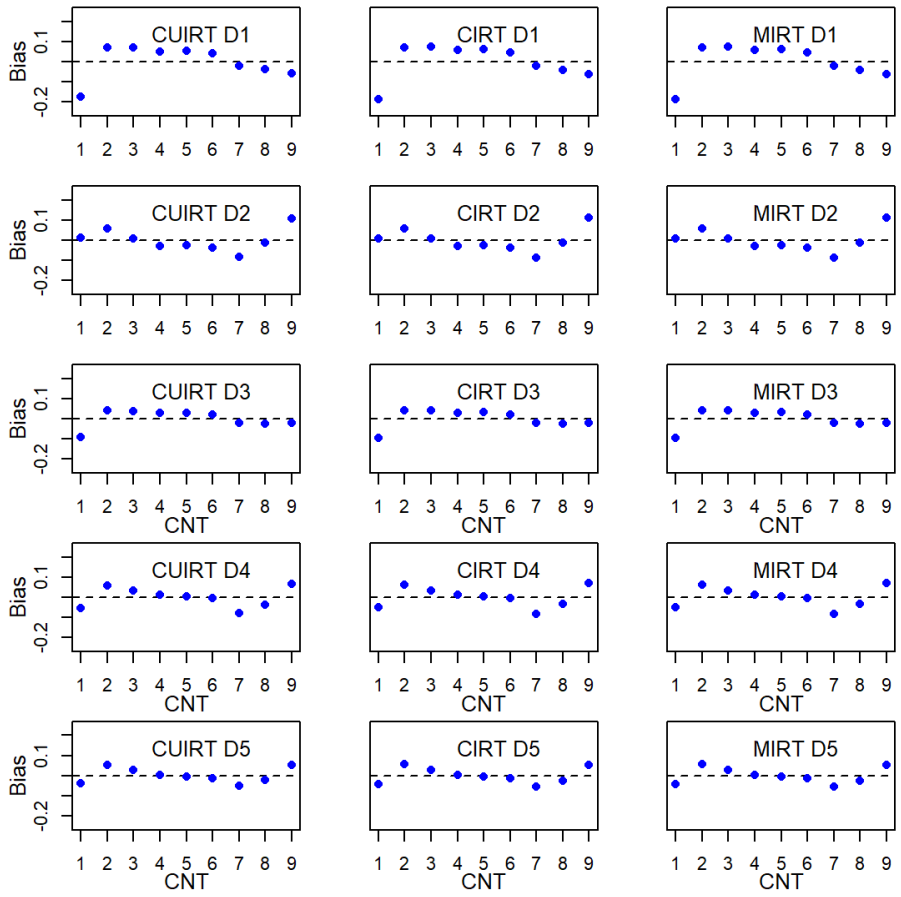


Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4; $D5$ = domain 5.

(see Table 5.8) showed that though all of the 5-subdomain test conditions had an average difficulty of around .01, the standard deviations were larger for the five- and 10-item subdomain tests. This suggests that the item-difficulties were

Figure 5.46

Subscale Score Bias for the 5-Domain, 15-Item, .45 Correlation Subdomain Tests: Multiple Groups



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4; $D5$ = domain 5.

more spread.

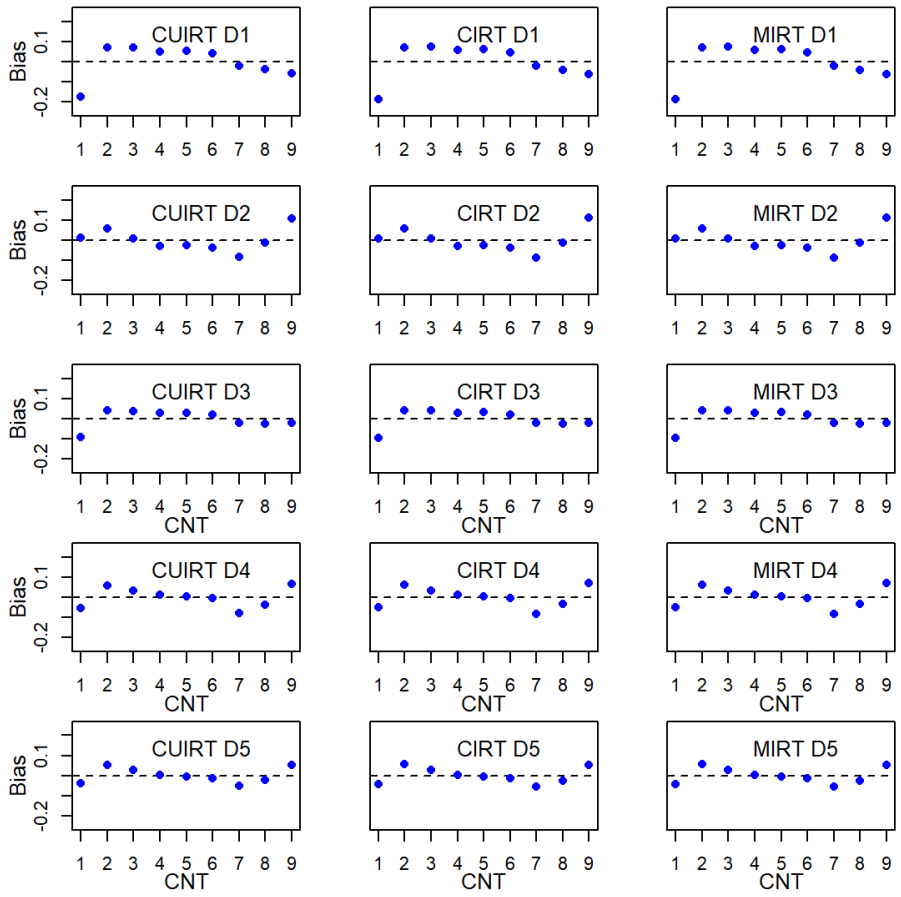
Table 5.8
Study 1 Average Item Difficulty of 5-Subdomain Tests: Multiple Groups

			Domain				
J	ρ	Model	1	2	3	4	5
5	.45	UIRT	.01 (.66)	.02 (1.15)	.01 (1.19)	.01 (.72)	.02 (1.24)
		CIRT	.01 (.70)	.02 (1.23)	.02 (1.27)	.01 (.77)	.02 (1.32)
		MIRT	.01 (.70)	.02 (1.23)	.01 (1.27)	.01 (.77)	.02 (1.33)
	.75	UIRT	.01 (.68)	.01 (1.19)	.02 (1.23)	.01 (.75)	.02 (1.28)
		CIRT	.01 (.70)	.02 (1.23)	.02 (1.27)	.01 (.77)	.02 (1.32)
		MIRT	.02 (.71)	.02 (1.25)	.02 (1.29)	.02 (.78)	.02 (1.34)
	.95	UIRT	.01 (.69)	.01 (1.22)	.01 (1.26)	.01 (.76)	.02 (1.31)
		CIRT	.01 (.70)	.02 (1.22)	.01 (1.27)	.01 (.77)	.02 (1.32)
		MIRT	.02 (.72)	.02 (1.26)	.02 (1.31)	.01 (.79)	.02 (1.36)
10	.45	UIRT	.01 (1.19)	.01 (1.33)	.01 (.99)	.01 (1.08)	.01 (1.23)
		CIRT	.01 (1.26)	.02 (1.41)	.01 (1.05)	.01 (1.14)	.01 (1.30)
		MIRT	.02 (1.26)	.02 (1.41)	.02 (1.05)	.01 (1.14)	.01 (1.30)
	.75	UIRT	.01 (1.23)	.01 (1.37)	.01 (1.03)	.01 (1.11)	.01 (1.27)
		CIRT	.01 (1.26)	.02 (1.41)	.02 (1.05)	.02 (1.14)	.01 (1.30)
		MIRT	.01 (1.27)	.02 (1.42)	.02 (1.06)	.02 (1.15)	.01 (1.31)
	.95	UIRT	.01 (1.25)	.01 (1.40)	.01 (1.05)	.01 (1.13)	.01 (1.30)
		CIRT	.01 (1.26)	.02 (1.41)	.01 (1.05)	.01 (1.14)	.01 (1.30)
		MIRT	.01 (1.28)	.01 (1.43)	.02 (1.07)	.01 (1.16)	.01 (1.32)
15	.45	UIRT	.01 (1.22)	.01 (1.14)	.01 (.98)	.01 (1.10)	.01 (1.12)
		CIRT	.01 (1.29)	.02 (1.20)	.01 (1.04)	.02 (1.16)	.01 (1.19)
		MIRT	.01 (1.29)	.02 (1.20)	.01 (1.04)	.01 (1.16)	.01 (1.19)
	.75	UIRT	.01 (1.26)	.01 (1.17)	.01 (1.01)	.01 (1.13)	.01 (1.16)
		CIRT	.01 (1.29)	.02 (1.20)	.01 (1.04)	.02 (1.16)	.01 (1.19)
		MIRT	.01 (1.30)	.02 (1.21)	.02 (1.04)	.01 (1.16)	.01 (1.19)
	.95	UIRT	.01 (1.29)	.01 (1.20)	.01 (1.03)	.01 (1.15)	.01 (1.18)
		CIRT	.01 (1.29)	.02 (1.20)	.01 (1.04)	.02 (1.16)	.01 (1.19)
		MIRT	.01 (1.31)	.01 (1.21)	.01 (1.05)	.01 (1.17)	.01 (1.20)

Note. J = subscale length; ρ = subscale correlation; the values in parentheses represent the standard deviations across replications.

Figure 5.47

Subscale Score Bias for the 5-Domain, 15-Item, .75 Correlation Subdomain Tests: Multiple Groups



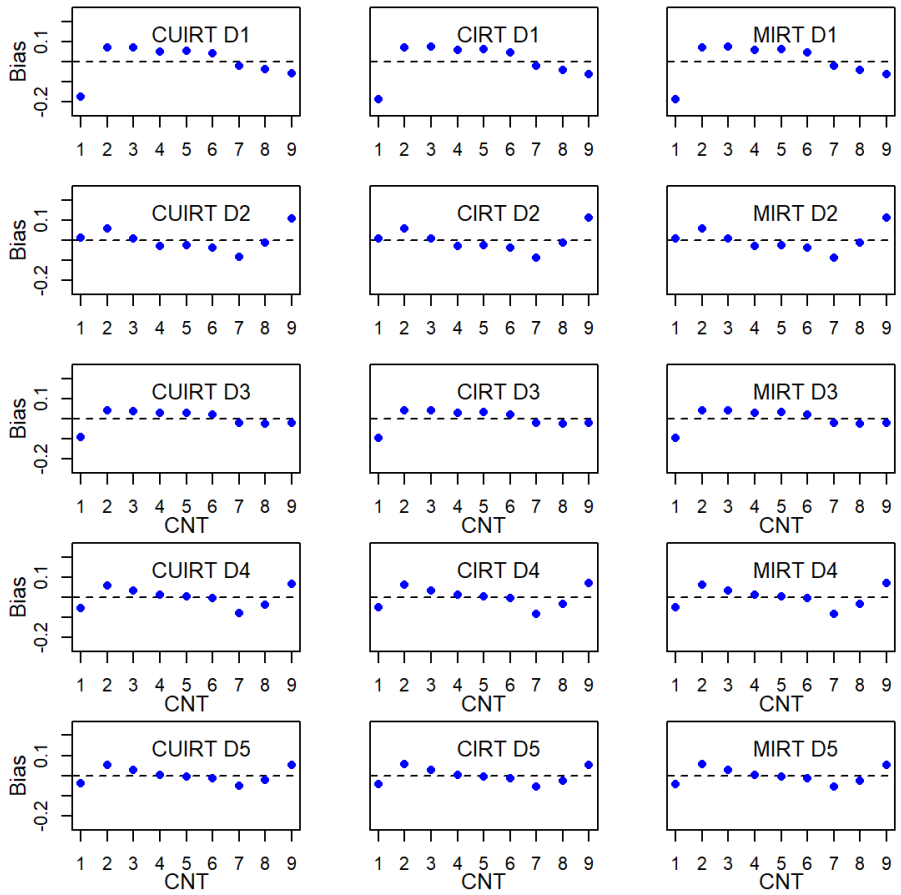
Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4; $D5$ = domain 5.

5.3.2.2 Subscale Length

In all the simulated conditions, bias and ABS were closer to 0 as the number of items per subdomain increased. In other words, there was less of a discrepancy

Figure 5.48

Subscale Score Bias for the 5-Domain, 15-Item, .95 Correlation Subdomain Tests: Multiple Groups



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4; $D5$ = domain 5.

between the estimated and true population scores. Also, the results seemed to indicate that bias and ABS suggested similar results (see Figure J.1 to J.16 in Appendix J). In this case, ABS was higher where bias was furthest from 0.

Bias, ABS and RMSE were also lower where the spread of the item-difficulty parameters was large.

5.3.3 Score Recovery for Study 2: Single Groups

In addition to the three models studied in Study 1 (i.e., CUIRT, CIRT, MIRT), Study 2 included the CUIRT-Op model which resembled how scores were estimated on TIMSS 2015 (see Section 3.3.4 in Chapter 3 for details). To estimate subscale scores, CUIRT-Op fixes item parameters estimated from CUIRT, and fits a MIRT model that estimates subscale correlations. Three and four domain-proficiency estimates from CUIRT, CUIRT-Op, CIRT, and MIRT were compared with true values, the generating population score estimates. Sections 5.3.3.1 and 5.3.3.2 present the results from the specified subscale correlations and lengths, respectively.

5.3.3.1 Subscale Correlation

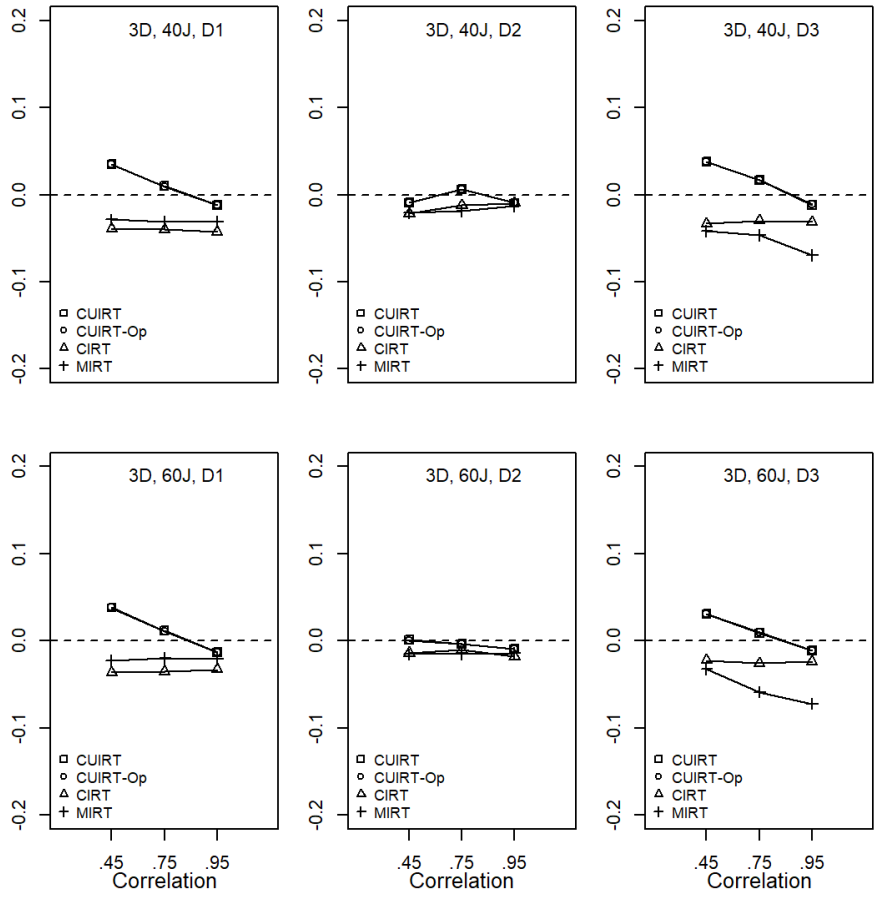
Three-Subdomain Tests.
Figure 5.49 plots the biases for all of the estimated subscale scores by domain. Figure K.1 and K.3 in Appendix K show plots of the corresponding ABS and RMSE, respectively. Figure 5.49 shows that CUIRT and CUIRT-Op bias were closest to 0 compared to CIRT and MIRT across all studied correlations in domain 2. Figure 5.49 shows that CUIRT and CUIRT-Op biases[15] were all closer to 0 on where subscale score correlation was high (i.e., .95). This was particularly evident on domains 1 and 2 for both the 40 and 60 subdomain tests. Whilst the CUIRT group of models showed improvements in domains 1 and 2 as subscale correlation increased, MIRT showed higher bias, ABS and RMSE scores as subdomain correlation increased ($\rho = .95$). Across all of the tested correlations, the evaluation criteria showed that CIRT was least sensitive to changes in subdomain correlation. The figures showed that the bias, ABS and RMSE observed from the CIRT model did not show sharp increases or decreases as subdomain correlation increased. CIRT had the lowest bias, ABS, and RMSE in domain in domain 3 where the subdomain correlations were low ($\rho = .45$).

Four-Subdomain Tests.
Figure 5.50 plots the biases for all of the estimated subscale scores by domain. Figures K.2 and K.4 in Appendix K show plots of the corresponding ABS and RMSE, respectively. The observed trends were similar to those observed

[15]The ABS and RMSE plots showed the same pattern. See Figures K.1 and K.3 in Appendix K.

Figure 5.49

Subscale Score Bias for the 3 Domain Subtests Tests: Single Groups

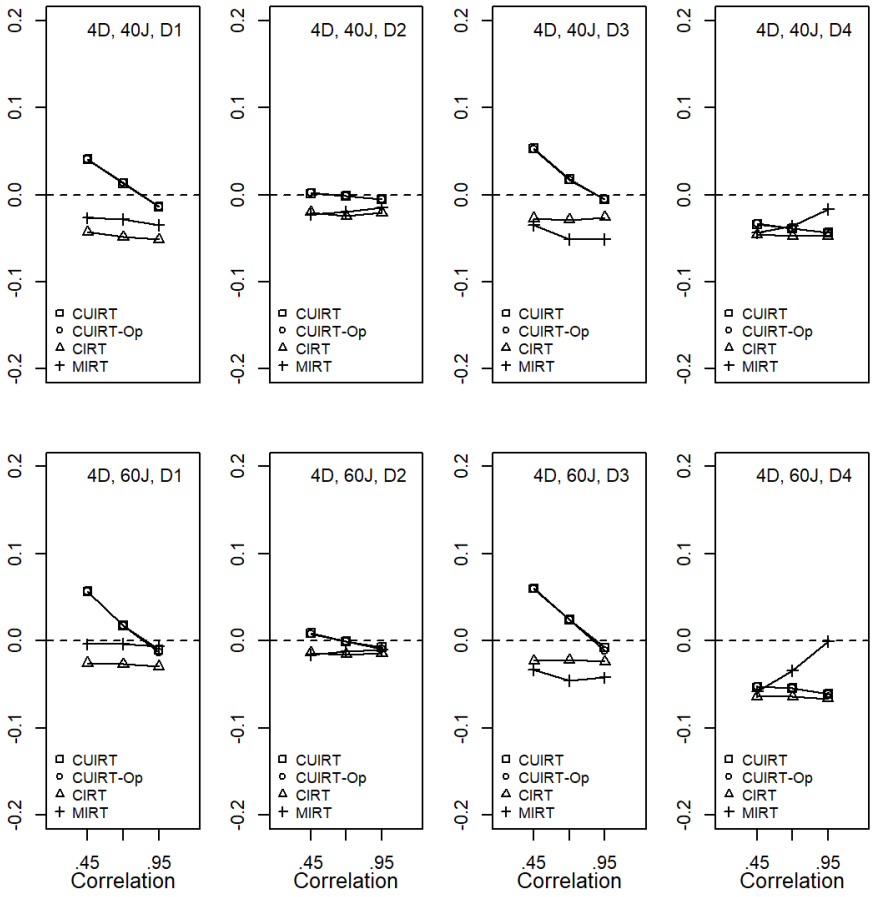


Note. D = number of domains; J = number of items; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

on the three domain tests. That is, CUIRT, CUIRT-Op, and MIRT reported subscale scores with the smallest bias.

Figure 5.50

Subscale Score Bias for the 4 Domain Subtests Tests: Single Groups



Note. D = number of domains; J = number of items; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4.

5.3.3.2 Subscale Length

The results presented in Figures 5.49 and 5.50 did not report any trends that were consistent for all subdomains as subscale length increased. For instance, in the 3- and 4-subdomain test condition, both 40- and 60-subscale

Table 5.9
Study 2 Average Item Discrimination of 3-Subdomain Tests by Subdomain: Single Groups

			Domain			
J	ρ	Model	1	2	3	4
40	.45	UIRT	.65 (.20)	.55 (.18)	.79 (.21)	.66 (.19)
		CIRT	.96 (.32)	.83 (.32)	1.09 (.31)	.93 (.28)
		MIRT	.84 (.27)	.87 (.33)	1.15 (.33)	1.00 (.30)
	.75	UIRT	.82 (.24)	.70 (.23)	.94 (.25)	.80 (.21)
		CIRT	.97 (.32)	.83 (.31)	1.09 (.32)	.93 (.28)
		MIRT	.79 (.25)	.83 (.30)	1.22 (.35)	1.14 (.34)
	.95	UIRT	.93 (.27)	.80 (.27)	1.05 (.28)	.90 (.24)
		CIRT	.97 (.33)	.83 (.32)	1.09 (.32)	.94 (.28)
		MIRT	.79 (.25)	.80 (.29)	1.24 (.35)	1.33 (.39)
60	.45	UIRT	.89 (.24)	.74 (.26)	.97 (.31)	.80 (.21)
		CIRT	1.05 (.30)	.87 (.33)	1.13 (.41)	.94 (.26)
		MIRT	.80 (.23)	.80 (.30)	1.27 (.45)	1.21 (.34)
	.75	UIRT	.89 (.24)	.74 (.26)	.97 (.31)	.80 (.21)
		CIRT	1.05 (.30)	.87 (.33)	1.13 (.41)	.94 (.26)
		MIRT	.80 (.23)	.80 (.30)	1.27 (.45)	1.21 (.34)
	.95	UIRT	1.00 (.27)	.85 (.30)	1.09 (.36)	.90 (.24)
		CIRT	1.05 (.31)	.88 (.33)	1.14 (.41)	.94 (.27)
		MIRT	.78 (.22)	.79 (.29)	1.23 (.43)	1.37 (.38)

Note. J = subscale length; ρ = subscale correlation; the values in parentheses represent the standard deviations across replications.

test conditions did not consistently show improvement of the score estimates due to an increase or decrease in subscale length. The figures showed that CUIRT reported the best score estimates on all domains in the test conditions except in domain 4 on all 4 subdomain test conditions with a subscale correlation of .95, where MIRT performed better than CUIRT and CIRT. The average item discrimination parameters presented in Table 5.9 show that MIRT produced large item discriminations on domain 4 of the 4-subdomain test conditions where subscale correlation was .95. In this condition MIRT's performance also improved as subscale length increased.

5.3.4 Score Recovery for Study 2: Multiple Groups

Figure 5.51 to 5.62 plot the biases of each country's model specific subdomain score. The ABS and RMSE results are presented in Appendix L. Like all other plots outlining the three evaluation criteria, each row (a) represents a specific domain; three and five depending on number of subdomains; (b) has four subplots that show the results from the four studied models: CUIRT, CUIRT-Op, CIRT and MIRT.

5.3.4.1 Subscale Correlation

Three-Subdomain Tests.
The studied models did not show alarming differences with respect to the observed patterns. The results showed that the middle performing country scores had larger bias than the high- and low-performers. The scores for the middle performing countries were consistently underestimated whereas those of the high- and low-performing countries were shown to be overestimated. On the first and second subdomains, the middle performing countries showed high biases, between -.11 and -.19, depending on test condition. On the second subdomain, the biases were between -.06 and -.08. Scores for the middle performers also showed the largest ABS across all studied conditions.

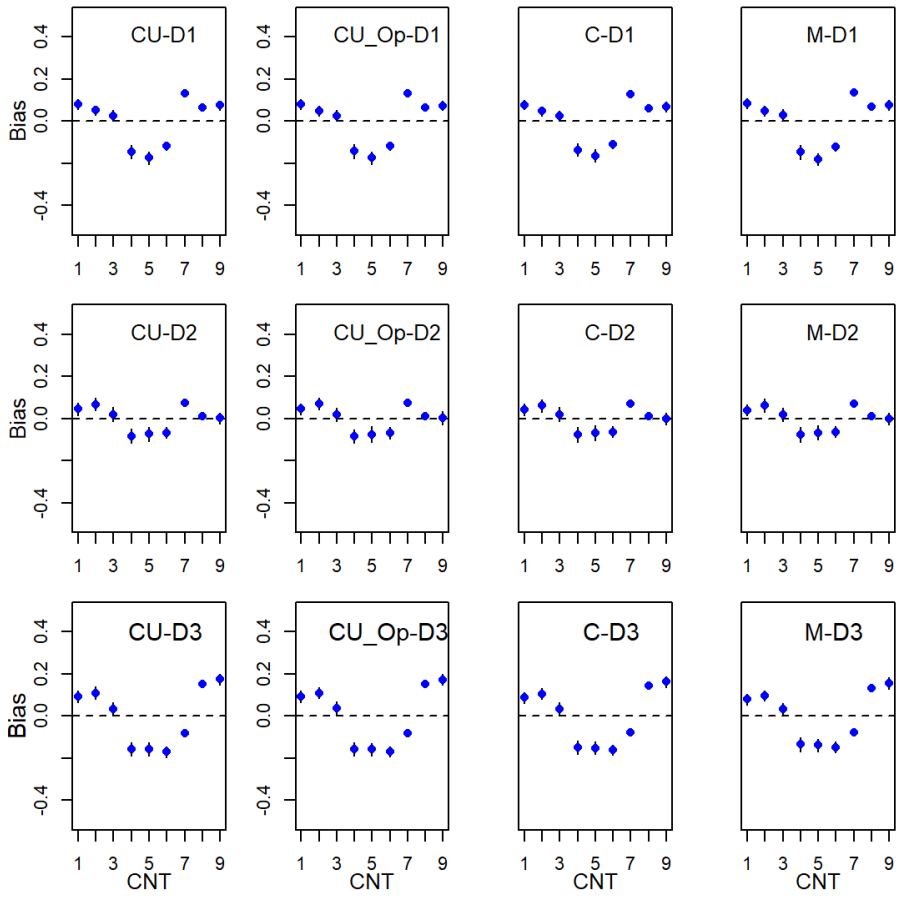
On Subdomain 2 of all 3 subdomain tests, MIRT scores showed the least deviation from the true values regardless of test length where subscale correlation was .45. For example, the MIRT biases ranged from -.06 to .06 in subdomain 2 on the 40-item-per-subdomain-tests as opposed to values between -.08 to .07 for CUIRT, -.08 to .08 for CUIRT-Op, and -.08 to .07 for CIRT. MIRT also produced a shorter range of biases on subdomain 3 regardless of test length, where subscale correlation was .45. In contrast, the CIRT had the shortest range of biases (i.e., between -.10 and -.08) on the 60-item-per-subdomain-tests. CIRT also produced lower biases on subdomain 1 under the same subscale correlation.

On the 40-item-per-subdomain tests, all of the studied models reported comparable biases on subscale 2 regardless of test length where subscale correlations were .75 and .95. The same results were observed on subdomain 3 where subscale correlation was .75. In contrast, CUIRT and CUIRT-Op produced ranges of bias closest to 0 where subscale score correlation was .75. CUIRT also showed a lower range of bias where subscale correlation was .95. CUIRT, CUIRT-Op, and CIRT all resulted in comparable, smaller range, biases where subscale correlation was .95.

On the 60-item-per-subdomain tests, all of the studied models reported comparable biases on subscale 3, regardless of test length, where subscale

Figure 5.51

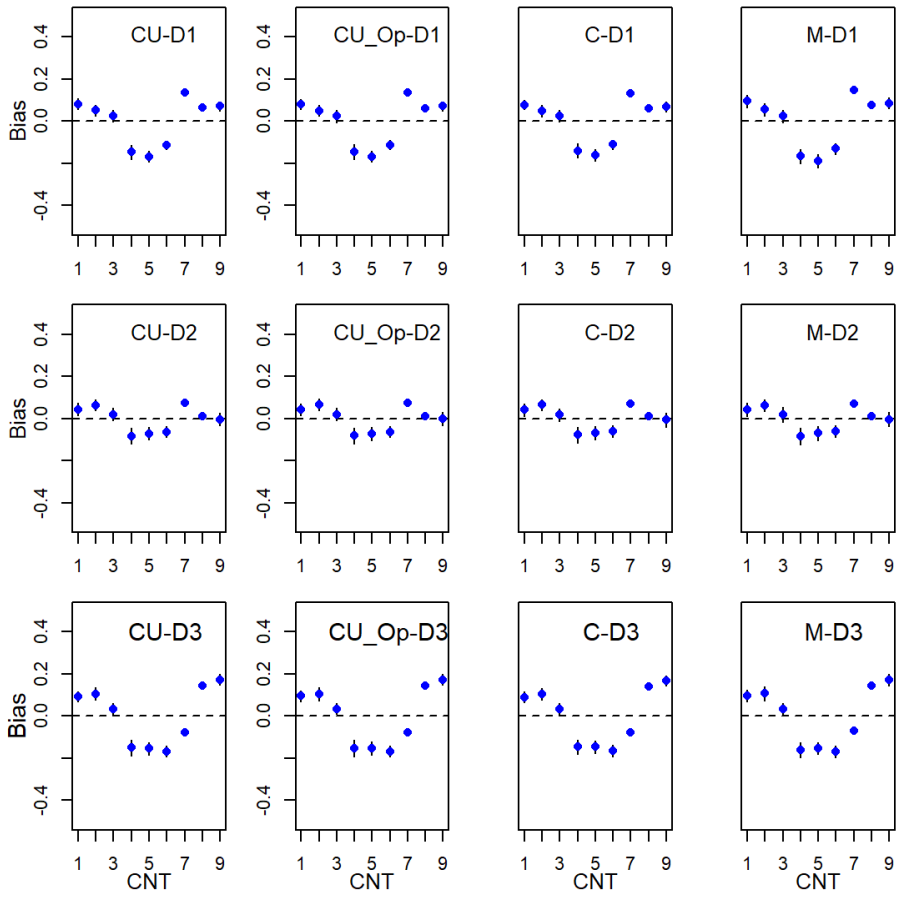
Subscale Score Bias for the 3-Domain, 40-item, .45 Correlation Subdomain Tests: Multiple Groups



Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

correlations were .75. CIRT had the lowest range of bias on subdomain 1 where subscale correlations were .75 and .95. CIRT also had biases closest to 0 (based on the range across all countries) on subdomain 2 where subscale correlation

Figure 5.52
Subscale Score Bias for the 3-Domain, 40-item, .75 Correlation Subdomain Tests: Multiple Groups

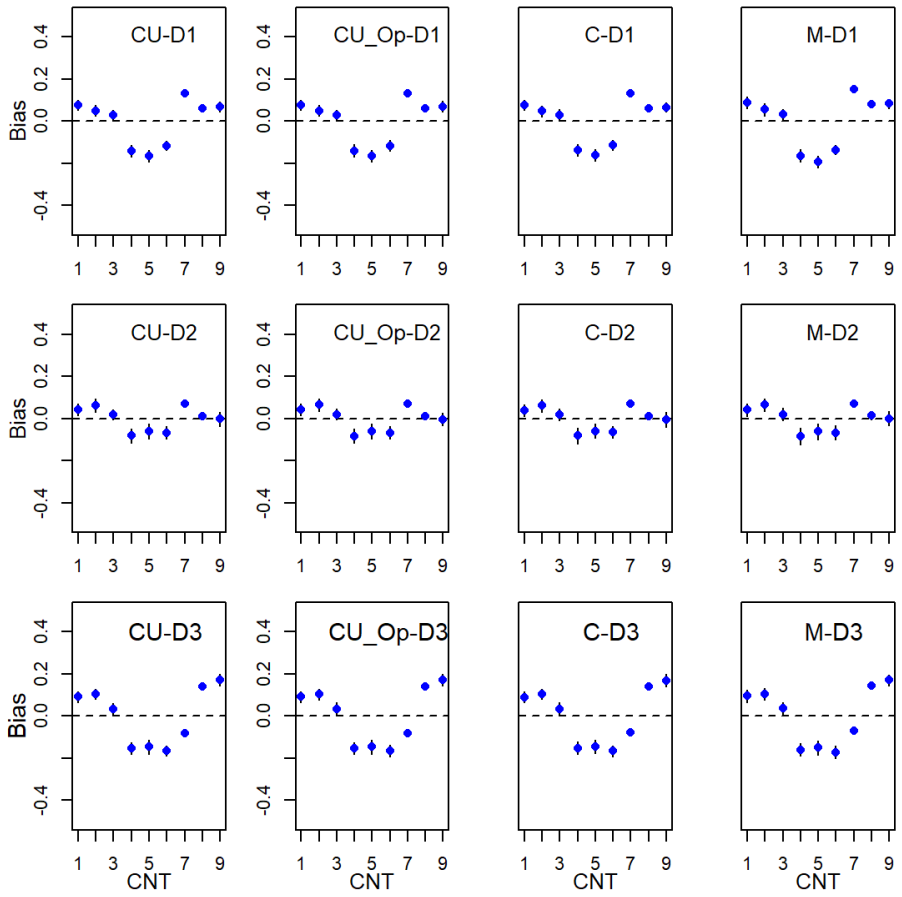


Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

was .75. CUIRT and CIRT all resulted in comparable, smaller range, biases where subscale correlation was .95.

To gain a deeper understanding as to why subdomain 2 showed the lowest biases for all countries, I examined the item parameters. First, subdomain

Figure 5.53
Subscale Score Bias for the 3-Domain, 40-item, .95 Correlation Subdomain Tests: Multiple Groups

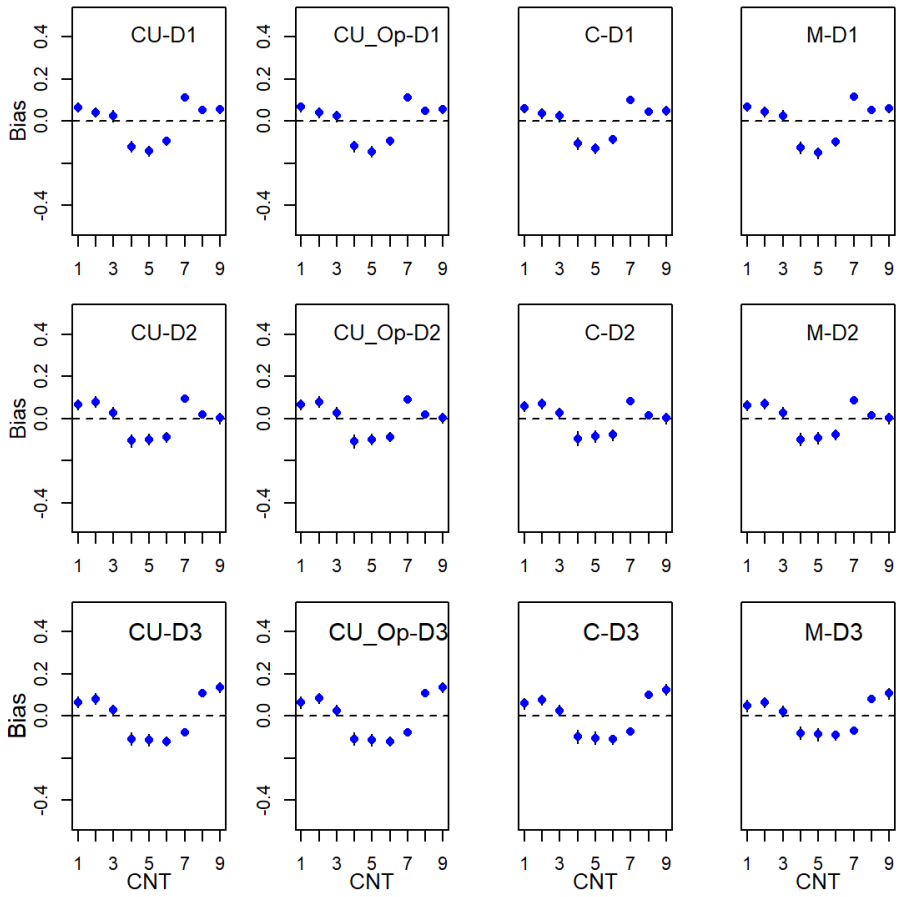


Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

2 reported the large average item discrimination; ranges between $[1.33, 1.62]$ and $[1.50, 1.81]$ on the 40- and 60-items-per-subdomain tests, respectively (see Table 5.10). According to DeAyala (2013, p. 101), "good" values of

Figure 5.54

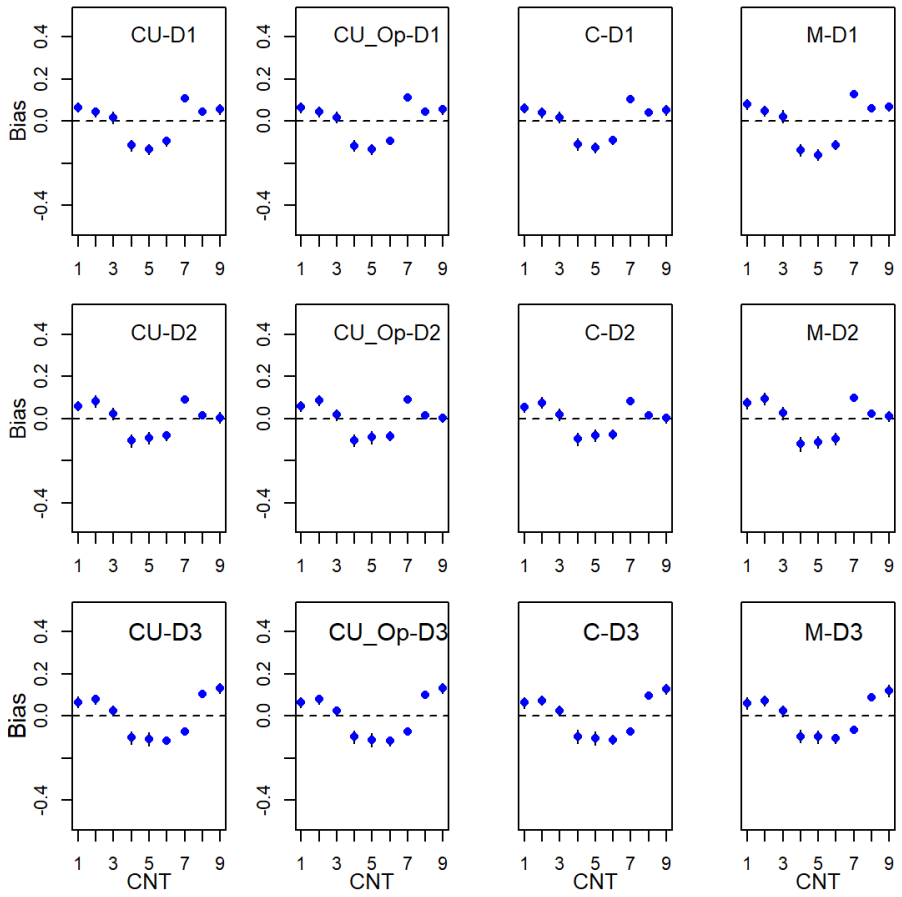
Subscale Score Bias for the 3-Domain, 60-item, .45 Correlation Subdomain Tests: Multiple Groups



Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

discrimination parameters range from approximately .8 to 2.5. Second, it was observed that the average item difficulties were the lowest on subdomain 2 for all four-subdomain conditions. The average item difficulties for each of the

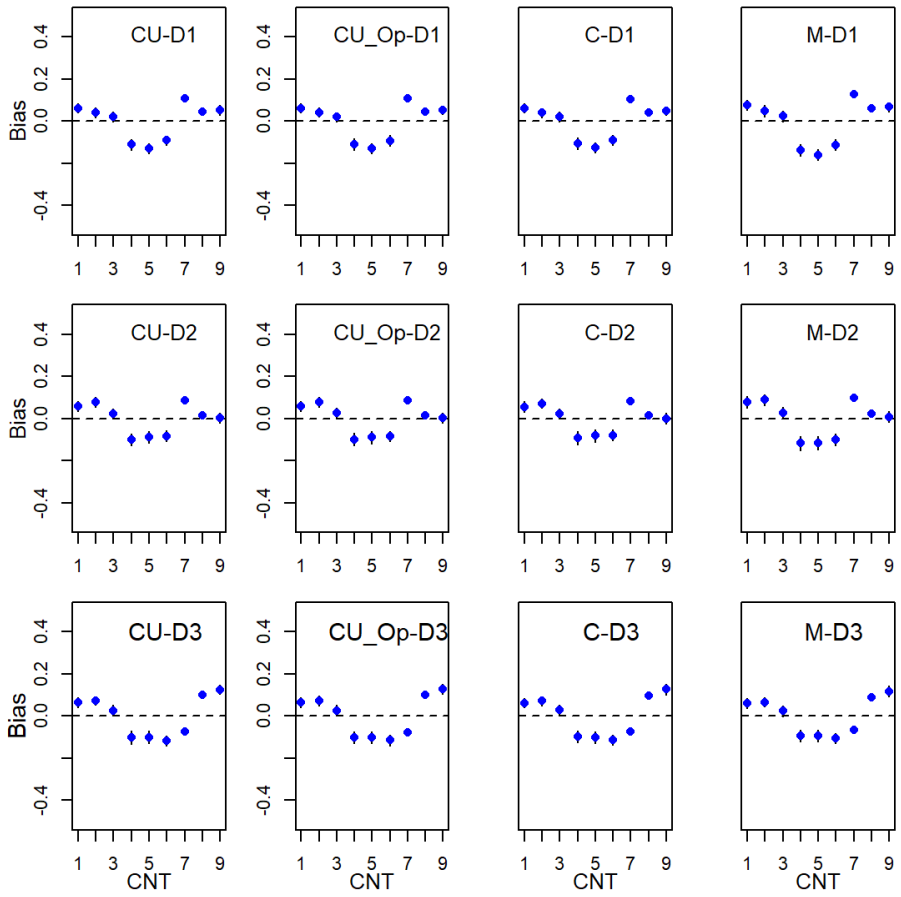
Figure 5.55
Subscale Score Bias for the 3-Domain, 60-item, .75 Correlation Subdomain Tests: Multiple Groups



Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

conditions ranged between [.82, .90] and [1.02, 1.14] on the 40- and 60-items-per-subdomain tests, respectively (see Table 5.11). In contrast, the average item difficulties for all other subdomains were greater than 1.14. This suggests that

Figure 5.56
Subscale Score Bias for the 3-Domain, 60-item, .95 Correlation Subdomain Tests: Multiple Groups



Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

subdomains 1 and 3 were more difficult than subdomain 2 which was easier. As such, more information may have been collected from subdomain 2 because the items were accessible to all the candidates on the proficiency continuum.

Table 5.10
Study 2 Average Item Discrimination of 3-Subdomain Tests by Subdomain: Multiple Groups

			Domain		
J	ρ	Model	1	2	3
40	.45	UIRT	1.71 (.45)	1.33 (.41)	1.52 (.45)
		CIRT	2.03 (.57)	1.62 (.59)	1.82 (.63)
		MIRT	1.54 (.43)	1.60 (.58)	2.33 (.81)
	.75	UIRT	1.84 (.49)	1.45 (.47)	1.65 (.52)
		CIRT	2.03 (.57)	1.62 (.59)	1.82 (.62)
		MIRT	1.50 (.42)	1.57 (.57)	2.42 (.83)
	.95	UIRT	1.93 (.52)	1.53 (.52)	1.75 (.58)
		CIRT	2.04 (.57)	1.62 (.59)	1.82 (.62)
		MIRT	1.48 (.41)	1.55 (.57)	2.47 (.86)
60	.45	UIRT	1.56 (.38)	1.50 (.49)	1.46 (.37)
		CIRT	1.87 (.51)	1.81 (.67)	1.72 (.48)
		MIRT	1.38 (.37)	1.72 (.63)	2.22 (.62)
	.75	UIRT	1.68 (.43)	1.62 (.55)	1.57 (.41)
		CIRT	1.88 (.51)	1.81 (.67)	1.72 (.48)
		MIRT	1.34 (.36)	1.66 (.60)	2.30 (.64)
	.95	UIRT	1.78 (.47)	1.71 (.61)	1.65 (.45)
		CIRT	1.88 (.51)	1.80 (.67)	1.72 (.48)
		MIRT	1.31 (.35)	1.62 (.59)	2.35 (.65)

Note. J = subscale length; ρ = subscale correlation; the values in parentheses represent the standard deviations across replications.

Table 5.11
Study 2 Average Item Difficulty of 3-Subdomain Tests by Subdomain: Multiple Groups

			Domain		
J	ρ	Model	1	2	3
40	.45	UIRT	1.40 (.65)	.82 (.58)	1.24 (.76)
		CIRT	1.52 (.72)	.90 (.65)	1.38 (.90)
		MIRT	1.53 (.71)	.90 (.66)	1.38 (.92)
	.75	UIRT	1.45 (.68)	.85 (.61)	1.30 (.81)
		CIRT	1.52 (.72)	.90 (.66)	1.37 (.89)
		MIRT	1.53 (.71)	.90 (.67)	1.37 (.92)
	.95	UIRT	1.48 (.70)	.87 (.63)	1.34 (.87)
		CIRT	1.52 (.72)	.90 (.65)	1.38 (.90)
		MIRT	1.53 (.71)	.90 (.67)	1.38 (.93)
60	.45	UIRT	1.35 (.63)	1.02 (.63)	1.14 (.70)
		CIRT	1.46 (.71)	1.14 (.73)	1.24 (.79)
		MIRT	1.47 (.70)	1.12 (.72)	1.22 (.80)
	.75	UIRT	1.40 (.67)	1.06 (.66)	1.18 (.74)
		CIRT	1.46 (.71)	1.14 (.73)	1.24 (.80)
		MIRT	1.48 (.70)	1.12 (.72)	1.22 (.81)
	.95	UIRT	1.44 (.69)	1.09 (.69)	1.21 (.78)
		CIRT	1.46 (.71)	1.14 (.73)	1.24 (.79)
		MIRT	1.48 (.70)	1.12 (.72)	1.22 (.80)

Note. J = subscale length; ρ = subscale correlation; the values in parentheses represent the standard deviations across replications.

Four-Subdomain Tests.
The studied models showed patterns and trends similar to those observed in the 3 subdomain tests. That is, the middle performing country scores showed more bias than the high- and low-performers. The scores for the middle performing countries were consistently underestimated whereas those of the high- and low-performing countries were shown to be overestimated. However, all countries showed the least bias and ABS (closest to 0) in subdomains 4 on the 4-subdomain test conditions. On subdomain 4, MIRT showed the least bias, ABS and RMSE. The range of biases was smaller and closer to 0 on MIRT, than they were on CUIRT, CUIRT-Op, and CIRT. In other words, MIRT performed better on these subdomains regardless of subscale length and subscale correlation.

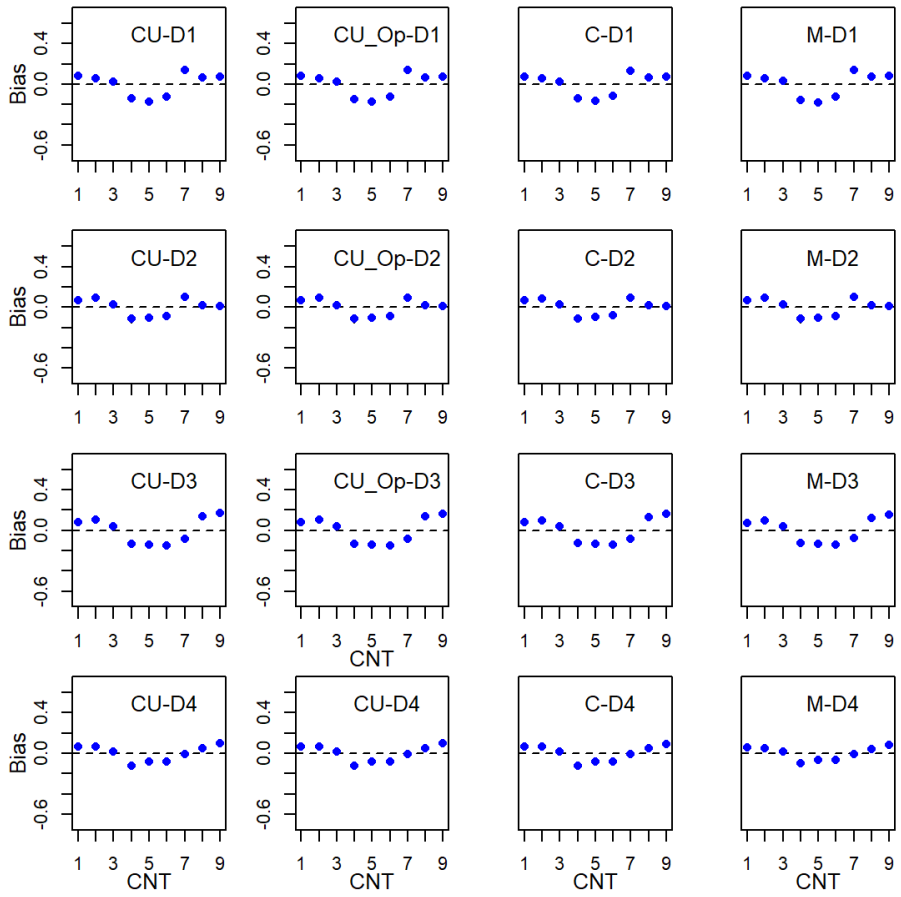
Though the biases were larger on the other subdomains, CIRT performed better on the first and second subdomains. In other words, the biases of all of the country scores reported from the CIRT model had a smaller range and were closer to 0, regardless of test length and subscale correlation. In contrast, MIRT performed better on subdomain 3 in all of the 4-subdomain test conditions. The results from the four subdomain tests also suggested that bias, ABS and RMSE improved as (a) subscale score correlation increased, and (b) tests with more items per subdomain showed evaluation criteria closer to 0.

To gain a deeper understanding as to why subdomain 4 showed the lowest biases for all countries, I examined the item parameters. First, subdomain 4 reported the largest average item discrimination; ranges between $[1.55, 2.83]$ and $[1.66, 2.82]$ on the 40- and 60-items-per-subdomain tests, respectively (see Table 5.12). DeAyala (2013) and Hambleton and Swaminathan (2013) noted that the larger the item discrimination, the greater the the information. More information translates in better estimation of proficiency scores (i.e., subscale score estimates with lower bias). Second, it was observed that the average item difficulties were the lowest on subdomain 4 for all four-subdomain conditions. The average item difficulties for each of the conditions ranged between $[.86, .94]$ and $[.83, .87]$ on the 40- and 60-items-per-subdomain tests, respectively (see Table 5.13). In contrast, the average item difficulties for all other subdomains were greater than .96. This suggests that subdomains 1 to 3 were more difficult than subdomain 4. As a result, less information may have been collected from subdomains 1 and 3 since items may have been too difficult for the populations.

5.3.4.2 Subscale Length

Figure 5.51 to 5.62 did not report any trends that were consistent for all subdomains as subscale length increased. That is, all of the subscale score

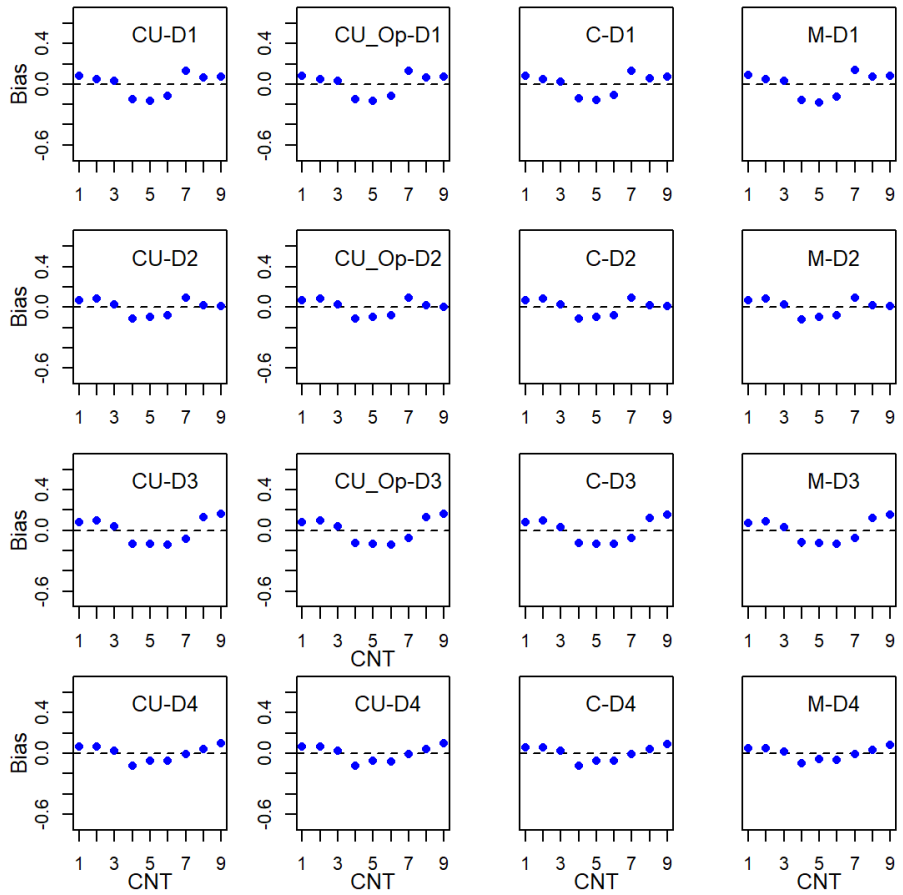
Figure 5.57
Subscale Score Bias for the 4-Domain, 40-item, .45 Correlation Subdomain Tests: Multiple Groups



Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4.

estimation models were not sensitive to the change in subscale length (from 40- to 60-item subscales).

Figure 5.58
Subscale Score Bias for the 4-Domain, 40-item, .75 Correlation Subdomain Tests: Multiple Groups



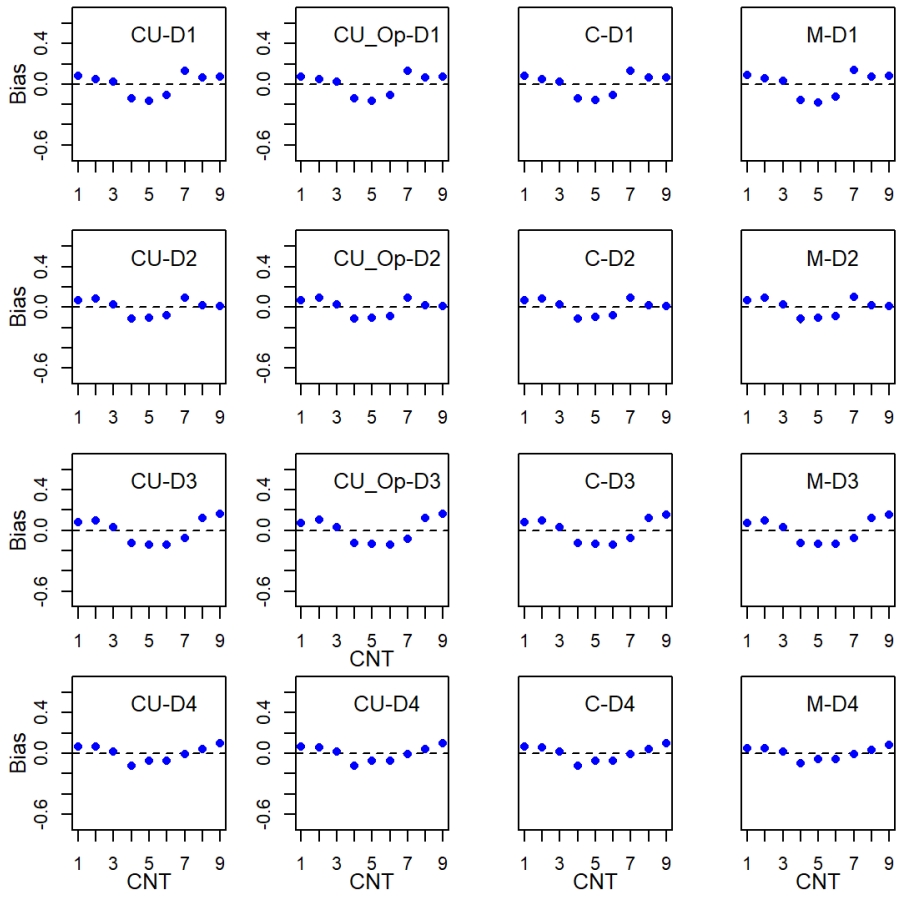
Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4.

5.3.5 Synthesis of Score Recovery

The results presented in Section 5.3 suggested that the studied subscale score estimation models were sensitive to subscale correlation regardless of the

Figure 5.59

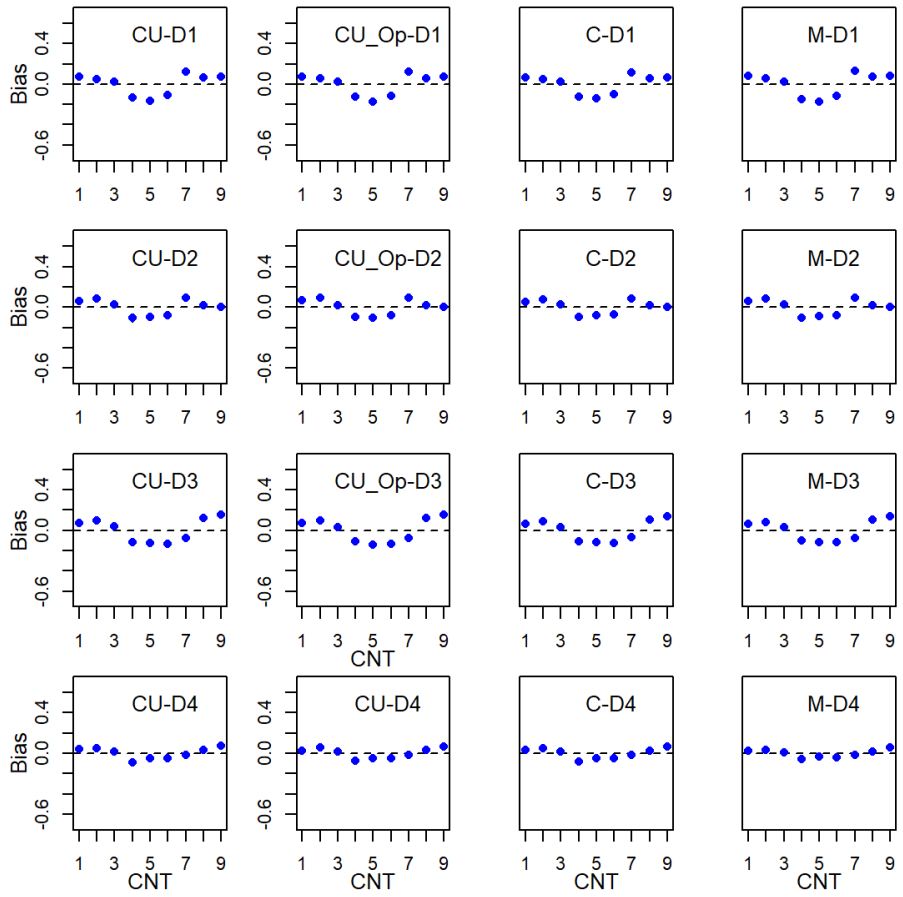
Subscale Score Bias for the 4-Domain, 40-item, .95 Correlation Subdomain Tests: Multiple Groups



Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4.

number of subscales. That is, some models were likely to report better score estimates than others under certain scubscale correlations. In general, the models performed comparatively at different subscale lengths regardless of the

Figure 5.60
Subscale Score Bias for the 4-Domain, 60-item, .45 Correlation Subdomain Tests: Multiple Groups



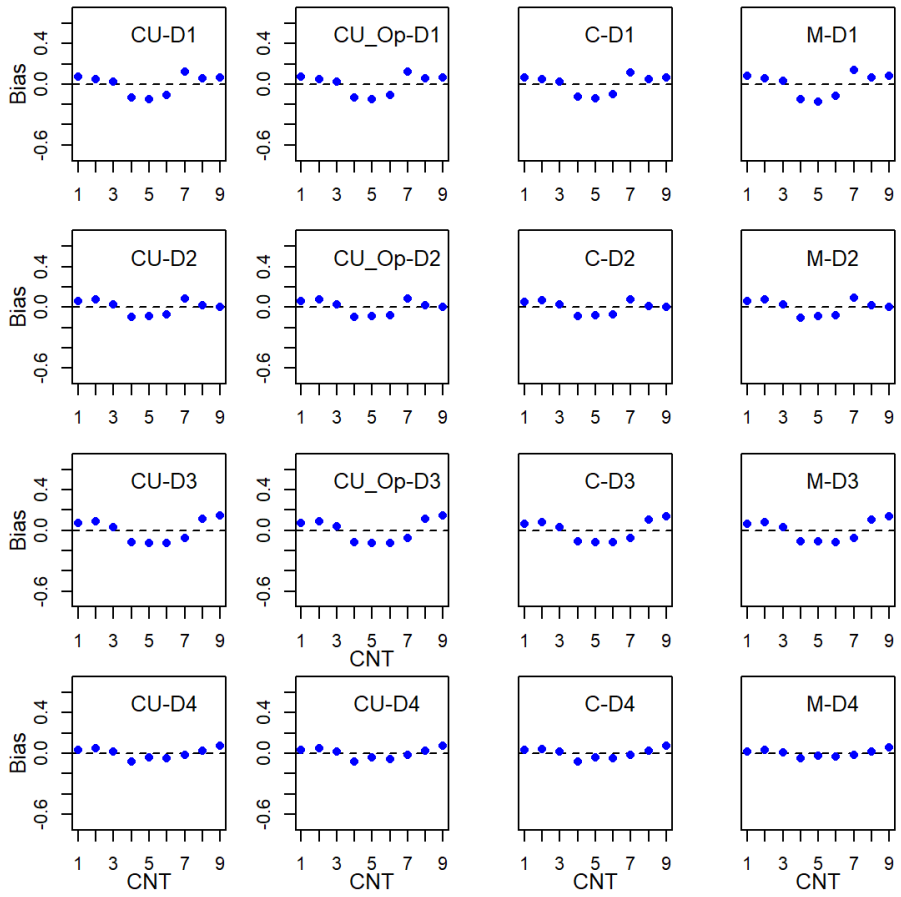
Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4.

number of subscales.

It would have been expected that since MIRT was the generating model, the model would have resulted in better item parameter estimates over all simulated test conditions. However, the results presented in Section 5.2 showed that this

Figure 5.61

Subscale Score Bias for the 4-Domain, 60-item, .75 Correlation Subdomain Tests: Multiple Groups

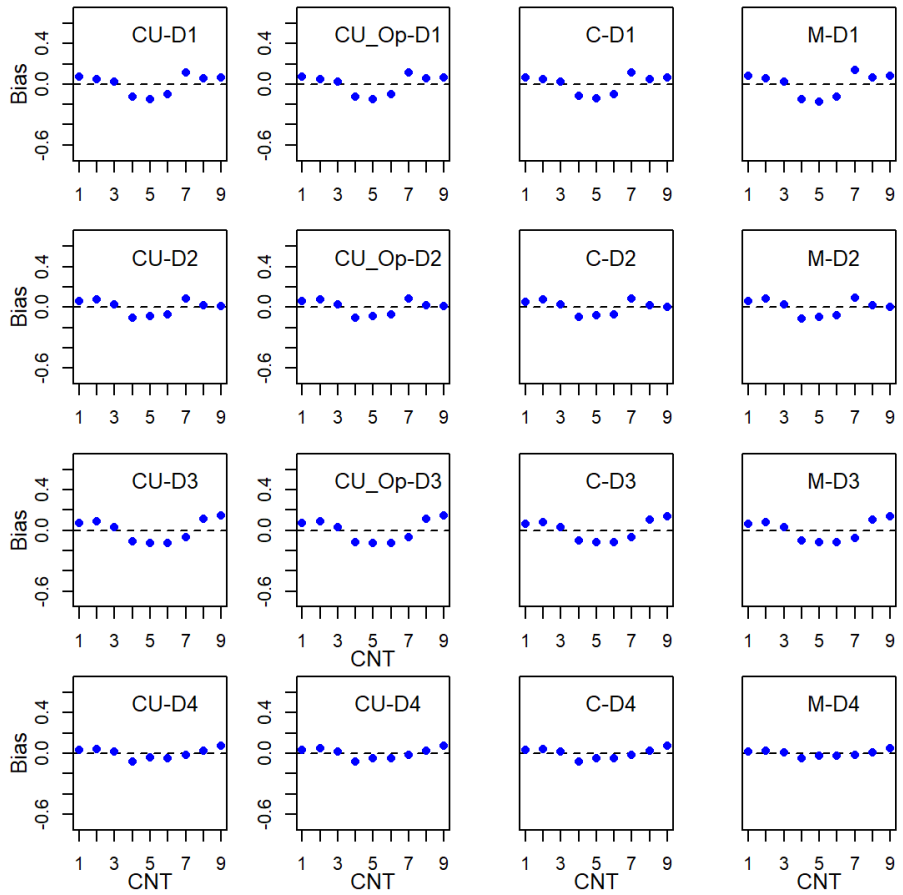


Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4.

was not always the case. That is, MIRT was not the best performing model across all simulated yest conditions. For example, CUIRT and CIRT performed better than MIRT where subscale correlation was .95. One likely reason that

Figure 5.62

Subscale Score Bias for the 4-Domain, 60-item, .95 Correlation Subdomain Tests: Multiple Groups



Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4.

this was the case is that the underlying "true" data generating mechanism of the chosen data where subscale correlations were high represented a more unidimensional model. That is, since the subscales were highly correlated,

Table 5.12
Study 2 Average Item Discrimination of 4-Subdomain Tests by Subdomain: Multiple Groups

			Domain			
J	ρ	Model	1	2	3	4
40	.45	CUIRT	1.58 (.41)	1.41 (.39)	1.49 (.34)	1.55 (.37)
		CIRT	1.98 (.57)	1.73 (.57)	1.84 (.48)	1.89 (.50)
		MIRT	1.41 (.40)	1.48 (.48)	1.95 (.50)	2.60 (.69)
	.75	CUIRT	1.74 (.46)	1.54 (.46)	1.63 (.38)	1.69 (.41)
		CIRT	1.99 (.58)	1.73 (.57)	1.85 (.48)	1.89 (.50)
		MIRT	1.37 (.39)	1.44 (.48)	1.91 (.49)	2.75 (.72)
	.95	CUIRT	1.84 (.51)	1.63 (.51)	1.72 (.42)	1.79 (.45)
		CIRT	2.00 (.58)	1.73 (.57)	1.86 (.49)	1.89 (.50)
		MIRT	1.35 (.39)	1.42 (.48)	1.88 (.48)	2.83 (.75)
60	.45	CUIRT	1.80 (.45)	1.58 (.57)	1.55 (.49)	1.66 (.39)
		CIRT	2.08 (.56)	1.79 (.71)	1.79 (.63)	1.86 (.48)
		MIRT	1.35 (.36)	1.42 (.56)	1.77 (.60)	2.73 (.70)
	.75	CUIRT	1.80 (.45)	1.58 (.57)	1.55 (.49)	1.66 (.39)
		CIRT	2.08 (.56)	1.79 (.71)	1.79 (.63)	1.86 (.48)
		MIRT	1.35 (.36)	1.42 (.56)	1.77 (.60)	2.73 (.70)
	.95	CUIRT	1.91 (.50)	1.68 (.64)	1.64 (.55)	1.76 (.43)
		CIRT	2.10 (.57)	1.78 (.71)	1.81 (.64)	1.85 (.47)
		MIRT	1.32 (.35)	1.38 (.55)	1.73 (.59)	2.82 (.72)

Note. J = subscale length; ρ = subscale correlation; the values in parentheses represent the standard deviations across replications.

they may inherently be measuring the same construct thus rendering the data unidimensional. Indeed, at such high correlations, the data that were generated in the DGP may have essentially been unidimensional (UIRT) as opposed to being multidimensional (MIRT).

5.3.5.1 Study 1

The results presented in Study 1 were not consistent across subdomains. For example, the results presented in Figure 5.30 showed that some domains had a positive bias (domain 4) and some exhibited negative bias (domain 1). One other observation from Study 1's single groups, 5-subdomain test conditions,

Table 5.13
Study 2 Average Item Difficulty of 4-Subdomain Tests by Subdomain: Multiple Groups

			Domain			
J	ρ	Model	1	2	3	4
40	.45	CUIRT	1.38 (.68)	.98 (.61)	1.14 (.68)	.86 (.57)
		CIRT	1.53 (.81)	1.08 (.70)	1.25 (.78)	.94 (.65)
		MIRT	1.55 (.80)	1.07 (.70)	1.24 (.79)	.94 (.67)
	.75	CUIRT	1.44 (.73)	1.01 (.64)	1.18 (.72)	.89 (.61)
		CIRT	1.53 (.81)	1.08 (.70)	1.24 (.78)	.94 (.65)
		MIRT	1.56 (.80)	1.08 (.71)	1.24 (.79)	.93 (.67)
	.95	CUIRT	1.49 (.77)	1.04 (.67)	1.22 (.76)	.92 (.63)
		CIRT	1.53 (.81)	1.08 (.70)	1.24 (.79)	.94 (.65)
		MIRT	1.56 (.80)	1.08 (.71)	1.25 (.80)	.94 (.67)
60	.45	CUIRT	1.57 (.84)	.96 (.67)	1.19 (.74)	.83 (.58)
		CIRT	1.65 (.91)	1.04 (.74)	1.25 (.82)	.86 (.61)
		MIRT	1.67 (.89)	1.02 (.73)	1.25 (.81)	.84 (.61)
	.75	CUIRT	1.57 (.84)	.96 (.67)	1.19 (.74)	.83 (.58)
		CIRT	1.65 (.91)	1.04 (.74)	1.25 (.82)	.86 (.61)
		MIRT	1.67 (.89)	1.02 (.73)	1.25 (.81)	.84 (.61)
	.95	CUIRT	1.62 (.88)	.99 (.70)	1.23 (.78)	.85 (.60)
		CIRT	1.64 (.92)	1.05 (.74)	1.24 (.83)	.87 (.62)
		MIRT	1.68 (.89)	1.02 (.74)	1.25 (.82)	.84 (.62)

Note. J = subscale length; ρ = subscale correlation; the values in parentheses represent the standard deviations across replications.

was that on some domains (i.e., domain 1), the 5 subscale item tests reported less bias than the 15 subdomain item test. One likely reason was that the generating item parameters for corresponding test lengths (and subdomains) were different.

5.3.5.2 Study 2

Similar to Study 1, Study 2's results were also not generalizable across subdomains because the data generating item parameters were different for each. The results for the single groups analyses pointed out five key results. First, CUIRT and CUIRT-Op bias, ABS and RMSE decreased as subscale

correlation increased from .45 to .95. Second, CUIRT and CUIRT-Op subscale scores showed the least bias, ABS and RMSE at $\rho = .95$ in domains 1, 2, and 3 compared to domain 4. The CUIRT group of model results observed on domain 3 showed that bias, ABS and RMSE was the closest to 0 compared to the MIRT models across all correlations. Third, MIRT scores showed bias, ABS and RMSE's closer to 0 where $\rho = .95$ in subdomain 4. Fourth, CIRT bias, ABS and RMSE were the least sensitive to subscale correlation. Fifth, MIRT consistently showed bias, ABS and RMSE closest to 0 across all subdomains in subdomain 1 where $D = 4$, $J = 60$.

In summary of the the multiple groups analyses, countries that were in the the middle ranges showed bias, ABS, and RMSE that were nearly equal regardless of the estimation model. However, these middle performing countries systematically showed biases and ABS furthest from 0. The negative biases for these middle performers meant that their simulated country scores were consistently underestimated. The high and low performing countries showed the lowest ABS in both the three- and four-subdomain test conditions. Overall, countries 3 and 4 showed score biases, and ABS closest to 0. The pattern of results that were observed in the multiple groups analyses may have been a result of the overestimated item discrimination estimates that were presented in Section 5.2.4. The results also showed that bias, ABS and RMSE were closest to 0 when subscale correlation increased within specified test condition (i.e., from $\rho = .45$ to $\rho = .75$). Slight improvements were also observed as number of items per subdomain increased (i.e., from $J = 40$ to $J = 60$).

5.4 Subsale Score Value

In order to examine which of the three subscale score methods (i.e., CUIRT, CIRT, MIRT) produced the most valuable subscale scores, I examined their PRMSE's. That is, for each studied condition, I compared the PRMSEs of several indicators of a true subscale score; subscale scores estimated from the studied models. Values of the PRMSE lie between 0 and 1. Conceptually, a subscale score estimate with the highest PRMSE provides a more valuable subscale score (Haberman, 2008). Ideally, CUIRT may not result in large PRMSE since it does not model subscales. Nonetheless, I was consistent in my calculation of the PRMSE for all of the models. In what follows, I summarize Tables 5.14 and 5.17 which allows for the average PRMSE values across all 100 replications to be compared.

5.4.1 PRMSE for Study 1

5.4.1.1 Single Groups

Table 5.14 shows the values of PRMSE for all the studied conditions in Study 1's single groups example. The reported PRMSE's for all of the models in the single group's simulation were generally comparable regardless of subscale correlation. In most of the conditions, CIRT and MIRT had larger and equal estimates of subscale score value regardless of the number of subdomains, subscale length and subscale correlation. The estimated PRMSE's for MIRT based subscale scores were larger in some domains in conditions where $D = 5$ and subdomain correlations were moderate and high (i.e., $\rho = .75$ and $\rho = .95$, respectively). All three models provided equally valuable subscale scores when $D = 3$, $J = 15$, $\rho = .95$. Also, when $D = 5$, $J = 15$, $\rho = .95$ the three models resulted in similar PRMSE's on multiple domains. Table 5.14 showed that the studied models reported larger PRMSEs on longer subscale tests for all simulated conditions regardless of test length and subscale correlation.

Table 5.14
Study 1 Single Groups' Simulation Average PRMSE

			$J = 5$			$J = 10$			$J = 15$		
D	ρ	d	CUIRT	CIRT	MIRT	CUIRT	CIRT	MIRT	CUIRT	CIRT	MIRT
3	.45	1	.465	.467	.467	.631	.634	.634	.710	.714	.714
		2	.461	.464	.464	.629	.632	.632	.708	.713	.713
		3	.445	.449	.449	.608	.612	.612	.715	.719	.719
	.75	1	.467	.468	.468	.633	.634	.634	.712	.714	.714
		2	.461	.463	.463	.631	.633	.633	.710	.712	.712
		3	.446	.448	.448	.610	.612	.612	.717	.719	.719
	.95	1	.467	.467	.468	.634	.634	.634	.713	.713	.713
		2	.463	.463	.464	.633	.633	.633	.712	.712	.712
		3	.448	.448	.449	.612	.612	.613	.718	.718	.718
5	.45	1	.486	.488	.488	.617	.622	.622	.702	.708	.708
		2	.451	.455	.455	.604	.610	.610	.710	.714	.714
		3	.454	.457	.457	.633	.637	.637	.723	.726	.726
		4	.482	.484	.484	.627	.631	.631	.714	.718	.718
		5	.449	.452	.452	.614	.619	.619	.712	.716	.716
	.75	1	.485	.486	.486	.617	.620	.620	.703	.705	.705
		2	.451	.453	.453	.605	.608	.608	.709	.711	.711
		3	.455	.457	.457	.637	.639	.639	.727	.729	.729
		4	.482	.483	.483	.627	.629	.629	.713	.714	.714
		5	.450	.451	.452	.617	.620	.620	.713	.715	.715
	.95	1	.486	.486	.487	.619	.619	.620	.705	.705	.706
		2	.452	.453	.454	.606	.607	.608	.708	.709	.709
		3	.458	.459	.460	.643	.643	.644	.733	.734	.734
		4	.483	.483	.483	.626	.627	.627	.710	.710	.710
		5	.452	.452	.453	.619	.620	.620	.713	.714	.714

Note. D = number of subscales; ρ = subscale correlation; d = subdomain; J = subscale length

5.4.1.2 Multiple Groups

Table 5.15 shows the values of PRMSE for all the studied conditions in Study 1's multiple groups (MG) example. In this example, the reported model specific PRMSEs were comparable (to the third decimal) regardless of subscale correlation. On the three subdomain tests, MIRT scores had the largest PRMSEs. The five subdomain tests told a different story. When subscale correlations were low (i.e., $\rho = .45$), CIRT showed larger PRMSEs on four of the five subdomains where $D = 10$ and $J = 15$. Higher PRMSEs were also reported on multiple domains for the $D = 5$, $J = 15$, $\rho = .75$ condition. In no case were all PRMSEs equal beyond the third decimal. Table 5.15 showed that the studied models reported larger PRMSEs on longer subscale tests for all simulated conditions regardless of test length and subscale correlation.

Table 5.15
Study 1 Multiple Groups' Simulation Average PRMSE

			$J = 5$			$J = 10$			$J = 15$		
D	ρ	d	CUIRT	CIRT	MIRT	CUIRT	CIRT	MIRT	CUIRT	CIRT	MIRT
3	.45	1	.520	.522	.522	.700	.702	.702	.779	.781	.781
		2	.516	.519	.519	.700	.702	.702	.779	.781	.781
		3	.504	.507	.507	.683	.686	.686	.784	.786	.786
	.75	1	.520	.521	.522	.701	.702	.702	.780	.781	.781
		2	.518	.519	.519	.701	.702	.702	.780	.781	.781
		3	.505	.506	.507	.685	.686	.686	.784	.786	.786
	.95	1	.521	.521	.522	.702	.702	.702	.780	.780	.780
		2	.518	.519	.519	.702	.702	.702	.780	.781	.781
		3	.506	.507	.508	.686	.686	.687	.785	.785	.786
5	.45	1	.535	.536	.536	.687	.689	.689	.771	.773	.773
		2	.508	.510	.511	.678	.682	.682	.775	.778	.778
		3	.512	.514	.514	.709	.711	.711	.795	.796	.796
		4	.533	.535	.535	.694	.696	.696	.775	.777	.777
		5	.506	.509	.509	.689	.693	.693	.779	.781	.781
	.75	1	.534	.535	.535	.689	.689	.690	.773	.774	.774
		2	.508	.510	.511	.678	.680	.680	.774	.775	.775
		3	.514	.515	.516	.714	.715	.715	.800	.801	.801
		4	.533	.534	.534	.693	.694	.694	.772	.773	.773
		5	.507	.508	.509	.690	.692	.692	.778	.779	.779
	.95	1	.535	.534	.535	.690	.690	.690	.775	.775	.775
		2	.508	.509	.510	.679	.679	.680	.773	.773	.773
		3	.516	.516	.518	.718	.718	.719	.804	.804	.804
		4	.534	.534	.535	.692	.692	.693	.769	.769	.770
		5	.508	.508	.510	.691	.692	.693	.776	.776	.777

Note. D = number of subscales; ρ = subscale correlation; d = subdomain; J = subscale length

5.4.2 PRMSE for Study 2

5.4.2.1 Single Groups

Table 5.16 shows the values of PRMSE for all the studied conditions in Study 2's single groups (SG) example. The reported MIRT score PRMSE values were larger than those reported for CUIRT, CUIRT-Op and CIRT on all domains in the simulated four subdomain test conditions regardless of test length and subscale correlation. However, CUIRT-Op produced larger PRMSE values in subdomain one on several test conditions. For example, CUIRT-Op produced a score with the largest PRMSE in subdomain one on the three subdomain test that had 40 items-per-subdomain and a correlation of .95. CUIRT-Op also produced scores with the largest PRMSE in subdomain one on the three subdomain test that had 60 items-per-subdomain and correlations of .75 and .95, respectively. Table 5.16 showed that the studied models reported larger PRMSEs on longer subscale tests for all simulated conditions regardless of test length and subscale correlation.

5.4.2.2 Multiple Groups

Table 5.17 shows the values of PRMSE for all the studied conditions in Study 2's multiple groups (MG) example. The MIRT-based PRMSE's were the largest for most domains on the simulated conditions where $D = 3$, regardless of subscale correlation and subscale length. However, CUIRT-Op produced the highest PRMSE in subdomain 1 on all three subdomain test conditions that comprised of multiple groups and regardless of subscale correlation and length. Wang et al. (2019) argued that the proportion of valuable subscale scores on a test is related to the item parameters on a test. That is, item parameter distributions may be related to the prevalence of subscale score value on a test. As such, review of the item parameters showed that the average item discriminations, in situations where CUIRT-Op outperformed MIRT, were larger and more variable than those reported for MIRT (see Table 5.12). The item difficulties were on average lower than those reported by MIRT (see Table 5.13). hlThis may have translated into more information pertaining to the simulated examinees being collected and more subscale value.

Further inspection of Table 5.17 showed that MIRT and CIRT based PRMSE's were largest on two of the four dimensions, each, for a majority of the conditions; where (a) $D = 4$, $J = 40$ (on all studied correlations) and (b) $D = 4$, $J = 40$ $\rho = .45$. In the multiple group's simulation, the CIRT model reported larger PRMSE on three-subdomains on tests that had five-subdomains, so items per subdomain and subscale correlation was .75 and .95. Table 5.12

Table 5.16
Study 2: Single Groups' Simulation Average PRMSE

			Three Domains			Four Domains		
J	ρ	d	CUIRT	CIRT	MIRT	CUIRT	CIRT	MIRT
40	.45	1	.41	.49	.52	.37	.47	.52
		2	.34	.45	.49	.36	.48	.51
		3	.47	.53	.55	.47	.54	.57
		4	—	—	—	.44	.53	.57
	.75	1	.46	.49	.56	.43	.47	.56
		2	.41	.46	.51	.43	.48	.59
		3	.50	.53	.57	.51	.54	.61
		4	—	—	—	.49	.53	.62
	.95	1	.48	.49	.58	.46	.47	.57
		2	.44	.46	.55	.46	.48	.63
		3	.52	.53	.60	.53	.54	.69
		4	—	—	—	.52	.53	.68
60	.45	1	.47	.56	.58	.52	.59	.63
		2	.51	.59	.62	.50	.59	.63
		3	.57	.62	.64	.56	.62	.65
		4	—	—	—	.55	.63	.66
	.75	1	.53	.56	.62	.57	.60	.65
		2	.56	.59	.66	.56	.59	.69
		3	.60	.62	.69	.60	.63	.72
		4	—	—	—	.59	.63	.72
	.95	1	.55	.56	.64	.59	.60	.66
		2	.59	.59	.69	.59	.60	.72
		3	.61	.62	.71	.62	.63	.77
		4	—	—	—	.62	.63	.76

Note. J = subscale length; ρ = subscale correlation; d = subdomain

also showed that the CIRT item parameters exhibited the highest average item discriminations where the model out performed all others. Table 5.17 showed that the studied models reported larger PRMSEs on longer subscale tests for all simulated conditions regardless of test length and subscale correlation.

Table 5.17
Study 2 Multiple Groups' Simulation Average PRMSE

			Three Domains			Four Domains		
J	ρ	d	CUIRT	CIRT	MIRT	CUIRT	CIRT	MIRT
40	.45	1	.66	.68	.71	.65	.67	.64
		2	.64	.67	.74	.65	.68	.66
		3	.64	.66	.73	.65	.67	.68
		4	—	—	—	.68	.70	.72
	.75	1	.67	.68	.72	.66	.67	.63
		2	.65	.67	.75	.66	.68	.66
		3	.65	.66	.74	.66	.67	.68
		4	—	—	—	.69	.70	.72
	.95	1	.67	.68	.72	.66	.67	.63
		2	.66	.67	.76	.67	.68	.65
		3	.66	.66	.75	.66	.67	.67
		4	—	—	—	.69	.70	.72
60	.45	1	.72	.74	.76	.72	.74	.72
		2	.73	.75	.80	.73	.75	.74
		3	.72	.74	.79	.71	.73	.73
		4	—	—	—	.76	.77	.78
	.75	1	.73	.74	.77	.73	.74	.71
		2	.74	.75	.81	.74	.75	.73
		3	.73	.74	.79	.72	.73	.73
		4	—	—	—	.77	.77	.78
	.95	1	.73	.74	.77	.73	.74	.71
		2	.74	.75	.82	.75	.75	.73
		3	.73	.74	.80	.72	.73	.72
		4	—	—	—	.77	.77	.78

Note. J = subscale length; ρ = subscale correlation; d = subdomain

5.5 Model Fit

In this section, I compared CUIRT, CIRT and MIRT model fit within each condition. I evaluated three criteria: $-2ll$, AIC, and BIC. These comparisons were possible since the models were fit to the same set of responses within each studied condition (Singer & Willett, 2003). Smaller values of $-2ll$, AIC, and BIC indicate better relative model fit. In what follows, I compared CUIRT, CIRT and MIRT model fit within each condition since the models were fit to the same set of responses.

5.5.1 Model Fit for Study 1

5.5.1.1 Single Groups

As a reminder, smaller values of $-2ll$, AIC, and BIC indicate better model fit. Results presented in Table 5.18 indicate that for many simulated conditions in Study 1's single groups analysis, MIRT showed better fit (compared to CUIRT and CIRT) regardless of test length in the single groups simulations where subscale correlation was .45 and .75. However, all of the fit measures generally suggested better fit for CUIRT on test conditions where subscale correlation was .95. All of the fit indices suggested better fit for MIRT on one high correlation test condition that had 3 subdomains and 15 items in each subdomain. On several other high correlation conditions, the different fit indices reported different conclusions. For example, on the 3- and 5 subdomain test conditions that had 10- and 15-items per subdomain, respectively. AIC and BIC showed that CUIRT fit the data better on the 3 subdomain test that had a subscale length of 10, and a subscale correlation of .95. $-2ll$ showed that MIRT fit the data better. BIC and $-2ll$ showed that MIRT fit the data better on the 5 subdomain test that had a subscale length of 15, and a subscale correlation of .95. AIC showed that CUIRT fit the data better. One likely reason that the three fit indices reached different conclusions is that some indices penalized the number of parameters more than others. In all of Study 1's single groups conditions, CIRT showed some poor fit.

Table 5.18
Simulation Study 1 Model Fit: Single Groups

			$-2ll$			AIC			BIC		
D	J	ρ	CUIRT	CIRT	MIRT	CUIRT	CIRT	MIRT	CUIRT	CIRT	MIRT
3	5	.45	103899	103773	103068	103931	103809	103110	104039	103930	103251
		.75	102061	103787	101861	102093	103823	101903	102200	103943	102044
		.95	100665	103806	100702	100697	103842	100744	100804	103962	100884
	10	.45	203464	201855	200561	203526	201921	200633	203734	202143	200875
		.75	198966	201849	198194	199028	201915	198266	199236	202136	198507
		.95	195689	201835	195685	195751	201901	195757	195959	202122	195998
	15	.45	302599	298513	296845	302691	298609	296947	302999	298931	297289
		.75	295342	298614	293730	295434	298710	293832	295743	299031	294174
		.95	290066	298591	290007	290158	298687	290109	290466	299008	290451
5	5	.45	177156	177469	175454	177208	177529	175534	177383	177730	175802
		.75	172791	177506	172407	172843	177566	172487	173018	177767	172755
		.95	169507	177495	169635	169559	177555	169715	169734	177756	169983
	10	.45	335183	333063	329618	335285	333173	329748	335627	333541	330183
		.75	325773	333225	324294	325875	333335	324424	326217	333703	324860
		.95	318734	333168	318775	318836	333278	318905	319178	333646	319341
	15	.45	508496	501617	497110	508648	501777	497290	509157	502313	497893
		.75	493079	501868	489867	493231	502028	490047	493740	502564	490650
		.95	481821	501856	481730	481973	502016	481910	482482	502552	482513

Note. D = number of subscales; J = subscale length; ρ = subscale correlation

5.5.1.2 Multiple Groups

With respect to, $-2ll$, AIC and BIC, MIRT generally showed better fit, regardless of test length, on all test conditions where subscale correlation was .45 and .75. In contrast, CUIRT showed better fit compared to the other studied models in conditions where subscale correlation was .95. Put differently, in situations where MIRT did not show better fit, the CUIRT model showed better fit. Overall, CIRT did not fit the data in each condition best in the multiple groups conditions.

Table 5.19
Simulation Study 1 Model Fit: Multiple Groups

			$-2ll$			AIC			BIC		
D	J	r	CUIRT	CIRT	MIRT	CUIRT	CIRT	MIRT	CUIRT	CIRT	MIRT
3	5	.45	499573	515782	497595	499605	515818	497637	499738	515968	497812
		.75	491753	515767	491366	491785	515803	491408	491918	515953	491582
		.95	486413	515817	486689	486445	515853	486731	486578	516002	486905
	10	.45	963489	985747	956141	963551	985813	956213	963809	986087	956512
		.75	946427	985801	944604	946489	985867	944676	946747	986141	944975
		.95	934641	985893	934840	934703	985959	934912	934960	986233	935211
	15	.45	1424804	1446435	1409845	1424896	1446531	1409947	1425278	1446930	1410371
		.75	1398003	1446470	1394078	1398095	1446566	1394180	1398477	1446964	1394604
		.95	1379869	1446798	1379921	1379961	1446894	1380023	1380343	1447293	1380447
5	5	.45	835248	876532	831250	835300	876592	831330	835516	876841	831663
		.75	818124	876507	817435	818176	876567	817515	818392	876816	817847
		.95	806093	876586	806939	806145	876646	807019	806361	876895	807351
	10	.45	1576167	1632235	1562054	1576269	1632345	1562184	1576692	1632802	1562724
		.75	1540655	1632390	1537219	1540757	1632500	1537349	1541181	1632957	1537889
		.95	1516350	1632916	1517043	1516452	1633026	1517173	1516876	1633483	1517713
	15	.45	2369398	2427299	2339718	2369550	2427459	2339898	2370182	2428124	2340646
		.75	2314108	2428246	2306263	2314260	2428406	2306443	2314892	2429071	2307190
		.95	2275758	2428740	2276071	2275910	2428900	2276251	2276541	2429565	2276999

Note. D = number of subscales; J = subscale length; ρ = subscale correlation

Table 5.20
Simulation Study 2 Model Fit (−2ll): Single Groups

			−2ll			
D	J	ρ	CUIRT	CUIRT-Op	CIRT	MIRT
3	40	.45	122801	136105	122591	122183
		.75	120904	135424	122627	121705
		.95	119414	135106	122605	120883
	60	.45	183365	196241	182446	181617
		.75	179988	195180	182389	180029
		.95	177536	194896	182412	178910
4	40	.45	167719	184912	167603	166775
		.75	164397	184398	167602	165187
		.95	161902	181829	167640	163437
	60	.45	248242	264825	246818	245237
		.75	242393	262272	246881	242443
		.95	238043	264322	246896	240282

Note. D = number of subscales; J = subscale length; ρ = subscale correlation

5.5.2 Model Fit for Study 2

5.5.2.1 Single Groups

Results in Table 5.20 to 5.22 ($-2ll$, AIC, and BIC, respectively), for Study 2, showed results that contradicted those presented in Table 5.18 (results for simulation Study 1). With respect to $-2ll$, AIC and BIC, CUIRT showed better fit, regardless of test length, in the single groups simulations where subscale correlation was .75 and .95. However, all of the fit measures suggested better fit for MIRT on test conditions where subscale correlation was .45. In all of Study 2's single groups conditions, CUIRT-Op showed some poor fit.

5.5.2.2 Multiple Groups

With respect to, $-2ll$, AIC and BIC (see Tables 5.20 to 5.22), CUIRT generally showed better fit, regardless of test length, on the multiple groups analyses where subscale correlation was .75 and .95. In addition, all of the fit measures suggested that CUIRT had better model fit on the four subdomain tests that comprised of 40 items-per-subdomain where subscale correlation was .45. Only the BIC fit measure showed that CUIRT fit the data best on the three

Table 5.21
Simulation Study 2 Model Fit (AIC): Single Groups

			AIC			
D	J	ρ	CUIRT	CUIRT-Op	CIRT	MIRT
3	40	.45	123293	136279.2	123083	122681
		.75	121396	135597.5	123119	122203
		.95	119906	135280.4	123097	121381
	60	.45	184103	196415.2	183184	182361
		.75	180726	195354.4	183127	180773
		.95	178274	195070.4	183150	179654
4	40	.45	168375	185148.4	168259	167443
		.75	165053	184633.6	168258	165855
		.95	162558	182053	168296	164105
	60	.45	249226	265060.7	247802	246233
		.75	243377	262496.2	247865	243439
		.95	239027	264558.3	247880	241278

Note. D = number of subscales; J = subscale length; ρ = subscale correlation

Table 5.22
Simulation Study 2 Model Fit (BIC): Single Groups

			BIC			
D	J	ρ	CUIRT	CUIRT-Op	CIRT	MIRT
3	40	.45	124941	136862.1	124731	124349
		.75	123044	136180.4	124767	123872
		.95	121554	135863.3	124745	123049
	60	.45	186575	196998	185656	184853
		.75	183198	195937.3	185599	183265
		.95	180747	195653.3	185622	182146
4	40	.45	170573	185939	170457	169681
		.75	167250	185424.2	170455	168093
		.95	164756	182803.9	170493	166343
	60	.45	252523	265851.3	251098	249569
		.75	246673	263246.5	251162	246776
		.95	242323	265349.3	251176	244614

Note. D = number of subscales; J = subscale length; ρ = subscale correlation

Table 5.23
Simulation Study 2 Model Fit (−2ll): Multiple Groups

			−2ll			
D	J	ρ	CUIRT	CUIRT-Op	CIRT	MIRT
3	40	.45	545407	549449.4	565911.2	545395
		.75	533349.5	544635.3	566056.6	540852.1
		.95	524519.7	541385.9	565936.2	537360.7
	60	.45	796740.7	797500.8	817551.4	791936.2
		.75	777879.7	793981.1	817511.1	785602.1
		.95	764409	784364.6	817605.1	781010.1
4	40	.45	720898.6	728609.3	757352.9	721262.4
		.75	701492.7	720436.1	757449.2	713104.7
		.95	687539.7	722602.8	757437.6	707139.5
	60	.45	1059779	1063232	1096848	1052904
		.75	1030045	1048669	1097094	1042286
		.95	1008747	1046043	1097138	1034321

Note. D = number of subscales; J = subscale length; ρ = subscale correlation

subdomain tests that comprised of 40 items-per-subdomain where subscale correlation was .45; on the contrary, $-2ll$ and AIC suggested that MIRT fit the data best when compared to the other models. However, all of the fit measures suggested that MIRT had better model fit regardless of the number of subdomains on the tests conditions that comprised of 60 items-per-subdomain and subscale correlation was .45. In all of Study 2's multiple groups conditions, CIRT showed some poor fit. This results were confirmed by all of the fit measures.

5.6 Summary

This chapter presented the results of all the simulation studies that were designed to resemble SACMEQ and TIMSS data, respectively. The difference between the two simulation studies is that Study 1 does not employ matrix sampled test booklets whilst Study 2 does. All of the results were intended to show how well competing IRT methods perform in subscale score estimation under different test specifications. The results are summarized in Figures 5.63 and 5.64. These two figures show all of the simulated conditions. The figures show which of the competing models performed best in either item- or

Table 5.24
Simulation Study 2 Model Fit (AIC): Multiple Groups

			AIC			
D	J	ρ	CUIRT	CUIRT-Op	CIRT	MIRT
3	40	.45	545899	549449.4	566403.2	545893
		.75	533841.5	544635.3	566548.6	541350.1
		.95	525011.7	541385.9	566428.2	537858.7
	60	.45	797478.7	797500.8	818289.4	792680.2
		.75	778617.7	793981.1	818249.1	786346.1
		.95	765147	785108.6	818343.1	781754.1
4	40	.45	721554.6	728609.3	758008.9	721930.4
		.75	702148.7	720436.1	758105.2	713772.7
		.95	688195.7	722602.8	758093.6	707807.5
	60	.45	1060763	1063232	1097832	1053900
		.75	1031029	1049665	1098078	1043282
		.95	1009731	1046043	1098122	1035317

Note. D = number of subscales; J = subscale length; ρ = subscale correlation

Table 5.25
Simulation Study 2 Model Fit (BIC): Multiple Groups

			BIC			
D	J	ρ	CUIRT	CUIRT-Op	CIRT	MIRT
3	40	.45	547943	549449.4	568447.2	547961.9
		.75	535885.5	544635.3	568592.6	543419
		.95	527055.7	541385.9	568472.2	539927.6
	60	.45	800544.7	797500.8	821355.4	795771.1
		.75	781683.7	793981.1	821315.1	789437
		.95	768213	788199.6	821409.2	784845
4	40	.45	724280	728609.3	760734.2	724705.6
		.75	704874.1	720436.1	760830.6	716547.9
		.95	690921.1	722602.8	760818.9	710582.7
	60	.45	1064851	1063232	1101920	1058038
		.75	1035117	1053803	1102166	1047420
		.95	1013819	1046043	1102210	1039455

Note. D = number of subscales; J = subscale length; ρ = subscale correlation

score-recovery and PRMSE. These were presented by comprising sample and subdomain where necessary.

Figure 5.63

Heat Map (with Number Reference) of Study 1 Results

			Item par's		Scores					PRMSE															
			b		Domains																				
					SG					MG					SG					MG				
D	J	R	SG	MG	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
3	5	.45	2	2	6	1	1			7	7	7			2	2	2			3	3	3		
		.75	2	2	6	1	1			7	7	7			3	3	3			3	3	3		
		.95	4	2	6	1	1			7	7	7			3	3	3			3	3	3		
	10	.45	2	2	6	7	1			7	7	7			2	2	2			3	3	2		
		.75	2	2	6	1	1			7	7	7			2	2	3			3	3	3		
		.95	4	2	6	1	1			7	7	7			3	3	3			3	3	3		
	15	.45	2	2	6	7	7			7	7	7			2	2	2			3	3	3		
		.75	2	2	6	1	7			7	7	7			2	2	2			3	3	3		
		.95	4	2	6	3	7			7	7	7			3	3	3			3	3	3		
5	5	.45	2	2	7	6	6	6	6	7	7	7	7	7	2	3	2	2	2	3	3	3	3	3
		.75	2	2	1	1	2	6	2	7	7	7	7	7	3	3	3	3	3	3	3	3	3	3
		.95	2	2	1	3	1	6	2	7	7	7	7	7	3	3	3	3	3	3	3	3	3	3
	10	.45	2	2	1	7	7	2	7	7	7	7	7	7	2	2	2	2	2	2	3	2	2	3
		.75	2	2	1	6	1	2	3	7	7	7	7	7	3	3	3	3	3	3	3	3	3	3
		.95	2	2	1	3	1	2	3	7	7	7	7	7	3	3	3	3	3	3	3	3	3	3
	15	.45	2	2	3	7	7	3	4	7	7	7	7	7	2	2	2	2	2	3	2	2	2	2
		.75	2	2	1	3	1	2	1	7	7	7	7	7	3	2	2	2	2	3	2	2	2	3
		.95	2	2	1	1	1	1	1	7	7	7	7	7	3	3	3	3	3	1	3	3	3	3

Note. Item par = item parameter; D = number of domains; J = number of items per domain; R = subscale correlation; SG = single groups; MG = multiple groups; b = item difficulty; 1 = CUIRT; 2 = CIRT; 3 = MIRT; 4 = CUIRT and CIRT; 5 = CUIRT and MIRT; 6 = CIRT and MIRT; 7 = all models comparable.

Figure 5.63 summarises the key findings of simulation Study 1. The numbers in each figure represent the best performing model at (a) item parameter recovery, (b) score recovery, and (c) PRMSE. In other words, the numbers correspond to a specific model, or a combination of models that result in comparable performance, that perform best. In Figure 5.63: *1* = CUIRT; *2* = CIRT; *3* = MIRT; *4* = CUIRT and CIRT; *5* = CUIRT and MIRT; *6* = CIRT and MIRT; *7* = All models. The results are presented over all of Study 1's simulated test conditions, single- or multiple-groups.

When comparing item parameter recovery, the results showed that CIRT generally performed better than all other models on both the single- and multiple-groups studies. The performance of CUIRT was consistently related to subscale correlation in that results showed better item parameter recovery where subscale correlation was .95 as compared to correlations of .45 and .75. As a result, CUIRT produced similar results as CIRT on the three-subdomain tests where subscale correlation was .95.

Studies 1 and 2 did not present generalizable score recovery patterns across subdomains across the different subdomains. This was largely because the generating item parameters were different for all of the subdomains on a test condition. As a result, the subdomains had different items which translated manifested into subscales with different test properties. Study 1's score recovery seemed to suggest that CIRT and MIRT resulted in better subscale score estimates on subdomain 1 on the single groups simulation, on tests that had three-items-per-subdomain. The single groups simulations showed that CUIRT improved in some other conditions (i.e., there was notable improvement in score estimation where subscale score correlation was .95). Study 1's score recovery results for the multiple groups simulations were generally comparable regardless of test specification and subscale score estimation model.

The results from Study 1 further showed that MIRT had higher PRMSE compared to CUIRT. In other words, MIRT produced more valuable subscale scores. CIRT consistently produced comparable PRMSE's to MIRT in the single groups simulations where subscale score correlation was .45 and .75. MIRT, generally resulted in the more valuable subscale scores in the multiple groups simulation. MIRT also showed better model fit whereas CIRT did not fit the data in each condition best.

Figure 5.64 summarises the key findings of simulation Study 2. The numbers in each figure represent the best performing model at (a) item parameter recovery, (b) score recovery, and (c) PRMSE. In other words, the numbers correspond to a specific model, or a combination of models that result in comparable performance, that perform best. In Figure 5.64: *1* = CUIRT; *2* = CUIRT-Op; *3* = CIRT; *4* = MIRT; *5* = CUIRT and CUIRT-Op; *6* =

CUIRT and CIRT; *7* = CUIRT and MIRT; *8* = CIRT and CUIRT; *9* = CIRT and MIRT; *10* = CUIRT-Op and MIRT; *11* = CIRT and MIRT. The results are presented over all off Study 2's simulated test conditions, single- or multiple-groups.

When comparing item parameter recovery, the results showed that CIRT and MIRT generally produced better difficulty and discrimination parameters than CIRT, on both the single- and multiple-groups studies, where subscale correlation was .45 and .75. In contrast, CUIRT showed better discrimination and difficulty recovery where subscale correlation was .95. In addition, the models resulted in similar item location parameters for the GPCM items (see Appendix D for the bias plots for $d1$ and $d2$).

Figure 5.64

Heat (with Number Reference) Map of Study 2 Results

			Item Parameters								Scores								PRMSE							
			SG				MG				Domain															
			SG				MG				SG				MG				SG				MG			
D	J	R	a	b	d1	d2	a	b	d1	d2	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
3	40	.45	9	9	11	11	9	9	11	11	5	5	5		3	4	4		4	4	4		2	4	4	
		.75	9	9	11	11	9	9	11	11	5	5	5		3	4	3		4	4	4		2	4	4	
		.95	1	1	11	11	1	1	11	11	5	5	5		3	3	9		2	4	4		2	4	4	
	60	.45	9	9	11	11	9	9	11	11	1	5	5		3	3	4		4	4	4		2	4	4	
		.75	9	9	11	11	9	9	11	11	5	5	5		3	3	4		2	4	4		2	4	4	
		.95	1	1	11	11	1	1	11	11	5	5	5		3	3	4		2	4	4		2	4	4	
4	40	.45	9	9	11	11	9	9	11	11	4	5	4	5	3	3	4	4	4	4	4	4	3	3	4	4
		.75	9	9	11	11	9	9	11	11	5	5	5	4	3	3	4	4	4	4	4	4	3	3	4	4
		.95	1	1	11	11	1	1	11	11	5	5	5	5	3	3	4	4	4	4	4	4	3	3	4	4
	60	.45	9	9	11	11	9	9	11	11	4	5	5	4	3	3	4	4	4	4	4	4	3	3	3	4
		.75	9	9	11	11	9	9	11	11	5	5	5	4	3	3	4	4	4	4	4	4	3	3	3	4
		.95	1	1	11	11	1	1	11	11	5	5	5	4	3	3	9	4	4	4	4	4	3	3	3	4

Note. D = number of domains; J = number of items per domain; R = subscale correlation; SG = single groups; MG = multiple groups; a = item discrimination; b = item difficulty; d1 = GPCM first item location; d2 = GPCM second item location; 1 = CUIRT; 2 = CUIRT-Op; 3 = CIRT; 4 = MIRT; 5 = CUIRT and CUIRT-Op; 6 = CUIRT and CIRT; 7 = CUIRT and MIRT; 8 = CIRT and CUIRT; 9 = CIRT and MIRT; 10 = CUIRT-Op and MIRT; 11 = CIRT and MIRT.

Score recovery seemed to suggest that CUIRT and CUIRT-Op resulted in better subscale score estimates on all subdomains on the single groups simulation. Where these two models did not do well, MIRT recovered the scores better. In contrast, the multiple groups simulations suggested that CIRT produced better subscale score estimates on the first and second subdomain. MIRT resulted in better subscale score estimates on subdomains 3 and 4.

The MIRT model also resulted in better PRMSE over a majority of the conditions on Study 2's single groups study. CUIRT-Op produced more valuable scores on subdomain 1 of the three subdomain test conditions where subscale correlations were .75 and .95. All of the 4 subdomain test conditions of simulation study 2's single groups study showed that MIRT produced more valuable subscores. The results from simulation Study 2's multiple groups test conditions showed a slightly different trend. First, CUIRT-Op reported the highest PRMSEs on subdomain 1 of all three subdomain test conditions. Second, CIRT produced more valuable subscale scores in subdomain 1 and 2 on the 4-items-per-subdomain, multiple-groups simulation regardless of test length. Third, CIRT also produced higher PRMSE in the third subdomain of the 60-item-per-subdomain test conditions. Fourth, where CUIRT-Op and CIRT did not perform well, MIRT scores were reported as being more valuable. Fifth, CUIRT generally reported the lowest PRMSE's; a result that was to be expected because of the models inherent properties. Sixth, as the number of subdomains increased, t

One common result from Studies 1 and 2 was that the studied models showed larger PRMSEs on longer subscale tests for all simulated conditions. In other words, longer subscales had larger PRMSE. This result was expected since PRMSE is analogous to the marginal reliability of a subscale, and the reliability of a test increases as the number of items increases.

The results from simulation Study 1 suggested that MIRT fit the data best where subscale correlation was .45 and .75. These findings were expected since MIRT was the assumed data generation model. In contrast, CUIRT generally showed better model fit where subscale correlation was .95. The results form Study 2 contradicted expected results that the generating model, MIRT, would result in better fit across all of the conditions. CUIRT generally showed better in Study 2 model fit where subscale correlation was .75 and .95. In addition, the results from simulation Study 2's single groups conditions suggested that MIRT fit the data best where subscale correlation was .45. This trend was not consistent in Study 2's multiple groups analyses. There were some contradicting results in the three subdomain tests that comprised of 40 items-per-subscale and subscale correlation was .45. In this condition, $-2ll$ and AIC suggested that MIRT fit the data better than the other models, and BIC suggested that

CUIRT fit the data better. Also, all of the fit criteria suggested that CUIRT fit the data better in the four subdomain tests that comprised of 40 items-per-subscale and subscale correlation was .45. However, one would have expected MIRT to fit all of the data best since MIRT was the assumed generating model. But then, depending on the study, the results suggest that this was not the case at the medium to high correlations. One likely reason that this was the case is that the underlying “true" data generating mechanism of the chosen data where subscale correlations were high (and in some cases moderate) represented a more unidimensional model. In both studies 1 and 2, CIRT showed poor fit when all three models (CUIRT, CIRT, MIRT) were compared. The findings from simulation study 2 further suggested that CUIRT-Op and CIRT fit the data poorly depending on the sample composition. That is, CUIRT-Op and CIRT fit the data poorly in the single- and multiple-groups conditions, respectively. In the next chapter, I present the results of the empirical study.

Chapter 6

Empirical Results

6.1 Introduction

Chapter 5 presented the results of two simulation studies. One of the simulation studies, Study 2, was designed to resemble TIMSS 2015's mathematics test. As such, to demonstrate how the studied models (i.e, CUIRT, CUIRT-OP, CIRT, and MIRT) may be used in practice, I conducted an empirical study using TIMSS 2015 data from 9 countries. The empirical study was conducted in order to validate the findings from the simulation studies. That is, the study was conducted to respond to research questions 2 and 3 presented in Section 1.4 in Chapter 1.

This chapter presents results obtained from the data analysis that was carried out in the empirical study that was described in Chapter 4. The data for the analysis were drawn from TIMSS 2015 eighth grade mathematics. In total, the test had 209 items. Each item on the test belonged to one of four subdomains that were defined in the test blueprint. These were: algebra, data and chance, geometry, and numbers.

6.2 Achievement

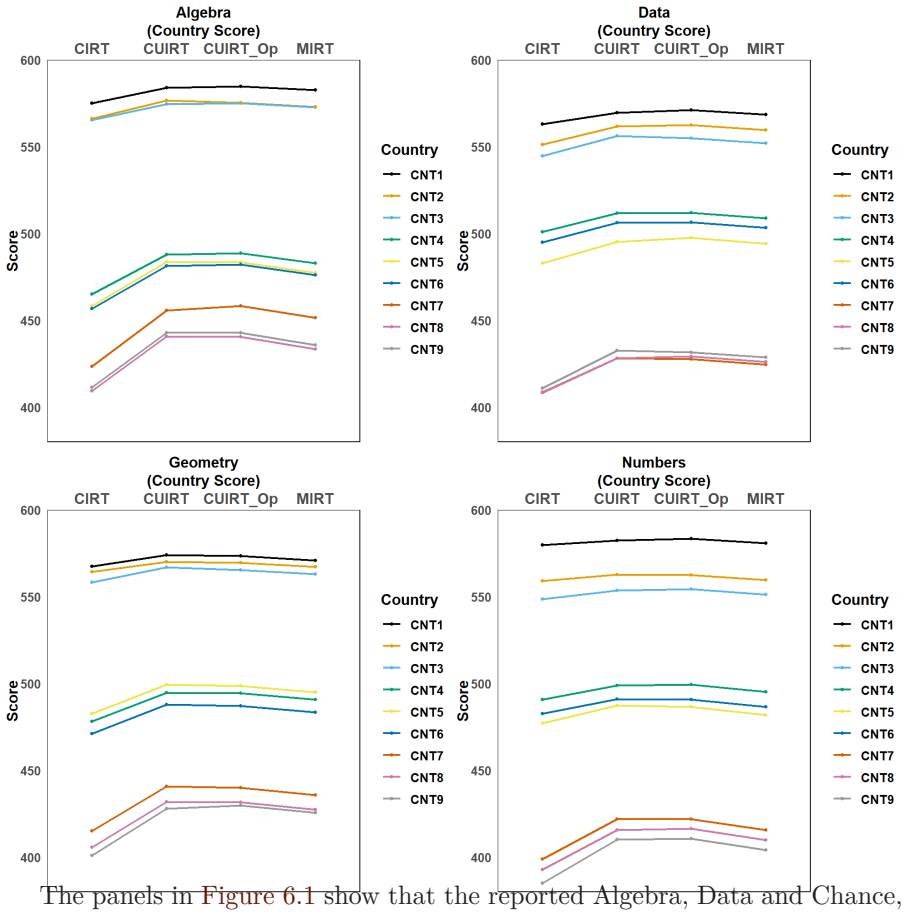
Items parameter estimates that were used in the study were obtained from the CUIRT, CIRT, and MIRT models. The estimated item parameters were then fixed and used in the scoring process. Scores were estimated assuming four subscale score models: CUIRT, CUIRT-Op, CIRT, and MIRT as described in Section 4.4 of Chapter 4. As a reminder, CUIRT-Op is the model that is closest to what was used operationally on TIMSS 2015.

6.2.1 Achievement by Population

In this section, I report the results from my investigation as to whether there were differences in reported country scores by model. In this study, I did not run significance tests to make broader inferences related to these differences. As such, Figure 6.1 plots the sampled countries observed subscale score. Each of the four panels, on the figure, corresponds to a different subscale. The points across the nine spaghetti plots show each country's average subdomain score

estimated from a specification of either the CIRT, CUIRT, CUIRT-Op, or MIRT models. Each country is represented by a different colour. In turn, each panel in Figure 6.1 represents a different subdomain. The panels in Figure 6.1 will make it possible to visualize the score differences.

Figure 6.1
Estimated Population Scores



The panels in Figure 6.1 show that the reported Algebra, Data and Chance, Geometry, and Numbers scores followed the same pattern. The panels show that CUIRT and CUIRT-Op produce higher scores on all subdomains. Put differently, the observed CUIRT and CUIRT-Op scores were higher than those reported by MIRT and CIRT. In contrast, the panels in Figure 6.1 also showed

that CIRT generally produced the lowest scores. Figure 6.1 also showed that there were larger differences between the reported CUIRT and CIRT scores. This pattern was observed on the country scores for all countries. However, these differences were larger in the middle- to low-performing countries as compared to the top three performers: countries 1, 2, and 3.

In general, the differences between the CIRT subscale scores and those of the other studied models were large (see Figure 6.1). This was observed on all subscales. To gain an insight as to why, I inspected the averages of the item discriminations that were used to estimate subscale scores (see Table 6.1). The table showed that CIRT item discrimations were on average lower than those reported for CUIRT and MIRT on all subdomains. DeAyala (2013) pointed out that as item discrimination increases, the maximum information for estimating proficiency increases thus resulting in better proficiency estimates. This may explain why CUIRT, CUIRT-Op, and MIRT perform similarly, and may not report largely underestimated scores like those observed from CIRT[1]. Coincidentally, CIRT reported more biased item parameters compared to CUIRT and MIRT in simulation Study 2's multiple groups test conditions that comprised of 4 subdomains, and subscale correlation of .95 (see Section 5.2.4). As such, these respective differences and biases may have resulted in underestimated subscale scores from CIRT.

Table 6.1
Summary of the Item Discrimination Parameters

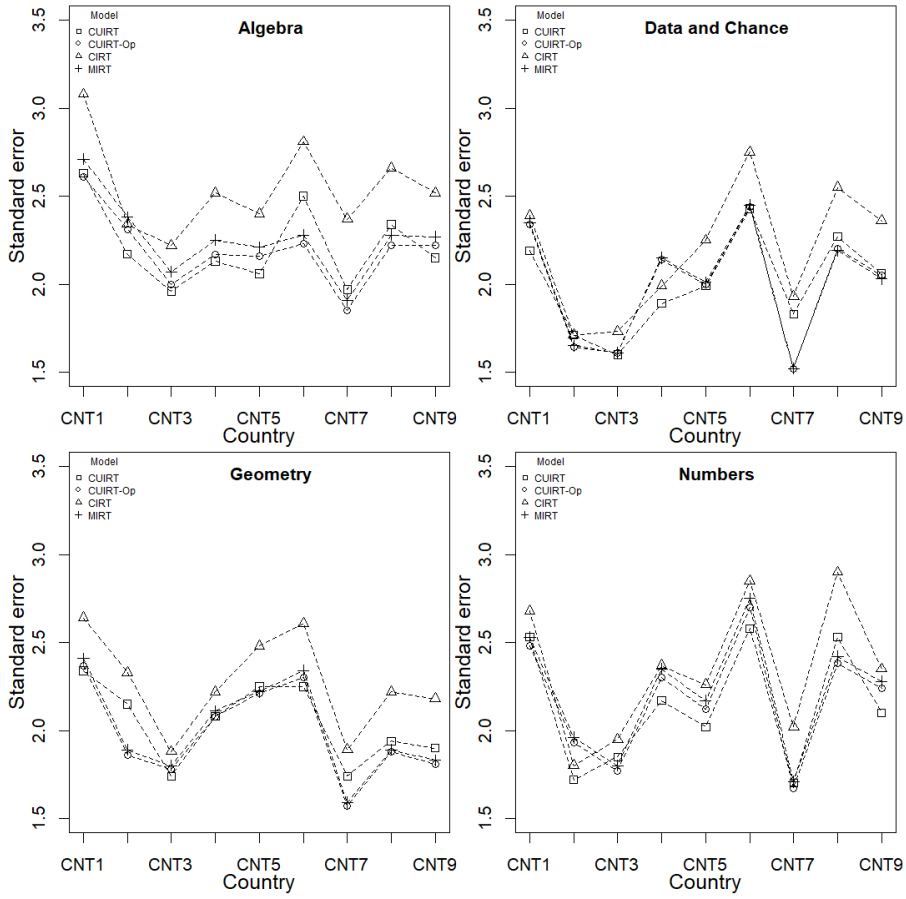
	Domain							
	1		2		3		4	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD
CUIRT	2.66	(.72)	2.21	(.87)	2.27	(.73)	2.45	(.68)
CIRT	2.39	(.70)	2.14	(1.12)	2.12	(.79)	2.36	(.71)
MIRT	2.55	(.70)	2.28	(.95)	2.30	(.83)	2.37	(.67)

Figure 6.1 only suggested differences in the magnitude of the reported score. However, it is from the estimated scores that countries are ranked. As such, the results presented in Figure 6.1 did not show that there were any rank changes due to the scores reported from the studied models. In other words, the countries were ranked the same regardless of scoring model.

[1]The results presented in the next paragraphs show that CIRT also resulted in scores with larger SEs

Each panel in Figure 6.2 shows the standard errors (SEs) that correspond to the model specific subscale scores presented in Figure 6.1. The four lines in each panel on Figure 6.2 represent each of the studied models: CUIRT, CUIRT-Op, CIRT, and MIRT; and the points are the specific SE estimates for each country. From Figure 6.2, it can be seen that CIRT consistently produces country score estimates with the largest SE on all subdomains. However, MIRT reported the largest SEs in the few cases where CIRT had smaller SEs (i.e., Country 2's Algebra and Numbers scores).

Figure 6.2
Standard Error of the Population Scores



6.2.2 Achievement by Subpopulation

Countries that participate in ILSA's are often interested in subscale scores. To that effect, I used the empirical dataset to compare how well the three subscale score estimation models performed. That is, which model provided the better estimates. In doing so, I wanted to determine whether the results, with regards to the estimated scores, were similar to those observed at the country-level as well as the simulation study. This analysis was conducted to respond to research question 2 presented in Section 1.4.

To that effect, in this section, I report country performance by (a) gender and (b) number of books at home. These analyses were done in order to examine each model's reported country-subpopulation-scores. The gender variable had two levels: boys and girls. The number of books at home comprised of five levels: 0–10 books; 11–25 books; 26–100 books; 101–200 books; and more than 200. Table 6.2 summarizes the samples that were included in the analyses for both subpopulations. The total samples for all subpopulations were lower than those included in the analyses of the country-level subscale scores (see Table 4.1). This was likely because several of the participants did not specify which category they belonged to.

Table 6.2
Subpopulation Sample Sizes

	Gender		Books at home				
Country	Girls	Boys	0-10	11-25	26-100	101-200	Over 200
1	2976	3132	1136	1662	1854	841	610
2	2604	2704	355	413	1195	1368	1975
3	2795	2912	1150	1317	1556	779	903
4	5201	5072	1188	1798	2478	2041	2133
5	2224	2257	640	1060	1112	738	895
6	4291	3810	1017	1340	2346	1667	1577
7	4278	3585	2238	2691	1739	530	465
8	6424	6082	5135	4709	1654	459	415
9	1997	1760	1342	1118	732	236	259

Upon conducting analyses, it was observed that in some countries, there was some clustering with respect to the number of books at home. That is, scores in the latter three levels in Countries 7, 8 and 9 for students that reported to having 0–10 books and 11–25 books were not too different (i.e., the scores in the respective categories were similar, almost indistinguishable). However,

I did not collapse the categories for the number of books since I wanted to observe the reported scores in a situation where there are more than two levels.

In the two sections that follow, Sections 6.2.2.1 and 6.2.2.2, I examine whether there are any differences in the reported scores between respective categories. Without running significance tests, I also examine the difference between category scores due to model specification. Figures 6.3 and 6.6 show the differences in performance between boys and girls, and Figures 6.8 and 6.11 show the differences in performance between the number of books at home. I used these figures to identify if the models reported large subpopulation-subscale score differences (characterized by the distances between the points). I then examined each subscale score's SE to identify the model that performed better (i.e., reported smaller SEs). The better performing model would be expected to report scores with the lowest SE. As such, I intend to validate the findings of simulation study 2 by identifying whether the same model that performed well in simulation Study 2's multiple groups study was better than the other models in the empirical analysis. The analyses conducted in Sections 6.2.2.1 and 6.2.2.2 respond to research question 2 (see Section 1.4).

6.2.2.1 Scores by Gender

The average subscale scores for boys and girls in each country were estimated for each subdomain. Figures 6.3 and 6.6 show the differences in performance between boys and girls. In all of the plots, the y-axis shows the country code. These range from 1 to 9. The x-axis, on each of the plots, shows the mean scores. The figures make it possible to observe the score differences between the two genders.

Figures 6.3 and 6.6 did not show any trends or patterns. Visually, the figures showed that the observed differences did not follow a specific pattern. That is to say, no model consistently reported large or small subpopulation differences. The model-specific differences were the same for some countries on particular domains and different on others.

Figure 6.7 shows the SEs of all of the subscale scores that were estimated from each of the studied models by gender. For example, the top-left panel shows the SEs of each country's estimated Algebra subscale score for the female subpopulation. Each of the four lines join the SEs of the subscale score estimates calculated from the CUIRT, CUIRT-Op, CIRT and MIRT models. The findings presented in Figure 6.7 show that CIRT generally produces subpopulation subscale scores with large SEs. However, there are a few exceptions. For example, the SEs were not the largest for: females in country 2, and males in countries 2, 4, 5, and 9 on the Numbers subdomain. Figure 6.7 also showed

Figure 6.3
Estimated Gender Subscale Scores for Algebra

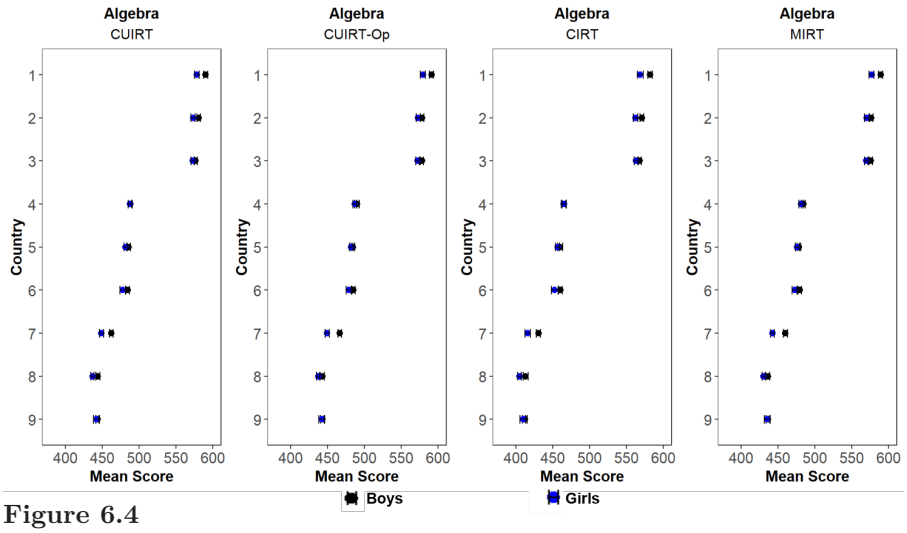
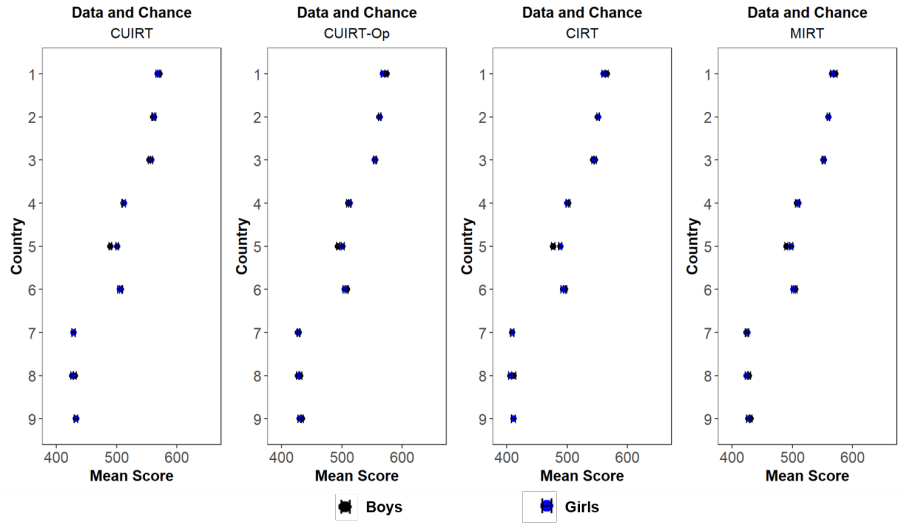


Figure 6.4
Estimated Gender Subscale Scores for Data and Chance



that CIRT scores did not have the largest SE for females in country 6 on the Geometry subdomain. It was also noted that, in all cases where CIRT reported

Figure 6.5
Estimated Gender Subscale Scores for Geometry

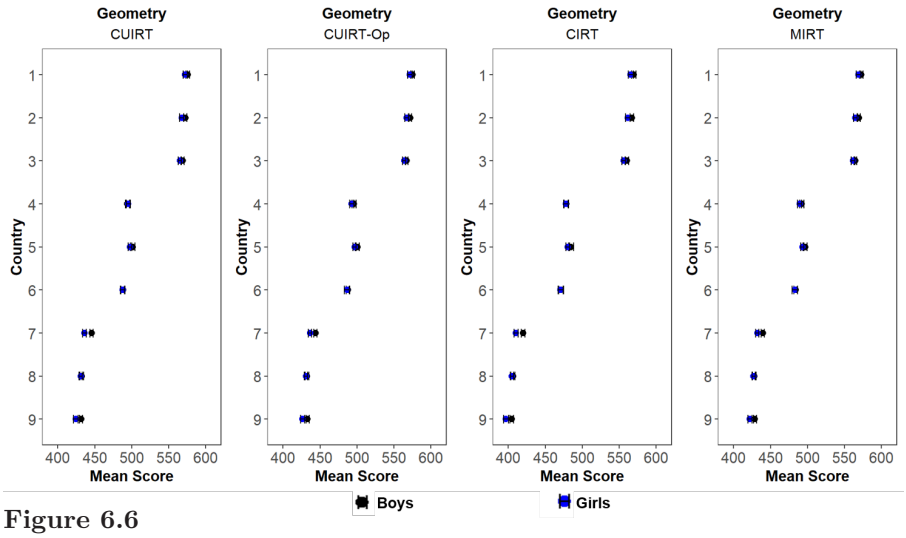
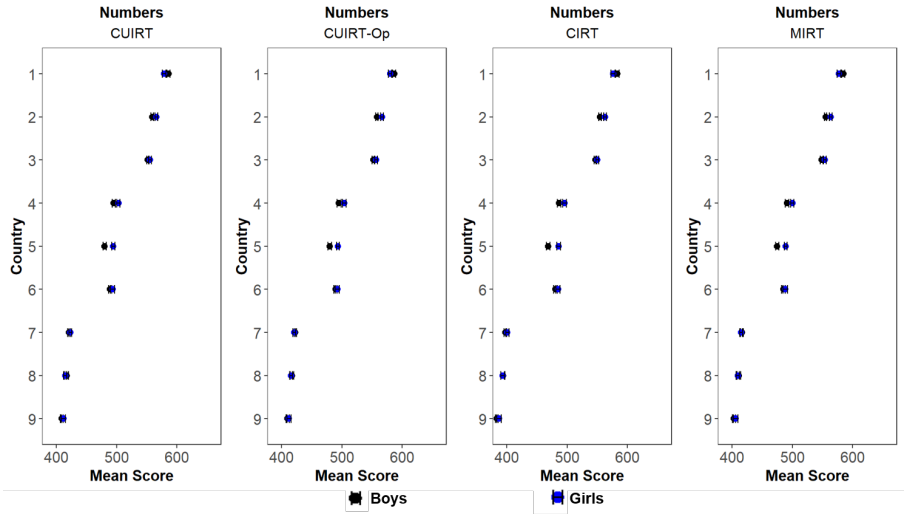


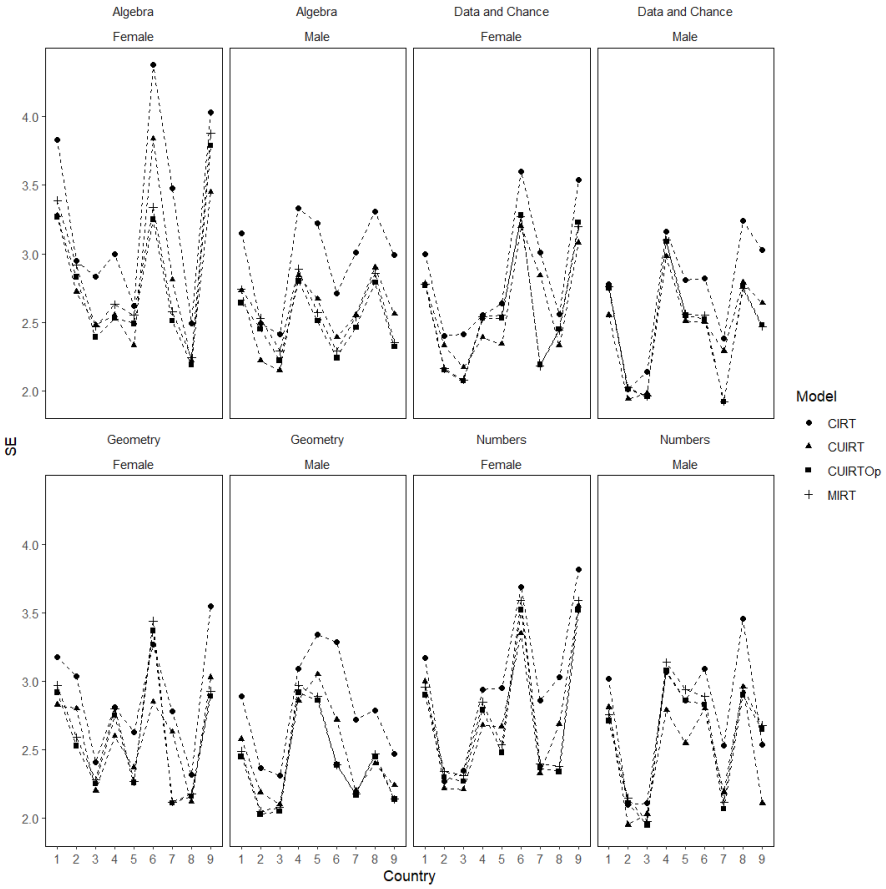
Figure 6.6
Estimated Gender Subscale Scores for Numbers



lowed SEs, MIRT produced the largest SEs.

Table 6.3 shows the average SEs of each models subdomain score. It

Figure 6.7
Standard Error of the Sub-Population Scores: Gender



was observed that SEs were the lowest on all models for males. However, CUIRT-Op produced the lowest SEs across all subdomains and subpopulations. Note that the MIRT average SEs on the Data and Chance subdomain were comparable to the CUIRT-Op averages (2.58 and 2.45, for the females and males, respectively). In contrast, CIRT had the highest average SEs across all subdomains. The repeated-measures ANOVA results show whether the observed differences between standard errors were statistically significant. These results are presented in the paragraph that follows.

Table 6.3
Average Standard Errors of the Subscale Scores Reported by Gender

	Algebra		Data and Chance		Geometry		Numbers	
Model	Female	Male	Female	Male	Female	Male	Female	Male
CUIRT	2.85	2.56	2.61	2.46	2.6	2.48	2.74	2.47
CUIRT-Op	2.81	2.49	2.58	2.45	2.58	2.38	2.72	2.57
CIRT	3.29	2.96	2.86	2.71	2.89	2.81	3.01	2.75
MIRT	2.89	2.56	2.58	2.45	2.62	2.41	2.77	2.62

The repeated-measures ANOVA with a Greenhouse-Geisser correction determined that mean standard errors differed statistically significantly when the estimation method was changed for the domain ability estimates $F_{(1.17,19.87)\text{Algebra}} = 37.99$, $p < .05$, partial $\eta^2 = .15$; $F_{(1.35,22.89)\text{Data and Chance}} = 18.343$, $p < .05$, partial $\eta^2 = .08$; $F_{(1.3,22.14)\text{Geometry}} = 25.49$, $p < .05$, partial $\eta^2 = .14$; and $F_{(1.37,23.22)\text{Number}} = 15.25$, $p < .05$, partial $\eta^2 = .05$. Post-hoc tests using the Bonferroni correction revealed that not all pairwise comparisons were statistically significantly different from each other. There were statistically significant differences between CIRT and all other models: CUIRT, CUIRT-Op, CIRT, and MIRT on all subdomains. According to the results in Table 6.3, CIRT had the highest SEs for all subpopulation subdomain proficiency scores. In other words, subdomain scores from the CIRT model were not as accurate as the other three methods. There were also statistically significant differences between CUIRT-Op and MIRT on the Algebra, Geometry and Numbers subdomains. According to the results in Table 6.3, when compared with MIRT, the CUIRT-Op method had the lowest standard errors for the subdomain scores reported on the three domains. Table 6.3 showed that the differences were not statistically significant on the Numbers subdomain.

Therefore, it can be concluded that CUIRT, CUIRT-Op and MIRT elicited a statistically significant reduction in standard errors of subdomain score estimates compared to CIRT. Likewise, CUIRT-Op showed statistically significant reduction in SEs compared to MIRT on the Algebra, Geometry and Numbers subdomains. Post hoc tests using the Bonferroni correction revealed that some pairwise comparisons were significantly different from each other. The CIRT had the highest mean for standard errors.

6.2.2.2 Scores by Books at Home

Figure 6.8 to 6.11 show the average subpopulation scores by number of books-at-home. Each figure plots 4 panels, and each panel plots the subscale scores estimated for nine countries. The subpopulation subscale scores are color-coded. These figures were used to inspect whether the magnitudes of the subpopulations differed conditional on the subscale score estimation model. To complement these findings, I also inspected whether model choice resulted in different magnitudes of subpopulation -subscale score differences. As a reminder, I observed each subscale score's SE to identify the model that performed better (i.e., reported smaller SEs).

Figure 6.8

Estimated Algebra Subscale Scores by Books in the Home

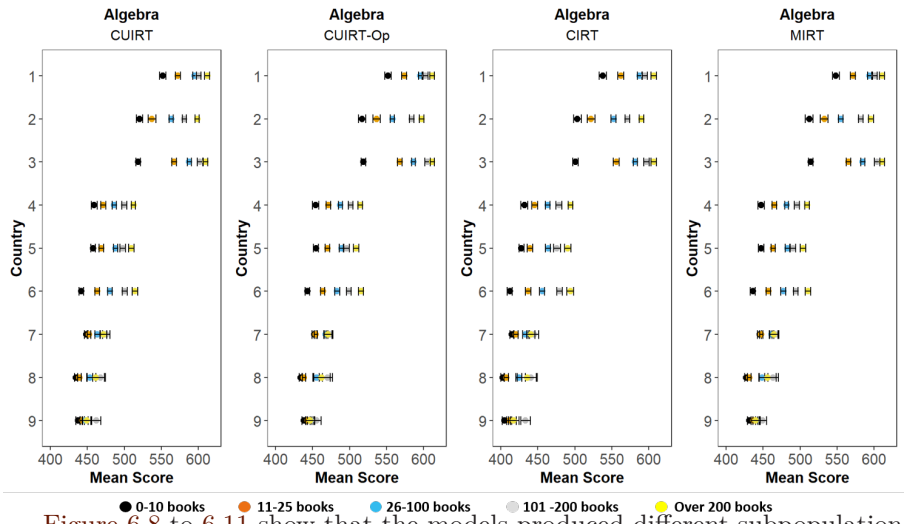


Figure 6.8 to 6.11 show that the models produced different subpopulation scores. Each figure plots 4 panels, and each panel plots the subscale scores estimated for nine countries. The subpopulation subscale scores are color-coded.

Upon visual inspection of the figures, subpopulation score differences were noted. The magnitude of these subpopulation differences differed by country. In other words, the differences did not follow a specific pattern, and the score differences were not large, except for country 3 in the MIRT model.

I also plotted the standard errors of the reported subscale scores (see Figure 6.12). The results presented in Figure 6.12 show the SEs of all of the subscale scores that were estimated from each of the studied models by the

Figure 6.9
Estimated Data and Chance Subscale Scores by Books in the Home

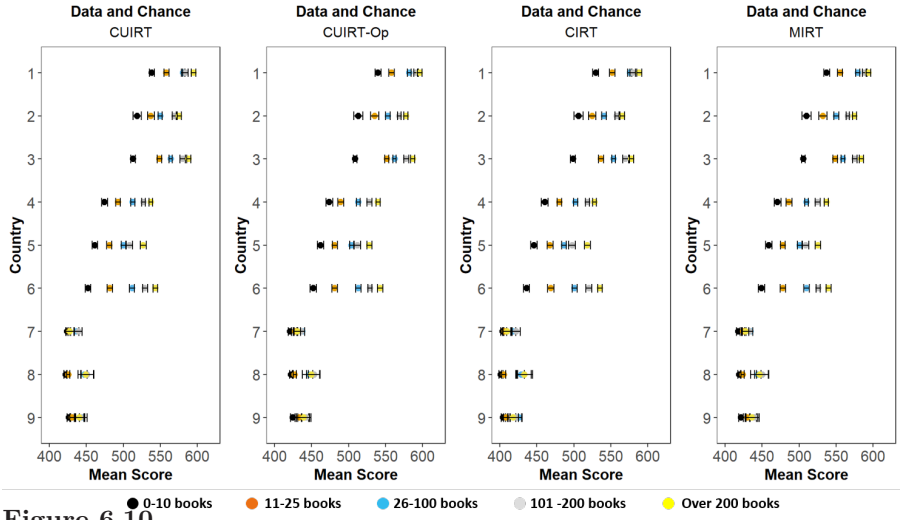
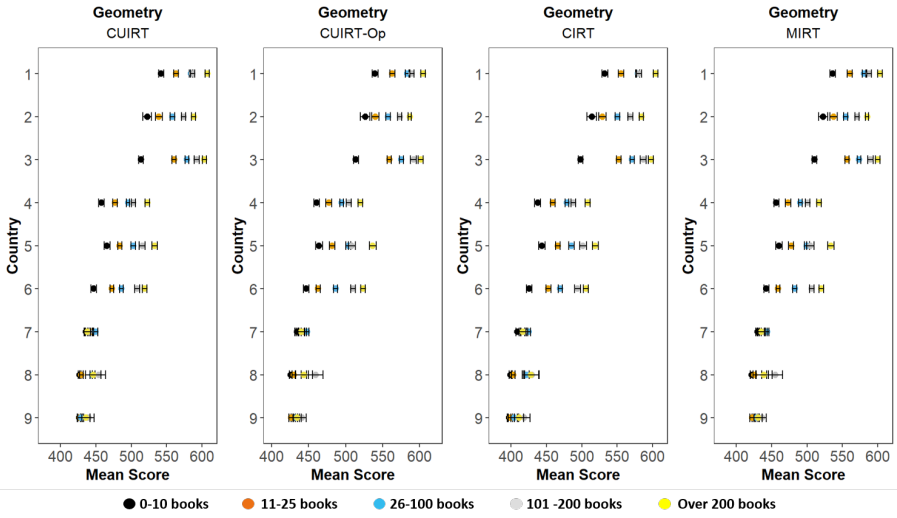
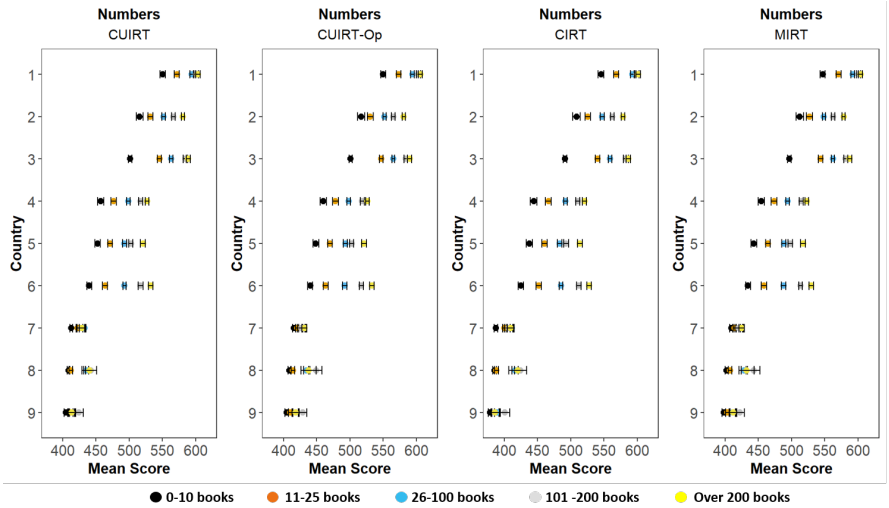


Figure 6.10
Estimated Geometry Subscale Scores by Books in the Home



number of books at home. The panels on each row in Figure 6.12 correspond

Figure 6.11
Estimated Numbers Subscale Scores by Books in the Home

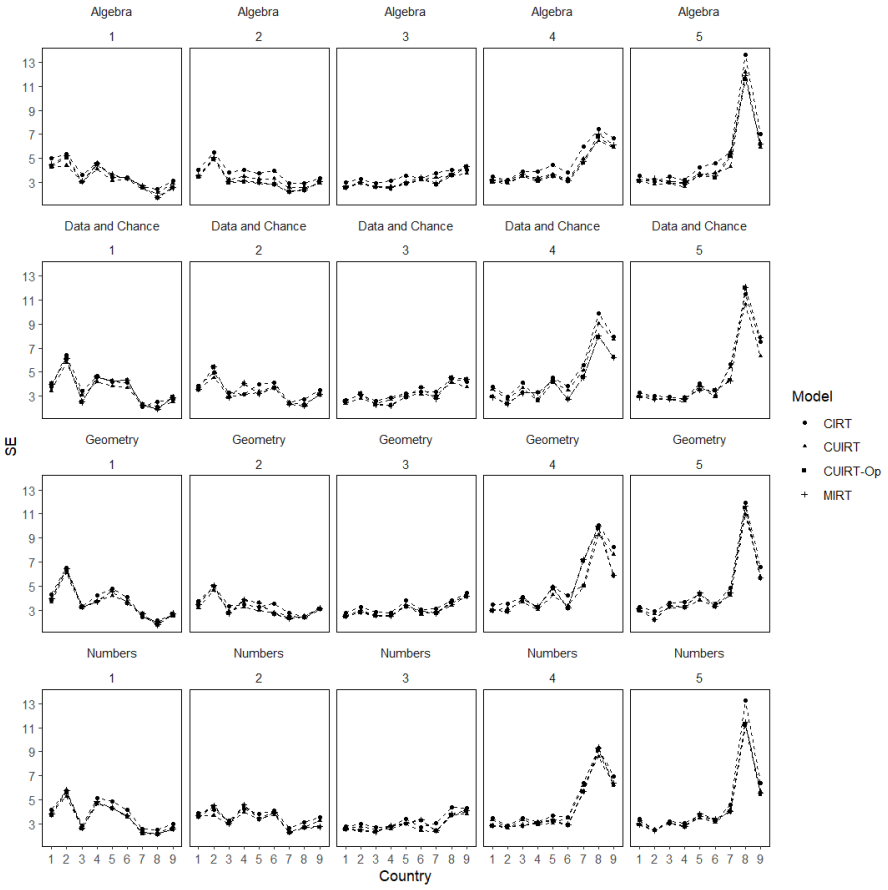


to the four subdomains that were assessed; and each panel in the rows contains the SE for the five, respective, subpopulations' scores by model. For example, the top-left panel shows the SEs of each country's estimated Algebra subscale score for students that reported owning 0–10 books at home. Each of the four lines join the SEs of the subscale score estimates calculated from the CUIRT, CUIRT-Op, CIRT and MIRT models.

Several patterns related to the SEs of the scores reported for "number of books at home" subscale scores were observed. First, the findings presented in Figure 6.12 show that CIRT produces scores with the largest SEs. In contrast, the other three models show comparable SEs. Second, Countries 1 and 2 showed larger SEs regardless of subdomain where students reported to owning 0-100 books. The SEs for these countries decreased as the reported number of books at home increased. Third, Countries 8 and 9 reported larger SEs regardless of subdomain where students reported to owning many books (i.e., 101-200 books and, more than 200). The SEs for these countries was smallest, regardless of subdomain, where students owned 0-10 books, and 11-25 books.

Table 6.4 shows the average SEs of each models subdomain score. It should be seen that SEs were the lowest on all models in Group 3, "26-100 books at home". The results presented in the Table 6.4 show that CUIRT-Op mostly produces the smallest average SEs. In only few circumstances, CUIRT produced lower average SEs than CUIRT-Op. That is:

Figure 6.12
Standard Error of the Sub-Population Scores: Books at Home



Note: 1 = "0–10 books"; 2 = "11–25 books"; 3 = "26–100 books"; 4 = "101–200 books"; 5 = "More than 200"

(a) Algebra; 5.

(b) Data and Chance; 2 and 3.

(c) Geometry; 2, 4 and 5.

(d) Num; 2 and 3.

Similar to the results presented in Section 6.2.2.1, CIRT subscale scores had the largest average SEs compared to CUIRT, CUIRT-Op and MIRT across all subdomains and subpopulations (see Table 6.4). The repeated-measures ANOVA results whether the difference between standard errors are statistically significant are presented in the paragraph that follows.

Table 6.4
Average Standard Errors of the Subscale Scores Reported by Number of Books at Home

	Algebra				
Model	1	2	3	4	5
CUIRT	3.31	3.32	3.06	4.12	4.60
CUIRT-Op	3.41	3.08	3.04	4.08	4.67
CIRT	3.74	3.82	3.45	4.77	5.37
MIRT	3.49	3.17	3.12	4.20	4.80
	Data and Chance				
CUIRT	3.44	3.25	3.06	4.69	4.49
CUIRT-Op	3.65	3.39	3.18	4.18	4.72
CIRT	3.77	3.56	3.32	5.05	4.86
MIRT	3.66	3.40	3.19	4.18	4.72
	Geometry				
CUIRT	3.49	3.06	2.99	4.69	4.46
CUIRT-Op	3.58	3.21	2.98	4.85	4.54
CIRT	3.81	3.39	3.30	5.18	4.95
MIRT	3.62	3.26	3.02	4.92	4.60
	Numbers				
CUIRT	3.51	3.29	2.87	4.40	4.38
CUIRT-Op	3.50	3.37	2.94	4.28	4.31
CIRT	3.89	3.65	3.22	4.76	4.82
MIRT	3.57	3.44	3.00	4.36	4.40

The repeated-measures ANOVA with a Greenhouse-Geisser correction determined that mean standard errors differed statistically significantly when the estimation method was changed for the domain ability estimates $F_{(1.56,68.77)\text{Algebra}} = 60.77$, $p < .05$, partial $\eta^2 =$

.02; $F_{(1.15,50.65)\text{Data and Chance}} = 8.01$, $p < .05$, partial $\eta^2 = .01$; $F_{(1.17,51.55)\text{Geometry}} = 12.19$, $p < .05$, partial $\eta^2 = .01$; and $F_{(1.64,72.09)\text{Number}} = 30.84$, $p < .05$, partial $\eta^2 = .01$. Post-hoc tests using the Bonferroni correction revealed that not all pairwise comparisons were statistically significantly different from each other. There were statistically significant differences between CIRT and all other models: CUIRT, CUIRT-Op, CIRT, and MIRT on all subdomains. According to the results in Table 6.4, CIRT had the highest SEs for all subpopulation subdomain proficiency scores. In other words, subdomain scores from the CIRT model were not as accurate as the other three methods. There were also statistically significant differences between CUIRT-Op and MIRT on the Algebra, Geometry and Numbers subdomains. According to the results in Table 6.4, when compared with MIRT, the CUIRT-Op method had the lowest standard errors for the subdomain scores reported on the three domains. Table 6.4 showed that the differences were not statistically significant on the Numbers subdomain.

Therefore, it can be concluded that CUIRT, CUIRT-Op and MIRT elicited a statistically significant reduction in standard errors of subdomain score estimates compared to CIRT. Likewise, CUIRT-Op showed statistically significant reduction in SEs compared to MIRT on the Algebra, Geometry and Numbers subdomains. Post hoc tests using the Bonferroni correction revealed that some pairwise comparisons were significantly different from each other. The CIRT had the highest mean for standard errors.

6.2.3 Preliminary Summary

All of the results presented in Sections 6.2.1 and 6.2.2 When compared, CUIRT, CUIRT-Op and MIRT performed better than CIRT. The analyses showed that CIRT produced population- and subpopulation-subscale scores with the largest SE. This was often characterized by CIRT resulting in lower scores than CUIRT, CUIRT-Op and MIRT.

6.3 Proportional Reduction in Mean Squared Error

Results of the PRMSE based on the entire sample and at country-level are shown in Figures 6.13 and 6.14, respectively. Each of the nine panels in Figure 6.14 shows the subscale PRMSEs for each of the participating countries. Also, the single panel in Figure 6.13, and each panel in Figure 6.14 compares the PRMSEs for the true subscale score predicted from CUIRT, CUIRT-Op, CIRT and MIRT models, respectively. Each line graph joins points that present model-specific PRMSEs.

Figure 6.13
Subscale PRMSE Based on the Entire Sample.

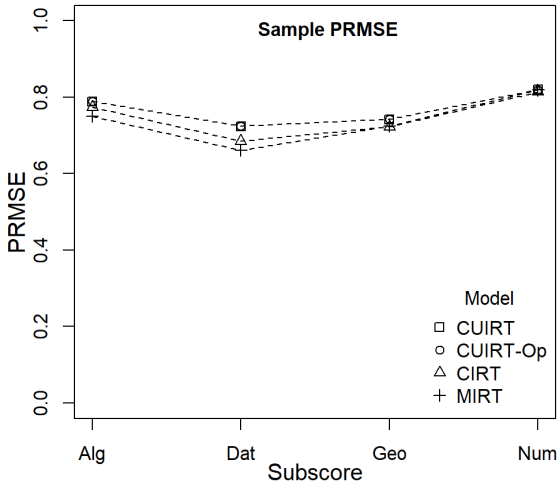


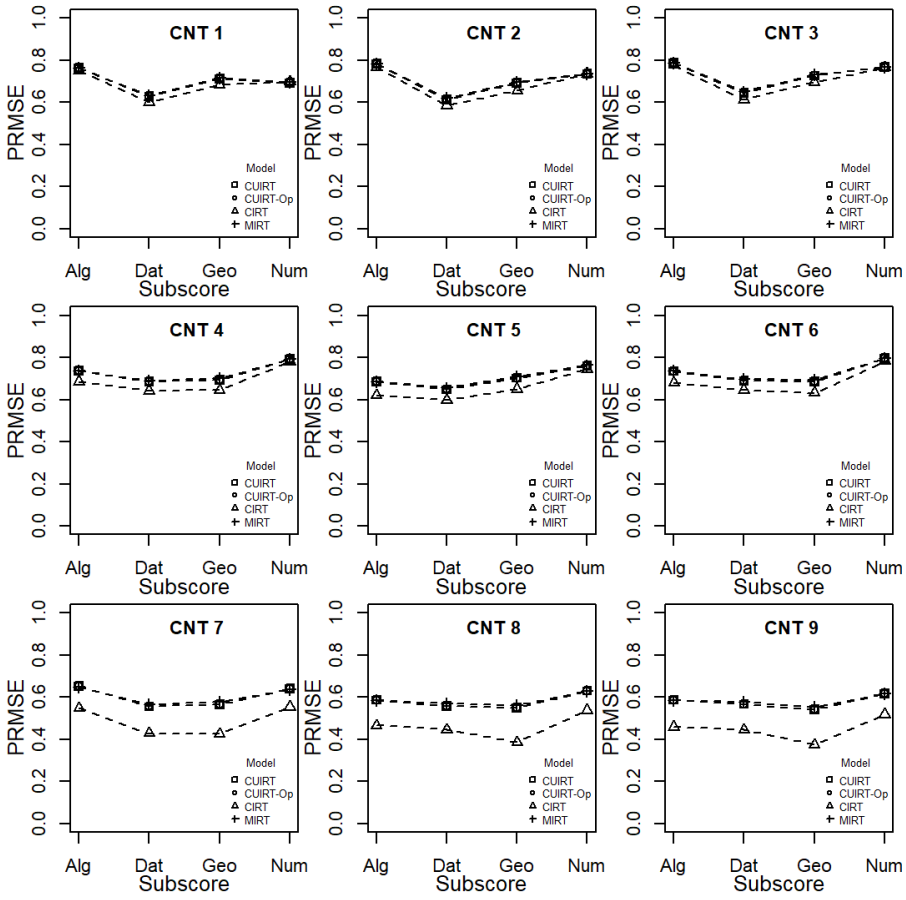
Figure 6.13 shows that CUIRT and CUIRT-Op generally produced higher PRMSE in all subdomains over the entire sample[2]. In contrast, MIRT produced lower PRMSEs on the Algebra, and Data and Chance subdomains. In addition, MIRT, alongside CIRT, reported scores with the lowest subscale score value on these subdomains. However, all of the models reported comparable PRMSEs on the Numbers subscale. Therefore, based on the results presented in Figure 6.13, CUIRT and CUIRT-Op produce larger PRMSE values across all domains. That is, CUIRT and CUIRT-Op produce more valuable subscale scores on this data set.

In contrast, when the PRMSE's were calculated for the country level[3], all of the panels in Figure 6.14 show that the CIRT produced the lowest PRMSE across all subscales. However, the country-specific PRMSE values for CUIRT,

[2]In this case, I considered the entire dataset, comprising of 9 countries, to calculate the PRMSE.

[3]In this case, I considered each country as a standalone sample and calculated their respective PRMSE's.

Figure 6.14
Subscale PRMSE Based on Each Country.



CUIRT-Op, and MIRT were, similar and, higher than those for CIRT. The results seemed to suggest that the CUIRT, CUIRT-Op, and MIRT subscale scores were more valuable than CIRT scores. In part, these findings confirmed the results from simulation Study 2 (i.e., the multiple groups conditions) where MIRT produced the largest PRMSE in some domains (see Section 5.4.2) but clashes with the findings that were reported from the entire sample where MIRT produced the lowest PRMSE's on two of the four subscales.

6.4 Model Fit

I also evaluated the data to identify the model that fit the data the best. First, I compared the relative fit indices to identify the model that fit the entire sample best. Second, I compared the relative fit indices for each country. A model with the lowest $-2ll$, AIC and BIC showed better model fit compared to the other models that were studied. Table 6.5 show the $-2ll$, AIC and BIC for the entire test. Tables 6.6 to 6.8 show the $-2ll$, AIC and BIC for each country, respectively.

All of the criteria suggested that the MIRT model fit the entire data-set better than CUIRT, CUIRT-Op, and CIRT (see Table 6.5). Since empirical subdomain correlations were between .75 and .95 (see Table A.1 in Appendix A), these findings contradicted the results from Simulation study 2 that reported better model fit for CUIRT where subscale correlations are high (i.e., .95; see Section 5.5.2). Of all the models, $-2ll$, AIC and BIC suggested that CUIRT-Op did not fit the entire data-set well compared to CUIRT, CIRT and MIRT. These results were similar to those presented in simulation study 2's single groups test simulations (see Section 5.5.2).

Table 6.5
Model-Fit Based on the Entire Test

Model	$-2ll$	AIC	BIC
CUIRT	108222723.8	108222723.8	108222723.8
CUIRT-Op	110910760.9	110910760.9	110910760.9
CIRT	100952327.9	100953187.9	100958754.1
MIRT	95030870.3	95031742.3	95037386.1

Tables 6.6 to 6.8 show that all of the fit indices reported that CIRT fit all the countries better than all other models. These results were not consistent with the findings of simulation study 2's multiple group's test conditions. The simulation study suggested that CUIRT fit the data better than all other models. These simulation study results were observed where subscale correlations were high (i.e., .75 and .95; see Section 5.5.2). Such high correlations are also observed on the empirical TIMSS 2015 mathematics' test (see Table A.1 in Appendix A). However, it should be noted that the contexts presented in the simulation and empirical studies were different (i.e., different samples).

Table 6.6
Model-Fit for all Countries: −2ll

	Model			
Country	CUIRT	CUIRT-Op	CIRT	MIRT
1	294950.9	245055.4	222210.8	242128.6
2	204534.8	215271.5	199667.1	212946.1
3	215058.0	238464.3	209451.5	235825.3
4	410665.7	419064.2	392268.8	418108.3
5	184347.2	185315.9	174706.0	184847.8
6	324104.6	330994.8	309857.8	330431.3
7	297175.4	294881.5	270809.5	298251.2
8	449633.7	446086.7	407935.8	452724.2

Table 6.7
Model-Fit for all Countries: AIC

	Model			
Country	CUIRT	CUIRT-Op	CIRT	MIRT
1	294950.9	245055.4	222210.8	242128.6
2	204534.8	215271.5	199667.1	212946.1
3	215058.0	238464.3	209451.5	235825.3
4	410665.7	419064.2	392268.8	418108.3
5	184347.2	185315.9	174706.0	184847.8
6	324104.6	330994.8	309857.8	330431.3
7	297175.4	294881.5	270809.5	298251.2
8	449633.7	446086.7	407935.8	452724.2

Table 6.8
Model-Fit for all Countries: BIC

	Model			
Country	CUIRT	CUIRT-Op	CIRT	MIRT
1	306368.7	256385.4	233540.8	253458.6
2	218032.8	228769.5	213165.1	226444.2
3	228490.4	251896.7	222883.9	249257.7
4	423712.2	432110.7	405315.3	431154.8
5	197806.1	198774.8	188164.9	198306.7
6	335529.3	342419.5	321282.5	341855.9
7	310917.3	308623.4	284551.4	311993.2
8	464017.4	460470.4	422319.5	467107.9
9	152178.8	151068.1	139951.0	152924.0

6.5 Summary

The results that were presented were obtained from analyses that were conducted on a data set that comprised of nine countries that participated on TIMSS 2015's mathematics test. In general, the empirical study showed that the magnitude of each sampled-countries reported subscale score was dependent on the subscale score estimation model. The findings showed that CUIRT and CUIRT-Op produced higher score estimates than CIRT and MIRT. It was also reported that CIRT scores were lowest. The difference between the CIRT scores and the other models were larger in the low performing countries. In other words, CIRT underestimated the low performers subdomain scores more than for the middle- and high-performers. In addition, CIRT reported subscale sores with the highest SE. As such, CUIRT, CUIRT-Op, and MIRT resulted in better subscale score estimates than CIRT becaus etheir SEs were lower.

Several other patterns were observed. First, though the reported scores were different, the rank order of the countries remained unchanged. Second, the subpopulation scores differed as well and the trend was similar to that which was observed on the population subscale scores. Third, CIRT also produced subpopulation-subscale-scores with the largest SE.

Results of the analyzed data set suggested that CUIRT subscale scores were more valuable on the Algebra, Data and Chance, and Geometry subscale scores. In contrast, all of the models reported comparable subscale score value on the numbers subdomain. These results were not consistent with the findings

from simulation study 2's results which suggested that CUIRT-Op, CIRT, and MIRT would report more valuable subscale scores. However, when evaluated at the country level, CUIRT, CUIRT-Op, and MIRT subscale scores were more valuable than those reported from CIRT. These results, were consistent with the findings from simulation Study 2.

The results from the empirical study showed that MIRT fit the entire data better than CUIRT, CUIRT-Op, and CIRT. It was also reported that the CIRT model showed the best fit when fit to eight of the nine countries (MIRT fit country 9 better). It should be noted that CUIRT-Op showed the poorest model fit. Also recall that CUIRT showed better fit in simulation study 2, much like TIMSS 2015's mathematics test, where the subscale correlations were high.

Chapter 7

Discussion and Conclusion

7.1 Introduction

This dissertation intended to answer three main research questions through two simulation studies and an empirical study. Both of the simulation studies focused on evaluating how well different subscale score estimation models (i.e., CUIRT, CUIRT-Op, CIRT, and MIRT) performed item- and score-parameter estimation in an ILSA setting. The simulation studies also aimed to identify which model reported the most valuable subscale scores. Simulation studies 1 and 2 were designed to resemble SACMEQ and TIMSS data, respectively. The difference between the two simulation studies is that Study 1 does not employ matrix sampled test booklets whilst Study 2 does. Three design characteristics (i.e., number of subscales, correlation between subscales, subscale length) were varied to create conditions of various test characteristics. The empirical study further illustrated how the studied IRT models perform using the TIMSS eighth grade mathematics dataset. The performance of the subscale scoring methods was evaluated on three study outcomes: bias, ABS, and RMSE. In order to examine the added value of subscale scores, I examined the PRMSE of each model-specific subscale score.

In this dissertation, I also conducted an empirical study that was intended to show how well the four studied models (i.e., CUIRT, CUIRT-Op, CIRT and MIRT) performed in subscale score estimation using the TIMSS 2015 mathematics dataset. To evaluate the results, I compared the magnitudes of the reported population and subpopulation subscale scores under the four different models. I complemented these analyses by looking at the subscale score SEs as well as PRMSE and model fit across the four studied models.

In what follows, I provide recommendations for practitioners (Section 7.2). Then, I discuss the significance and contributions of this dissertation (Section 7.3). At last, I note the limitations of this dissertation as well as suggest potential directions of future research (Section 7.4).

7.2 Summary of Recommendations

This section presents some recommendations for practitioners that are based on the findings of the simulation- and empirical-studies. The simulated test

design characteristics were motivated and informed by the empirical structures of TIMSS and SACMEQ. Each of the research questions I responded to considered a single- and multiple-group context in order to explore the effect of achievement heterogeneity. I also tested my findings on TIMSS 2015 eighth grade mathematics data. As such, the findings of this study may be generalized to tests with similar properties as those presented in this study.

Tables 7.1 and 7.4 provide a summary of the recommendations based on the results of Studies 1 and 2. Tables 7.1 and 7.2 provide recommendations for Study 1's single- and multiple-groups conditions, respectively. Tables 7.3 and 7.4 provide recommendations for Study 2's single- and multiple-groups conditions, respectively. Each of the tables provides suggestions pertaining to the better model to use in (a) item parameter estimation; (b) score estimation; (c) reporting valuable subscale scores; and (d) model fit. These results are presented where the comparisons are made among three (i.e., CUIRT, CIRT, and MIRT) and four models (i.e., CUIRT, CUIRT-Op, CIRT, and MIRT) in Studies 1 and 2, respectively. The virgule or forward slash between the models means that one of the models is the best or provides a better value on one or more subdomains and the other model for the remaining subdomains. For instance CUIRT/MIRT means that CUIRT is better on subdomain 1 and MIRT is better on subdomains 2 and 3 on a three subdomain test. As a suggestion, where two models perform best on separate subdomains at a specific criteria (i.e., score estimation), practitioner's decisions could be based on how well the successful model performs on other criteria (i.e., item parameter estimation, PRMSE and/or model fit).

7.2.1 Summary of Recommendations from Study 1

7.2.1.1 Single Group Conditions

If item parameter recovery is the primary concern, results presented in Table 7.1 suggest that CIRT is more suitable, regardless of test length and subscale correlation. However, in the single group conditions, CUIRT may also be a model of choice where subscale correlation is high. Similar recommendations were proposed by Yao (2010). In her study, it was noted that models that assume test is multidimensionality (i.e., CIRT and MIRT) generally outperformed CUIRT when subscale correlation was low or moderate. Nevertheless, Yao (2010) also pointed out that CUIRT may be suitable where subscale correlation is high.

On the basis of the findings of simulation Study 1, it does not matter which model is used in score estimation. This recommendation was reached

Table 7.1
Summary of Recommendations for Simulation Study 1's Single Group Conditions

						Model fit		
D	J	ρ	IP	Scores	PRMSE	$-2ll$	AIC	BIC
3	5	.45	CIRT	CIRT/MIRT/CUIRT	CIRT	MIRT	MIRT	MIRT
		.75	CIRT	CIRT/MIRT/CUIRT	MIRT	MIRT	MIRT	MIRT
		.95	CIRT/CUIRT	CIRT/MIRT/CUIRT	MIRT	CUIRT	CUIRT	CUIRT
	10	.45	CIRT	CIRT/MIRT/CUIRT	CIRT	MIRT	MIRT	MIRT
		.75	CIRT	CIRT/MIRT/CUIRT	CIRT/MIRT	MIRT	MIRT	MIRT
		.95	CIRT/CUIRT	CIRT/MIRT/CUIRT	MIRT	MIRT	CUIRT	CUIRT
	15	.45	CIRT	CIRT/MIRT/CUIRT	CIRT	MIRT	MIRT	MIRT
		.75	CIRT	CIRT/MIRT/CUIRT	CIRT	MIRT	MIRT	MIRT
		.95	CIRT/CUIRT	CIRT/MIRT/CUIRT	MIRT	MIRT	MIRT	MIRT
5	5	.45	CIRT	CIRT/MIRT/CUIRT	CIRT/MIRT	MIRT	MIRT	MIRT
		.75	CIRT	CIRT/MIRT/CUIRT	MIRT	MIRT	MIRT	MIRT
		.95	CIRT	CIRT/MIRT/CUIRT	MIRT	CUIRT	CUIRT	CUIRT
	10	.45	CIRT	CIRT/MIRT/CUIRT	CIRT	MIRT	MIRT	MIRT
		.75	CIRT	CIRT/MIRT/CUIRT	MIRT	MIRT	MIRT	MIRT
		.95	CIRT	CIRT/MIRT/CUIRT	MIRT	CUIRT	CUIRT	CUIRT
	15	.45	CIRT	CIRT/MIRT/CUIRT	CIRT	MIRT	MIRT	MIRT
		.75	CIRT	CIRT/MIRT/CUIRT	CIRT/MIRT	MIRT	MIRT	MIRT
		.95	CIRT	CIRT/MIRT/CUIRT	MIRT	MIRT	MIRT	CUIRT

Note. D = Number of subscales, J = Subscale length; ρ = Subscale correlation, IP = Item parameters.

because the scores did not show large sensitivity to the specified subscale score estimation model. This conclusion was reached regardless of CUIRT showing slightly more bias (to the third, fourth decimal, or more) than the other studied models where subscale correlation was .95[1]. As a result, either CUIRT, CIRT or MIRT are optimal regardless of number of subscales, test length and subscale correlation. This recommendation seemed to align well with the sufficient statistic principle of the Rasch model. Conceptually, the sufficiency principle in statistics says that if $t(x)$ is a sufficient statistic for a parameter theta, θ, then $t(x)$ contains all the information that is possible to use from x to infer θ (Andersen, 1977). In the case of a number of item scores $x_1, x_2, x_3, \ldots, x_K$ that follow the Rasch model, the sum score, $s(x)$, of a K item test $s(x) = x_1 + x_2 + x_3 + \cdots + x_K$ is a sufficient statistic for the underlying proficiency parameter, θ. That being the case, we would not expect any differences in the estimated scores since the models were applied to the

[1]Recall that CUIRT based item parameters had larger bias compared to those estimated from CIRT and MIRT where subscale correlation was low to moderate.

same datasets which had the same underlying sufficient statistic, the total score.

If subscale value is of primary concern, Table 7.1 shows that either MIRT or CIRT are the methods of choice. Nonetheless, CIRT will generally provide more valuable subscale scores, regardless of test length, where subscale correlation is low. But then, the results presented in Table 7.1 suggest that practitioners may also use MIRT to obtain more valuable subscale scores on a test with more subdomains with shorter within-domain scales and low correlations. In addition, practitioners can use either CIRT or MIRT on tests with longer subscales and moderate correlation, regardless of subdomains (e.g., 3 compared to 5).

With regards to model fit, MIRT shows better fit than CUIRT and CIRT on tests that have short subscale lengths when subscale correlations are low and moderate. In contrast, CUIRT is the best fitting model compared to all other studied models on all short subscale test conditions, when subscale correlation is high. For the longer subscale tests (i.e., $J = 15$), MIRT is generally the the best fitting model compared to all other studied models since it is the better fitting model regardless of subscale correlation. Depending on the model fit index used to make decisions, practitioners may fit the CUIRT model on five subdomain tests that comprise of 15 items-per-subdomain since BIC suggests better fit.

7.2.1.2 Multiple Groups Conditions

If item parameter recovery is the primary concern, results presented in Table 7.2 suggest that CIRT is optimal compared to the other models regardless of test length and subscale correlation. Similar recommendations were provided by Yao (2010).

Where score recovery is of primary concern, all of the models provide comparable results across all test conditions. Therefore, all models would be deemed optimal. A similar recommendation was given for Study 1's single group conditions.

MIRT generally provides more valuable subscale scores, regardless of test length and subscale correlation. However, CIRT may be used to obtain more valuable subscale scores in all 10 items-per-subdomain test conditions. Similarly, practitioners may use CIRT in all tests with more domains with longer subscales, regardless of subscale correlation to obtain valuable subscale scores.

When subscale correlations are low and moderate, the MIRT model shows better model fit. On the basis of the study results, this is true regardless of test length. Where the subscales are highly correlated, CUIRT results in better model fit regardless of test length.

Table 7.2
Summary of Recommendations for Simulation Study 1's Multiple Groups Conditions

						Model fit		
D	J	ρ	IP	Scores	PRMSE	$-2ll$	AIC	BIC
3	5	.45	CIRT	All	MIRT	MIRT	MIRT	MIRT
		.75	CIRT	All	MIRT	MIRT	MIRT	MIRT
		.95	CIRT	All	MIRT	CUIRT	CUIRT	CUIRT
	10	.45	CIRT	All	MIRT/CIRT	MIRT	MIRT	MIRT
		.75	CIRT	All	MIRT	MIRT	MIRT	MIRT
		.95	CIRT	All	MIRT	CUIRT	CUIRT	CUIRT
	15	.45	CIRT	All	MIRT	MIRT	MIRT	MIRT
		.75	CIRT	All	MIRT	MIRT	MIRT	MIRT
		.95	CIRT	All	MIRT	CUIRT	CUIRT	CUIRT
5	5	.45	CIRT	All	MIRT	MIRT	MIRT	MIRT
		.75	CIRT	All	MIRT	MIRT	MIRT	MIRT
		.95	CIRT	All	MIRT	CUIRT	CUIRT	CUIRT
	10	.45	CIRT	All	CIRT/MIRT	MIRT	MIRT	MIRT
		.75	CIRT	All	MIRT	MIRT	MIRT	MIRT
		.95	CIRT	All	MIRT	CUIRT	CUIRT	CUIRT
	15	.45	CIRT	All	CIRT/MIRT	MIRT	MIRT	MIRT
		.75	CIRT	All	CIRT/MIRT	MIRT	MIRT	MIRT
		.95	CIRT	All	CIRT/MIRT	CUIRT	CUIRT	CUIRT

Note. D = Number of subscales, J = Subscale length; ρ = Subscale correlation, IP = Item parameters.

7.2.2 Summary of Recommendations from Study 2

7.2.2.1 Single Group Conditions

If item parameter recovery is the primary concern, results from Table 7.3 suggest that CIRT and MIRT may be more suitable where subscales have low to moderate correlations. However, in the single group conditions (see Table 7.3), CUIRT would be more suitable at item parameter estimation where subscales are highly correlated.

In addition, if score recovery is of primary concern, either CUIRT or CUIRT-Op would be more suitable in the three and subdomain tests. Based on the results of the study, practitioners may also use MIRT to estimate scores on the

Table 7.3
Summary of Recommendations for Simulation Study 2's Single Group Conditions

						Model fit		
D	J	ρ	IP	Scores	PRMSE	$-2ll$	AIC	BIC
3	40	.45	CIRT/MIRT	CUIRT/CUIRT-Op	MIRT	MIRT	MIRT	MIRT
		.75	CIRT/MIRT	CUIRT/CUIRT-Op	MIRT	CUIRT	CUIRT	CUIRT
		.95	CUIRT	CUIRT/CUIRT-Op	MIRT/CUIRT-Op	CUIRT	CUIRT	CUIRT
	60	.45	CIRT/MIRT	CUIRT/CUIRT-Op	MIRT	MIRT	MIRT	MIRT
		.75	CIRT/MIRT	CUIRT/CUIRT-Op	MIRT/CUIRT-Op	CUIRT	CUIRT	CUIRT
		.95	CUIRT	CUIRT/CUIRT-Op	MIRT/CUIRT-Op	CUIRT	CUIRT	CUIRT
4	40	.45	CIRT/MIRT	MIRT/CUIRT/CUIRT-Op	MIRT	MIRT	MIRT	MIRT
		.75	CIRT/MIRT	CUIRT/CUIRT-Op/MIRT	MIRT	CUIRT	CUIRT	CUIRT
		.95	CUIRT	CUIRT/CUIRT-Op	MIRT	CUIRT	CUIRT	CUIRT
	60	.45	CIRT/MIRT	MIRT/CUIRT/CUIRT-Op	MIRT	MIRT	MIRT	MIRT
		.75	CIRT/MIRT	CUIRT/CUIRT-Op/MIRT	MIRT	CUIRT	CUIRT	CUIRT
		.95	CUIRT	CUIRT/CUIRT-Op/MIRT	MIRT	CUIRT	CUIRT	CUIRT

Note. D = Number of subscales, J = Subscale length; ρ = Subscale correlation, IP = Item parameters.

four subdomain tests (see Table 7.3). When score recovery is the primary issue, CIRT may not be optimal. The results presented in Section 5.3.3.1 showed that CUIRT, CUIRT-Op, and MIRT generally produced subscale score estimates with the smallest bias whilst CIRT reported subscale scores with larger bias.

In general, MIRT also produces more valuable subscales on all test conditions. These recommendations echo findings from previous studies by Thissen (2013) and Wedman and Lyrén (2015). The researchers suggested that MIRT produces more valuable subscale scores where subscale correlation is high and augmented methods (i.e., latent regression) are used. However, Table 7.3 shows that CUIRT-Op may also result in valuable subscale scores on three subdomain tests where subscales have low and moderate correlations.

Furthermore, if subscale correlation is low, MIRT fits the data better as compared to the other models. Where subscale correlations are moderate and high, CUIRT would result in better model fit.

7.2.2.2 Multiple Groups Conditions

If item parameter recovery is the primary concern, results from Table 7.4 suggest that CIRT and MIRT are more suitable, regardless of test length, where subscale correlation is low or moderate. However, in the single group conditions (see Table 7.4), CUIRT would be more suitable at item parameter estimation where subscale correlation is high.

Table 7.4
Summary of Recommendations from Simulation Study 2's Multiple Groups Conditions

						Model fit		
D	J	ρ	IP	Scores	PRMSE	$-2ll$	AIC	BIC
3	40	.45	CIRT/MIRT	CIRT/MIRT	MIRT/CUIRT-Op	MIRT	MIRT	CUIRT
		.75	CIRT/MIRT	CIRT/MIRT	MIRT/CUIRT-Op	CUIRT	CUIRT	CUIRT
		.95	CUIRT	CIRT/MIRT	MIRT/CUIRT-Op	CUIRT	CUIRT	CUIRT
	60	.45	CIRT/MIRT	CIRT/MIRT	MIRT/CUIRT-Op	MIRT	MIRT	MIRT
		.75	CIRT/MIRT	CIRT/MIRT	MIRT/CUIRT-Op	CUIRT	CUIRT	CUIRT
		.95	CUIRT	CIRT/MIRT	MIRT/CUIRT-Op	CUIRT	CUIRT	CUIRT
4	40	.45	CIRT/MIRT	CIRT/MIRT	CIRT/MIRT	CUIRT	CUIRT	CUIRT
		.75	CIRT/MIRT	CIRT/MIRT	CIRT/MIRT	CUIRT	CUIRT	CUIRT
		.95	CUIRT	CIRT/MIRT	CIRT/MIRT	CUIRT	CUIRT	CUIRT
	60	.45	CIRT/MIRT	CIRT/MIRT	CIRT/MIRT	MIRT	MIRT	MIRT
		.75	CIRT/MIRT	CIRT/MIRT	CIRT/MIRT	CUIRT	CUIRT	CUIRT
		.95	CUIRT	CIRT/MIRT	CIRT/MIRT	CUIRT	CUIRT	CUIRT

Note. D = Number of subscales, J = Subscale length; ρ = Subscale correlation, IP = Item parameters.

In addition, if score recovery is of primary concern, either CIRT or MIRT is more suitable in the three subdomain tests regardless of test condition. These two models that assumed multidimensionality of the test in the entire scoring process generally outperformed CUIRT and CUIRT-Op. These findings echoed the results from several studies that were conducted on tests for individual inference (e.g., de la Torre & Patz, 2005; de la Torre et al., 2011).

In general, MIRT and CUIRT-Op produce more valuable subscales for all conditions of the three subdomain test conditions. Practitioners may wish to know that these models assume that the subdomains are correlated when estimating subscale scores. In addition, a model similar to CUIRT-Op was operationalized on TIMSS 2015. However, on the basis of the results, MIRT and CIRT will produce more valuable subscale scores for all four subdomain test conditions.

Table 7.4 shows that either CUIRT or MIRT result in better model fit when compared to the other models. That is, when subscale correlation is moderate or high, CUIRT results in better model fit regardless of test length. In contrast, MIRT is expected to fit the data best on all low subscale correlation tests regardless of test length.

7.2.2.3 Consistency of the Results from the Empirical and Simulation Studies

The empirical study was conducted in order to validate the findings from the simulation studies. Though the Study 2 and the empirical study resembled each other, the contexts were slightly different. First, the sample compositions were different (64,112 examinees in the empirical study as opposed to 30,000 simulated examinees in Study 2's multiple-groups test conditions). Second, the empirical study included principal components in the conditioning model whilst the empirical study did not. Third, the instruments were different in that the empirical study included subscales of uneven lengths whilst simulation Study 2's subscales were even.

If score estimation is of primary concern, results from the empirical study and simulation Study 2 contradicted each other. On the basis of the results from the empirical study, MIRT, CUIRT-Op and CUIRT are the more optimal methods for population- and subpopulation-subscale score estimation. Results from the empirical study showed that CIRT may not be suitable for subscale score estimation because the CIRT scores had high SEs that would make it hard to distinguish the country achievement on these subdomains. In contrast, the results from Table 7.4 showed that CIRT would likely result in the best subscale score estimates. Though CIRT and MIRT results produced comparable score estimates in simulation Study 2's multiple groups study, evidence of higher SEs that were observed in the empirical study seemed to support the reporting of subdomain scores estimated using other methods. Though CIRT reported larger SEs for the population and subpopulation scores, practitioners should not expect any differences in rank ordering of countries on achievement, only differences in magnitude of the reported score if any of the models were operationalized.

In circumstances where subscale value is of primary concern, findings from the empirical study reported that CUIRT-Op, CUIRT and MIRT are more optimal amongst the compared models. In part, the recommendations from the empirical study conflicted the results of simulation Study 2's multiple groups simulations. Study 2's results generally showed that MIRT and, to an extent, CUIRT-Op and CIRT reported more valuable subscale scores thus contradicting findings from the empirical study. However, the findings from the empirical study echo findings from previous studies by Thissen (2013) and Wedman and Lyrén (2015). The researchers suggested that a model likely produces more valuable subscale scores where subscale correlation is high and augmented methods (i.e., latent regression) are used. To that effect, it would be expected that CUIRT would result in higher PRMSEs when latent regression

is used than when collateral information is not included in the scoring model. However, improvements in PRMSE were more evident from CUIRT than the other studied models in this empirical example.

Based on the empirical analyses, CIRT generally showed better fit compared to the other models. These results partially contradicted the findings of simulation study 2's multiple groups simulations which suggested that MIRT and CUIRT may be more suitable where subscale correlation is low and moderate/high, respectively. Coincidentally, CUIRT-Op showed the poorest model fit for countries 2 to 6. These results did not come as a surprise since CUIRT-Op presents a model mismatch. That is, the item parameters are estimated from a CUIRT model, but scoring employs a MIRT model. This result seems to call to question TIMSS' choice of a model that resembles CUIRT-Op as their preferred subscale score estimation model. However, based on the results of the empirical study, the best fitting model at country level (i.e., CIRT) is likely to produce scores with larger SEs.

7.3 Significance and Contributions

The results of this dissertation contribute to the existing subscale score estimation literature. Subscore reporting is a relatively novel research area in educational measurement. Though there is a body of research into subscale score reporting at the individual level, there is a paucity of research into subscale score estimation in an ILSA context. Eminently, there is little subscale score research in contexts where the emphasis is at the population level, and where sophisticated booklet designs require specialized achievement methods. There is limited research regarding how well different subscale score estimation models perform in item- and population-parameter estimation. Furthermore, not many studies have been conducted to examine the added value of subscale scores. This dissertation extended some of the studied conditions that were evaluated in research aimed at subscale score reporting at the individual level to the ILSA context. That is, the test design factors included: number of subscales, subscale length, and [balanced] subscale correlations. Hence, this dissertation begins to fill the void of research and bring methodological contributions to subscale score reporting in an ILSA contest under different conditions.

The results of the simulation studies also inform test practitioners as to the selection of the most appropriate subscale score estimation method. Information about the best model at item- and score-parameter recovery, as well as a consideration of the PRMSE, and model fit would empower practitioners to make a good decision about which model would be optimal. Having these results from a single- or multiple-groups perspective may further empower practitioners

in their decision making. For example, Tables 7.1 and 7.4 could be used as reference to make decisions when tests have certain characteristics similar to those studied in this manuscript. For instance, if a test is administered to multiple populations, has relatively few subdomains, and a moderate number of items-per-subdomain (i.e., in Study 2), and the subdomains are moderately correlated, then Table 7.4 would provide insight as to which subscale score estimation model is optimal for (a) item parameter estimation, (b) score estimation, (c) subscore value, and (d) model fit.

In addition, the results may be used to provide headway on the performance of the subscale score estimation models from tests that collect data using the booklet designs and use latent regression methods to estimate subscale scores. The results from simulation Study 2's multiple groups studies suggested that CIRT and MIRT were the best, among the studied models, for estimating subscale scores. These findings contradicted the single-group's designs which favoured CUIRT and CUIRT-Op to report better subscale score estimates and seemed to suggest that specifying multidimensionality of the construct had more benefits when the sample comprised of several populations. The advantages of CIRT and MIRT extended to reporting subscale scores that were more valuable. However, MIRT had an edge over CIRT in that it performed better on the empirical study. The results suggested that the currently operationalized model, CUIRT-Op, performed best at subscale score estimation where the sample is comprised of a single group. This means that there may not be any need to run two separate item calibration and scoring processes to estimate the overall- and subscale-scores had one model performed significantly better than the other. Since subscales on most ILSA are highly correlated, the choice of CUIRT may be further justified considering that the model performs as good as CIRT and MIRT.

The advantage of one model over the other seemed to relate with subscale correlation, item parameter properties and distributions, and model fit. That is, the advantage was more prominent for the models that specified multidimensionality when the subscale correlation was lower, and closely competing models would easily be differentiated based on the distribution of item parameters. In test conditions where subscale correlation was low, MIRT showed better fit. In cases when subscale correlations are higher, the unidimensional family of models showed some improvements. High correlations coincided with CUIRT showing better model fit. Better fitting models also corresponded with better subscale score estimates. As a result, recommendations are made that practitioners should consider how much effort they would like to put in subscale scoring. However, it was a surprising result that MIRT was not the best performing model across all the simulated

test conditions since it was the generating model. The results presented in Chapter 5 showed that MIRT was not the best performing model where subscale correlations were high, and some instances where the correlations were moderate. One likely reason was that the underlying “true" data generating mechanism of the chosen data, where high subscale correlations were high, represented a more unidimensional model.

Study 2’s results showed that it may be useful to consider the model choice when the sample becomes more diverse with regards to performance. The results from item parameter estimation were comparable, regardless of specified sample, in that CIRT and MIRT performed best. But from a score estimation point of view, the unidimensional models were the proposed models for use when the sample being assessed is homogeneous. Whereas CIRT and MIRT were preferable when the samples comprised of countries that were placed on separate positions on the score continuum. However, if practitioners are to take subscale value into consideration, MIRT showed an edge over CIRT. The results of the empirical studies also showed that the magnitude of the reported scores may also be different. The observed score differences were greater for certain groups than others. For example, the differences between CIRT scores and those estimated from CUIRT, CUIRT-Op, and MIRT was higher for the low performing countries. These results suggested that if CIRT were used in subscale score estimation, the magnitude of the scores would be even lower in the low-performing countries. But then, these findings may have been particular to this empirical study. The consequences of such under- or over-estimation, and bias of scores may have implications over decisions that are made from the subscale scores. This is true where the over- or under-estimated may be used to in the evaluation of educational systems with the aim of amending an educational system’s educational policies.

7.4 Limitations and Future Research

The dissertation has several limitations. First, much like any simulation study, the general test design choices limit the generalization of the findings. In the current simulation studies, I looked at test specification factors such as: number of subscale scores, subscale length, and subscale correlation. However, given the large number of ILSAs and their various characteristics, it is impossible to include all various factors in design. For example, this study did not look at: unbalanced subscale lengths and subscale correlations. Also, simulation Study 1 looked at the Rasch model, whereas Study 2 employed the 2PL and GPCM models. It is not uncommon for studies to have widely different test specifications and use other IRT models. Given these design limitations, future

research should consider more conditions. For example, future studies could include unbalanced subscale lengths and subscale correlations.

Second, the generating parameter estimates were not exhaustive of what may be empirically observed. For instance, the generating item difficulty parameters used in Study 1 were assumed to be drawn from a uniform distribution (i.e., $\beta_{di} \sim \mathcal{U}(-2.35, 2.35)$). Empirically, the item parameters may be distributed differently (i.e., as given in test specification and design). In addition, each country's generating subscale score distribution was equal on each subdomain. These scores were based on the observed overall, HAKT test score and its subsequent standard deviation. For example, if Country X had an observed subscale score of .45 and a standard deviation of .83, then this was the specification for all subdomains. Empirically, participating countries may not have the same subsale scores on all subdomains. The reason for this specification was that SACMEQ III did not report subscale scores; and test-level information was not made public due to test security reasons. Therefore, future research should consider studying tests with different generating parameters. For example, simulation studies could include item parameter distribution as a study condition in order to identify conditions where benefits exist. As such, future studies may extend Study 1 to include different population and subpopulation generating subscale score distributions. These distributions may be theoretical or empirically observed from ILSAs. Furthermore, in my simulation studies, the generated item parameters were fixed for all 100 replications on each specific test condition. That is, one single draw of item parameters was made, and the selected item parameters were kept for the specified replications. Such a specification makes it a challenge to tease out the effects of varied conditions, since the item parameters are specific to a subdomain. As such, the tests at different subscale lengths not only differed in the simulation design factors (number of subscales, subscale length, and subscale correlation), but also in the item parameters. This may explain why the patterns observed in the simulation studies were unclear. To that effect, strong conclusions as to how the studied models performed in different test conditions may not be warranted. Future studies could vary item parameters across replications thus strengthening the generalizability of the findings.

Third, in simulation Study 2's data generation did not fully resemble TIMSS 2015. For example, though the study managed to capture most of the test design aspects, the dichotomously scored items were assumed to be 2PL items. von Davier (2016) argued that the results obtained from the Rasch, 2PL, or 3PL models were highly correlated, close to 1. But then, OECD (2017) and Mazzeo and von Davier (2008) noted that there are some concerns over the insufficiencies of the Rasch model to adequately address the complexity of data

from ILSA such as TIMSS and PISA. In addition, the latent regression in simulation Study 2 specified 10 background variables, that were not subject to principle component analysis. As such, the selected variables may not have explained as much variance in the scores compared to the numerous principal components that are incorporated in TIMSS' empirical conditioning model. To that effect, the conditioning model specified in simulation Study 2 may not have had the same consequences in the estimation of proficiency compared to TIMSS' operationalized model. Following which, this selection of variables may have consequences over how the study results may be generalized to TIMSS. The selection of fewer variables for the latent regression model were extended to the empirical study; much unlike the TIMSS study that uses all available background information. However, unlike simulation Study 2, principal components comprised the empirical study's conditioning model. These design and scoring changes, although not strictly in line with operational procedures, provide insights into expectations around different methods of subscale score estimation.

Fourth, the data were generated using the MIRT model. The chosen model assumed that the studied subscales had underlying relationships. In other words, the simulation studies assumed that the subscales were correlated by specfying a compensatory MIRT model. This assumption was reasonable considering that the subscales on ILSA are often correlated. However, other data generation models may assume different subscale relationships. For example, if a CUIRT model were adopted, it assumes that the assessment measures one construct. Or, if a higher-order IRT (HO-IRT) model were used for data generation, it assumes that an overarching general proficiency above the subscale scores (i.e., a general mathematics construct above the algebra, data and chance, geometry, and numbers subscales). Given the numerous IRT models, there is an uncertainty as to whether MIRT fully captures the subscale relationships or not. Since MIRT was one of the four compared subscale scoring models, it may be unclear whether its benefited from being the data generating model. Future research could examine the performance of the four subscale score estimation models but adopt different data generation models. This would make it possible to gauge the performance of MIRT over tests that assume different subscale relations. For instance, generating data from HO-IRT and comparing the performance of several models at subscale score estimation. Ideally, HO-IRT would be more suitable since the model captures the relationship between subscales, as well as their relationship with the general construct from which an overall score is also reported. As such, future research could could be extended to simulate data from HO-IRT, and compare model performance in the simultaneous estimation of subscale- and overall-scores.

Fifth, the simulation studies only compared four subscale subscoring methods. There may exist other subscale score estimation methods such as bi-factor model, HO-IRT (and extensions thereof). Future studies may extend the simulation- and empirical-studies by including the bi-factor model and HO-IRT in the comparison. Since subscales are estimated for cognitive subdomains, future studies may also be extended to include the cognitive diagonostic models (CDMs).

Lastly, simulation Study 1 did not consider augmented extensions of the IRT methods in simulation Study 1. This was because SACMEQ did not specify, in its documentation, whether latent regression techniques were used in score estimation. The augmented methods were only fit in Study 2 since it resembled TIMSS 2015 and scores were estimated from latent regression techniques on that assessment. According to Tao (2009), collateral information that could potentially be utilized in subscale score estimation include (a) information from other subscale scores, (b) information about schools the students attended, and (c) school-level subscore information on the same test obtained from previous students in that school. Though the MIRT model specified in Study 1 is inherently augmented in that it takes subscale correlation into account, the dissertation only considered non-augmented CUIRT and CIRT models. Since SACMEQ also collects volumes of contextual data, future studies may be extended to estimate scores from augmented models that take into account all sources of information in the estimation of subscale scores. Future studies may distinguish from the currently operationalized latent regression techniques by including information from other subscales in the conditioning model.

References

Ackerman, T. A. (1994). Using multidimensional item response theory to understand what items and tests are measuring. *Applied Measurement in Education*, *7*(4), 255–278. https://doi.org/10.1207/s15324818ame0704_1

Adams, R. J., Wilson, M., & Wu, M. (1997). Multilevel item response models: An approach to errors in variables regression. *Journal of Educational and Behavioral Statistics*, *22*(1), 47–76. https://doi.org/10.3102/10769986022001047

Adams, R. J., & Wu, M. L. (2007). The mixed-coefficients multinomial logit model: A generalized form of the Rasch model. In M. von Davier & C. H. Carstensen (Eds.), *Multivariate and mixture distribution rasch models* (pp. 57–75). Springer.

Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, *19*(6), 716–723. https://doi.org/10.1109/TAC.1974.1100705

Andersen, E. B. (1977). Sufficient statistics and latent trait models. *Psychometrika*, *42*(1), 69–81. https://doi.org/10.1007/BF02293746

Andersen, E. B. (1997). The rating scale model. *Handbook of modern item response theory* (pp. 67–84). Springer.

Baird, J., Johnson, S., Hopfenbeck, T. N., Isaacs, T., Sprague, T., Stobart, G., & Yu, G. (2016). On the supranational spell of PISA in policy. *Educational Research*, *58*(2), 121–138. https://doi.org/10.1080/00131881.2016.1165410

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., & Mislevy, R. J. (1982). Adaptive EAP estimation of ability in a microcomputer environment. *Applied Psychological Measurement*, *6*(4), 431–444. https://doi.org/10.1177/014662168200600405

Brennan, R. L. (2012). Utility indexes for decisions about subscores. *Center for Advanced Studies in Measurement and Assessment (CASMA). Research Report*, *33*.

Cai, L. (2010a). High-dimensional exploratory item factor analysis by a Metropolis–Hastings Robbins–Monro algorithm. *Psychometrika*, *75*(1), 33–57. https://doi.org/10.1007/s11336-009-9136-x

Cai, L. (2010b). Metropolis-Hastings Robbins-Monro algorithm for confirmatory item factor analysis. *Journal of Educational and Behavioral Statistics*, *35*(3), 307–335. https://doi.org/doi.org/10.3102/1076998609353115

Camilli, G., & Dossey, J. A. (2019). Multidimensional national profiles for TIMSS 2007 and 2011 mathematics. *The Journal of Mathematical Behavior*, *55*, 100693. https://doi.org/10.1016/j.mex.2019.06.015

Carstens, R., & Hastedt, D. (2010). The effect of not using plausible values when they should be: An illustration using TIMSS 2007 grade 8 mathematics data.

Chalmers, R. P. (2012). mirt: A multidimensional item response theory package for the R environment. *Journal of Statistical Software*, *48*(6), 1–29. https://doi.org/10.18637/jss.v048.i06

Chmielewski, A. K., & Dhuey, E. (2017). The analysis of international large-scale assessments to address causal questions in education policy. *Commissioned paper. Washington, DC. National Academy of Education.*

Costa, P. T., & McCrae, R. R. (1992). Normal personality assessment in clinical practice: The NEO personality inventory. *Psychological Assessment*, *4*(1), 5–3.

Crocker, L., & Algina, J. (1986). *Introduction to classical and modern test theory.* Holt, RineHart, Winston, Inc.

de la Torre, J., & Hong, Y. (2010). Parameter estimation with small sample size a higher-order IRT model approach. *Applied Psychological Measurement*, *34*(4), 267–285.

de la Torre, J., & Patz, R. J. (2005). Making the most of what we have: A practical application of multidimensional item response theory in test scoring. *Journal of Educational and Behavioral Statistics*, *30*(3), 295–311.

de la Torre, J., & Song, H. (2009). Simultaneous estimation of overall and domain abilities: A higher-order IRT model approach. *Applied Psychological Measurement*, *33*(8), 620–639. https://doi.org/10.1177/0146621608326423

de la Torre, J., Song, H., & Hong, Y. (2011). A comparison of four methods of IRT subscoring. *Applied Psychological Measurement*, *35*(4), 296–316. https://doi.org/10.1177/0146621610378653

DeAyala, R. J. (2013). *The theory and practice of item response theory.* Guilford Publications.

Debanne, S. M. (2000). The planning of clinical studies: Bias and precision. *Gastrointestinal Endoscopy*, *52*(6), 821. https://doi.org/10.1067/mge.2000.110757

DeMars, C. E. (2006). Application of the Bi-Factor multidimensional item response theory model to Testlet-Based tests. *Journal of Educational Measurement*, *43*(2), 145–168.

Dwyer, A., Boughton, K., Yao, L., Steffen, M., & Lewis, D. (2006, April). A comparison of subscale score augmentation methods using empirical data. *Presented at the annual meeting of the National Council on Measurement in Education, San Francisco, CA*.

Edwards, M. C., & Vevea, J. L. (2006). An empirical Bayes approach to subscore augmentation: How much strength can we borrow? *Journal of Educational and Behavioral Statistics*, *31*(3), 241–259.

Embretson, S. E., & Reise, S. P. (2000). *Item response theory for psychologists*. Lawrence Erlbaum Associates Publishers.

Erdemir, A., & Atar, H. Y. (2020). Simultaneous estimation of overall score and subscores using MIRT, HO-IRT and Bi-factor model on TIMSS data. *Eğitimde ve Psikolojide Ölçme ve Değerlendirme Dergisi*, *11*(1), 61–75. https://doi.org/10.21031/epod.645478

Fayers, P. M., Hays, R., & Hays, R. D. (2005). *Assessing quality of life in clinical trials: Methods and practice* (Second). Oxford University Press: Oxford.

Feinberg, R. A., & Jurich, D. P. (2017). Guidelines for interpreting and reporting subscores. *Educational Measurement: Issues and Practice*, *36*(1), 5–13.

Frey, A., Hartig, J., & Rupp, A. A. (2009). An ncme instructional module on booklet designs in large-scale assessments of student achievement: Theory and practice. *Educational Measurement: Issues and Practice*, *28*(3), 39–53. https://doi.org/doi.org/10.1111/j.1745-3992.2009.00154.x

Gonzalez, E., & Rutkowski, L. (2010). Practical approaches for choosing multiple-matrix sample designs. *IEA-ETS Research Institute Monograph*, *3*, 125–156.

Goodman, D. P., & Hambleton, R. K. (2004). Student test score reports and interpretive guides: Review of current practices and suggestions for future research. *Applied Measurement in Education*, *17*(2), 145–220. https://doi.org/10.1207/s15324818ame1702_3

Green, B. F., Bock, R. D., Humphreys, L. G., Linn, R. L., & Reckase, M. D. (1984). Technical guidelines for assessing computerized adaptive tests. *Journal of Educational Measurement*, *21*(4), 347–360. https://doi.org/10.1111/j.1745-3984.1984.tb01039.x

Haberman, S. J. (2005). When can subscores have value? *ETS Research Report Series*, *2005*(1), i–15.
Haberman, S. J. (2008a). Subscores and validity. *ETS Research Report Series*, *2008*(2), i–11.
Haberman, S. J. (2008b). When can subscores have value? *Journal of Educational and Behavioral Statistics*, *33*(2), 204–229. https://doi.org/10.3102/1076998607302636
Haberman, S. J., & Sinharay, S. (2010). Reporting of subscores using multidimensional item response theory. *Psychometrika*, *75*(2), 209–227. https://doi.org/10.1007/s11336-010-9158-4
Haberman, S. J., Sinharay, S., & Puhan, G. (2009). Reporting subscores for institutions. *British Journal of Mathematical and Statistical Psychology*, *62*(1), 79–95. https://doi.org/10.1348/000711007X248875
Haladyna, T. M., & Kramer, G. A. (2004). The validity of subscores for a credentialing test. *Evaluation & the Health Professions*, *27*(4), 349–368. https://doi.org/10.1177/0163278704270010
Hambleton, R. K., Jones, R. W., & Rogers, H. J. (1993). Influence of item parameter estimation errors in test development. *Journal of Educational Measurement*, *30*(2), 143–155.
Hambleton, R. K., & Swaminathan, H. (2013). *Item response theory: Principles and applications*. Springer Science & Business Media.
Hambleton, R. K., Swaminathan, H., & Rogers, H. J. (1991). *Fundamentals of item response theory*. Sage.
IEA. (2020). Idb analyzer (version 4.0.35). https://www.iea.nl/data-tools/tools
Kahraman, N. (2013). Unidimensional interpretations for multidimensional test items. *Journal of Educational Measurement*, *50*(2), 227–246.
Kahraman, N., & Kamata, A. (2004). Increasing the precision of subscale scores by using out-of-scale information. *Applied Psychological Measurement*, *28*(6), 407–426. https://doi.org/10.1177/0146621604268736
Kane, M. T. (2013). Validating the interpretations and uses of test scores. *Journal of Educational Measurement*, *50*(1), 1–73.
Kaplan, J. (2019). *Fastdummies: Fast creation of dummy (binary) columns and rows from categorical variables* [R package version 1.6.1].
Kassambara, A. (2020). *rstatix: Pipe-friendly framework for basic statistical tests* [R package version 0.6.0].
Kelley, T. L. (1947). *Fundamentals of statistics*. Harvard University Press.
Kim, J. K., & Nicewander, W. A. (1993). Ability estimation for conventional tests. *Psychometrika*, *58*(4), 587–599. https://doi.org/10.1007/BF02294829

Kim, S., & Lee, W. (2006). An extension of four IRT linking methods for mixed-format tests. *Journal of Educational Measurement*, *43*(1), 53–76. https://doi.org/10.1111/j.1745-3984.2006.00004.x

Kolen, M. J., & Brennan, R. L. (2014). *Test equating, scaling, and linking: Methods and practices*. Springer Science & Business Media.

Kotz, S., & Johnson, N. L. (1982). *Encyclopedia of statistical sciences*. John Wiley & Sons.

Lindblad, S., Pettersson, D., & Popkewitz, T. (2015). International comparisons of school results: A systematic review of research on large scale assessments in education.

Liu, F. (2015). *Comparisons of subscoring methods in computerized adaptive testing: A simulation study* (Doctoral dissertation). The University of North Carolina at Greensboro.

Longabach, T. (2015). *A comparison of subscore reporting methods for a state assessment of English language proficiency* (Doctoral dissertation). University of Kansas.

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

Loyd, B. H., & Hoover, H. (1980). Vertical equating using the Rasch model. *Journal of Educational Measurement*, 179–193.

Lu, I. R., Thomas, D. R., & Zumbo, B. D. (2005). Embedding IRT in structural equation models: A comparison with regression based on IRT scores. *Structural Equation Modeling*, *12*(2), 263–277. https://doi.org/10.1207/s15328007sem1202_5

Luecht, R. M. (2003). Applications of multidimensional diagnostic scoring for certification and licensure tests.

Martin, M. O., & Mullis, I. V. S. (2019). TIMSS 2015: Illustrating advancements in large-scale international assessments. *Journal of Educational and Behavioral Statistics*, *44*(6), 752–781. https://doi.org/10.3102/1076998619882030

Martin, M. O., Mullis, I. V. S., & Foy, P. (2016). TIMSS 2015 assessment design. In I. V. S. Mullis & M. O. Martin (Eds.), *TIMSS 2015 assessment frameworks* (pp. 85–99). TIMSS & PIRLS International Study Center, Lynch School of Education, Boston College; International Association for the Evaluation of Educational Achievement (IEA).

Martin, M. O., Mullis, I. V. S., & Hooper, M. (2016). *Methods and procedures in TIMSS 2015*. TIMSS & PIRLS International Study Center, Boston College.

Masters, G. N. (1982). A Rasch model for partial credit scoring. *Psychometrika*, *47*(2), 149–174.

Matta, T. H., Rutkowski, L., Rutkowski, D., & Liaw, Y. L. (2018). lsasim: An R package for simulating large-scale assessment data. *Large-Scale Assessments in Education*, *6*(1), 15. https://doi.org/10.1186/s40536-018-0068-8

Maughan-Brown, B., & Spaull, N. (2014). HIV-related discrimination among grade six students in nine southern african countries. *PLoS One*, *9*(8), e102981. https://doi.org/10.1371/journal.pone.0102981

Mazzeo, J., & von Davier, M. (2008). Review of the Programme for International Student Assessment (PISA) test design: Recommendations for fostering stability in assessment results. *Education Working Papers EDU/PISA/GB (2008)*, *28*, 23–24.

McDonald, R. P. (2000). A basis for multidimensional item response theory. *Applied Psychological Measurement*, *24*(2), 99–114. https://doi.org/10.1177/01466210022031552

Meijer, R. R., Boevé, A. J., Tendeiro, J. N., Bosker, R. J., & Albers, C. J. (2017). The use of subscores in higher education: When is this useful? *Frontiers in Psychology*, *8*, 305. https://doi.org/10.3389/fpsyg.2017.00305

Mislevy, R. J. (1985). Estimation of latent group effects. *Journal of the American Statistical Association*, *80*(392), 993–997. https://doi.org/10.1080/01621459.1985.10478215

Mislevy, R. J. (1991). The role of collateral information about examinees in item parameter estimation. *Psychometrika*, *56*(2), 177–196. https://doi.org/10.1007/BF02294457

Mislevy, R. J., Beaton, A. E., Kaplan, B., & Sheehan, K. M. (1992). Estimating population characteristics from sparse matrix samples of item responses. *Journal of Educational Measurement , 29*(2), 133–161. https://doi.org/10.1111/j.1745-3984.1992.tb00371.x

Mislevy, R. J., Johnson, E. G., & Muraki, E. (1992). Chapter 3: Scaling procedures in NAEP. *Journal of Educational Statistics*, *17*(2), 131–154. https://doi.org/10.3102/10769986017002131

Mislevy, R. J., & Sheehan, K. M. (1989). The role of collateral information about examinees in item parameter estimation. *Psychometrika*, *54*(4), 661–679. https://doi.org/10.1007/BF02296402

Moloi, M., & Chetty, M. (2014). SACMEQ III, policy brief. *Avaiable at http://www. sacmeq. org/sacmeq–projects/sacmeq–iii/reports. Date of access*, *31*.

Monaghan, W. (2006). The facts about subscores. *R&D Connections*.

Morris, T. P., White, I. R., & Crowther, M. J. (2019). Using simulation studies to evaluate statistical methods. *Statistics in medicine*, *38*(11), 2074–2102. https://doi.org/10.1002/sim.8086

Muraki, E. (1992). A generalized partial credit model: Application of an EM algorithm. *ETS Research Report Series*, *1992*(1), i–30.
OECD. (2002). *PISA 2000 technical report*. OECD Publishing.
OECD. (2005). *PISA 2003 technical report*. OECD Publishing.
OECD. (2009). *PISA 2006 technical report*. OECD Publishing.
OECD. (2017). *PISA 2015 technical report*. OECD Publishing.
Oliveri, M. E., & von Davier, M. (2011). Investigation of model fit and score scale comparability in international assessments. *Psychological Test and Assessment Modeling*, *53*(3), 315.
Puhan, G., Sinharay, S., Haberman, S. J., & Larkin, K. (2010). The utility of augmented subscores in a licensure exam: An evaluation of methods using empirical data. *Applied Measurement in Education*, *23*(3), 266–285. https://doi.org/10.1080/08957347.2010.486287
R Core Team. (2013). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing. Vienna, Austria. http://www.R-project.org/
Reckase, M. D. (2009). *Multidimensional item response theory*. Springer.
Rijmen, F., Jeon, M., von Davier, M., & Rabe-Hesketh, S. (2014). A third-order item response theory model for modeling the effects of domains and subdomains in large-scale educational assessment surveys. *Journal of Educational and Behavioral Statistics*, *39*(4), 235–256. https://doi.org/doi.org/10.3102/1076998614531045
Rosa, K., Swygert, K. A., Nelson, L., & Thissen, D. (2001). Item response theory applied to combinations of multiple-choice and constructed-response items—scale scores for patterns of summed scores. In D. E. Thissen & H. E. Wainer (Eds.), *Test scoring*. Lawrence Erlbaum Associates Publishers.
Rubin, D. B. (1987). *Multiple imputation for nonresponse in surveys*. John Wiley & Sons.
Rutkowski, L., Gonzales, E., von Davier, M., & Zhou, Y. (2014). Assessment design for international large-scale assessments. In L. Rutkowski, M. von Davier, & D. Rutkowski (Eds.), *Handbook of international large-scale assessment: Background, technical issues, and methods of data analysis* (pp. 75–95). CRC Press.
Samejima, F. (2006). Graded response models. *Handbook of item response theory* (pp. 95–107). CRC Press.
Sandefur, J. (2018). Internationally comparable mathematics scores for fourteen African countries. *Economics of Education Review*, *62*, 267–286. https://doi.org/10.1016/j.econedurev.2017.12.003

Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics*, *6*(2), 461–464. https://doi.org/10.1214/aos/1176344136
Sha, S., & McCoy, T. (2014). A comparison of two augmented subscore methods and the role of score distribution. *Paper presented at annual meeting of National Council on Measurement and Education, Philadelphia*.
Sheehan, K. M. (1985). *M-GROUP: Estimation of group effects in multivariate models [computer program]*. Educational Testing Service.
Sheng, Y., & Wikle, C. K. (2007). Comparing multiunidimensional and unidimensional item response theory models. *Educational and Psychological Measurement*, *67*(6), 899–919. https://doi.org/10.1177/0013164406296977
Shin, C. D. (2004). *A comparison of methods of estimating objective scores* (Doctoral dissertation). The University of Iowa.
Shin, D. (2007). A comparison of methods of estimating subscale scores for mixed-format tests. *Report for Pearson Educational Measurement*.
Singer, J. D., & Willett, J. B. (2003). *Applied longitudinal data analysis: Modeling change and event occurrence*. Oxford university press.
Sinharay, S. (2010). How often do subscores have added value? results from operational and simulated data. *Journal of Educational Measurement*, *47*(2), 150–174. https://doi.org/10.1111/j.1745-3984.2010.00106.x
Sinharay, S., Haberman, S. J., & Puhan, G. (2007). Subscores based on classical test theory: To report or not to report. *Educational Measurement: Issues and Practice*, *26*(4), 21–28.
Skorupski, W. P. (2008). A review and empirical comparison of approaches for improving the reliability of objective level scores. *Paper presented at the annual meeting of A Study of the Council of Chief State School Officers*.
Skorupski, W. P., & Carvajal, J. (2010). A comparison of approaches for improving the reliability of objective level scores. *Educational and Psychological Measurement*, *70*(3), 357–375. https://doi.org/10.1177/0013164409355694
Skrondal, A., & Rabe-Hesketh, S. (2004). *Generalized latent variable modeling: Multilevel, longitudinal, and structural equation models*. CRC Press.
Spaull, N. (2011). *A preliminary analysis of SACMEQ III South Africa*. Stellenbosch University.
Stone, C. A., Ye, F., Zhu, X., & Lane, S. (2009). Providing subscale scores for diagnostic information: A case study when the test is essentially unidimensional. *Applied Measurement in Education*, *23*(1), 63–86. https://doi.org/10.1080/08957340903423651

Tao, S. (2009). *Using collateral information in the estimation of sub-scores—a fully Bayesian approach* (Doctoral dissertation). University of Iowa.

Tate, R. L. (2004). Implications of multidimensionality for total score and subscore performance. *Applied Measurement in Education*, *17*(2), 89–112. https://doi.org/10.1207/s15324818ame1702_1

Thissen, D. (2013). Using the testlet response model as a shortcut to multidimensional item response theory subscore computation. In R. E. Millsap, A. van der Ark, D. M. Bolt, & C. M. Woods (Eds.), *New developments in quantitative psychology* (pp. 29–40). Springer.

Thissen, D., & Edwards, M. C. (2005). Diagnostic scores augmented using multidimensional item response theory: Preliminary investigation of MCMC strategies. *Paper presented at the annual meeting of the National Council on Measurement in Education, Montreal, PQ, Canada.*

Thomas, N. (1993). The E-step of the MGROUP EM algorithm. *ETS Research Report Series*, *1993*(2), i–71.

Torney-Purta, J., & Amadeo, J. (2013). International large-scale assessments: Challenges in reporting and potentials for secondary analysis. *Research in Comparative and International Education*, *8*(3), 248–258.

van der Linden, W. J. (1999). Multidimensional adaptive testing with a minimum error-variance criterion. *Journal of Educational and Behavioral Statistics*, *24*(4), 398–412. https://doi.org/10.3102/10769986024004398

von Davier, M., & Sinharay, S. (2010). Stochastic approximation methods for latent regression item response models. *Journal of Educational and Behavioral Statistics*, *35*(2), 174–193. https://doi.org/10.3102/1076998609346970

von Davier, M. (2016). The Rasch model: Chapter 3. In W. van der Linden (Ed.), *Handbook of item response theory, vol. 1* (pp. 31–48). CRC Press.

von Davier, M., Gonzalez, E., & Mislevy, R. (2009). What are plausible values and why are they useful? *IERI Monograph Series*, *2*(1), 9–36.

Wainer, H., Sheehan, K. M., & Wang, X. (2000). Some paths toward making Praxis scores more useful. *Journal of Educational Measurement*, *37*(2), 113–140.

Wainer, H., Vevea, J. L., Camacho, F., Reeve, B. B., Rosa, K., Nelson, L., Swygert, K. A., & Thissen, D. (2001). Augmented scores-"borrowing strength"-to compute scores based on small numbers of items. In D. Thissen & H. Wainer (Eds.), *Test scoring* (pp. 343–387). Lawrence Erlbaum Associates, Inc.

Wang, W., Chen, P., & Cheng, Y. (2004). Improving measurement precision of test batteries using multidimensional item response models. *Psychological Methods*, *9*(1), 116. https://doi.org/10.1037/1082-989X.9.1.116

Wang, X. (2017). *When do subscores add value and what method do we use to estimate subscores?* (Doctoral dissertation). Indiana University, Bloomington.

Wang, X., Svetina, D., & Dai, S. (2019). Exploration of factors affecting the added value of test subscores. *The Journal of Experimental Education*, *87*(2), 179–192. https://doi.org/10.1080/00220973.2017.1409182

Wedman, J., & Lyrén, P. (2015). Methods for examining the psychometric quality of subscores: A review and application. *Practical Assessment, Research, and Evaluation*, *20*(1), 21.

West, M. J. (1999). Stereological methods for estimating the total number of neurons and synapses: Issues of precision and bias. *Trends in Neurosciences*, *22*(2), 51–61. https://doi.org/10.1016/S0166-2236(98)01362-9

Wolf, R. (2014). *Assessing the impact of characteristics of the test, common-items, and examinees on the preservation of equity properties in mixed-format test equating* (Doctoral dissertation). University of Pittsburgh.

Yamamoto, K., & Kulick, E. (2000). Scaling methodology and procedures for the TIMSS mathematics and science scales. In M. O. Martin, K. D. Gregory, & S. E. Stemler (Eds.), *TIMSS 1999 technical report: Description of the methods and procedures used in IEA's repeat of the third international mathematics and science study at the eighth grade* (pp. 237–263). TIMSS & PIRLS International Study Center, Boston College.

Yao, L. (2010). Reporting valid and reliable overall scores and domain scores. *Journal of Educational Measurement*, *47*(3), 339–360.

Yao, L., & Boughton, K. A. (2007). A multidimensional item response modeling approach for improving subscale proficiency estimation and classification. *Applied Psychological Measurement*, *31*(2), 83–105. https://doi.org/10.1177/0146621606291559

Yen, W. M. (1987). A Bayesian/IRT index of objective performance. *Annual meeting of the Psychometric Society, Montreal, Quebec, Canada.*

Appendices

Appendix A

Empirical Subscale Correlations

Table A.1
Empirical Correlations: Mathematics Domains by Country

Country	mat-sci	alg-dat	alg-num	alg-geo	dat-num	dat-geo	geo-num
Singapore	.88	.89	.91	.92	.92	.89	.92
Korea	.82	.90	.91	.92	.91	.89	.91
China	.89	.91	.94	.94	.93	.90	.92
Sweden	.82	.81	.88	.83	.87	.83	.83
Italy	.83	.84	.89	.88	.87	.85	.87
New Zealand	.88	.92	.93	.91	.95	.91	.92
Morocco	.83	.78	.86	.81	.82	.79	.84
South Africa	.87	.84	.89	.84	.89	.86	.88
Saudi Arabia	.80	.71	.84	.74	.80	.71	.75

Note. "mat" = Mathematics; "alg" = Algebra; "dat" = Data and Chance; "geo" = Geometry; "num" = Number.

Table A.2
Empirical Correlations: Science Domains by Country

Country	che-ear	che-bio	che-phy	ear-bio	ear-phy	bio-phy
Singapore	.92	.89	.91	.93	.92	.92
Korea	.85	.90	.91	.88	.87	.90
China	.88	.89	.89	.90	.89	.87
Sweden	.86	.89	.89	.85	.89	.89
Italy	.88	.88	.86	.86	.88	.86
New Zealand	.91	.90	.93	.92	.93	.92
Morocco	.78	.83	.83	.80	.81	.87
South Africa	.89	.92	.87	.90	.89	.88
Saudi Arabia	.83	.87	.85	.84	.80	.83

Note. "che" = Chemistry; "ear" = Earth Science; "bio" = Biology; "phy" = Physics.

Table A.3
Empirical Correlations: Algebra and Science Domains by Country

Country	alg-che	alg-ear	alg-bio	alg-phy
Singapore	.81	.77	.78	.78
Korea	.63	.59	.62	.66
China	.74	.67	.71	.70
Sweden	.66	.61	.64	.65
Italy	.63	.60	.60	.61
New Zealand	.73	.71	.72	.73
Morocco	.54	.47	.54	.54
South Africa	.64	.65	.67	.66
Saudi Arabia	.49	.49	.50	.51

Note. "alg" = Algebra; "che" = Chemistry; "ear" = Earth Science; "bio" = Biology; "phy" = Physics.

Table A.4
Empirical Correlations: Data and Chance and Science Domains by Country

Country	dat-che	dat-ear	dat-bio	dat-phy
Singapore	.77	.75	.76	.77
Korea	.60	.59	.61	.64
China	.71	.67	.70	.70
Sweden	.64	.63	.64	.66
Italy	.63	.61	.62	.62
New Zealand	.74	.74	.73	.76
Morocco	.54	.48	.56	.57
South Africa	.70	.69	.71	.70
Saudi Arabia	.56	.55	.57	.60

Note. "dat" = Data and Chance; "che" = Chemistry; "ear" = Earth Science; "bio" = Biology; "phy" = Physics.

Table A.5
Empirical Correlations: Numbers and Science Domains by Country

Country	num-che	num-ear	num-bio	num-phy
Singapore	.78	.75	.76	.77
Korea	.61	.60	.62	.65
China	.72	.68	.71	.71
Sweden	.66	.65	.66	.68
Italy	.62	.60	.60	.61
New Zealand	.74	.73	.72	.75
Morocco	.54	.49	.56	.55
South Africa	.69	.69	.70	.70
Saudi Arabia	.53	.52	.51	.53

Note. "num" = Number; "che" = Chemistry; "ear" = Earth Science; "bio" = Biology; "phy" = Physics.

Table A.6
Empirical Correlations: Geometry and Science Domains by Country

Country	geo-che	geo-ear	geo-bio	geo-phy
Singapore	.78	.76	.77	.78
Korea	.66	.61	.63	.68
China	.72	.67	.70	.70
Sweden	.62	.60	.61	.63
Italy	.62	.60	.63	.61
New Zealand	.74	.73	.72	.74
Morocco	.52	.45	.53	.54
South Africa	.65	.66	.66	.67
Saudi Arabia	.50	.48	.49	.51

Note. "geo" = Geometry; "che" = Chemistry; "ear" = Earth Science; "bio" = Biology; "phy" = Physics.

Appendix B

Relationships between Background Variables

Table B.1
Chinese Taipei

Variable	Algebra	Data	Geometry	Numbers	BSBG01	BSBG04	BSBG06E	BSBG06H	BSBG06I	BSBG06J	BSBG07A	BSBG07B	BSBG10B	BSBM17A
Algebra	1.00													
Data	.95	1.00												
Geometry	.95	.95	1.00											
Numbers	.95	.95	.95	1.00										
BSBG01	-.09	-.06	-.07	-.05	1.00									
BSBG04	.40	.44	.41	.42	-.29	1.00								
BSBG06E	-.11	-.19	-.14	-.18	.06	-.08	1.00							
BSBG06H	.00	-.02	-.05	.02	.22	-.26	.07	1.00						
BSBG06I	-.06	-.07	-.07	-.04	-.06	-.18	-.01	.27	1.00					
BSBG06J	-.37	-.31	-.41	-.39	.14	-.47	.07	.15	.21	1.00				
BSBG07A	-.18	-.19	-.13	-.24	-.10	.01	.04	-.12	-.08	-.01	1.00			
BSBG07B	-.18	-.22	-.16	-.21	-.07	.02	-.01	.00	-.05	-.04	.44	1.00		
BSBG10B	.48	.37	.51	.40	.14	.08	.06	.06	-.01	-.16	.09	.02	1.00	
BSBM17A	-.35	-.30	-.30	-.30	-.17	-.06	.07	.08	.06	.10	.21	-.02	-.17	1.00

Table B.2
Italy

Variable	Algebra	Data	Geometry	Numbers	BSBG01	BSBG04	BSBG06E	BSBG06H	BSBG06I	BSBG06J	BSBG07A	BSBG07B	BSBG10B	BSBM17A
Algebra	1.00													
Data	.95	1.00												
Geometry	.95	.95	1.00											
Numbers	.95	.95	.95	1.00										
BSBG01	-.08	-.02	-.12	.05	1.00									
BSBG04	.21	.25	.27	.25	-.07	1.00								
BSBG06E	-.10	-.02	-.06	-.12	.05	-.13	1.00							
BSBG06H	-.07	-.07	-.09	-.07	-.07	-.27	.01	1.00						
BSBG06I	.20	.24	.19	.21	-.01	-.11	.02	.15	1.00					
BSBG06J	-.02	.03	.00	-.01	-.04	-.23	-.03	.25	.18	1.00				
BSBG07A	-.13	-.08	-.11	-.08	.04	.08	.09	-.08	-.10	-.28	1.00			
BSBG07B	-.13	-.04	-.11	-.07	-.03	.04	.06	-.09	-.10	-.12	.58	1.00		
BSBG10B	.12	.11	.29	.09	-.17	.18	.03	-.16	-.02	-.04	-.06	-.17	1.00	
BSBM17A	-.34	-.28	-.27	-.31	-.12	-.01	-.11	.00	-.06	.06	.06	.14	.00	1.00

Table B.3
Korea, Republic of

Variable	Algebra	Data	Geometry	Numbers	BSBG01	BSBG05	BSBG04	BSBG06H	BSBG06I	BSBG07B	BSBG15A	BSBG16D	BSBM17B	BSBM20B
Algebra	1.00													
Data	.95	1.00												
Geometry	.95	.95	1.00											
Numbers	.95	.95	.95	1.00										
BSBG01	-.01	.03	.02	.10	1.00									
BSBG05	.35	.32	.35	.35	-.02	1.00								
BSBG04	.10	.13	.12	.14	.00	.17	1.00							
BSBG06H	-.03	-.02	-.04	-.04	.00	-.04	-.19	1.00						
BSBG06I	-.07	-.04	-.08	-.08	-.04	-.20	-.12	.11	1.00					
BSBG07B	.00	-.02	.00	.00	.02	.00	.00	.03	-.03	1.00				
BSBG15A	-.12	-.08	-.11	-.10	-.01	-.08	.00	.00	.05	.01	1.00			
BSBG16D	.01	.00	-.01	-.01	-.06	-.02	-.05	.02	.02	.04	-.05	1.00		
BSBM17B	.22	.14	.19	.14	.03	.03	-.03	.00	.00	.00	-.17	-.01	1.00	
BSBM20B	-.25	-.21	-.25	-.19	-.02	-.09	-.01	-.04	.05	.02	.20	.00	-.31	1.00

Table B.4
Morocco

Variable	Algebra	Data	Geometry	Numbers	BSBG01	BSBG04	BSBG06E	BSBG06H	BSBG06I	BSBG06J	BSBG07A	BSBG07B	BSBG10B	BSBM17A
Algebra	1.00													
Data	.95	1.00												
Geometry	.95	.95	1.00											
Numbers	.95	.95	.95	1.00										
BSBG01	.05	.11	.08	.13	1.00									
BSBG04	.13	.11	.18	.15	.20	1.00								
BSBG06E	-.20	-.14	-.23	-.20	-.15	-.27	1.00							
BSBG06H	.08	.12	.11	.09	-.14	-.14	.12	1.00						
BSBG06I	-.12	-.04	-.11	-.07	-.13	-.20	.24	.05	1.00					
BSBG06J	.05	.19	.03	.08	.12	-.22	.18	.22	.19	1.00				
BSBG07A	.07	.11	.11	.05	.15	.23	-.02	-.15	-.17	-.12	1.00			
BSBG07B	.10	.16	.06	.11	.02	.09	-.08	.02	-.02	-.07	.43	1.00		
BSBG10B	-.12	-.16	-.09	-.12	.06	.09	.04	-.19	.04	.06	.11	.16	1.00	
BSBM17A	-.37	-.19	-.28	-.24	.12	-.03	-.02	-.02	.04	-.04	.11	.00	.11	1.00

Table B.5
New Zealand

Variable	Algebra	Data	Geometry	Numbers	BSBG01	BSBG04	BSBG06E	BSBG06H	BSBG06I	BSBG06J	BSBG07A	BSBG07B	BSBG10B	BSBM17A
Algebra	1.00													
Data	.95	1.00												
Geometry	.95	.95	1.00											
Numbers	.95	.95	.95	1.00										
BSBG01	-.04	-.04	-.01	.05	1.00									
BSBG04	.34	.41	.39	.36	-.07	1.00								
BSBG06E	-.16	-.20	-.17	-.20	.06	-.18	1.00							
BSBG06H	-.19	-.19	-.18	-.14	.17	-.25	.04	1.00						
BSBG06I	-.05	-.03	-.01	-.05	.03	-.10	.05	.06	1.00					
BSBG06J	-.12	-.12	-.11	-.12	.01	-.17	.06	.06	.28	1.00				
BSBG07A	.03	.03	.02	.03	.07	.05	.02	-.07	.03	-.01	1.00			
BSBG07B	.01	-.01	-.01	.00	.03	.02	.04	-.02	.08	-.01	.58	1.00		
BSBG10B	.03	.09	.08	.04	-.02	.15	-.06	-.08	.02	-.09	.03	.01	1.00	
BSBM17A	-.24	-.15	-.19	-.19	-.11	.10	-.05	-.04	-.06	-.03	-.07	-.07	.16	1.00

Table B.6
Saudi Arabia

Variable	Algebra	Data	Geometry	Numbers	BSBG01	BSBG05	BSBG04	BSBG06H	BSBG6I	BSBG07B	BSBG15A	BSBG16D	BSBM17B	BSBM20D
Algebra	1.00													
Data	.95	1.00												
Geometry	.95	.95	1.00											
Numbers	.95	.95	.95	1.00										
BSBG01	-.07	-.08	-.10	.03	1.00									
BSBG05	.16	.25	.16	.21	-.07	1.00								
BSBG04	.09	.15	.08	.15	-.01	.17	1.00							
BSBG06H	-.01	-.05	.01	-.05	.04	-.11	-.13	1.00						
BSBG06I	.11	.10	.07	.07	-.03	-.09	-.10	.27	1.00					
BSBG07B	.11	.16	.11	.15	.02	.11	.15	-.09	-.08	1.00				
BSBG15A	-.02	.01	-.06	-.03	.05	.13	.02	-.04	-.05	.02	1.00			
BSBG16D	.02	.05	.01	.01	-.19	.00	-.03	.04	.07	.04	-.04	1.00		
BSBM17B	.15	.23	.13	.18	-.01	-.01	.06	.04	.00	.01	-.12	.07	1.00	
BSBM20D	-.13	-.11	-.07	-.08	.05	.03	-.02	-.01	-.02	-.03	.19	.00	-.10	1.00

Table B.7
Singapore

Variable	Algebra	Data	Geometry	Numbers	BSBG01	BSBG04	BSBG06E	BSBG06H	BSBG06I	BSBG06J	BSBG07A	BSBG07B	BSBG10B	BSBM17A
Algebra	1.00													
Data	.95	1.00												
Geometry	.95	.95	1.00											
Numbers	.95	.95	.95	1.00										
BSBG01	-.14	-.05	-.08	-.08	1.00									
BSBG04	.26	.29	.30	.26	.01	1.00								
BSBG06E	-.13	-.15	-.11	-.13	.05	-.05	1.00							
BSBG06H	-.13	-.11	-.15	-.09	.10	-.21	.06	1.00						
BSBG06I	-.08	-.08	-.09	-.05	.05	-.15	.07	.25	1.00					
BSBG06J	-.17	-.16	-.16	-.14	.04	-.14	.12	.16	.13	1.00				
BSBG07A	.03	.03	.04	.03	.03	.11	-.05	-.03	-.03	-.10	1.00			
BSBG07B	.07	.06	.08	.07	.03	.10	-.03	-.03	-.06	-.09	.61	1.00		
BSBG10B	-.01	.07	.02	.05	-.01	.08	-.03	-.12	-.02	.01	.04	.07	1.00	
BSBM17A	-.24	-.15	-.21	-.18	.06	-.08	.01	.07	.03	.07	.03	.01	.07	1.00

Table B.8
South Africa

Variable	Algebra	Data	Geometry	Numbers	BSBG01	BSBG04	BSBG06E	BSBG06H	BSBG06I	BSBG06J	BSBG07A	BSBG07B	BSBG10B	BSBM17A
Algebra	1.00													
Data	.95	1.00												
Geometry	.95	.95	1.00											
Numbers	.95	.95	.95	1.00										
BSBG01	-.07	-.05	-.05	-.01	1.00									
BSBG04	.36	.38	.39	.37	.07	1.00								
BSBG06E	-.25	-.36	-.32	-.39	-.15	-.24	1.00							
BSBG06H	-.30	-.30	-.26	-.30	.12	-.15	.22	1.00						
BSBG06I	-.11	-.13	-.12	-.15	.15	-.15	.21	.13	1.00					
BSBG06J	-.33	-.30	-.29	-.33	.08	-.11	.25	.22	.20	1.00				
BSBG07A	.36	.37	.39	.36	-.19	.14	-.09	-.24	-.21	-.31	1.00			
BSBG07B	.31	.34	.34	.36	-.12	.08	-.23	-.24	-.20	-.24	.48	1.00		
BSBG10B	.11	.17	.06	.16	.00	.18	-.10	-.02	-.05	-.12	.06	.09	1.00	
BSBM17A	-.11	.02	.03	.01	.10	.00	-.08	.05	-.01	.05	.02	.17	.05	1.00

Table B.9
Sweden

Variable	Algebra	Data	Geometry	Numbers	BSBG01	BSBG04	BSBG06E	BSBG06H	BSBG06I	BSBG06J	BSBG07A	BSBG07B	BSBG10B	BSBM17A
Algebra	1.00													
Data	.95	1.00												
Geometry	.95	.95	1.00											
Numbers	.95	.95	.95	1.00										
BSBG01	.05	.07	.07	.12	1.00									
BSBG04	.37	.38	.36	.38	-.06	1.00								
BSBG06E	-.05	-.03	-.05	-.06	.02	-.08	1.00							
BSBG06H	-.16	-.14	-.14	-.13	-.12	-.31	.09	1.00						
BSBG06I	-.05	-.05	-.08	-.04	.09	-.19	.09	.20	1.00					
BSBG06J	-.27	-.28	-.27	-.29	.01	-.32	.10	.26	.30	1.00				
BSBG07A	-.03	-.03	.00	-.03	-.03	.05	-.12	-.01	-.23	.00	1.00			
BSBG07B	-.01	-.04	-.02	-.04	.00	.02	-.07	-.05	-.12	-.03	.63	1.00		
BSBG10B	.10	.16	.15	.13	-.05	.23	.02	-.17	-.17	-.27	.03	.11	1.00	
BSBM17A	-.22	-.09	-.12	-.16	-.13	-.05	-.03	.09	-.03	-.04	.09	.02	.25	1.00

Table B.10
Example Block Design for Simulation Study 2

	Blocks													
	b1	b2	b3	b4	b5	b6	b7	b8	b9	b10	b11	b12	b13	b14
Item	1	5	9	13	17	21	25	29	33	37	41	45	49	53
	2	6	10	14	18	22	26	30	34	38	42	46	50	54
	3	7	11	15	19	23	27	31	35	39	43	47	51	55
	4	8	12	16	20	24	28	32	36	40	44	48	52	56
	57	58	59	60	77	81	85	89	93	97	101	105	109	113
	61	65	69	73	78	82	86	90	94	98	102	106	110	114
	62	66	70	74	79	83	87	91	95	99	103	107	111	115
	63	67	71	75	80	84	88	92	96	100	104	108	112	116
	64	68	72	76	117	118	119	120	153	157	161	165	169	173
	121	125	129	133	137	141	145	149	154	158	162	166	170	174
	122	126	130	134	138	142	146	150	155	159	163	167	171	175
	123	127	131	135	139	143	147	151	156	160	164	168	172	176
	124	128	132	136	140	144	148	152	177	178	179	180	229	233
	181	185	189	193	197	201	205	209	213	217	221	225	230	234
	182	186	190	194	198	202	206	210	214	218	222	226	231	235
	183	187	191	195	199	203	207	211	215	219	223	227	232	236
	184	188	192	196	200	204	208	212	216	220	224	228	237	238
	239	240	0	0	0	0	0	0	0	0	0	0	0	0
Total	18	18	17	17	17	17	17	17	17	17	17	17	17	17

Appendix C

Sample Simulation Code

C.1 Sample Code for Study 1's Single Group Simulation

```
# Do not run

options(width = 100)
rm(list = ls())

library(mirt)
library(mvtnorm)
library(lsasim)
library(asbio)

# set working directory
setwd("C:/workingDirectory")

#----------------------------------------------------#
## Define the condition
#----------------------------------------------------#
Study         <- 1 # study 1a - single group
no_factors    <- 3  # 3 or 5    number of factors
items_per_fac <- 5 # 5, 10, or 15 number of items per factor
rr <- cors    <- .45 # correlations, e.g., .45, .75, 95
no_reps       <- 100 # 100 replications
total_items   <- max(items_per_fac) * max(no_factors) # Total number
    of items
n_examinees   <- 6000 # Single groups sample
# kk = a specific replication
# rr = a specific covariance matrix

#----------------------------------------------------#
# Specify all possible models
#----------------------------------------------------#
# (a) uirt = single factor model,
# (b) cuirt = D uncorrelated factor model,
# (c) mirt = D correlated factor model,
#  where D = number of factors

uirt_3_5  <- mirt::mirt.model('F1 = 1 - 15')
cuirt_3_5 <- mirt::mirt.model("
                              F1 =  1 - 5
                              F2 =  6 - 10
                              F3 = 11 - 15
```

```
                              ") # 5 items per domain

mirt_3_5 <-  mirt::mirt.model("
                              F1 =  1 - 5
                              F2 =  6 - 10
                              F3 = 11 - 15
                              COV = F1 * F2, F2 * F3, F1 * F3
                              ") # 5 items per domain

uirt_3_10    <- mirt::mirt.model('F1 = 1 - 30')
cuirt_3_10  <- mirt::mirt.model("
                               F1 =  1 - 10
                               F2 =  11 - 20
                               F3 =  21 - 30
                               ") # 10 items per domain

mirt_3_10 <- mirt::mirt.model("
                              F1 =  1 - 10
                              F2 =  11 - 20
                              F3 =  21 - 30
                              COV = F1 * F2, F2 * F3, F1 * F3
                              ") # 10 items per domain

uirt_3_15    <- mirt::mirt.model('F1 = 1 - 45')
cuirt_3_15  <- mirt::mirt.model("
                               F1 =  1 - 15
                               F2 = 16 - 30
                               F3 = 31 - 45
                               ") # 15 items per domain

mirt_3_15 <- mirt::mirt.model("
                              F1 =  1 - 15
                              F2 = 16 - 30
                              F3 = 31 - 45
                              COV = F1 * F2, F2 * F3, F1 * F3
                              ") # 15 items per domain

uirt_5_5  <- mirt::mirt.model('F1 = 1 - 25')
cuirt_5_5  <- mirt::mirt.model("
                               F1 =  1 - 5
                               F2 =  6 - 10
                               F3 = 11 - 15
                               F4 = 16 - 20
                               F5 = 21 - 25
                               ") # 5 items per domain

mirt_5_5 <- mirt::mirt.model("
                             F1 =  1 - 5
                             F2 =  6 - 10
                             F3 = 11 - 15
                             F4 = 16 - 20
                             F5 = 21 - 25
                             COV = F1 * F2, F1 * F3, F1 * F4, F1 * F5
                                 , F2 * F3, F2 * F4, F2 * F5, F3 * F4
```

```
                                , F3 * F5,  F4 * F5
                              ") # 5 items per domain

uirt_5_10    <- mirt::mirt.model('F1 = 1 - 50')
cuirt_5_10  <- mirt::mirt.model("'
                                F1  =  1 - 10
                                F2 = 11 - 20
                                F3 = 21 - 30
                                F4 = 31 - 40
                                F5 = 41 - 50
                                ") # 10 items per domain
mirt_5_10 <- mirt::mirt.model("
                               F1  =  1 - 10
                               F2 = 11 - 20
                               F3 = 21 - 30
                               F4 = 31 - 40
                               F5 = 41 - 50
                               COV = F1 * F2, F1 * F3, F1 * F4, F1 *
                                   F5, F2 * F3, F2 * F4, F2 * F5, F3 *
                                    F4, F3 * F5,  F4 * F5
                               ") # 10 items per domain

uirt_5_15    <- mirt::mirt.model('F1 = 1 - 75')
cuirt_5_15  <- mirt::mirt.model("
                                F1 =  1 - 15
                                F2 = 16 - 30
                                F3 = 31 - 45
                                F4 = 46 - 60
                                F5 = 61 - 75
                                ") # 15 items per domain
mirt_5_15 <- mirt::mirt.model("
                               F1 =  1 - 15
                               F2 = 16 - 30
                               F3 = 31 - 45
                               F4 = 46 - 60
                               F5 = 61 - 75
                               COV = F1 * F2, F1 * F3, F1 * F4, F1 *
                                   F5, F2 * F3, F2 * F4, F2 * F5, F3 *
                                    F4, F3 * F5,  F4 * F5
                               ") # 15 items per domain

# save all of the models as a list to be used for each condition
mods <- list("uirt_3_5"  = uirt_3_5,
             "cuirt_3_5"  = cuirt_3_5,
             "mirt_3_5"  = mirt_3_5,
             "uirt_3_10" = uirt_3_10,
             "cuirt_3_10" = cuirt_3_10,
             "mirt_3_10" = mirt_3_10,
             "uirt_3_15" = uirt_3_15,
             "cuirt_3_15" = cuirt_3_15,
             "mirt_3_15" = mirt_3_15,
             "uirt_5_5"  = uirt_5_5,
             "cuirt_5_5"  = cuirt_5_5,
             "mirt_5_5"  = mirt_5_5,
```

```
              "uirt_5_10" = uirt_5_10,
              "cuirt_5_10" = cuirt_5_10,
              "mirt_5_10" = mirt_5_10,
              "uirt_5_15" = uirt_5_15,
              "cuirt_5_15" = cuirt_5_15,
              "mirt_5_15" = mirt_5_15)

#-----------------------------------------------------#
## Generate item parameters
#-----------------------------------------------------#
items <- list()
set.seed( round(5653 + Study + no_factors + items_per_fac ) ) # seed
    for generated item parameters
for ( zz in 1: no_factors){
  items[[zz]] <- lsasim::item_gen(n_1pl = items_per_fac, b_bounds = c
      (-2, 2))
} # generate true item difficulty parameters  from a uniform
    distribution (-2, 2)
# The loop allows me to generate the same distribution for all
    domains

#--- combined sub-test items into a single data frame
test_items <- do.call("rbind", items)

#--- re-write items numbers
test_items$item <- 1:nrow(test_items)

#--- asssign items to a domains
test_items$domain <- rep(1:no_factors, each = items_per_fac )

test_items$Study  <- Study # specify the study
test_items$no_factors <- no_factors # specify the number of factors
test_items$items_per_fac <- items_per_fac # specify the number of
    items per factor
test_items$cors <- rr # specify the correlations

# Save a table of the generating item parameters
save(test_items, file = paste0("test_items_", "ss", Study,
"_zz", no_factors, "_rr", rr, "_cc", items_per_fac, "_trueip.Rdata"))

#-----------------------------------------------------#
# Specify the parallelization
#-----------------------------------------------------#
#--- determine replications
replications <- vector("list", no_reps) # when running replication
    1:100
for (rep in 1:no_reps) {
  replications[[rep]] <- rep
} # useful for instances where I conduct my analysis in parallel
    coding

# specify my function to be distributed across all of the specified
    nodes
# the function includes all of the input
```

```
myFunction <- function(X, Study, no_factors, items, test_items,
                       items_per_fac, rr, n_examinees, total_items,
                       mods, verbose = TRUE) {
  library(mirt)
  library(mvtnorm)
  library(lsasim)
  library(asbio)

  # specify the number of replications
  kk <- X # specify the replication

  #-----------------------------------------------------#
  # Generate thetas
  #-----------------------------------------------------#
  #--- generate correlation matrix from factor loadings
  p2_vec   <- rep(sqrt(rr), no_factors)
  p2_X_tp2 <- p2_vec %*% t(p2_vec)
  diag_p2_X_tp2_matrix <- diag(diag(p2_X_tp2))
  i1_matrix <- diag(no_factors)
  u1_matrix <- i1_matrix - diag_p2_X_tp2_matrix
  r1 <- p2_X_tp2 + u1_matrix # results in a correlation matrix
  # direct deerivation from an inverse Wishart

  set.seed((kk + rr) * 100) # set seed for theta, kk changes with
      every rep.
  # generate true examinee theta based on population mean and the
      specified correlation matrix, sigma = r1
  theta_df <- data.frame(mvtnorm::rmvnorm(n_examinees, mean = rep(0,
      no_factors), sigma = r1))
  # specify number of examinees, number of factors and the
      correlation matrix (sigma)
  colnames(theta_df) <- paste0("theta", 1:no_factors) # rename
      columns for saving

  true_theta_df <- data.frame(Study = Study, # specify study
                              no_factors = no_factors, # specify the
                                  number of factors
                              items_per_fac = items_per_fac, #
                                  specify the number of items per
                                  factor
                              cors = rr, # specify the correlations
                              replication = kk, # specify the
                                  replication
                              theta_df) # prepare the true theta for
                                  saving

  save(true_theta_df, file = paste0("True_theta_df_", "ss", Study, "_
      zz", no_factors, "_rr", rr,
                                    "_cc", items_per_fac, "_kk", kk,
                                        ".RData")) # save the true
                                        theta


  #-----------------------------------------------------#
```

```
# Generate item responses
#-----------------------------------------------------#
# create a container for the responses
resp <- data.frame(matrix(NA, nrow= n_examinees, ncol = total_items
    ))
colnames(resp) <- paste0("i.", 1:total_items) # rename the items in
     each column

# generate response by each domain by specifying a loop that takes
    into account the:
# (a) domain specific item parameters
# (b) domain theta from theta_df(true theta)
# (c) booklet and booklet design
for (zz in 1:no_factors) { # zz = factors

  # assign items to block
  block_bk1 <- lsasim::block_design(n_blocks = 1,
                                    item_parameters = items[[zz]] )
                                         # select items in specific
                                         domain

  #assign block to booklet
  book_bk1 <- lsasim::booklet_design(item_block_assignment = block_
      bk1$block_assignment,
                                     book_design = matrix(1))
  #assign booklet to subjects
  book_samp <- lsasim::booklet_sample(n_subj = n_examinees,
                                      book_item_design = book_bk1,
                                      book_prob = NULL)

  # generate item responses
  cog <- lsasim::response_gen(subject = book_samp$subject,
                              item = book_samp$item,
                              theta = theta_df[, zz], # use theta
                                  for specific domain
                              b_par = items[[zz]]$b) # use item
                                  difficulty from specific domain
  # fill in subdomain responses
  resp[, c((zz-1)*items_per_fac+1):(zz*items_per_fac) ] <- cog[, c
      (1:nrow(items[[zz]]) ) ]

}

# save the test responses
save(resp, file = paste0("resp_", "ss", Study, "_zz", no_factors,
     "_rr", rr, "_cc", items_per_fac, "_kk", kk, ".RData"))

#-----------------------------------------------------#
# Fit model 1 - UIRT
#-----------------------------------------------------#
uirt_cc <- mods[[paste0("uirt_", no_factors, "_", items_per_fac)]]
    # model from line 55
# specify the UIRT model
uirt_fit <- mirt::mirt(resp, # specify test responses
```

```
                        model = uirt_cc, # specify the model
                        itemtype = "Rasch", # specify the IRT model
                        method = "SEM", # specify the estimation
                            method
                        draws = 5000, # the number of draws for
                            estimation
                        verbose = FALSE) # not to show the processes
                             and iterations in the background; run
                            discreetly

# obtain fit indices
logLik_uirt <- uirt_fit@Fit$logLik # extract the loglikelihood
AIC_uirt    <- uirt_fit@Fit$AIC # extract the AIC
BIC_uirt    <- uirt_fit@Fit$BIC # extract the BIC
uirt_indices <- data.frame(cbind(logLik_uirt, AIC_uirt, BIC_uirt))
    # dataframe of CUIRT model fit

# obtain item parameters and save standard errors
uirt_coef <- coef(uirt_fit, printSE = TRUE) # extract item
    parameter coefficients
uirt_ip <- data.frame(do.call("rbind", uirt_coef[paste0("i.", 1:
    total_items)])) # place them in a table

# specify which parameters to save and how
uirt_item <- data.frame( id = 1:total_items,
                         d =  uirt_ip$d)    # item difficulty, to
                              be converted in analysis, b = -d
uirt_item$b <-  - uirt_item$d # specify the b parameter conversion

#-----------------------------------------------------#
# Fit model 2 - CIRT
#-----------------------------------------------------#
cirt_cc <- mods[[paste0("cuirt_", no_factors, "_", items_per_fac)]]
# specify the CIRT model
cirt_fit <- mirt::mirt(resp, # specify test responses
                        model = cirt_cc, # specify the model
                        itemtype = "Rasch", # specify the IRT model
                        method = "SEM", # specify the estimation
                            method
                        draws = 5000, # the number of draws for
                            estimation
                        verbose = FALSE) # not to show the processes
                             and iterations in the background; run
                            discreetly

# obtain fit indices
logLik_cirt <- cirt_fit@Fit$logLik # extract the loglikelihood
AIC_cirt    <- cirt_fit@Fit$AIC # extract the AIC
BIC_cirt    <- cirt_fit@Fit$BIC # extract the BIC
cirt_indices <- data.frame(cbind(logLik_cirt, AIC_cirt, BIC_cirt))

# obtain item parameters and save standard errors
cirt_coef <- coef(cirt_fit, printSE = TRUE)
cirt_ip <- data.frame(do.call("rbind", cirt_coef[paste0("i.", 1:
```

```
    total_items)]))

# specify which parameters to save and how
cirt_item <- data.frame( id = 1:total_items,
                         d =  cirt_ip$d) # item difficulty, to be
                             converted in analysis, b = -d
cirt_item$b <-  - cirt_item$d # convert d to b

#-----------------------------------------------------#
# Fit model 3 - MIRT
#-----------------------------------------------------#
mirt_cc <- mods[[paste0("mirt_", no_factors, "_", items_per_fac)]]
# specify the MIRT model
mirt_fit <- mirt::mirt(resp, # specify test responses
                       model = mirt_cc, # specify the model
                       itemtype = "Rasch", # specify the IRT model
                       method = "SEM", # specify the estimation
                           method
                       draws = 5000, # the number of draws for
                           estimation
                       verbose = FALSE) # not to show the processes
                            and iterations in the background; run
                           discreetly

# obtain fit indices
logLik_mirt <- mirt_fit@Fit$logLik # extract the loglikelihood
AIC_mirt    <- mirt_fit@Fit$AIC # extract the AIC
BIC_mirt    <- mirt_fit@Fit$BIC # extract the BIC
mirt_indices <- data.frame(cbind(logLik_mirt, AIC_mirt, BIC_mirt))

# obtain item parameters and save standard errors
mirt_coef <- coef(mirt_fit, printSE = TRUE)
mirt_ip <- data.frame(do.call("rbind", mirt_coef[paste0("i.", 1:
    total_items)]))

# specify which parameters to save and how
mirt_item <- data.frame( id = 1:total_items,
                         d =  mirt_ip$d) # item difficulty standard
                             error
mirt_item$b <-  - mirt_item$d # convert d to b

#-----------------------------------------------------#
# Summarize and save estimated item parameters
#-----------------------------------------------------#
est_ip <- data.frame(test_items,
                     replication = kk,
                     uirt = uirt_item[,-1], # CUIRT item parameters
                         ; remove item ID
                     cirt = cirt_item[,-1], # CIRT item parameters;
                          remove item ID
                     mirt = mirt_item[,-1] ) # MIRT item parameters
                         ; remove item ID
save(est_ip, file = paste0("est_ip_", "ss", Study, "_zz", no_
    factors, "_rr", rr,
```

```
                          "_cc", items_per_fac, "_kk", kk, ".Rdata
                              "))

#-----------------------------------------------------#
# Summarize and save fit indices
#-----------------------------------------------------#
mod_fit <- data.frame(Study = Study,
                      no_factors = no_factors,
                      items_per_fac = items_per_fac,
                      cors = rr,
                      replication = kk,
                      uirt_indices, # CUIRT fit
                      cirt_indices, # CIRT fit
                      mirt_indices) # MIRT fit
save(mod_fit, file = paste0("mod_fit_", "ss", Study, "_zz", no_
    factors, "_rr", rr,
                            "_cc", items_per_fac, "_kk", kk, "_.
                                Rdata"))

#-----------------------------------------------------#
# Scoring
#-----------------------------------------------------#
sv_uirt <- mod2values( uirt_fit ) # obtain matrix of parameters to
    fix
sv_cirt <- mod2values( cirt_fit ) # obtain matrix of parameters to
    fix

# specifiy dimensionality similar to CIRT to score the dimensions (
    assume a testlet structure)
# specify sv_uirt_fx_from_cirt to specify matrix of parameters to
    fix similar to cirt
# this is done to resemble the two step procedure
# in this case, we assume the dimensions are uncorrelated
    unidimensional pieces
sv_uirt_fx_from_cirt <- sv_cirt

# Insert d from uirt to cirt
sv_uirt_fx_from_cirt[ sv_uirt_fx_from_cirt$name =="d", "value" ] <-
     sv_uirt[ sv_uirt$name =="d", "value" ]
# fix item parameters, not to be estimated (set True to False)
sv_uirt_fx_from_cirt[1: ((ncol(cirt_ip)-1) *total_items)  , "est"]
    <- FALSE #estimate covariances
# place item parameters into the model
uirt_fx_from_cirt <- mirt::mirt(resp, # Responses
                                model = cirt_cc, # cirt factor
                                    structure
                                itemtype = "Rasch", # IRT model
                                method = "SEM", # estimation method
                                pars = sv_uirt_fx_from_cirt, #
                                    saved item parametersspecify
                                    item parameters for sa
                                SE = TRUE, # standard error
                                    estimation
                                draws = 5000, # number of
```

```
                                    iterations and draws for
                                    maximum likelihood
                                verbose = FALSE) # operate
                                    discreetly

# estimate uirt EAP scores [Justification in Kim, S., & Lee, W. C.
    (2006); Lu, Thomas, & Zumbo, 2005; Von Davier, Gonzalez, &
    Mislevy, 2009]
uirt_score <- mirt::fscores( uirt_fx_from_cirt, # the UIRT fix of
    the item parameters
                            method = "EAP", # EAP method for
                                scoring
                            full.scores.SE = TRUE, # standard
                                error
                            QMC = TRUE ) # use quasi-Monte Carlo
                                integration, recommended for
                                multidimensional estimation
 # prepare and save UIRT scores
 u_score <- data.frame(Study = Study,
                       no_factors = no_factors,
                       items_per_fac = items_per_fac,
                       cors = rr,
                       replication = kk,
                       theta_df,
                       uirt_score)
save(u_score, file = paste0("est_individual_uirt_score_", "ss",
    Study, "_zz", no_factors, "_rr", rr,
                            "_cc", items_per_fac, "_kk", kk, ".
                                Rdata")) # save individual uirt
                                scores

uirt_score_mean <- apply(uirt_score, 2, mean) # estimate population
     means
uirt_score_sd <- apply(uirt_score, 2, sd) # estimate the standard
    deviations
uirt_prmse <- empirical_rxx(uirt_score) # obtain PRMSE

# estimate cirt EAP scores [Justification in Kim, S., & Lee, W. C.
    (2006); Lu, Thomas, & Zumbo, 2005; Von Davier, Gonzalez, &
    Mislevy, 2009]
cirt_score <- mirt::fscores( cirt_fit, method = "EAP", full.scores.
    SE=TRUE, QMC=TRUE )
c_score <- data.frame(Study = Study,
                       no_factors = no_factors,
                       items_per_fac = items_per_fac,
                       cors = rr,
                       replication = kk,
                       theta_df,
                       cirt_score)
save(c_score, file = paste0("est_individual_cirt_score_", "ss",
    Study, "_zz", no_factors, "_rr", rr,
                            "_cc", items_per_fac, "_kk", kk, ".
                                Rdata")) # save individual cirt
                                scores
```

```
cirt_score_mean <- apply(cirt_score, 2, mean) # estimate population
     means
cirt_score_sd <- apply(cirt_score, 2, sd) # estimate the standard
    deviations
cirt_prmse <- empirical_rxx(cirt_score) # obtain PRMSE

# estimate mirt EAP scores [Justification in Kim, S., & Lee, W. C.
    (2006); Lu, Thomas, & Zumbo, 2005; Von Davier, Gonzalez, &
    Mislevy, 2009]
mirt_score <- mirt::fscores( mirt_fit,  method = "EAP", full.scores
    .SE=TRUE, QMC=TRUE )
m_score <- data.frame(Study = Study,
                      no_factors = no_factors,
                      items_per_fac = items_per_fac,
                      cors = rr,
                      replication = kk,
                      theta_df,
                      mirt_score)
save(m_score, file = paste0("est_individual_mirt_score_", "ss",
    Study, "_zz", no_factors, "_rr", rr,
                            "_cc", items_per_fac, "_kk", kk, ".
                                Rdata")) # save individual mirt
                                scores

mirt_score_mean <- apply(mirt_score, 2, mean) # estimate population
     means
mirt_score_sd <- apply(mirt_score, 2, sd) # estimate the standard
    deviations
mirt_prmse <- empirical_rxx(mirt_score) # obtain PRMSE

#-----------------------------------------------------#
# Summarize and save estimated scores
#-----------------------------------------------------#
est_score <- data.frame(
  Study = Study,
  no_factors = no_factors,
  items_per_fac = items_per_fac,
  cors = rr,
  replications = kk,
  domain = 1: no_factors,
  uirt_score = uirt_score_mean, # CUIRT mean
  uirt_score_sd = uirt_score_sd, # CUIRT mean sd
  cirt_score = cirt_score_mean, # CIRT mean
  cirt_score_sd = cirt_score_sd, # CIRT mean sd
  mirt_score = mirt_score_mean, # MIRT mean
  mirt_score_sd = mirt_score_sd) # MIRT mean sd

save(est_score, file = paste0("est_score_", "ss", Study, "_zz", no_
    factors, "_rr", rr,
                              "_cc", items_per_fac, "_kk", kk, ".
                                  Rdata"))

#-----------------------------------------------------#
```

```
  # Summarize and save PRMSE
  #-----------------------------------------------------#
  prmse <- data.frame(
    Study = Study,
    no_factors = no_factors,
    items_per_fac = items_per_fac,
    cors = rr,
    replications = kk,
    domain = 1: no_factors,
    uirt_prmse = uirt_prmse,
    cirt_prmse = cirt_prmse,
    mirt_prmse = mirt_prmse)

  save(prmse, file = paste0("prmse_", "ss", Study, "_zz", no_factors,
       "_rr", rr,
                            "_cc", items_per_fac, "_kk", kk, ".Rdata"
                                ))

  #-----------------------------------------------------#
  # Specify the output
  #-----------------------------------------------------#
  output <- list(est_ip = est_ip,
                 mirt_cov = mirt_cov, cirt_cov = cirt_cov, uirt_cov =
                      uirt_cov,
                 test_items = test_items, est_score = est_score,
                     prmse = prmse, mod_fit = mod_fit )
  return(output)
  # saves all output to a file
}

#-----------------------------------------------------#
# Send the function to parallel nodes.
# The function contains the simulation code.
# In this case, if the computer has 4 nodes, use 3 to run seperate
    replications.
#-----------------------------------------------------#
library(parallel) # load the package
cl <- makeCluster(3) # specify the number of clusters
# serialize the analysis by sending it to multiple cores
sim_output <- parLapply(cl, replications, myFunction, Study = Study,
    no_factors = no_factors, items = items,
                        test_items = test_items, items_per_fac =
                            items_per_fac, rr = rr, n_examinees = n_
                            examinees,
                        total_items = total_items, mods = mods) #
                            rename the stored results
# rename each solution of the saved output based on its replication
names(sim_output) <- paste0("r", 1:no_reps)
# save the output to the folder
save(sim_output, file = paste0("sim_output_", "ss", Study, "_zz", no_
    factors,
                               "_rr", rr, "_cc", items_per_fac, ".
                                   Rdata"))
# stop cluster
```

```
stopCluster(cl)

# End run
```

C.2 Sample Code for Study 1's Multiple Groups Simulation

```
# Do not run

options(width = 100)
rm(list = ls())

library(mirt)
library(mvtnorm)
library(lsasim)

# set working directory
# setwd("C:/workingDirectory")
setwd("//kant/uv-ils-vit-u1/kondwanm/pc/Desktop/Try")

#-----------------------------------------------------#
## Define the conditions
#-----------------------------------------------------#
Study <- 1 # study 1b - multiple groups
no_factors <- 3 # 3 or 5    number of factors
items_per_fac <- 5 # 5, 10, or 15 number of items per factor
rr <- cors <- .45 # correlations, e.g., .45, .75, 95
no_reps <- 100 # 1:100 replications
total_items <- max(items_per_fac) * max(no_factors)
n_examinees <- 30000 # Multiple groups sample
# kk = a specific replication
# rr = a specific covariance matrix

#-----------------------------------------------------#
# Specify all possible models
#-----------------------------------------------------#
# (a) uirt = single factor model
# (b) cuirt = D uncorrelated factor model
# (c) mirt = D correlated factor model
#     where D = number of factors

uirt_3_5  <- mirt::mirt.model('F1 = 1 - 15')
cuirt_3_5  <- mirt::mirt.model("
                               F1 =  1 - 5
                               F2 =  6 - 10
                               F3 = 11 - 15
                               ") # 5 items per domain
mirt_3_5 <-  mirt::mirt.model("
                              F1 =  1 - 5
                              F2 =  6 - 10
```

```
                              F3 = 11 - 15
                              COV = F1 * F2, F2 * F3, F1 * F3
                              ") # 5 items per domain

uirt_3_10    <- mirt::mirt.model('F1 = 1 - 30')
cuirt_3_10  <- mirt::mirt.model("
                               F1 =  1 - 10
                               F2 =  11 - 20
                               F3 =  21 - 30
                               ") # 10 items per domain
mirt_3_10 <- mirt::mirt.model("
                              F1 =  1 - 10
                              F2 =  11 - 20
                              F3 =  21 - 30
                              COV = F1 * F2, F2 * F3, F1 * F3
                              ") # 10 items per domain

uirt_3_15    <- mirt::mirt.model('F1 = 1 - 45')
cuirt_3_15  <- mirt::mirt.model("
                               F1 =  1 - 15
                               F2 = 16 - 30
                               F3 = 31 - 45
                               ") # 15 items per domain
mirt_3_15 <- mirt::mirt.model("
                              F1 =  1 - 15
                              F2 = 16 - 30
                              F3 = 31 - 45
                              COV = F1 * F2, F2 * F3, F1 * F3
                              ") # 15 items per domain

uirt_5_5  <- mirt::mirt.model('F1 = 1 - 25')
cuirt_5_5  <- mirt::mirt.model("
                               F1 =  1 - 5
                               F2 =  6 - 10
                               F3 = 11 - 15
                               F4 = 16 - 20
                               F5 = 21 - 25
                               ") # 5 items per domain
mirt_5_5 <- mirt::mirt.model("
                             F1 =  1 - 5
                             F2 =  6 - 10
                             F3 = 11 - 15
                             F4 = 16 - 20
                             F5 = 21 - 25
                             COV = F1 * F2, F1 * F3, F1 * F4, F1 * F5
                                 , F2 * F3, F2 * F4, F2 * F5, F3 * F4
                                 , F3 * F5,  F4 * F5
                             ") # 5 items per domain

uirt_5_10    <- mirt::mirt.model('F1 = 1 - 50')
cuirt_5_10  <- mirt::mirt.model("'
                               F1  =  1 - 10
                               F2 = 11 - 20
                               F3 = 21 - 30
```

```
                                F4 = 31 - 40
                                F5 = 41 - 50
                                ") # 10 items per domain
mirt_5_10 <- mirt::mirt.model("
                               F1  =  1 - 10
                               F2 = 11 - 20
                               F3 = 21 - 30
                               F4 = 31 - 40
                               F5 = 41 - 50
                               COV = F1 * F2, F1 * F3, F1 * F4, F1 *
                                   F5, F2 * F3, F2 * F4, F2 * F5, F3 *
                                    F4, F3 * F5,  F4 * F5
                               ") # 10 items per domain

uirt_5_15    <- mirt::mirt.model('F1 = 1 - 75')
cuirt_5_15  <- mirt::mirt.model("
                                F1 =  1 - 15
                                F2 = 16 - 30
                                F3 = 31 - 45
                                F4 = 46 - 60
                                F5 = 61 - 75
                                ") # 15 items per domain
mirt_5_15 <- mirt::mirt.model("
                               F1 =  1 - 15
                               F2 = 16 - 30
                               F3 = 31 - 45
                               F4 = 46 - 60
                               F5 = 61 - 75
                               COV = F1 * F2, F1 * F3, F1 * F4, F1 *
                                   F5, F2 * F3, F2 * F4, F2 * F5, F3 *
                                    F4, F3 * F5,  F4 * F5
                               ") # 15 items per domain

# save all of the models as a list to be used for each condition
mods <- list("uirt_3_5"  = uirt_3_5,  "cuirt_3_5"  = cuirt_3_5,  "
    mirt_3_5"  = mirt_3_5,
             "uirt_3_10" = uirt_3_10, "cuirt_3_10" = cuirt_3_10, "
                 mirt_3_10" = mirt_3_10,
             "uirt_3_15" = uirt_3_15, "cuirt_3_15" = cuirt_3_15, "
                 mirt_3_15" = mirt_3_15,
             "uirt_5_5"  = uirt_5_5,  "cuirt_5_5"  = cuirt_5_5,  "
                 mirt_5_5"  = mirt_5_5,
             "uirt_5_10" = uirt_5_10, "cuirt_5_10" = cuirt_5_10, "
                 mirt_5_10" = mirt_5_10,
             "uirt_5_15" = uirt_5_15, "cuirt_5_15" = cuirt_5_15, "
                 mirt_5_15" = mirt_5_15)

#-----------------------------------------------------#
## Specify 9 countries
#-----------------------------------------------------#
# All 9 countries
ccc <- c( "CNT1", "CNT2", "CNT3",  # top performing
          "CNT4", "CNT5", "CNT6",  # middle performing
          "CNT7", "CNT8", "CNT9" ) # bottom performing
```

```
#-----------------------------------------------------#
## Population proficiency (sss)
#-----------------------------------------------------#
sss <- c( 1.76, 1.12, 0.44, 0.07, 0.04, -0.04, -0.83, -1.33, -1.71 )

#-----------------------------------------------------#
## Generate item parameters
#-----------------------------------------------------#
items <- list()
set.seed( round( 5653  + Study + no_factors + items_per_fac  )) #
    fixed across models
for ( zz in 1: no_factors){
  items[[zz]] <- lsasim::item_gen(n_1pl = items_per_fac, b_bounds = c
      (-2, 2))
} # generate true item difficulty parameters  from a uniform
    distribution (-2, 2)
# The loop allows me to generate the same distribution for all
    domains


#--- combined sub-test items into a single data frame
test_items <- do.call("rbind", items)

#--- re-write items numbers
test_items$item <- 1:nrow(test_items)

#--- assign items to a domains
test_items$domain <- rep(1:no_factors, each = items_per_fac )

test_items$Study  <- Study # specify the study
test_items$no_factors <- no_factors # specify the number of factors
test_items$items_per_fac <- items_per_fac # specify the number of
    items per factor
test_items$cors <- rr # specify the correlations

save(test_items, file = paste0("test_items_", "ss", Study, "_zz",
                               no_factors, "_rr", rr, "_cc", items_
                                   per_fac, "_trueip.Rdata"))

#-----------------------------------------------------#
## Specify replications
#-----------------------------------------------------#
replications <- vector("list", no_reps) # when running replication
    1:100
for (rep in 1:no_reps) {
  replications[[rep]] <- rep
}

#-----------------------------------------------------#
## Specify the function here
#-----------------------------------------------------#
# specify my function to be distributed across all of the specified
    nodes
```

```r
# the function includes all of the input:
# number of factors, items, items per factorm covariance, sample,
    total tems,
# country proficiency, country standard deviation
myFunction <- function(X, Study, no_factors, items, items_per_fac, rr
    , n_examinees,
                       total_items, test_items,  mods, ccc, sss,
                           verbose = TRUE) {
  #-----------------------------------------------------#
  ## Load packages
  #-----------------------------------------------------#
  library(mirt)
  library(mvtnorm)
  library(lsasim)

  #-----------------------------------------------------#
  ## Specify replications
  #-----------------------------------------------------#
  kk <- X

  #-----------------------------------------------------#
  ## Create function
  #-----------------------------------------------------#
  # correlation to covariance function for sigma
  cov_matrix <- function(xx){
    b <- xx %*% t(xx)
    covariance <- b * r1
    return(covariance)
  }

  #-----------------------------------------------------#
  ## Generate thetas
  #-----------------------------------------------------#
  #--- generate correlation matrix from factor loadings
  p2_vec   <- rep(sqrt(rr), no_factors)
  p2_X_tp2 <- p2_vec %*% t(p2_vec)
  diag_p2_X_tp2_matrix <- diag(diag(p2_X_tp2))
  i1_matrix <- diag(no_factors)
  u1_matrix <- i1_matrix - diag_p2_X_tp2_matrix
  r1 <- p2_X_tp2 + u1_matrix # results in a correlation matrix
  # direct deerivation from an inverse Wishart

  #-----------------------------------------------------#
  ## Correlation to covariance
  #-----------------------------------------------------#
  #--- Specify the standard deviations for wach country
  # The same across all domains because SACMEQ does not report
      subscale scores
  # Note: correlation is the standardized correlation
  sd_cnt1 <-  rep( .82, no_factors )
  sd_cnt2 <-  rep( .85, no_factors )
  sd_cnt3 <-  rep( .97, no_factors )
  sd_cnt4 <-  rep( .75, no_factors )
  sd_cnt5 <-  rep( .88, no_factors )
```

```
sd_cnt6 <-  rep( .88, no_factors )
sd_cnt7 <-  rep( .80, no_factors )
sd_cnt8 <-  rep( .87, no_factors )
sd_cnt9 <-  rep( .86, no_factors )

true_country_sd <- data.frame(rbind(sd_cnt1, sd_cnt2, sd_cnt3, sd_
    cnt4,
                                    sd_cnt5, sd_cnt6, sd_cnt7, sd_
                                        cnt8,
                                    sd_cnt9))
colnames(true_country_sd) <- paste0( "theta_sd_", 1:no_factors )
# save the standard deviations
save(true_country_sd, file = paste0("true_country_sd_", "ss",
                                    Study, "_zz", no_factors, "_rr"
                                        , rr,
                                    "_cc", items_per_fac, "_kk", kk
                                        , ".RData"))


#--- covariance matrices
# convert each countries correlation matrix to a covariance matrix
cnt1_r1 <- cov_matrix( sd_cnt1 )
cnt2_r1 <- cov_matrix( sd_cnt2 )
cnt3_r1 <- cov_matrix( sd_cnt3 )
cnt4_r1 <- cov_matrix( sd_cnt4 )
cnt5_r1 <- cov_matrix( sd_cnt5 )
cnt6_r1 <- cov_matrix( sd_cnt6 )
cnt7_r1 <- cov_matrix( sd_cnt7 )
cnt8_r1 <- cov_matrix( sd_cnt8 )
cnt9_r1 <- cov_matrix( sd_cnt9 )

set.seed( round((1234 + Study + no_factors +  items_per_fac + kk)*
    rr) )

# For each country, generate their true theta
cnt1 <- data.frame( mvtnorm::rmvnorm( 3656,
                                      mean = rep(1.76, no_factors),
                                           sigma = cnt1_r1)  )
cnt2 <- data.frame( mvtnorm::rmvnorm( 3491,
                                      mean = rep(1.12, no_factors),
                                           sigma = cnt2_r1)  )
cnt3 <- data.frame( mvtnorm::rmvnorm( 2242,
                                      mean = rep(0.44, no_factors),
                                           sigma = cnt3_r1)  )
cnt4 <- data.frame( mvtnorm::rmvnorm( 5859,
                                      mean = rep(0.07, no_factors),
                                           sigma = cnt4_r1)  )
cnt5 <- data.frame( mvtnorm::rmvnorm( 2253,
                                      mean = rep(0.04, no_factors),
                                           sigma = cnt5_r1)  )
cnt6 <- data.frame( mvtnorm::rmvnorm( 3329,
                                      mean = rep(-0.04, no_factors)
                                          , sigma = cnt6_r1) )
cnt7 <- data.frame( mvtnorm::rmvnorm( 2482,
```

```
                                        mean = rep(-0.83, no_factors)
                                            , sigma = cnt7_r1) )
cnt8 <- data.frame( mvtnorm::rmvnorm( 3701,
                                        mean = rep(-1.33, no_factors)
                                            , sigma = cnt8_r1) )
cnt9 <- data.frame( mvtnorm::rmvnorm( 2987,
                                        mean = rep(-1.71, no_factors)
                                            , sigma = cnt9_r1) )
theta_df <- rbind( cnt1, cnt2, cnt3, cnt4, cnt5, cnt6, cnt7, cnt8,
    cnt9 )
colnames(theta_df) <- paste0( "theta", 1:no_factors )

true_theta_df <- data.frame(Study = Study,
                            no_factors = no_factors,
                            items_per_fac = items_per_fac,
                            cors = rr,
                            replication = kk,
                            theta_df)

# save country true generating theta
save(true_theta_df, file = paste0("True_theta_df_", "ss",
                                  Study, "_zz", no_factors, "_rr",
                                      rr,
                                  "_cc", items_per_fac, "_kk", kk,
                                      ".RData"))

#-----------------------------------------------------#
# Generate item responses
#-----------------------------------------------------#
# Create a container to store the data
resp <- data.frame(matrix(NA, nrow= n_examinees, ncol = total_items
    ))
colnames(resp) <- paste0("i.", 1:total_items)

# generate response by each domain by specifying a loop that takes
    into account the:
# (a) domain specific item parameters
# (b) domain theta from theta_df (true theta)
# (c) booklet and booklet design
for (zz in 1:no_factors) {  #generate response by each domain

  # assign items to block
  block_bk1 <- lsasim::block_design(n_blocks = 1,
                                    item_parameters = items[[zz]] )
                                        #select items in specific
                                       domain

  #assign block to booklet
  book_bk1 <- lsasim::booklet_design(item_block_assignment = block_
      bk1$block_assignment,
                                     book_design = matrix(1))
  #assign booklet to subjects
  book_samp <- lsasim::booklet_sample(n_subj = n_examinees,
                                      book_item_design = book_bk1,
```

```
                                        book_prob = NULL)

  # generate item responses
  cog <- lsasim::response_gen(subject = book_samp$subject,
                              item = book_samp$item,
                              theta = theta_df[, zz], #use theta
                                  for specific domain
                              b_par = items[[zz]]$b) #use item
                                  difficulty from specific domain
  resp[, c((zz-1)*items_per_fac+1):(zz*items_per_fac) ] <- cog[, c
      (1:nrow(items[[zz]]) ) ] #fill in subdomain responses

}
# save the test responses
save(resp, file = paste0("resp_", "ss", Study, "_zz",
                         no_factors, "_rr", rr, "_cc",
                         items_per_fac, "_kk", kk, ".RData"))

#-----------------------------------------------------#
# Create country data sets
#-----------------------------------------------------#
# Specify each country as a dataset
dat1 <- resp[ 1    : 3656,  ]
dat2 <- resp[ 3657 : 7147,  ]
dat3 <- resp[ 7148 : 9389,  ]
dat4 <- resp[ 9390 : 15248, ]
dat5 <- resp[ 15249: 17501, ]
dat6 <- resp[ 17502: 20830, ]
dat7 <- resp[ 20831: 23312, ]
dat8 <- resp[ 23313: 27013, ]
dat9 <- resp[ 27014: 30000, ]

#-----------------------------------------------------#
# Fit model 1 - UIRT
#-----------------------------------------------------#
uirt_cc <- mods[[paste0("uirt_", no_factors, "_", items_per_fac)]]
# specify the UIRT model
uirt_fit <- mirt::mirt(resp, # specify test responses
                       model = uirt_cc, # specify the model
                       itemtype = "Rasch", # specify the IRT model
                       method = "SEM", # specify the estimation
                           method
                       draws = 5000, # the number of draws for
                           estimation
                       verbose = FALSE) # not to show the processes
                            and iterations in the background; run
                           discreetly

# obtain fit indices
logLik_uirt <- uirt_fit@Fit$logLik # extract the loglikelihood
AIC_uirt    <- uirt_fit@Fit$AIC # extract the AIC
BIC_uirt    <- uirt_fit@Fit$BIC # extract the BIC
uirt_indices <- data.frame(cbind(logLik_uirt, AIC_uirt, BIC_uirt))
    # dataframe of CUIRT model fit
```

```
# obtain item parameters and save standard errors
uirt_coef <- coef(uirt_fit, printSE = TRUE) # extract item
    parameter coefficients
uirt_ip <- data.frame(do.call("rbind", uirt_coef[paste0("i.", 1:
    total_items)])) # place them in a table

# specify which parameters to save and how
uirt_item <- data.frame( id = 1:total_items,
                         d =  uirt_ip$d) # item difficulty, to be
                             converted in analysis, b = -d
uirt_item$b <-  - uirt_item$d # specify the b parameter conversion

#-----------------------------------------------------#
# Fit model 2 - CIRT
#-----------------------------------------------------#
cirt_cc <- mods[[paste0("cuirt_", no_factors, "_", items_per_fac)]]
# specify the CIRT model
cirt_fit <- mirt::mirt(resp, # specify test responses
                       model = cirt_cc, # specify the model
                       itemtype = "Rasch", # specify the IRT model
                       method = "SEM", # specify the estimation
                           method
                       draws = 5000, # the number of draws for
                           estimation
                       verbose = FALSE) # not to show the processes
                            and iterations in the background; run
                           discreetly

# obtain fit indices
logLik_cirt <- cirt_fit@Fit$logLik # extract the loglikelihood
AIC_cirt    <- cirt_fit@Fit$AIC # extract the AIC
BIC_cirt    <- cirt_fit@Fit$BIC # extract the BIC
cirt_indices <- data.frame(cbind(logLik_cirt, AIC_cirt, BIC_cirt))

# obtain item parameters and save standard errors
cirt_coef <- coef(cirt_fit, printSE = TRUE)
cirt_ip <- data.frame(do.call("rbind", cirt_coef[paste0("i.", 1:
    total_items)]))

# specify which parameters to save and how
cirt_item <- data.frame( id = 1:total_items,
                         d =  cirt_ip$d) # item difficulty, to be
                             converted in analysis, b = -d
cirt_item$b <-  - cirt_item$d # convert d to b

#-----------------------------------------------------#
# Fit model 3 - MIRT
#-----------------------------------------------------#
mirt_cc <- mods[[paste0("mirt_", no_factors, "_", items_per_fac)]]
# specify the MIRT model
mirt_fit <- mirt::mirt(resp, # specify test responses
                       model = mirt_cc, # specify the model
                       itemtype = "Rasch", # specify the IRT model
```

```
                        method = "SEM", # specify the estimation
                            method
                        draws = 5000, # the number of draws for
                            estimation
                        verbose = FALSE) # not to show the processes
                             and iterations in the background; run
                            discreetly

# obtain fit indices
logLik_mirt <- mirt_fit@Fit$logLik # extract the loglikelihood
AIC_mirt    <- mirt_fit@Fit$AIC # extract the AIC
BIC_mirt    <- mirt_fit@Fit$BIC # extract the BIC
mirt_indices <- data.frame(cbind(logLik_mirt, AIC_mirt, BIC_mirt))

# obtain item parameters and save standard errors
mirt_coef <- coef(mirt_fit, printSE = TRUE)
mirt_ip <- data.frame(do.call("rbind", mirt_coef[paste0("i.", 1:
    total_items)]))

# specify which parameters to save and how
mirt_item <- data.frame( id = 1:total_items,
                         d =  mirt_ip$d) # item difficulty standard
                              error
mirt_item$b <-  - mirt_item$d # convert d to b

#-----------------------------------------------------#
# Rescale item parameters using the mean-mean method in order to
    put the items
# on the same scale in scoring.
#-----------------------------------------------------#
# rescale estimated item parameters using mean/sigma method
u <- mean(  1 )/ mean( 1 )
v <- mean(  test_items$b ) - u*mean( uirt_item$b )
uirt_item$rs_b <- u*uirt_item$b + v #rescaled item difficulty

u <- mean(  1 )/ mean( 1 )
v <- mean(  test_items$b ) - u*mean( cirt_item$b )
cirt_item$rs_b <- u*cirt_item$b + v # rescaled item difficulty

u <- mean(  1 )/ mean( 1 )
v <- mean(  test_items$b ) - u*mean( mirt_item$b )
mirt_item$rs_b <- u*mirt_item$b + v #rescaled item difficulty

#-----------------------------------------------------#
# Summarize and save estimated item parameters
#-----------------------------------------------------#
est_ip <- data.frame(test_items,
                     replication = kk,
                     uirt = uirt_item[,-1], # CUIRT item parameters
                     cirt = cirt_item[,-1], # CIRT item parameters
                     mirt = mirt_item[,-1] ) # MIRT item parameters

save(est_ip, file = paste0("est_ip_", "ss", Study, "_zz", no_
    factors, "_rr", rr,
```

```
                          "_cc", items_per_fac, "_kk", kk, ".Rdata
                              "))

#-----------------------------------------------------#
# Save fit indices
#-----------------------------------------------------#
all_mod_fit <- data.frame(Study = Study,
                          no_factors = no_factors,
                          items_per_fac = items_per_fac,
                          cors = rr,
                          replications = kk,
                          uirt_indices, # CUIRT fit
                          cirt_indices, # CIRT fit
                          mirt_indices) # MIRT fit
save(all_mod_fit, file = paste0("all_mod_fit_", "ss", Study, "_zz",
                                no_factors, "_rr", rr, "_cc", items
                                    _per_fac, "_kk", kk, "_.Rdata")
                                    )

#-----------------------------------------------------#
# Rescale item parameters using the mean-mean method in order to
    put the items
# on the same scale in scoring.
#-----------------------------------------------------#
## Insert d from uirt to cirt
sv_uirt <- mod2values( uirt_fit ) # obtain matrix of parameters to
    fix
sv_cirt <- mod2values( cirt_fit ) # obtain matrix of parameters to
    fix

# specifiy dimensionality similar to CIRT to score the dimensions (
    assume a testlet structure)
# specify sv_uirt_fx_from_cirt to specify matrix of parameters to
    fix similar to cirt
# this is done to resemble the two step procedure
# in this case, we assume the dimensions are uncorrelated
    unidimensional pieces
sv_uirt_fx_from_cirt <- sv_cirt

# Insert d from uirt to cirt
sv_uirt_fx_from_cirt[ sv_uirt_fx_from_cirt$name =="d", "value" ] <-
     sv_uirt[ sv_uirt$name =="d", "value" ]
# fix item parameters, not to be estimated (set True to False)
sv_uirt_fx_from_cirt[1: ((ncol(cirt_ip)-1) *total_items)  , "est"]
    <- FALSE #estimate covariances
# place item parameters into the model

#------- Country specific scores
# (a) fit model to country
cnt1_uirt <- mirt::mirt(dat1, # data for country 1
                        model = cirt_cc, # model
                        itemtype = "Rasch", # IRT estimation
                        method = "SEM", # estimation method
                        pars = sv_uirt_fx_from_cirt, # fix item
```

```
                        parameters
                    removeEmptyRows = TRUE, # remove all rows
                        with no responses
                    draws = 5000, # number of IRT draws
                    verbose = FALSE) # allow the process to tun
                         discreetly

# CNT 1 model fit
CNT_1_logLik_uirt  <- cnt1_uirt@Fit$logLik # extract logLikelihood
CNT_1_AIC_uirt     <- cnt1_uirt@Fit$AIC # extract AIC
CNT_1_BIC_uirt     <- cnt1_uirt@Fit$BIC # extract BIC
# Place fit indices in a table
CNT_1_uirt_indices <- data.frame(cbind(CNT_1_logLik_uirt, CNT_1_AIC
    _uirt, CNT_1_BIC_uirt))


cnt2_uirt <- mirt::mirt(dat2, # data for country 2
                    model = cirt_cc, # model
                    itemtype = "Rasch", # IRT estimation
                    method = "SEM", # estimation method
                    pars = sv_uirt_fx_from_cirt, # fix item
                        parameters
                    removeEmptyRows = TRUE, # remove all rows
                        with no responses
                    draws = 5000, # number of IRT draws
                    verbose = FALSE) # allow the process to tun
                         discreetly
# CNT 2 model fit
CNT_2_logLik_uirt  <- cnt2_uirt@Fit$logLik # extract logLikelihood
CNT_2_AIC_uirt     <- cnt2_uirt@Fit$AIC # extract AIC
CNT_2_BIC_uirt     <- cnt2_uirt@Fit$BIC # extract BIC
CNT_2_uirt_indices <- data.frame(cbind(CNT_2_logLik_uirt, CNT_2_AIC
    _uirt, CNT_2_BIC_uirt))

cnt3_uirt <- mirt::mirt(dat3, # data for country 3
                    model = cirt_cc, # model
                    itemtype = "Rasch", # IRT estimation
                    method = "SEM", # estimation method
                    pars = sv_uirt_fx_from_cirt, # fix item
                        parameters
                    removeEmptyRows = TRUE, # remove all rows
                        with no responses
                    draws = 5000, # number of IRT draws
                    verbose = FALSE) # allow the process to tun
                         discreetly
# CNT 3 model fit
CNT_3_logLik_uirt <- cnt3_uirt@Fit$logLik   # extract logLikelihood
CNT_3_AIC_uirt    <- cnt3_uirt@Fit$AIC # extract AIC
CNT_3_BIC_uirt    <- cnt3_uirt@Fit$BIC # extract BIC
CNT_3_uirt_indices <- data.frame(cbind(CNT_3_logLik_uirt, CNT_3_AIC
    _uirt, CNT_3_BIC_uirt))

cnt4_uirt <- mirt::mirt(dat4, # data for country 4
                    model = cirt_cc, # model
```

```
                        itemtype = "Rasch", # IRT estimation
                        method = "SEM", # estimation method
                        pars = sv_uirt_fx_from_cirt, # fix item
                            parameters
                        removeEmptyRows = TRUE, # remove all rows
                            with no responses
                        draws = 5000, # number of IRT draws
                        verbose = FALSE) # allow the process to tun
                             discreetly
# CNT 4 model fit
CNT_4_logLik_uirt <- cnt4_uirt@Fit$logLik  # extract logLikelihood
CNT_4_AIC_uirt    <- cnt4_uirt@Fit$AIC  # extract AIC
CNT_4_BIC_uirt    <- cnt4_uirt@Fit$BIC  # extract BIC
CNT_4_uirt_indices <- data.frame(cbind(CNT_4_logLik_uirt, CNT_4_AIC
    _uirt, CNT_4_BIC_uirt))

cnt5_uirt <- mirt::mirt(dat5, # data for country 5
                        model = cirt_cc, # model
                        itemtype = "Rasch", # IRT estimation
                        method = "SEM", # estimation method
                        pars = sv_uirt_fx_from_cirt, # fix item
                            parameters
                        removeEmptyRows = TRUE, # remove all rows
                            with no responses
                        draws = 5000, # number of IRT draws
                        verbose = FALSE) # allow the process to tun
                             discreetly
# CNT 5 model fit
CNT_5_logLik_uirt <- cnt5_uirt@Fit$logLik  # extract logLikelihood
CNT_5_AIC_uirt    <- cnt5_uirt@Fit$AIC  # extract AIC
CNT_5_BIC_uirt    <- cnt5_uirt@Fit$BIC  # extract BIC
CNT_5_uirt_indices <- data.frame(cbind(CNT_5_logLik_uirt, CNT_5_AIC
    _uirt, CNT_5_BIC_uirt))

cnt6_uirt <- mirt::mirt(dat6, # data for country 6
                        model = cirt_cc, # model
                        itemtype = "Rasch", # IRT estimation
                        method = "SEM", # estimation method
                        pars = sv_uirt_fx_from_cirt, # fix item
                            parameters
                        removeEmptyRows = TRUE, # remove all rows
                            with no responses
                        draws = 5000, # number of IRT draws
                        verbose = FALSE) # allow the process to tun
                             discreetly
# CNT 6 model fit
CNT_6_logLik_uirt <- cnt6_uirt@Fit$logLik  # extract logLikelihood
CNT_6_AIC_uirt    <- cnt6_uirt@Fit$AIC  # extract AIC
CNT_6_BIC_uirt    <- cnt6_uirt@Fit$BIC  # extract BIC
CNT_6_uirt_indices <- data.frame(cbind(CNT_6_logLik_uirt, CNT_6_AIC
    _uirt, CNT_6_BIC_uirt))

cnt7_uirt <- mirt::mirt(dat7, # data for country 7
                        model = cirt_cc, # model
```

```
                        itemtype = "Rasch", # IRT estimation
                        method = "SEM", # estimation method
                        pars = sv_uirt_fx_from_cirt, # fix item
                            parameters
                        removeEmptyRows = TRUE, # remove all rows
                            with no responses
                        draws = 5000, # number of IRT draws
                        verbose = FALSE) # allow the process to tun
                             discreetly

# CNT 7 model fit
CNT_7_logLik_uirt <- cnt7_uirt@Fit$logLik  # extract logLikelihood
CNT_7_AIC_uirt    <- cnt7_uirt@Fit$AIC  # extract AIC
CNT_7_BIC_uirt    <- cnt7_uirt@Fit$BIC  # extract BIC
CNT_7_uirt_indices <- data.frame(cbind(CNT_7_logLik_uirt, CNT_7_AIC
    _uirt, CNT_7_BIC_uirt))

cnt8_uirt <- mirt::mirt(dat8, # data for country 8
                        model = cirt_cc, # model
                        itemtype = "Rasch", # IRT estimation
                        method = "SEM", # estimation method
                        pars = sv_uirt_fx_from_cirt, # fix item
                            parameters
                        removeEmptyRows = TRUE, # remove all rows
                            with no responses
                        draws = 5000, # number of IRT draws
                        verbose = FALSE) # allow the process to tun
                             discreetly
# CNT 8 model fit
CNT_8_logLik_uirt <- cnt8_uirt@Fit$logLik # extract logLikelihood
CNT_8_AIC_uirt    <- cnt8_uirt@Fit$AIC # extract AIC
CNT_8_BIC_uirt    <- cnt8_uirt@Fit$BIC # extract BIC
CNT_8_uirt_indices <- data.frame(cbind(CNT_8_logLik_uirt, CNT_8_AIC
    _uirt, CNT_8_BIC_uirt))

cnt9_uirt <- mirt::mirt(dat9, # data for country 9
                        model = cirt_cc, # model
                        itemtype = "Rasch", # IRT estimation
                        method = "SEM", # estimation method
                        pars = sv_uirt_fx_from_cirt, # fix item
                            parameters
                        removeEmptyRows = TRUE, # remove all rows
                            with no responses
                        draws = 5000, # number of IRT draws
                        verbose = FALSE) # allow the process to tun
                             discreetly
# CNT 9 model fit
CNT_9_logLik_uirt <- cnt9_uirt@Fit$logLik # extract logLikelihood
CNT_9_AIC_uirt    <- cnt9_uirt@Fit$AIC  # extract AIC
CNT_9_BIC_uirt    <- cnt9_uirt@Fit$BIC  # extract BIC
CNT_9_uirt_indices <- data.frame(cbind(CNT_9_logLik_uirt, CNT_9_AIC
    _uirt, CNT_9_BIC_uirt))
```

```
# (b) score
#--- For each country
# (i) estimate factor scores
# (ii) calculate the standard deviation
# (iii) estimate the country mean
# (iv ) calculate the PRMSE
cnt1_uirt_score <- mirt::fscores( cnt1_uirt,  method = "EAP",
                                  full.scores.SE=TRUE, QMC=TRUE )
cnt1_uirt_score_sd <- apply(cnt1_uirt_score, 2, sd) # estimate the
    standard deviations
cnt1_uirt_score_mean <- apply(cnt1_uirt_score, 2, mean)
cnt1_uirt_prmse <- empirical_rxx(cnt1_uirt_score)

cnt2_uirt_score <- mirt::fscores( cnt2_uirt,  method = "EAP",
                                  full.scores.SE=TRUE, QMC=TRUE )
cnt2_uirt_score_sd <- apply(cnt2_uirt_score, 2, sd) # estimate the
    standard deviations
cnt2_uirt_score_mean <- apply(cnt2_uirt_score, 2, mean)
cnt2_uirt_prmse <- empirical_rxx(cnt2_uirt_score)

cnt3_uirt_score <- mirt::fscores( cnt3_uirt,  method = "EAP",
                                  full.scores.SE=TRUE, QMC=TRUE )
cnt3_uirt_score_sd <- apply(cnt3_uirt_score, 2, sd) # estimate the
    standard deviations
cnt3_uirt_score_mean <- apply(cnt3_uirt_score, 2, mean)
cnt3_uirt_prmse <- empirical_rxx(cnt3_uirt_score)

cnt4_uirt_score <- mirt::fscores( cnt4_uirt,  method = "EAP",
                                  full.scores.SE=TRUE, QMC=TRUE )
cnt4_uirt_score_sd <- apply(cnt4_uirt_score, 2, sd) # estimate the
    standard deviations
cnt4_uirt_score_mean <- apply(cnt4_uirt_score, 2, mean)
cnt4_uirt_prmse <- empirical_rxx(cnt4_uirt_score)

cnt5_uirt_score <- mirt::fscores( cnt5_uirt,  method = "EAP",
                                  full.scores.SE=TRUE, QMC=TRUE )
cnt5_uirt_score_sd <- apply(cnt5_uirt_score, 2, sd) # estimate the
    standard deviations
cnt5_uirt_score_mean <- apply(cnt5_uirt_score, 2, mean)
cnt5_uirt_prmse <- empirical_rxx(cnt5_uirt_score)

cnt6_uirt_score <- mirt::fscores( cnt6_uirt,  method = "EAP",
                                  full.scores.SE=TRUE, QMC=TRUE )
cnt6_uirt_score_sd <- apply(cnt6_uirt_score, 2, sd) # estimate the
    standard deviations
cnt6_uirt_score_mean <- apply(cnt6_uirt_score, 2, mean)
cnt6_uirt_prmse <- empirical_rxx(cnt6_uirt_score)

cnt7_uirt_score <- mirt::fscores( cnt7_uirt,  method = "EAP",
                                  full.scores.SE=TRUE, QMC=TRUE )
cnt7_uirt_score_sd <- apply(cnt7_uirt_score, 2, sd) # estimate the
    standard deviations
cnt7_uirt_score_mean <- apply(cnt7_uirt_score, 2, mean)
cnt7_uirt_prmse <- empirical_rxx(cnt7_uirt_score)
```

```
cnt8_uirt_score <- mirt::fscores( cnt8_uirt,  method = "EAP",
                                  full.scores.SE=TRUE, QMC=TRUE )
cnt8_uirt_score_sd <- apply(cnt8_uirt_score, 2, sd) # estimate the
    standard deviations
cnt8_uirt_score_mean <- apply(cnt8_uirt_score, 2, mean)
cnt8_uirt_prmse <- empirical_rxx(cnt8_uirt_score)

cnt9_uirt_score <- mirt::fscores( cnt9_uirt,  method = "EAP",
                                  full.scores.SE=TRUE, QMC=TRUE )
cnt9_uirt_score_sd <- apply(cnt9_uirt_score, 2, sd) # estimate the
    standard deviations
cnt9_uirt_score_mean <- apply(cnt9_uirt_score, 2, mean)
cnt9_uirt_prmse <- empirical_rxx(cnt9_uirt_score)

#--- UIRT individual scores
# Prepare data set of individual scores
uirt_cnt_individual_score <- data.frame(rbind(cnt1_uirt_score,
                                              cnt2_uirt_score,
                                              cnt3_uirt_score,
                                              cnt4_uirt_score,
                                              cnt5_uirt_score,
                                              cnt6_uirt_score,
                                              cnt7_uirt_score,
                                              cnt8_uirt_score,
                                              cnt9_uirt_score))

est_uirt_cnt_individual_score <- data.frame(Study = Study,
                                            no_factors = no_factors
                                                ,
                                            items_per_fac = items_
                                                per_fac,
                                            cors = rr,
                                            replication = kk,
                                            theta_df,
                                            uirt_cnt_individual_
                                                score)
# Save individual scores
save(est_uirt_cnt_individual_score, file = paste0("est_uirt_cnt_
    individual_score_",
                                                  "ss", Study, "_zz
                                                      ",
                                                  no_factors, "_rr"
                                                      ,
                                                  rr, "_cc", items_
                                                      per_fac,
                                                  "_kk", kk, ".
                                                      RData"))

#--- Score standard deviations
# Prepare data set of individual score standard deviation
est_uirt_individual_country_sd <- data.frame(rbind(cnt1_uirt_score_
    sd,
                                                   cnt2_uirt_score_
```

```
                                                    sd,
                                                cnt3_uirt_score_
                                                    sd,
                                                cnt4_uirt_score_
                                                    sd,
                                                cnt5_uirt_score_
                                                    sd,
                                                cnt6_uirt_score_
                                                    sd,
                                                cnt7_uirt_score_
                                                    sd,
                                                cnt8_uirt_score_
                                                    sd,
                                                cnt9_uirt_score_
                                                    sd))

uirt_individual_country_sd <- data.frame(Study = Study,
                                         CNT = ccc,
                                         true_country_sd = true_
                                             country_sd,
                                         no_factors = no_factors,
                                         items_per_fac = items_per_
                                             fac,
                                         cors = rr,
                                         replication = kk,
                                         est_uirt_individual_
                                             country_sd)
# Save individual score standard deviation
save(uirt_individual_country_sd, file = paste0("uirt_individual_
    country_sd_", "ss", Study, "_zz",
                                                no_factors, "_rr",
                                                    rr, "_cc", items
                                                    _per_fac, "_kk",
                                                     kk, ".RData"))


#--- Place uirt scores into a data frame
uirt_score_by_country <- data.frame(rbind( cnt1_uirt_score_mean,
                                           cnt2_uirt_score_mean,
                                           cnt3_uirt_score_mean,
                                           cnt4_uirt_score_mean,
                                           cnt5_uirt_score_mean,
                                           cnt6_uirt_score_mean,
                                           cnt7_uirt_score_mean,
                                           cnt8_uirt_score_mean,
                                           cnt9_uirt_score_mean ))

colnames(uirt_score_by_country)[1:no_factors] <- paste0( "uirt_",
    1:no_factors )
colnames(uirt_score_by_country)[(no_factors+1):(2*no_factors)] <-
    paste0( "uirt_se_", 1:no_factors )
rownames(uirt_score_by_country) <- NULL

#--- Place UIRT prmse into a data frame
```

```
uirt_PRMSE_by_country <- data.frame(rbind(cnt1_uirt_prmse, cnt2_
    uirt_prmse, cnt3_uirt_prmse,
                                          cnt4_uirt_prmse, cnt5_
                                              uirt_prmse, cnt6_uirt
                                              _prmse,
                                          cnt7_uirt_prmse, cnt8_
                                              uirt_prmse, cnt9_uirt
                                              _prmse ))
colnames(uirt_PRMSE_by_country)[1:no_factors] <- paste0( "uirt_
    prmse_", 1:no_factors )
rownames(uirt_PRMSE_by_country) <- NULL

# save uirt fit indices
uirt_fit_by_country <- data.frame(Study = Study,
                                  no_factors = no_factors,
                                  items_per_fac = items_per_fac,
                                  cors = rr,
                                  replication = kk,
                                  CNT_1_uirt_indices, CNT_2_uirt_
                                      indices, CNT_3_uirt_indices,
                                  CNT_4_uirt_indices, CNT_5_uirt_
                                      indices, CNT_6_uirt_indices,
                                  CNT_7_uirt_indices, CNT_8_uirt_
                                      indices, CNT_9_uirt_indices)
save(uirt_fit_by_country, file = paste0("uirt_fit_by_country_", "ss
    ", Study, "_zz", no_factors, "_rr",
                                        rr, "_cc", items_per_fac, "
                                            _kk", kk, "_.Rdata"))

#-------------------------------------------------------#
# Rescale item parameters using the mean-mean method in order to
    put the items
# on the same scale in scoring.
#-------------------------------------------------------#
## Insert d from uirt to cirt
sv_cirt <- mod2values(cirt_fit)
sv_cirt_score <- sv_cirt
# use rescaled_difficulty and change from b to d
sv_cirt_score[sv_cirt_score$name=="d", "value"] <- ( cirt_item$rs_b
    ) * -1
sv_cirt_score[1: ((ncol(cirt_ip)-1) *total_items)  , "est"] <-
    FALSE # do not estimate covariances

#------- Country specific scores
# (a) fit model to country
cnt1_cirt <- mirt::mirt(dat1, # data for country 1
                        model = cirt_cc, # model
                        itemtype = "Rasch", # IRT estimation
                        method = "SEM", # estimation method
                        pars = sv_cirt_score, # fix item parameters
                        removeEmptyRows = TRUE, # remove all rows
                            with no responses
                        draws = 5000, # number of IRT draws
                        verbose = FALSE) # allow the process to tun
```

```
                          discreetly

# CNT 1 model fit
CNT_1_logLik_cirt  <- cnt1_cirt@Fit$logLik # extract logLikelihood
CNT_1_AIC_cirt     <- cnt1_cirt@Fit$AIC # extract AIC
CNT_1_BIC_cirt     <- cnt1_cirt@Fit$BIC # extract BIC
# Place fit indices in a table
CNT_1_cirt_indices <- data.frame(cbind(CNT_1_logLik_cirt, CNT_1_AIC
    _cirt, CNT_1_BIC_cirt))

cnt2_cirt <- mirt::mirt(dat2, # data for country 2
                        model = cirt_cc, # model
                        itemtype = "Rasch", # IRT estimation
                        method = "SEM", # estimation method
                        pars = sv_cirt_score, # fix item parameters
                        removeEmptyRows = TRUE, # remove all rows
                            with no responses
                        draws = 5000, # number of IRT draws
                        verbose = FALSE) # allow the process to tun
                             discreetly
# CNT 2 model fit
CNT_2_logLik_cirt  <- cnt2_cirt@Fit$logLik # extract logLikelihood
CNT_2_AIC_cirt     <- cnt2_cirt@Fit$AIC # extract AIC
CNT_2_BIC_cirt     <- cnt2_cirt@Fit$BIC # extract BIC
CNT_2_cirt_indices <- data.frame(cbind(CNT_2_logLik_cirt, CNT_2_AIC
    _cirt, CNT_2_BIC_cirt))

cnt3_cirt <- mirt::mirt(dat3, # data for country 3
                        model = cirt_cc, # model
                        itemtype = "Rasch", # IRT estimation
                        method = "SEM", # estimation method
                        pars = sv_cirt_score, # fix item parameters
                        removeEmptyRows = TRUE, # remove all rows
                            with no responses
                        draws = 5000, # number of IRT draws
                        verbose = FALSE) # allow the process to tun
                             discreetly
# CNT 3 model fit
CNT_3_logLik_cirt <- cnt3_cirt@Fit$logLik   # extract logLikelihood
CNT_3_AIC_cirt    <- cnt3_cirt@Fit$AIC # extract AIC
CNT_3_BIC_cirt    <- cnt3_cirt@Fit$BIC # extract BIC
CNT_3_cirt_indices <- data.frame(cbind(CNT_3_logLik_cirt, CNT_3_AIC
    _cirt, CNT_3_BIC_cirt))

cnt4_cirt <- mirt::mirt(dat4, # data for country 4
                        model = cirt_cc, # model
                        itemtype = "Rasch", # IRT estimation
                        method = "SEM", # estimation method
                        pars = sv_cirt_score, # fix item parameters
                        removeEmptyRows = TRUE, # remove all rows
                            with no responses
                        draws = 5000, # number of IRT draws
                        verbose = FALSE) # allow the process to tun
                             discreetly
```

```
# CNT 4 model fit
CNT_4_logLik_cirt <- cnt4_cirt@Fit$logLik  # extract logLikelihood
CNT_4_AIC_cirt    <- cnt4_cirt@Fit$AIC  # extract AIC
CNT_4_BIC_cirt    <- cnt4_cirt@Fit$BIC  # extract BIC
CNT_4_cirt_indices <- data.frame(cbind(CNT_4_logLik_cirt, CNT_4_AIC
    _cirt, CNT_4_BIC_cirt))

cnt5_cirt <- mirt::mirt(dat5, # data for country 5
                        model = cirt_cc, # model
                        itemtype = "Rasch", # IRT estimation
                        method = "SEM", # estimation method
                        pars = sv_cirt_score, # fix item parameters
                        removeEmptyRows = TRUE, # remove all rows
                            with no responses
                        draws = 5000, # number of IRT draws
                        verbose = FALSE) # allow the process to tun
                             discreetly
# CNT 5 model fit
CNT_5_logLik_cirt <- cnt5_cirt@Fit$logLik  # extract logLikelihood
CNT_5_AIC_cirt    <- cnt5_cirt@Fit$AIC  # extract AIC
CNT_5_BIC_cirt    <- cnt5_cirt@Fit$BIC  # extract BIC
CNT_5_cirt_indices <- data.frame(cbind(CNT_5_logLik_cirt, CNT_5_AIC
    _cirt, CNT_5_BIC_cirt))

cnt6_cirt <- mirt::mirt(dat6, # data for country 6
                        model = cirt_cc, # model
                        itemtype = "Rasch", # IRT estimation
                        method = "SEM", # estimation method
                        pars = sv_cirt_score, # fix item parameters
                        removeEmptyRows = TRUE, # remove all rows
                            with no responses
                        draws = 5000, # number of IRT draws
                        verbose = FALSE) # allow the process to tun
                             discreetly
# CNT 6 model fit
CNT_6_logLik_cirt <- cnt6_cirt@Fit$logLik  # extract logLikelihood
CNT_6_AIC_cirt    <- cnt6_cirt@Fit$AIC  # extract AIC
CNT_6_BIC_cirt    <- cnt6_cirt@Fit$BIC  # extract BIC
CNT_6_cirt_indices <- data.frame(cbind(CNT_6_logLik_cirt, CNT_6_AIC
    _cirt, CNT_6_BIC_cirt))

cnt7_cirt <- mirt::mirt(dat7, # data for country 7
                        model = cirt_cc, # model
                        itemtype = "Rasch", # IRT estimation
                        method = "SEM", # estimation method
                        pars = sv_cirt_score, # fix item parameters
                        removeEmptyRows = TRUE, # remove all rows
                            with no responses
                        draws = 5000, # number of IRT draws
                        verbose = FALSE) # allow the process to tun
                             discreetly

# CNT 7 model fit
CNT_7_logLik_cirt <- cnt7_cirt@Fit$logLik  # extract logLikelihood
```

```
CNT_7_AIC_cirt    <- cnt7_cirt@Fit$AIC  # extract AIC
CNT_7_BIC_cirt    <- cnt7_cirt@Fit$BIC  # extract BIC
CNT_7_cirt_indices <- data.frame(cbind(CNT_7_logLik_cirt, CNT_7_AIC
    _cirt, CNT_7_BIC_cirt))

cnt8_cirt <- mirt::mirt(dat8, # data for country 8
                        model = cirt_cc, # model
                        itemtype = "Rasch", # IRT estimation
                        method = "SEM", # estimation method
                        pars = sv_cirt_score, # fix item parameters
                        removeEmptyRows = TRUE, # remove all rows
                            with no responses
                        draws = 5000, # number of IRT draws
                        verbose = FALSE) # allow the process to tun
                             discreetly
# CNT 8 model fit
CNT_8_logLik_cirt <- cnt8_cirt@Fit$logLik # extract logLikelihood
CNT_8_AIC_cirt    <- cnt8_cirt@Fit$AIC # extract AIC
CNT_8_BIC_cirt    <- cnt8_cirt@Fit$BIC # extract BIC
CNT_8_cirt_indices <- data.frame(cbind(CNT_8_logLik_cirt, CNT_8_AIC
    _cirt, CNT_8_BIC_cirt))

cnt9_cirt <- mirt::mirt(dat9, # data for country 9
                        model = cirt_cc, # model
                        itemtype = "Rasch", # IRT estimation
                        method = "SEM", # estimation method
                        pars = sv_cirt_score, # fix item parameters
                        removeEmptyRows = TRUE, # remove all rows
                            with no responses
                        draws = 5000, # number of IRT draws
                        verbose = FALSE) # allow the process to tun
                             discreetly
# CNT 9 model fit
CNT_9_logLik_cirt <- cnt9_cirt@Fit$logLik # extract logLikelihood
CNT_9_AIC_cirt    <- cnt9_cirt@Fit$AIC  # extract AIC
CNT_9_BIC_cirt    <- cnt9_cirt@Fit$BIC  # extract BIC
CNT_9_cirt_indices <- data.frame(cbind(CNT_9_logLik_cirt, CNT_9_AIC
    _cirt, CNT_9_BIC_cirt))


# (b) score
#--- For each country
# (i) estimate factor scores
# (ii) calculate the standard deviation
# (iii) estimate the country mean
# (iv) calculate the PRMSE
cnt1_cirt_score <- mirt::fscores( cnt1_cirt,  method = "EAP",
                                  full.scores.SE=TRUE, QMC=TRUE )
cnt1_cirt_score_sd <- apply(cnt1_cirt_score, 2, sd) # estimate the
    standard deviations
cnt1_cirt_score_mean <- apply(cnt1_cirt_score, 2, mean)
cnt1_cirt_prmse <- empirical_rxx(cnt1_cirt_score)

cnt2_cirt_score <- mirt::fscores( cnt2_cirt,  method = "EAP",
```

```
                                    full.scores.SE=TRUE, QMC=TRUE )
cnt2_cirt_score_sd <- apply(cnt2_cirt_score, 2, sd) # estimate the
    standard deviations
cnt2_cirt_score_mean <- apply(cnt2_cirt_score, 2, mean)
cnt2_cirt_prmse <- empirical_rxx(cnt2_cirt_score)

cnt3_cirt_score <- mirt::fscores( cnt3_cirt,  method = "EAP",
                                    full.scores.SE=TRUE, QMC=TRUE )
cnt3_cirt_score_sd <- apply(cnt3_cirt_score, 2, sd) # estimate the
    standard deviations
cnt3_cirt_score_mean <- apply(cnt3_cirt_score, 2, mean)
cnt3_cirt_prmse <- empirical_rxx(cnt3_cirt_score)

cnt4_cirt_score <- mirt::fscores( cnt4_cirt,  method = "EAP",
                                    full.scores.SE=TRUE, QMC=TRUE )
cnt4_cirt_score_sd <- apply(cnt4_cirt_score, 2, sd) # estimate the
    standard deviations
cnt4_cirt_score_mean <- apply(cnt4_cirt_score, 2, mean)
cnt4_cirt_prmse <- empirical_rxx(cnt4_cirt_score)

cnt5_cirt_score <- mirt::fscores( cnt5_cirt,  method = "EAP",
                                    full.scores.SE=TRUE, QMC=TRUE )
cnt5_cirt_score_sd <- apply(cnt5_cirt_score, 2, sd) # estimate the
    standard deviations
cnt5_cirt_score_mean <- apply(cnt5_cirt_score, 2, mean)
cnt5_cirt_prmse <- empirical_rxx(cnt5_cirt_score)

cnt6_cirt_score <- mirt::fscores( cnt6_cirt,  method = "EAP",
                                    full.scores.SE=TRUE, QMC=TRUE )
cnt6_cirt_score_sd <- apply(cnt6_cirt_score, 2, sd) # estimate the
    standard deviations
cnt6_cirt_score_mean <- apply(cnt6_cirt_score, 2, mean)
cnt6_cirt_prmse <- empirical_rxx(cnt6_cirt_score)

cnt7_cirt_score <- mirt::fscores( cnt7_cirt,  method = "EAP",
                                    full.scores.SE=TRUE, QMC=TRUE )
cnt7_cirt_score_sd <- apply(cnt7_cirt_score, 2, sd) # estimate the
    standard deviations
cnt7_cirt_score_mean <- apply(cnt7_cirt_score, 2, mean)
cnt7_cirt_prmse <- empirical_rxx(cnt7_cirt_score)

cnt8_cirt_score <- mirt::fscores( cnt8_cirt,  method = "EAP",
                                    full.scores.SE=TRUE, QMC=TRUE )
cnt8_cirt_score_sd <- apply(cnt8_cirt_score, 2, sd) # estimate the
    standard deviations
cnt8_cirt_score_mean <- apply(cnt8_cirt_score, 2, mean)
cnt8_cirt_prmse <- empirical_rxx(cnt8_cirt_score)

cnt9_cirt_score <- mirt::fscores( cnt9_cirt,  method = "EAP",
                                    full.scores.SE=TRUE, QMC=TRUE )
cnt9_cirt_score_sd <- apply(cnt9_cirt_score, 2, sd) # estimate the
    standard deviations
cnt9_cirt_score_mean <- apply(cnt9_cirt_score, 2, mean)
cnt9_cirt_prmse <- empirical_rxx(cnt9_cirt_score)
```

```
#--- CIRT individual student scores
cirt_cnt_individual_score <- data.frame(rbind(cnt1_cirt_score,
                                              cnt2_cirt_score,
                                              cnt3_cirt_score,
                                              cnt4_cirt_score,
                                              cnt5_cirt_score,
                                              cnt6_cirt_score,
                                              cnt7_cirt_score,
                                              cnt8_cirt_score,
                                              cnt9_cirt_score))

est_cirt_cnt_individual_score <- data.frame(Study = Study,
                                            no_factors = no_factors
                                                ,
                                            items_per_fac = items_
                                                per_fac,
                                            cors = rr,
                                            replication = kk,
                                            theta_df,
                                            cirt_cnt_individual_
                                                score)

save(est_cirt_cnt_individual_score, file = paste0("est_cirt_cnt_
    individual_score_",
                                                   "ss", Study, "_zz
                                                       ",
                                                   no_factors, "_rr"
                                                       , rr,
                                                   "_cc", items_per_
                                                       fac, "_kk",
                                                   kk, ".RData"))


#--- Score standard deviations
est_cirt_individual_country_sd <- data.frame(rbind(cnt1_cirt_score_
    sd,
                                                    cnt2_cirt_score_
                                                        sd,
                                                    cnt3_cirt_score_
                                                        sd,
                                                    cnt4_cirt_score_
                                                        sd,
                                                    cnt5_cirt_score_
                                                        sd,
                                                    cnt6_cirt_score_
                                                        sd,
                                                    cnt7_cirt_score_
                                                        sd,
                                                    cnt8_cirt_score_
                                                        sd,
                                                    cnt9_cirt_score_
                                                        sd))
```

```
cirt_individual_country_sd <- data.frame(Study = Study,
                                         CNT = ccc,
                                         true_country_sd = true_
                                             country_sd,
                                         no_factors = no_factors,
                                         items_per_fac = items_per_
                                             fac,
                                         cors = rr,
                                         replications = kk,
                                         est_cirt_individual_
                                             country_sd)

save(cirt_individual_country_sd, file = paste0("cirt_individual_
    country_sd_", "ss", Study, "_zz",
                                                no_factors, "_rr",
                                                    rr, "_cc", items
                                                    _per_fac, "_kk",
                                                     kk, ".RData"))

#--- Place cirt scores into a data frame
cirt_score_by_country <- data.frame(rbind( cnt1_cirt_score_mean,
                                           cnt2_cirt_score_mean,
                                           cnt3_cirt_score_mean,
                                           cnt4_cirt_score_mean,
                                           cnt5_cirt_score_mean,
                                           cnt6_cirt_score_mean,
                                           cnt7_cirt_score_mean,
                                           cnt8_cirt_score_mean,
                                           cnt9_cirt_score_mean ))

colnames(cirt_score_by_country)[1:no_factors] <- paste0( "cirt_",
    1:no_factors )
colnames(cirt_score_by_country)[(no_factors+1) : (2*no_factors)] <-
     paste0( "cirt_se_", 1:no_factors )
rownames(cirt_score_by_country) <- NULL


#--- Place CIRT prmse into a data frame
cirt_PRMSE_by_country <- data.frame(rbind( cnt1_cirt_prmse,
                                           cnt2_cirt_prmse,
                                           cnt3_cirt_prmse,
                                           cnt4_cirt_prmse,
                                           cnt5_cirt_prmse,
                                           cnt6_cirt_prmse,
                                           cnt7_cirt_prmse,
                                           cnt8_cirt_prmse,
                                           cnt9_cirt_prmse ))
colnames(cirt_PRMSE_by_country)[1:no_factors] <- paste0( "cirt_
    prmse_", 1:no_factors )
rownames(cirt_PRMSE_by_country) <- NULL

#--- save cirt fit
# create data frame of cirt fit indices
cirt_fit_by_country <- data.frame(Study = Study,
```

```
                                    no_factors = no_factors,
                                    items_per_fac = items_per_fac,
                                    cors = rr,
                                    replication = kk,
                                    CNT_1_cirt_indices,
                                    CNT_2_cirt_indices,
                                    CNT_3_cirt_indices,
                                    CNT_4_cirt_indices,
                                    CNT_5_cirt_indices,
                                    CNT_6_cirt_indices,
                                    CNT_7_cirt_indices,
                                    CNT_8_cirt_indices,
                                    CNT_9_cirt_indices)
save(cirt_fit_by_country, file = paste0("cirt_fit_by_country_", "ss
    ",
                                        Study, "_zz", no_factors, "
                                            _rr",
                                        rr, "_cc", items_per_fac, "
                                            _kk",
                                        kk, "_.Rdata"))

#-----------------------------------------------------#
# Rescale item parameters using the mean-mean method in order to
    put the items
# on the same scale in scoring.
#-----------------------------------------------------#
## Insert d from uirt to mirt
sv_mirt <- mod2values(mirt_fit)
sv_mirt_score <- sv_mirt
sv_mirt_score[sv_mirt_score$name=="d", "value"] <- ( mirt_item$rs_b
     ) * -1  #use rescaled_difficulty and change from b to d
sv_mirt_score[1: ((ncol(mirt_ip)-1) *total_items)  , "est"] <- TRUE
     # estimate covariances


cnt1_mirt <- mirt::mirt(dat1, # data for country 1
                        model = mirt_cc, # model
                        itemtype = "Rasch", # IRT estimation
                        method = "SEM", # estimation method
                        pars = sv_mirt_score, # fix item parameters
                        removeEmptyRows = TRUE, # remove all rows
                            with no responses
                        draws = 5000, # number of IRT draws
                        verbose = FALSE) # allow the process to tun
                             discreetly

# CNT 1 model fit
CNT_1_logLik_mirt  <- cnt1_mirt@Fit$logLik # extract logLikelihood
CNT_1_AIC_mirt     <- cnt1_mirt@Fit$AIC # extract AIC
CNT_1_BIC_mirt     <- cnt1_mirt@Fit$BIC # extract BIC
# Place fit indices in a table
CNT_1_mirt_indices <- data.frame(cbind(CNT_1_logLik_mirt, CNT_1_AIC
    _mirt, CNT_1_BIC_mirt))
```

```
cnt2_mirt <- mirt::mirt(dat2, # data for country 2
                        model = mirt_cc, # model
                        itemtype = "Rasch", # IRT estimation
                        method = "SEM", # estimation method
                        pars = sv_mirt_score, # fix item parameters
                        removeEmptyRows = TRUE, # remove all rows
                            with no responses
                        draws = 5000, # number of IRT draws
                        verbose = FALSE) # allow the process to tun
                             discreetly
# CNT 2 model fit
CNT_2_logLik_mirt  <- cnt2_mirt@Fit$logLik # extract logLikelihood
CNT_2_AIC_mirt     <- cnt2_mirt@Fit$AIC # extract AIC
CNT_2_BIC_mirt     <- cnt2_mirt@Fit$BIC # extract BIC
CNT_2_mirt_indices <- data.frame(cbind(CNT_2_logLik_mirt, CNT_2_AIC
    _mirt, CNT_2_BIC_mirt))

cnt3_mirt <- mirt::mirt(dat3, # data for country 3
                        model = mirt_cc, # model
                        itemtype = "Rasch", # IRT estimation
                        method = "SEM", # estimation method
                        pars = sv_mirt_score, # fix item parameters
                        removeEmptyRows = TRUE, # remove all rows
                            with no responses
                        draws = 5000, # number of IRT draws
                        verbose = FALSE) # allow the process to tun
                             discreetly
# CNT 3 model fit
CNT_3_logLik_mirt <- cnt3_mirt@Fit$logLik   # extract logLikelihood
CNT_3_AIC_mirt    <- cnt3_mirt@Fit$AIC # extract AIC
CNT_3_BIC_mirt    <- cnt3_mirt@Fit$BIC # extract BIC
CNT_3_mirt_indices <- data.frame(cbind(CNT_3_logLik_mirt, CNT_3_AIC
    _mirt, CNT_3_BIC_mirt))

cnt4_mirt <- mirt::mirt(dat4, # data for country 4
                        model = mirt_cc, # model
                        itemtype = "Rasch", # IRT estimation
                        method = "SEM", # estimation method
                        pars = sv_mirt_score, # fix item parameters
                        removeEmptyRows = TRUE, # remove all rows
                            with no responses
                        draws = 5000, # number of IRT draws
                        verbose = FALSE) # allow the process to tun
                             discreetly
# CNT 4 model fit
CNT_4_logLik_mirt <- cnt4_mirt@Fit$logLik  # extract logLikelihood
CNT_4_AIC_mirt    <- cnt4_mirt@Fit$AIC  # extract AIC
CNT_4_BIC_mirt    <- cnt4_mirt@Fit$BIC  # extract BIC
CNT_4_mirt_indices <- data.frame(cbind(CNT_4_logLik_mirt, CNT_4_AIC
    _mirt, CNT_4_BIC_mirt))

cnt5_mirt <- mirt::mirt(dat5, # data for country 5
                        model = mirt_cc, # model
```

```
                        itemtype = "Rasch", # IRT estimation
                        method = "SEM", # estimation method
                        pars = sv_mirt_score, # fix item parameters
                        removeEmptyRows = TRUE, # remove all rows
                            with no responses
                        draws = 5000, # number of IRT draws
                        verbose = FALSE) # allow the process to tun
                             discreetly
# CNT 5 model fit
CNT_5_logLik_mirt <- cnt5_mirt@Fit$logLik  # extract logLikelihood
CNT_5_AIC_mirt    <- cnt5_mirt@Fit$AIC  # extract AIC
CNT_5_BIC_mirt    <- cnt5_mirt@Fit$BIC  # extract BIC
CNT_5_mirt_indices <- data.frame(cbind(CNT_5_logLik_mirt, CNT_5_AIC
    _mirt, CNT_5_BIC_mirt))

cnt6_mirt <- mirt::mirt(dat6, # data for country 6
                        model = mirt_cc, # model
                        itemtype = "Rasch", # IRT estimation
                        method = "SEM", # estimation method
                        pars = sv_mirt_score, # fix item parameters
                        removeEmptyRows = TRUE, # remove all rows
                            with no responses
                        draws = 5000, # number of IRT draws
                        verbose = FALSE) # allow the process to tun
                             discreetly
# CNT 6 model fit
CNT_6_logLik_mirt <- cnt6_mirt@Fit$logLik  # extract logLikelihood
CNT_6_AIC_mirt    <- cnt6_mirt@Fit$AIC  # extract AIC
CNT_6_BIC_mirt    <- cnt6_mirt@Fit$BIC  # extract BIC
CNT_6_mirt_indices <- data.frame(cbind(CNT_6_logLik_mirt, CNT_6_AIC
    _mirt, CNT_6_BIC_mirt))

cnt7_mirt <- mirt::mirt(dat7, # data for country 7
                        model = mirt_cc, # model
                        itemtype = "Rasch", # IRT estimation
                        method = "SEM", # estimation method
                        pars = sv_mirt_score, # fix item parameters
                        removeEmptyRows = TRUE, # remove all rows
                            with no responses
                        draws = 5000, # number of IRT draws
                        verbose = FALSE) # allow the process to tun
                             discreetly

# CNT 7 model fit
CNT_7_logLik_mirt <- cnt7_mirt@Fit$logLik  # extract logLikelihood
CNT_7_AIC_mirt    <- cnt7_mirt@Fit$AIC  # extract AIC
CNT_7_BIC_mirt    <- cnt7_mirt@Fit$BIC  # extract BIC
CNT_7_mirt_indices <- data.frame(cbind(CNT_7_logLik_mirt, CNT_7_AIC
    _mirt, CNT_7_BIC_mirt))

cnt8_mirt <- mirt::mirt(dat8, # data for country 8
                        model = mirt_cc, # model
                        itemtype = "Rasch", # IRT estimation
                        method = "SEM", # estimation method
```

```
                        pars = sv_mirt_score, # fix item parameters
                        removeEmptyRows = TRUE, # remove all rows
                            with no responses
                        draws = 5000, # number of IRT draws
                        verbose = FALSE) # allow the process to tun
                             discreetly
# CNT 8 model fit
CNT_8_logLik_mirt <- cnt8_mirt@Fit$logLik # extract logLikelihood
CNT_8_AIC_mirt    <- cnt8_mirt@Fit$AIC # extract AIC
CNT_8_BIC_mirt    <- cnt8_mirt@Fit$BIC # extract BIC
CNT_8_mirt_indices <- data.frame(cbind(CNT_8_logLik_mirt, CNT_8_AIC
    _mirt, CNT_8_BIC_mirt))

cnt9_mirt <- mirt::mirt(dat9, # data for country 9
                        model = mirt_cc, # model
                        itemtype = "Rasch", # IRT estimation
                        method = "SEM", # estimation method
                        pars = sv_mirt_score, # fix item parameters
                        removeEmptyRows = TRUE, # remove all rows
                            with no responses
                        draws = 5000, # number of IRT draws
                        verbose = FALSE) # allow the process to tun
                             discreetly
# CNT 9 model fit
CNT_9_logLik_mirt <- cnt9_mirt@Fit$logLik # extract logLikelihood
CNT_9_AIC_mirt    <- cnt9_mirt@Fit$AIC  # extract AIC
CNT_9_BIC_mirt    <- cnt9_mirt@Fit$BIC  # extract BIC
CNT_9_mirt_indices <- data.frame(cbind(CNT_9_logLik_mirt, CNT_9_AIC
    _mirt, CNT_9_BIC_mirt))


# (b) score
#--- For each country
# (i) estimate factor scores
# (ii) calculate the standard deviation
# (iii) estimate the country mean
# (iv) calculate the PRMSE
cnt1_mirt_score <- mirt::fscores( cnt1_mirt,  method = "EAP", full.
    scores.SE=TRUE, QMC=TRUE )
cnt1_mirt_score_sd <- apply(cnt1_mirt_score, 2, sd) # estimate the
    standard deviations
cnt1_mirt_score_mean <- apply(cnt1_mirt_score, 2, mean)
cnt1_mirt_prmse <- empirical_rxx(cnt1_mirt_score)

cnt2_mirt_score <- mirt::fscores( cnt2_mirt,  method = "EAP", full.
    scores.SE=TRUE, QMC=TRUE )
cnt2_mirt_score_sd <- apply(cnt2_mirt_score, 2, sd) # estimate the
    standard deviations
cnt2_mirt_score_mean <- apply(cnt2_mirt_score, 2, mean)
cnt2_mirt_prmse <- empirical_rxx(cnt2_mirt_score)

cnt3_mirt_score <- mirt::fscores( cnt3_mirt,  method = "EAP", full.
    scores.SE=TRUE, QMC=TRUE )
cnt3_mirt_score_sd <- apply(cnt3_mirt_score, 2, sd) # estimate the
```

```
    standard deviations
cnt3_mirt_score_mean <- apply(cnt3_mirt_score, 2, mean)
cnt3_mirt_prmse <- empirical_rxx(cnt3_mirt_score)

cnt4_mirt_score <- mirt::fscores( cnt4_mirt,  method = "EAP", full.
    scores.SE=TRUE, QMC=TRUE )
cnt4_mirt_score_sd <- apply(cnt4_mirt_score, 2, sd) # estimate the
    standard deviations
cnt4_mirt_score_mean <- apply(cnt4_mirt_score, 2, mean)
cnt4_mirt_prmse <- empirical_rxx(cnt4_mirt_score)

cnt5_mirt_score <- mirt::fscores( cnt5_mirt,  method = "EAP", full.
    scores.SE=TRUE, QMC=TRUE )
cnt5_mirt_score_sd <- apply(cnt5_mirt_score, 2, sd) # estimate the
    standard deviations
cnt5_mirt_score_mean <- apply(cnt5_mirt_score, 2, mean)
cnt5_mirt_prmse <- empirical_rxx(cnt5_mirt_score)

cnt6_mirt_score <- mirt::fscores( cnt6_mirt,  method = "EAP",
                                  full.scores.SE=TRUE, QMC=TRUE )
cnt6_mirt_score_sd <- apply(cnt6_mirt_score, 2, sd) # estimate the
    standard deviations
cnt6_mirt_score_mean <- apply(cnt6_mirt_score, 2, mean)
cnt6_mirt_prmse <- empirical_rxx(cnt6_mirt_score)

cnt7_mirt_score <- mirt::fscores( cnt7_mirt,  method = "EAP",
                                  full.scores.SE=TRUE, QMC=TRUE )
cnt7_mirt_score_sd <- apply(cnt7_mirt_score, 2, sd) # estimate the
    standard deviations
cnt7_mirt_score_mean <- apply(cnt7_mirt_score, 2, mean)
cnt7_mirt_prmse <- empirical_rxx(cnt7_mirt_score)

cnt8_mirt_score <- mirt::fscores( cnt8_mirt,  method = "EAP",
                                  full.scores.SE=TRUE, QMC=TRUE )
cnt8_mirt_score_sd <- apply(cnt8_mirt_score, 2, sd) # estimate the
    standard deviations
cnt8_mirt_score_mean <- apply(cnt8_mirt_score, 2, mean)
cnt8_mirt_prmse <- empirical_rxx(cnt8_mirt_score)

cnt9_mirt_score <- mirt::fscores( cnt9_mirt,  method = "EAP",
                                  full.scores.SE=TRUE, QMC=TRUE )
cnt9_mirt_score_sd <- apply(cnt9_mirt_score, 2, sd) # estimate the
    standard deviations
cnt9_mirt_score_mean <- apply(cnt9_mirt_score, 2, mean)
cnt9_mirt_prmse <- empirical_rxx(cnt9_mirt_score)

#--- Individual scores
mirt_cnt_individual_score <- data.frame(rbind(cnt1_mirt_score,
                                              cnt2_mirt_score,
                                              cnt3_mirt_score,
                                              cnt4_mirt_score,
                                              cnt5_mirt_score,
                                              cnt6_mirt_score,
                                              cnt7_mirt_score,
```

```
                                                cnt8_mirt_score,
                                                cnt9_mirt_score))

est_mirt_cnt_individual_score <- data.frame(Study = Study,
                                            no_factors = no_factors
                                                ,
                                            items_per_fac = items_
                                                per_fac,
                                            cors = rr,
                                            replication = kk,
                                            theta_df,
                                            mirt_cnt_individual_
                                                score)

save(est_mirt_cnt_individual_score, file = paste0("est_mirt_cnt_
    individual_score_",
                                                    "ss", Study, "_zz
                                                        ",
                                                    no_factors, "_rr"
                                                        , rr,
                                                    "_cc", items_per_
                                                        fac,
                                                    "_kk", kk, ".
                                                        RData"))

#--- Score standard deviations
est_mirt_individual_country_sd <- data.frame(rbind(cnt1_mirt_score_
    sd,
                                                   cnt2_mirt_score_
                                                       sd,
                                                   cnt3_mirt_score_
                                                       sd,
                                                   cnt4_mirt_score_
                                                       sd,
                                                   cnt5_mirt_score_
                                                       sd,
                                                   cnt6_mirt_score_
                                                       sd,
                                                   cnt7_mirt_score_
                                                       sd,
                                                   cnt8_mirt_score_
                                                       sd,
                                                   cnt9_mirt_score_
                                                       sd))

mirt_individual_country_sd <- data.frame(Study = Study,
                                         CNT = ccc,
                                         true_country_sd = true_
                                             country_sd,
                                         no_factors = no_factors,
                                         items_per_fac = items_per_
                                             fac,
                                         cors = rr,
                                         replications = kk,
```

```
                                        est_mirt_individual_
                                            country_sd)

save(mirt_individual_country_sd, file = paste0("mirt_individual_
    country_sd_",
                                                "ss", Study, "_zz",
                                                no_factors, "_rr",
                                                    rr,
                                                "_cc", items_per_fac
                                                    ,
                                                "_kk", kk, ".RData")
                                                    )

#--- Place mirt scores into a data frame
mirt_score_by_country <- data.frame(rbind( cnt1_mirt_score_mean,
                                           cnt2_mirt_score_mean,
                                           cnt3_mirt_score_mean,
                                           cnt4_mirt_score_mean,
                                           cnt5_mirt_score_mean,
                                           cnt6_mirt_score_mean,
                                           cnt7_mirt_score_mean,
                                           cnt8_mirt_score_mean,
                                           cnt9_mirt_score_mean ))

colnames(mirt_score_by_country)[1:no_factors] <- paste0( "mirt_",
    1:no_factors )
colnames(mirt_score_by_country)[(no_factors+1) : (2*no_factors)] <-
     paste0( "mirt_se_", 1:no_factors )
rownames(mirt_score_by_country) <- NULL

#--- Place MIRT prmse into a data frame
mirt_PRMSE_by_country <- data.frame(rbind( cnt1_mirt_prmse,
                                           cnt2_mirt_prmse,
                                           cnt3_mirt_prmse,
                                           cnt4_mirt_prmse,
                                           cnt5_mirt_prmse,
                                           cnt6_mirt_prmse,
                                           cnt7_mirt_prmse,
                                           cnt8_mirt_prmse,
                                           cnt9_mirt_prmse ))
colnames(mirt_PRMSE_by_country)[1:no_factors] <- paste0( "mirt_
    prmse_", 1:no_factors )
rownames(mirt_PRMSE_by_country) <- NULL

#--- save mirt fit
# create data frame of mirt fit indices
mirt_fit_by_country <- data.frame(Study = Study,
                                  no_factors = no_factors,
                                  items_per_fac = items_per_fac,
                                  cors = rr,
                                  replication = kk,
                                  CNT_1_mirt_indices,
                                  CNT_2_mirt_indices,
                                  CNT_3_mirt_indices,
```

```
                                    CNT_4_mirt_indices,
                                    CNT_5_mirt_indices,
                                    CNT_6_mirt_indices,
                                    CNT_7_mirt_indices,
                                    CNT_8_mirt_indices,
                                    CNT_9_mirt_indices)
save(mirt_fit_by_country, file = paste0("mirt_fit_by_country_", "ss
    ", Study, "_zz", no_factors, "_rr",
                                          rr, "_cc", items_per_fac, "
                                              _kk", kk, "_.Rdata"))

#-----------------------------------------------------#
# Summarize and save estimated country scores
#-----------------------------------------------------#
cnt_domain_score <- data.frame(
  CNT = ccc,
  Study = Study,
  no_factors = no_factors,
  items_per_fac = items_per_fac,
  theta_T = sss,
  cors = rr,
  replication = kk,
  uirt_score_by_country,
  cirt_score_by_country,
  mirt_score_by_country )
save(cnt_domain_score, file = paste0("est_cnt_score_",
                                     "ss", Study, "_zz", no_factors
                                         ,
                                     "_rr", rr, "_cc", items_per_
                                         fac,
                                     "_kk", kk, ".Rdata"))

#-----------------------------------------------------#
# Summarize and save estimated country PRMSE
#-----------------------------------------------------#
PRMSE_by_country <- data.frame(
  CNT = ccc,
  Study = Study,
  no_factors = no_factors,
  items_per_fac = items_per_fac,
  cors = rr,
  replication = kk,
  uirt_PRMSE_by_country,
  cirt_PRMSE_by_country,
  mirt_PRMSE_by_country )
save(PRMSE_by_country, file = paste0("PRMSE_by_country_", "ss",
                                     Study, "_zz", no_factors, "_rr
                                         ",
                                     rr, "_cc", items_per_fac, "_kk
                                         ",
                                     kk, ".Rdata"))

#-----------------------------------------------------#
# Entire data PRMSE
```

```
#-----------------------------------------------------#
uirt_all_fit <- mirt::mirt(resp,
                           model = cirt_cc,
                           itemtype = "Rasch",
                           method = "SEM",
                           pars = sv_uirt_fx_from_cirt,
                           draws = 5000,
                           verbose = FALSE)

uirt_all_score <- mirt::fscores(uirt_all_fit, method = "EAP",
                                full.scores.SE=TRUE, QMC=TRUE )
uirt__all_score_sd <- apply(uirt_all_score, 2, sd) # estimate the
    standard deviations
uirt_all_score_mean <- apply(uirt_all_score, 2, mean)
uirt_all_prmse <- empirical_rxx(uirt_all_score)

cirt_all_score <- mirt::fscores( cirt_fit, method = "EAP",
                                 full.scores.SE=TRUE, QMC=TRUE )
cirt__all_score_sd <- apply(cirt_all_score, 2, sd) # estimate the
    standard deviations
cirt_all_score_mean <- apply(cirt_all_score, 2, mean)
cirt_all_prmse <- empirical_rxx(cirt_all_score)

mirt_all_score <- mirt::fscores( mirt_fit,  method = "EAP",
                                 full.scores.SE=TRUE, QMC=TRUE )
mirt__all_score_sd <- apply(mirt_all_score, 2, sd) # estimate the
    standard deviations
mirt_all_score_mean <- apply(mirt_all_score, 2, mean)
mirt_all_prmse <- empirical_rxx(mirt_all_score)


#-----------------------------------------------------#
# Save scores
#-----------------------------------------------------#
temp_all_data_uirt_scores <- data.frame(uirt_all_score)
all_data_uirt_scores <- data.frame(no_factors = no_factors,
                                   items_per_fac = items_per_fac,
                                   cors = rr,
                                   replication = kk,
                                   temp_all_data_uirt_scores)

save(all_data_uirt_scores, file = paste0("all_data_uirt_scores_", "
    ss",
                                         Study, "_zz", no_factors,
                                         "_rr", rr, "_cc", items_
                                             per_fac,
                                         "_kk", kk, ".Rdata"))

temp_all_data_cirt_scores <- data.frame(cirt_all_score)
all_data_cirt_scores <- data.frame(no_factors = no_factors,
                                   items_per_fac = items_per_fac,
                                   cors = rr,
                                   replication = kk,
                                   temp_all_data_cirt_scores)
```

```
save(all_data_cirt_scores, file = paste0("all_data_cirt_scores_", "
    ss",
                                         Study, "_zz", no_factors,
                                         "_rr", rr, "_cc", items_
                                             per_fac,
                                         "_kk", kk, ".Rdata"))

temp_all_data_mirt_scores <- data.frame(mirt_all_score)
all_data_mirt_scores <- data.frame(no_factors = no_factors,
                                   items_per_fac = items_per_fac,
                                   cors = rr,
                                   replication = kk,
                                   temp_all_data_mirt_scores)

save(all_data_mirt_scores, file = paste0("all_data_mirt_scores_", "
    ss",
                                         Study, "_zz", no_factors,
                                         "_rr", rr, "_cc", items_
                                             per_fac,
                                         "_kk", kk, ".Rdata"))

#-----------------------------------------------------#
# Save standard deviations
#-----------------------------------------------------#
all_sd <- data.frame(uirt_all_score_sd, cirt_all_score_sd,
                     mirt_all_score_sd)

all_standard_deviations <- data.frame(Study = Study,
                                      no_factors = no_factors,
                                      items_per_fac = items_per_fac
                                          ,
                                      cors = rr,
                                      replication = kk,
                                      all_sd)

save(all_standard_deviations, file = paste0("all_standard_
    deviations_", "ss",
                                            Study, "_zz", no_
                                                factors,
                                            "_rr", rr, "_cc", items
                                                _per_fac,
                                            "_kk", kk, ".Rdata"))

#-----------------------------------------------------#
# Summarize population PRMSE and save PRMSE
#-----------------------------------------------------#
#---- Entire population
prmse_all <- data.frame(
  Study = Study,
  no_factors = no_factors,
  items_per_fac = items_per_fac,
  cors = rr,
  replication = kk,
```

```
    domain = 1: no_factors,
    uirt_all_prmse = uirt_all_prmse,
    cirt_all_prmse = cirt_all_prmse,
    mirt_all_prmse = mirt_all_prmse)
  save(prmse_all, file = paste0("prmse_all_", "ss", Study, "_zz", no_
      factors,
                                "_rr", rr, "_cc", items_per_fac, "_kk
                                    ", kk, ".Rdata"))

  #  } #close the replication
} # close function

#-----------------------------------------------------#
# Send the function to parallel nodes.
# The function contains the simulation code.
# In this case, if the computer has 4 nodes, use 3 to run seperate
    replications.
#-----------------------------------------------------#
library(parallel)
cl <- makeCluster(3)
Analysis <- parLapply(cl, replications, myFunction, Study = Study,
                      no_factors = no_factors, items = items,
                      items_per_fac = items_per_fac, rr = rr,
                      n_examinees = n_examinees, total_items = total_
                          items,
                      test_items = test_items, mods = mods, ccc = ccc
                          , sss = sss) # rename the stored results

names(Analysis) <- paste0("r", 1:no_reps)

stopCluster(cl)

save(Analysis, file = paste0("Analysis_", "ss", Study,
                             "_zz", no_factors, "_rr", rr, "_cc",
                                 items_per_fac, ".Rdata"))

# End run
```

Appendix D

Study 2 $d1$- and $d2$-Parameter Bias: Multiple Groups

Figure D.1

Bias of d1-Parameter for the 3 Domain, 40 Items per Domain Tests: Single Groups

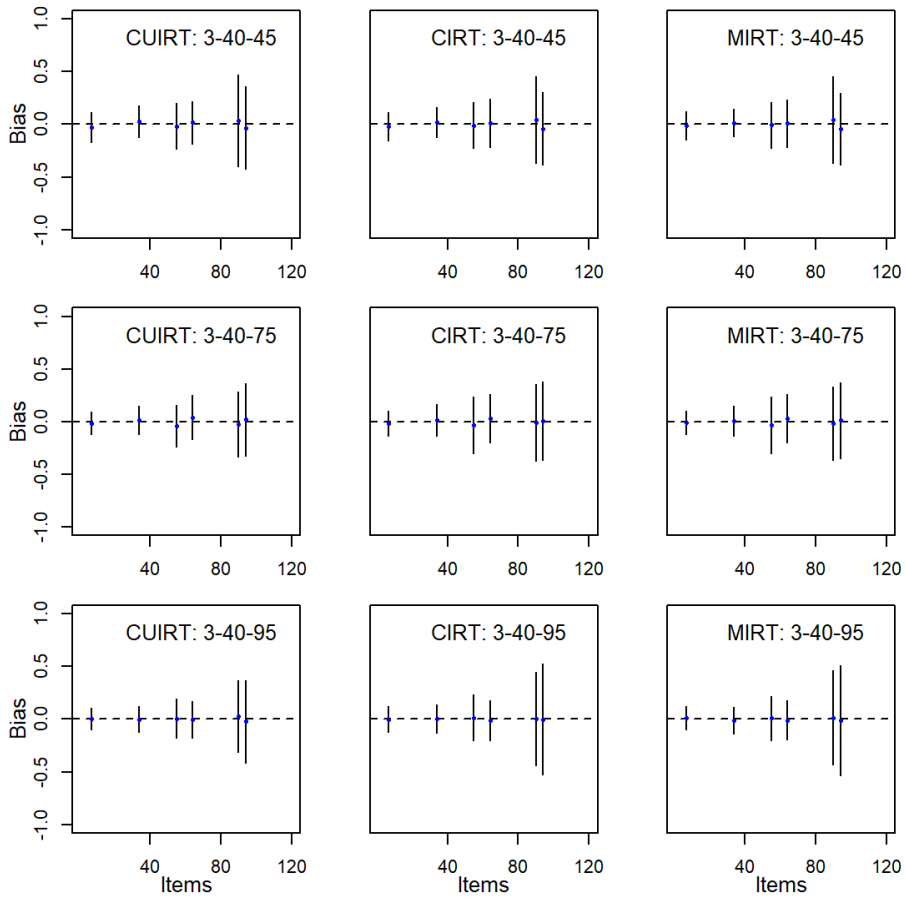


Figure D.2

Bias of d1-Parameter for the 3 Domain, 60 Items per Domain Tests: Single Groups

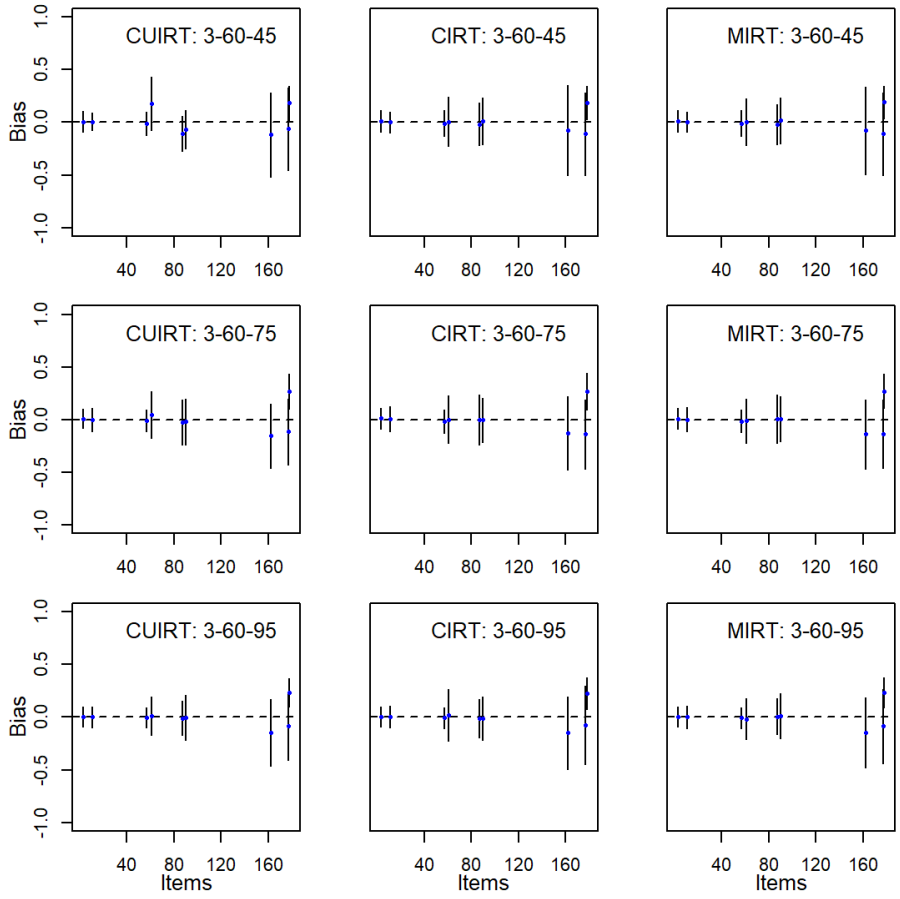


Figure D.3

Bias of d1-Parameter for the 4 Domain, 40 Items per Domain Tests: Single Groups

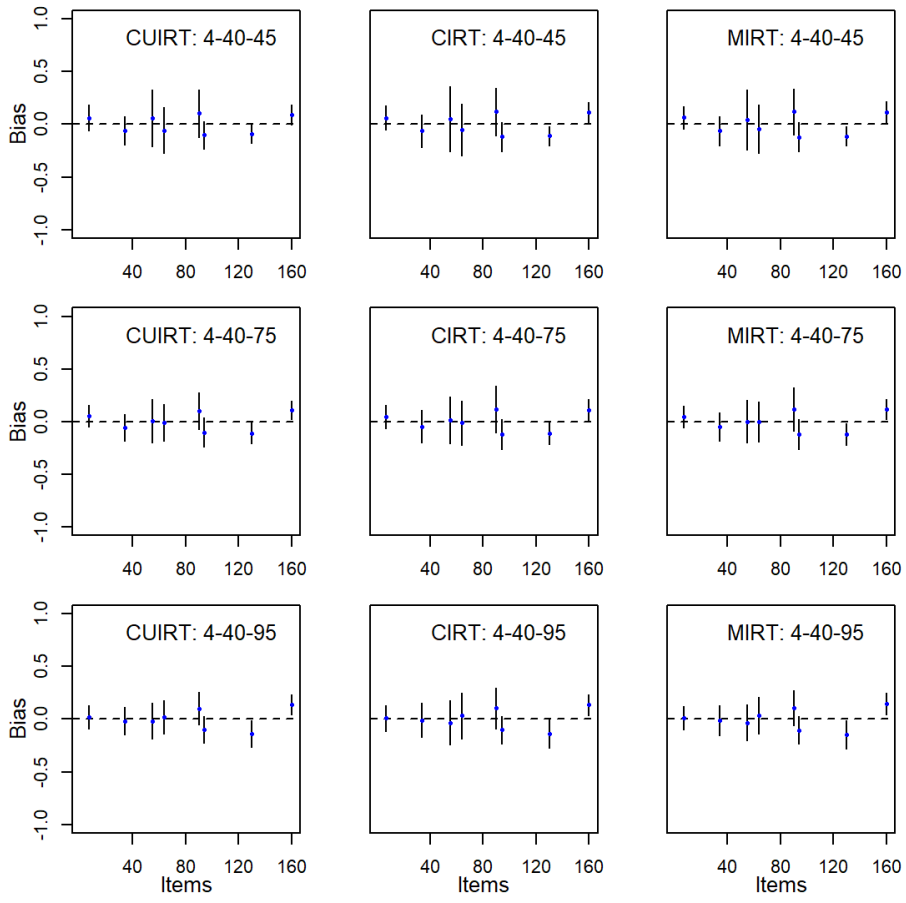


Figure D.4

Bias of d1-Parameter for the 4 Domain, 60 Items per Domain Tests: Single Groups

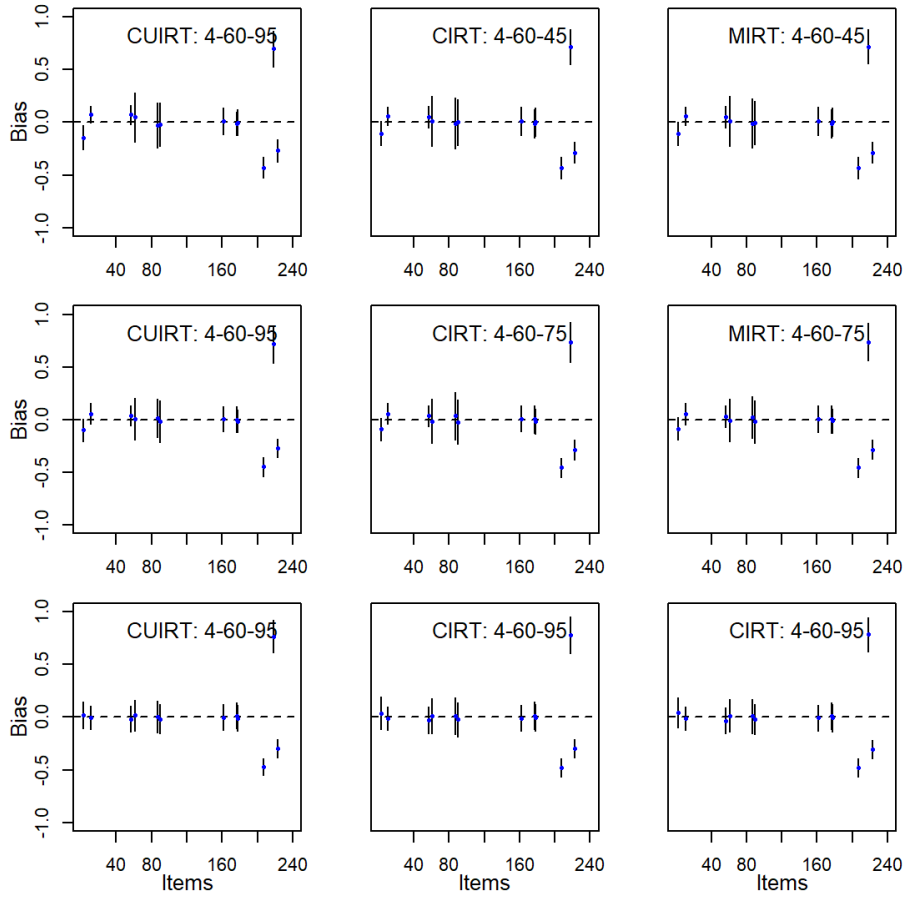


Figure D.5

Bias of d2-Parameter for the 3 Domain, 40 Items per Domain Tests: Single Groups

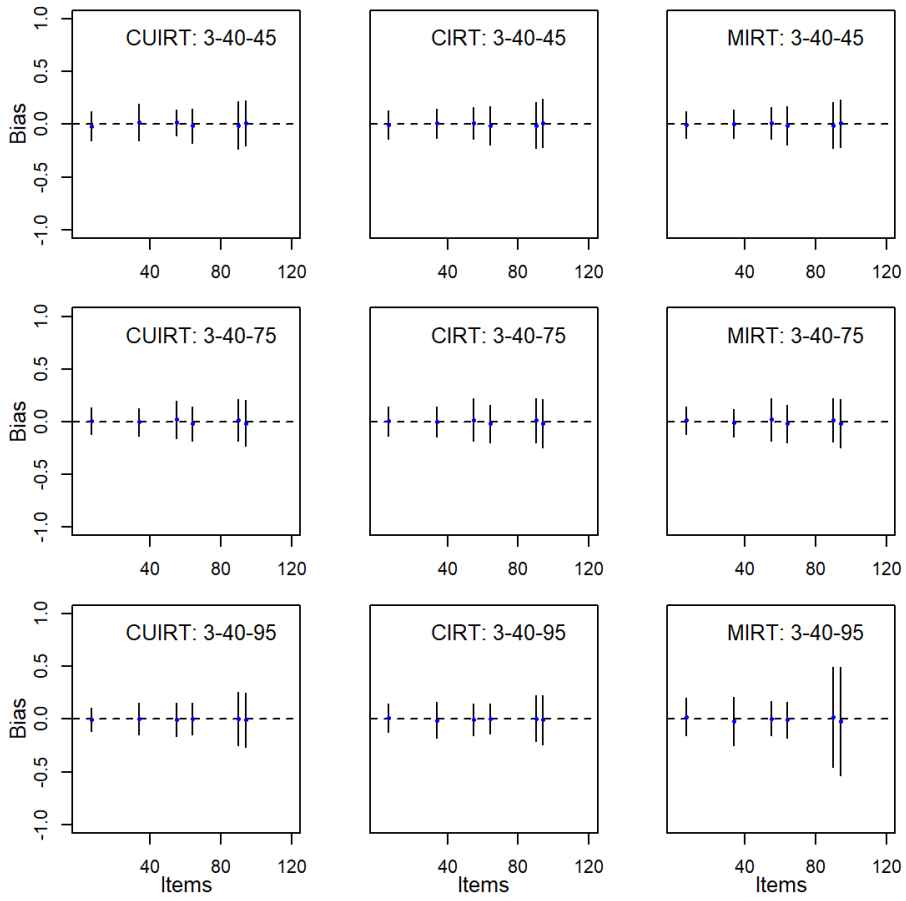


Figure D.6

Bias of d2-Parameter for the 3 Domain, 60 Items per Domain Tests: Single Groups

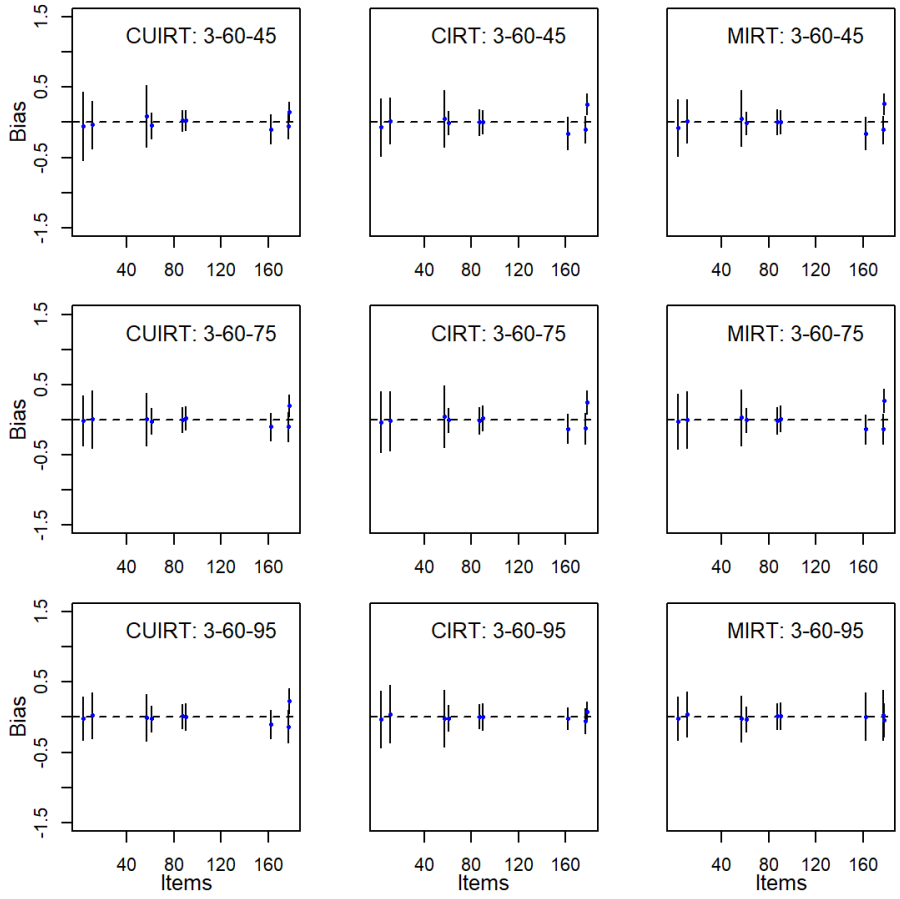


Figure D.7

Bias of d2-Parameter for the 4 Domain, 40 Items per Domain Tests: Single Groups

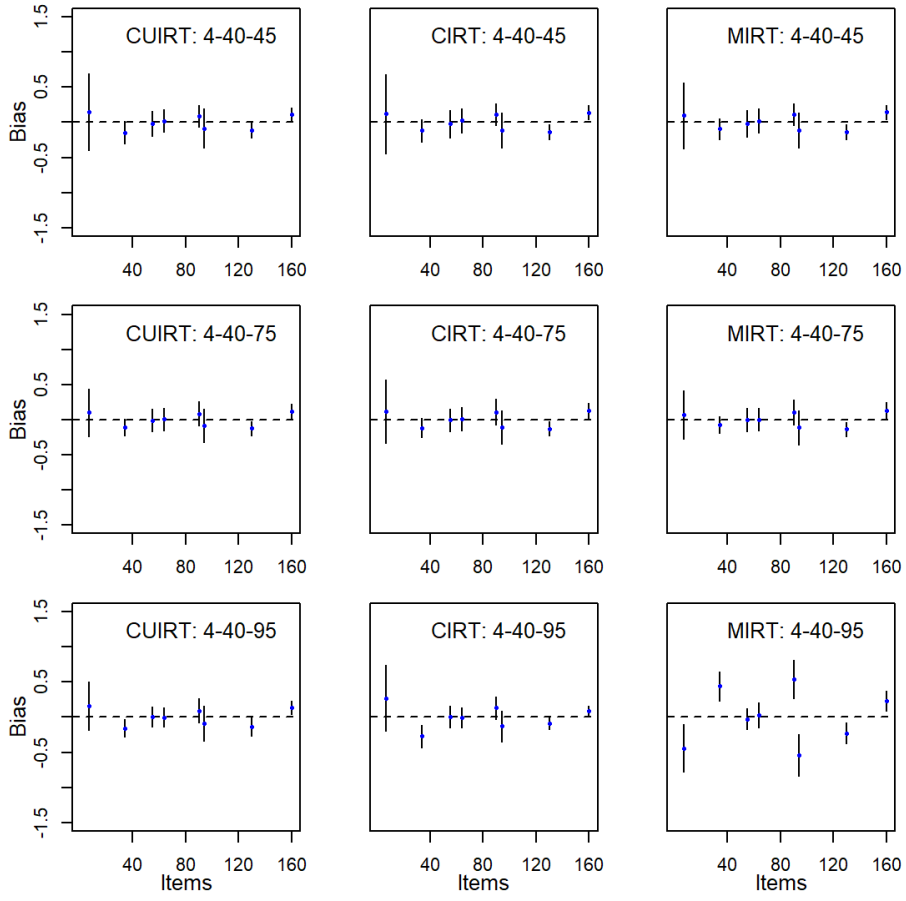


Figure D.8

Bias of d2-Parameter for the 4 Domain, 60 Items per Domain Tests: Single Groups

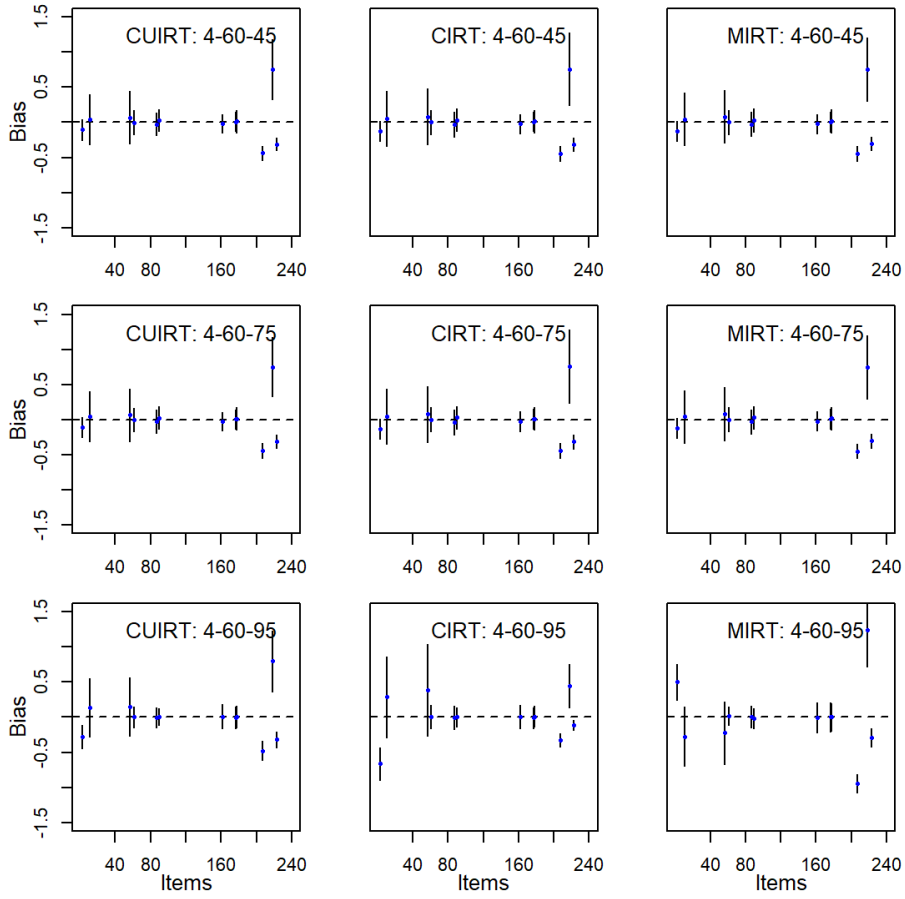


Figure D.9

Bias of d1-Parameter for the 3 Domain, 40 Items per Domain Tests: Multiple Groups

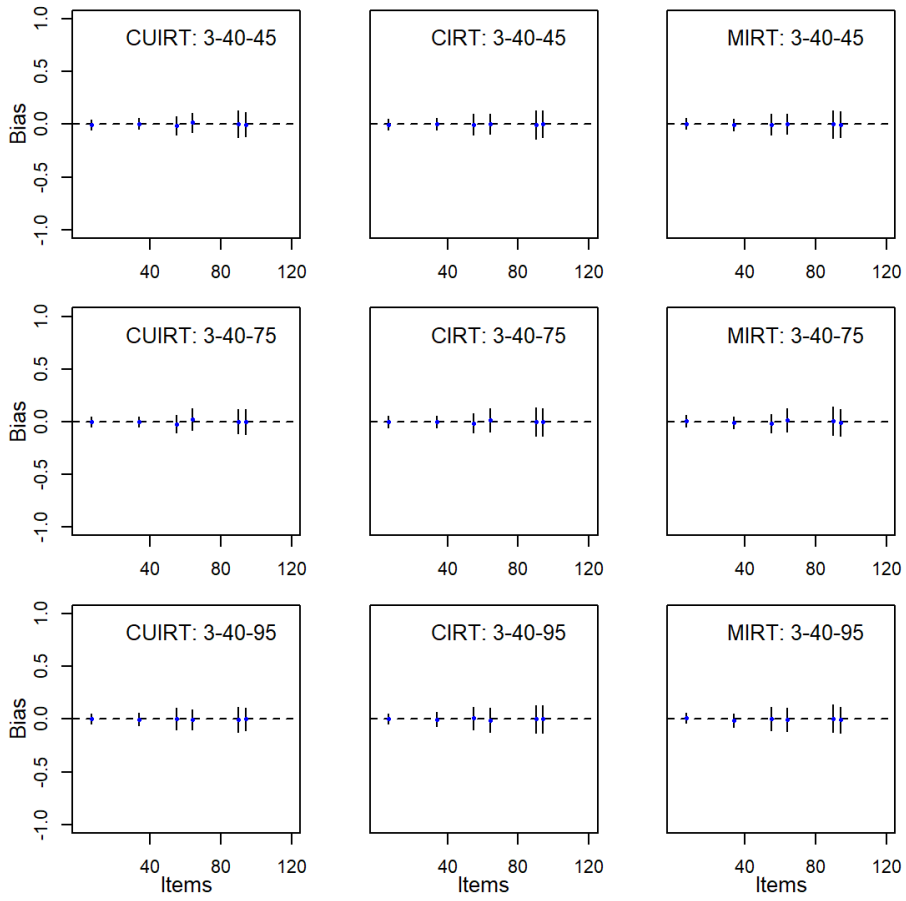


Figure D.10

Bias of d1-Parameter for the 3 Domain, 60 Items per Domain Tests: Multiple Groups

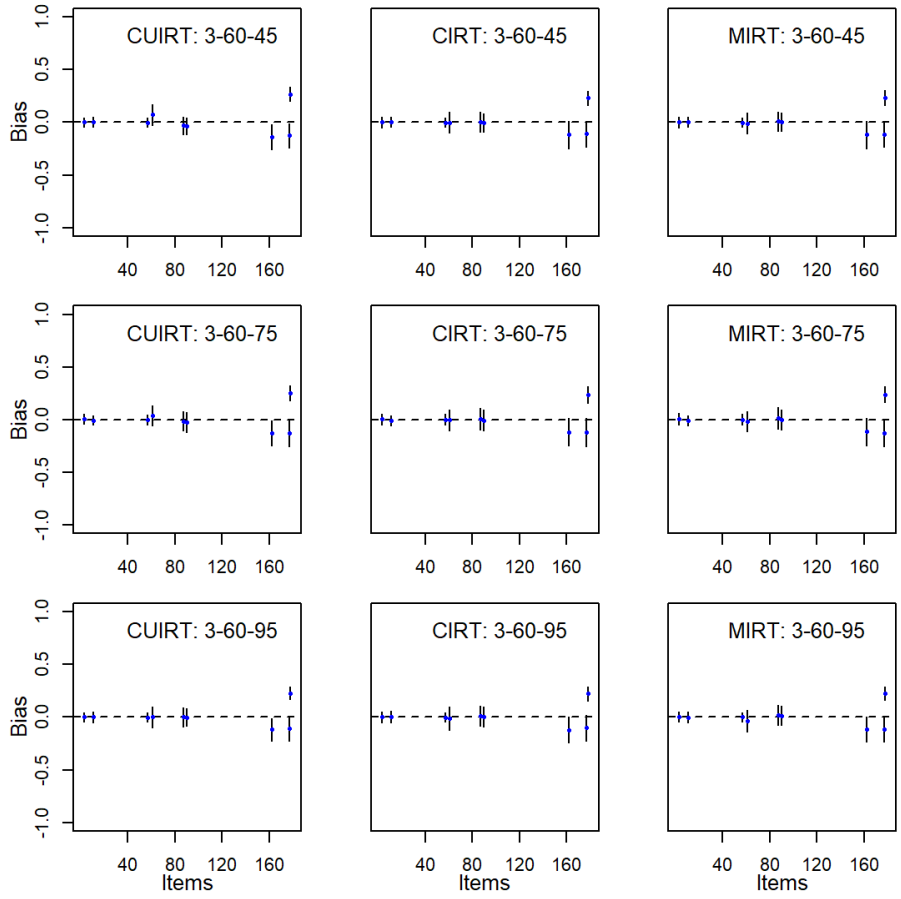


Figure D.11

Bias of d1-Parameter for the 4 Domain, 40 Items per Domain Tests: Multiple Groups

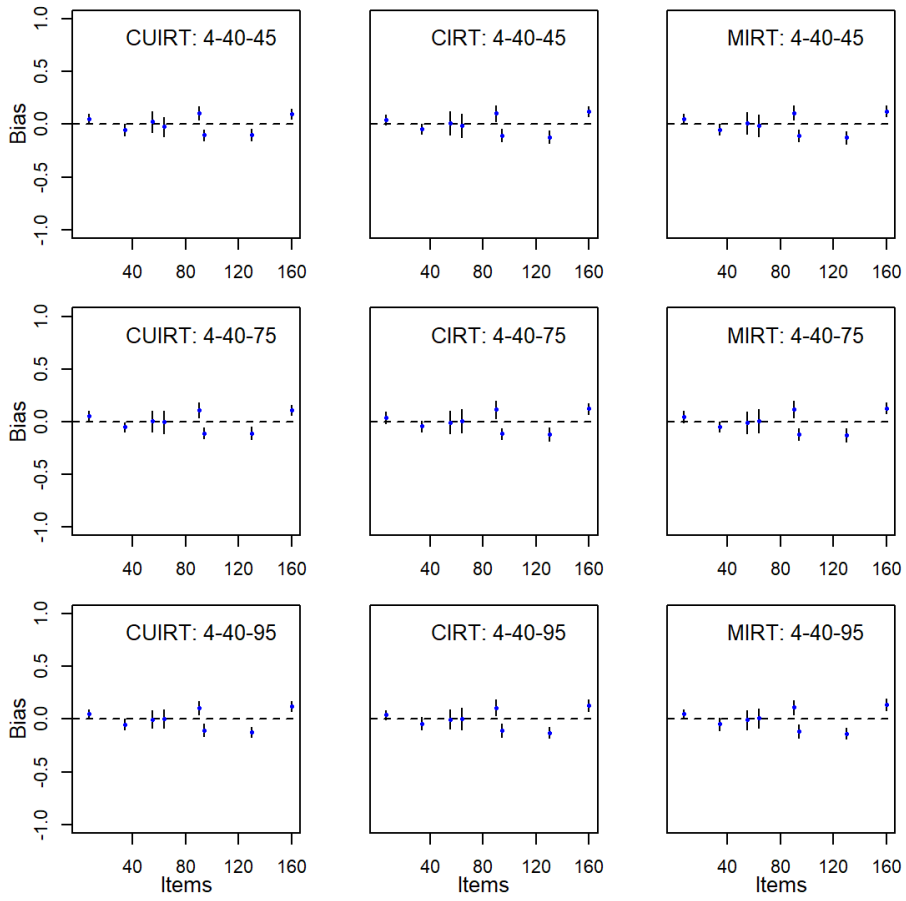


Figure D.12

Bias of d1-Parameter for the 4 Domain, 60 Items per Domain Tests: Multiple Groups

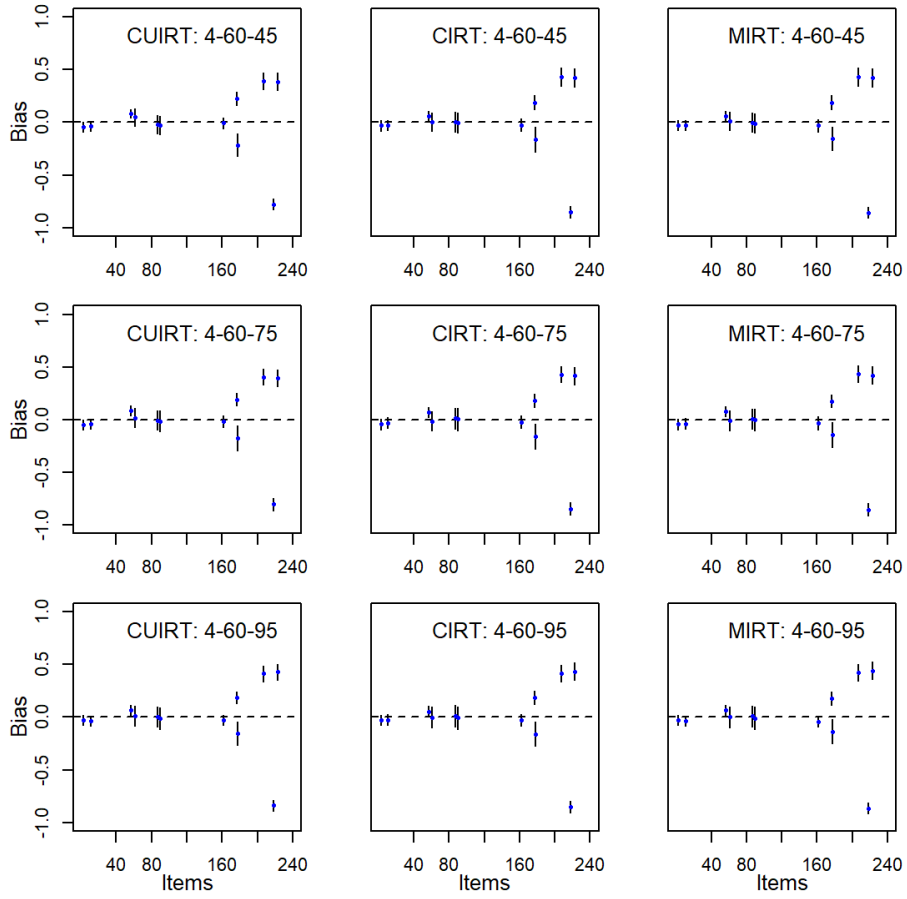


Figure D.13

Bias of d2-Parameter for the 3 Domain, 40 Items per Domain Tests: Multiple Groups

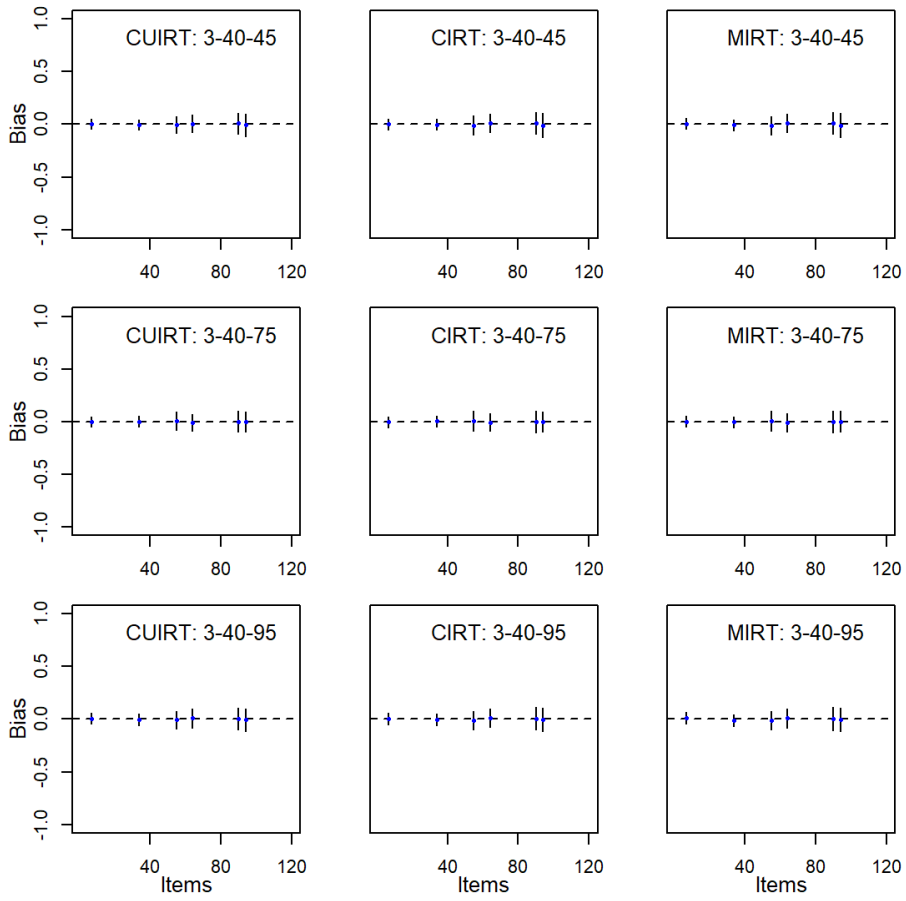


Figure D.14

Bias of d2-Parameter for the 3 Domain, 60 Items per Domain Tests: Multiple Groups

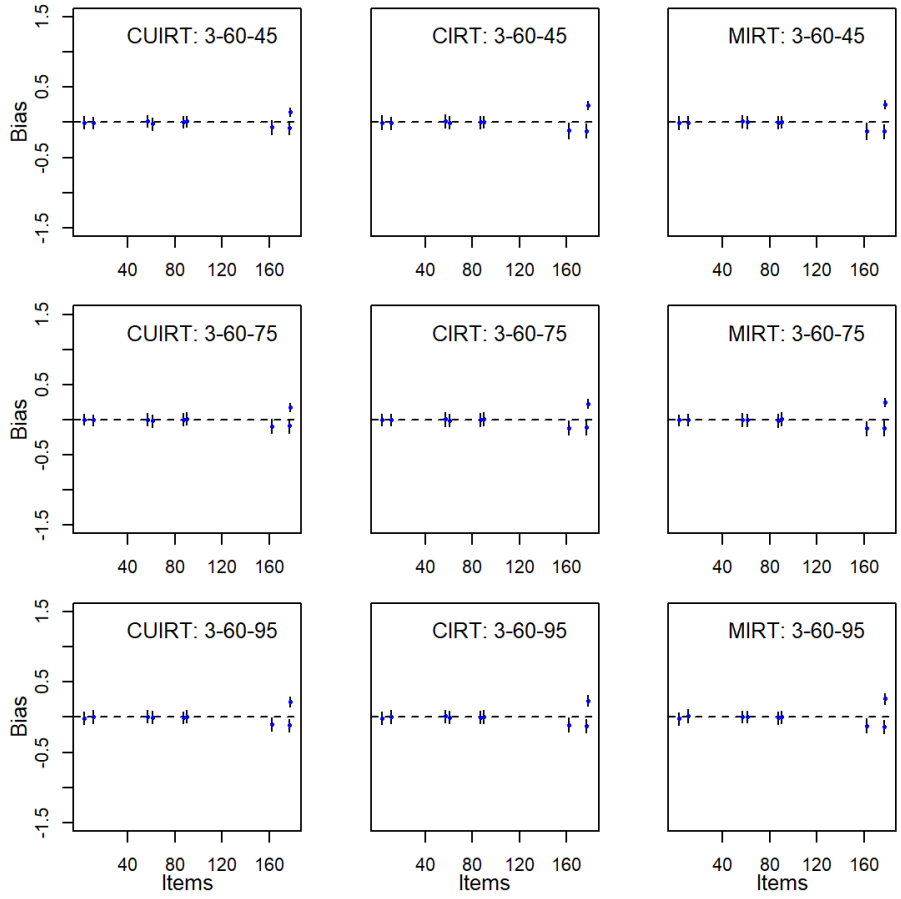


Figure D.15

Bias of d2-Parameter for the 4 Domain, 40 Items per Domain Tests: Multiple Groups

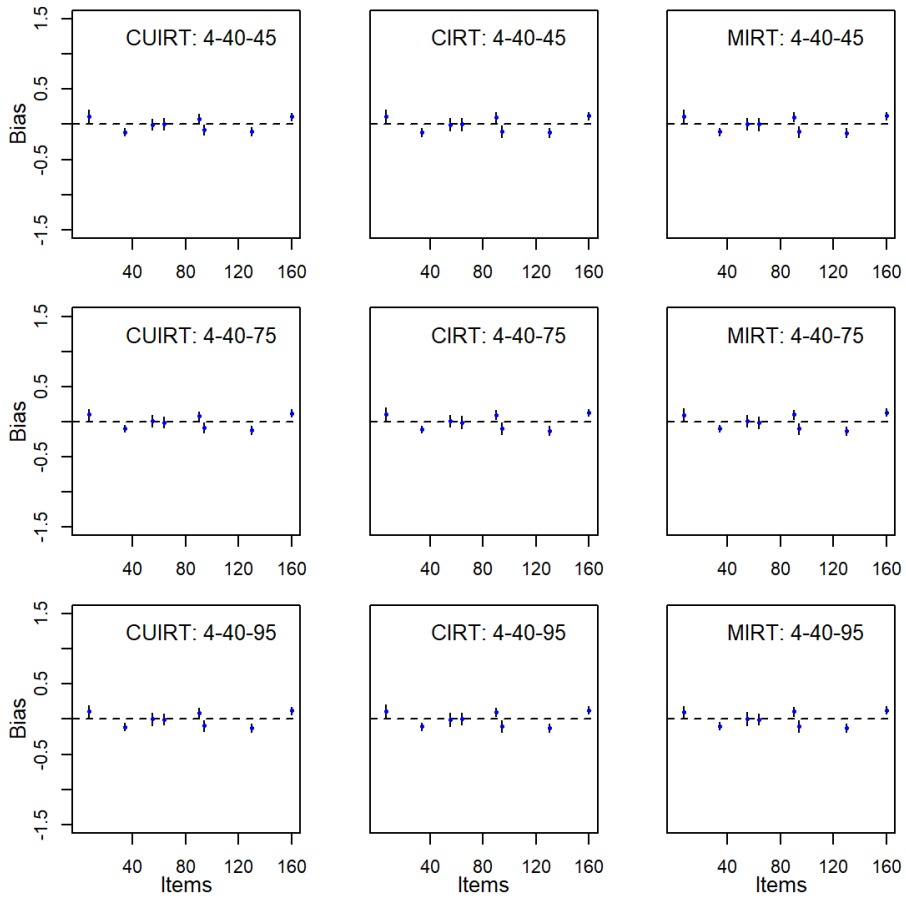
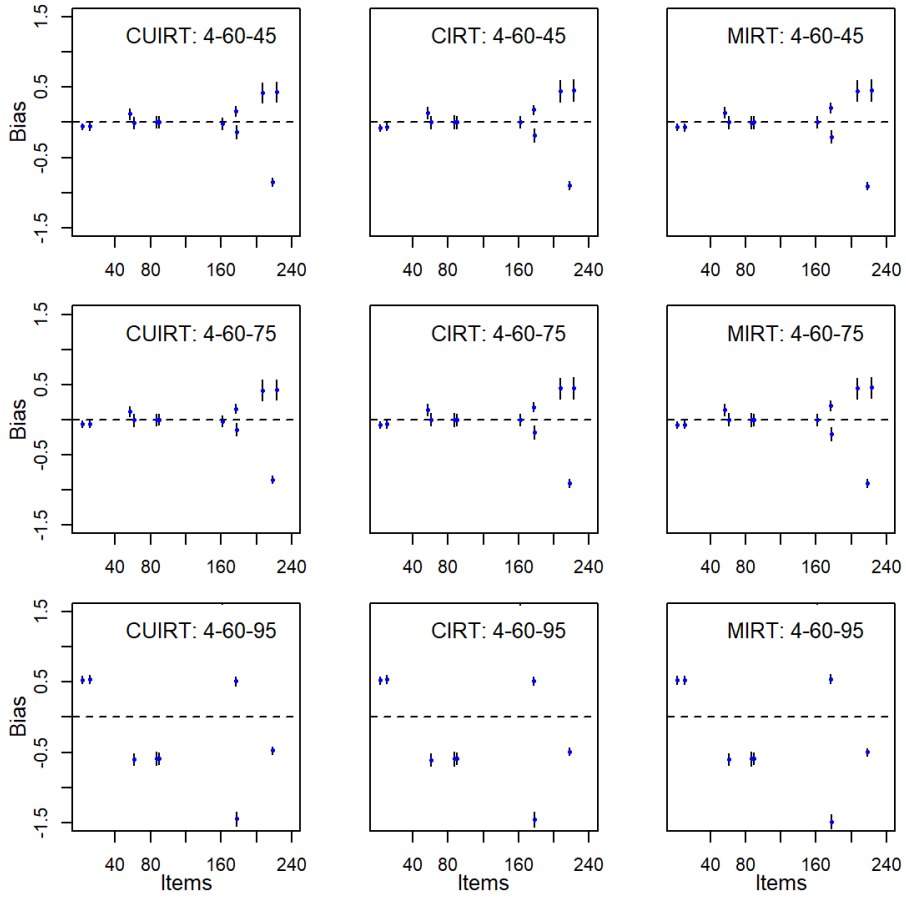


Figure D.16

Bias of d2-Parameter for the 4 Domain, 60 Items per Domain Tests: Multiple Groups



Appendix E

Study 1 Item Parameter ABS and RMSE: Single Groups

Figure E.1

Item Difficulty Absolute Bias for the 3 Domain, 5 Items per Domain Tests

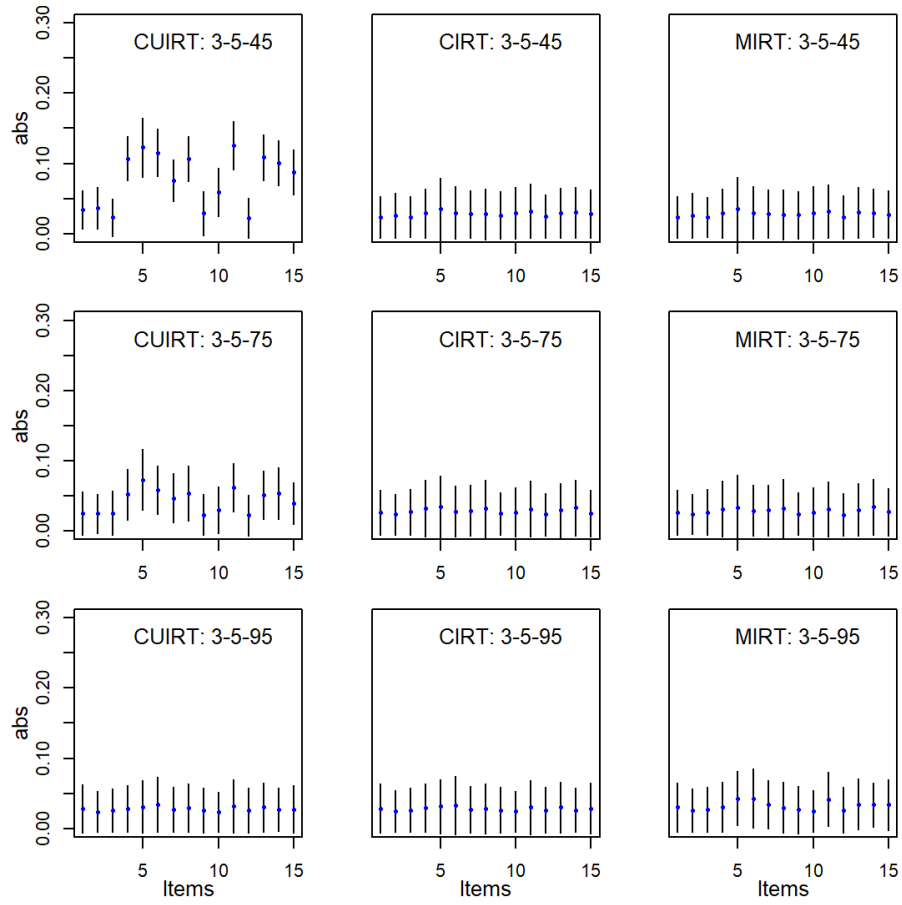


Figure E.2

Item Difficulty Absolute Bias for the 3 Domain, 10 Items per Domain Tests

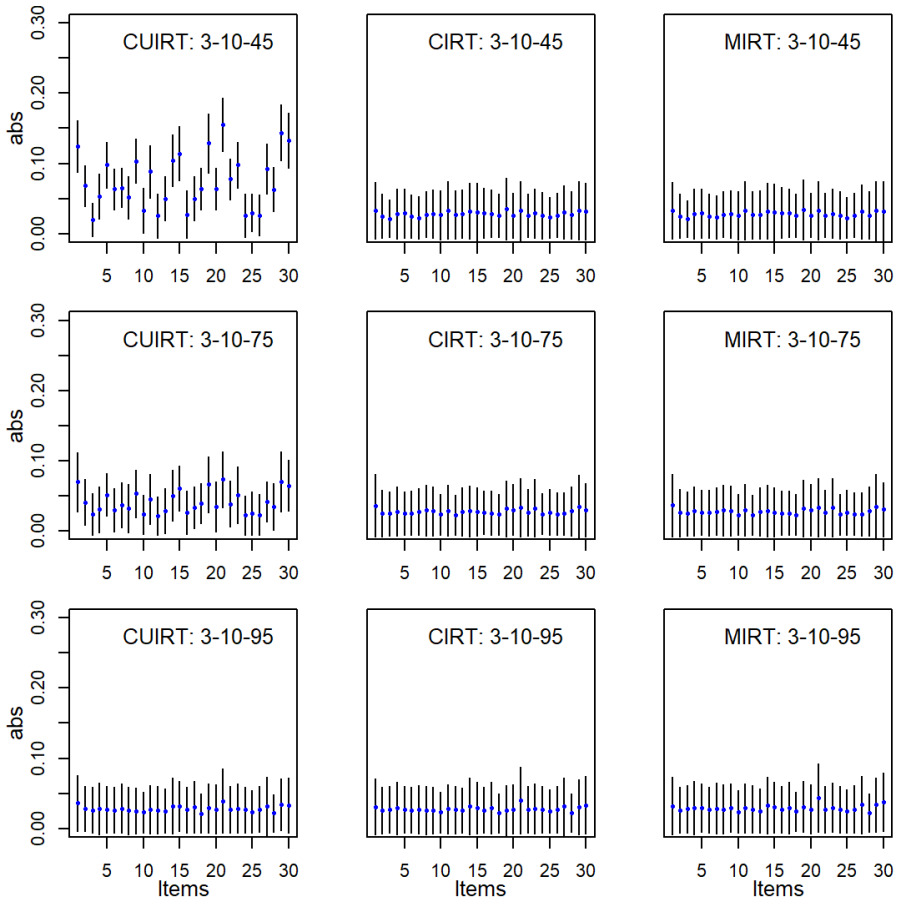


Figure E.3

Item Difficulty Absolute Bias for the 3 Domain, 15 Items per Domain Tests

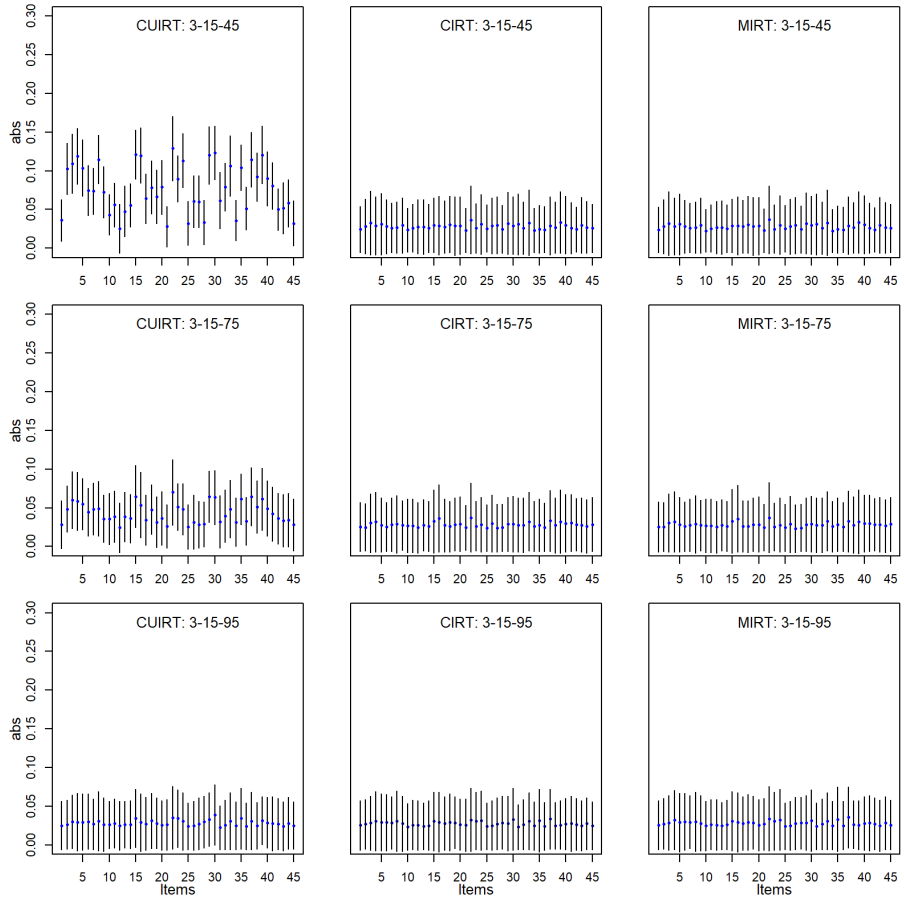


Figure E.4

Item Difficulty Absolute Bias for the 5 Domain, 5 Items per Domain Tests

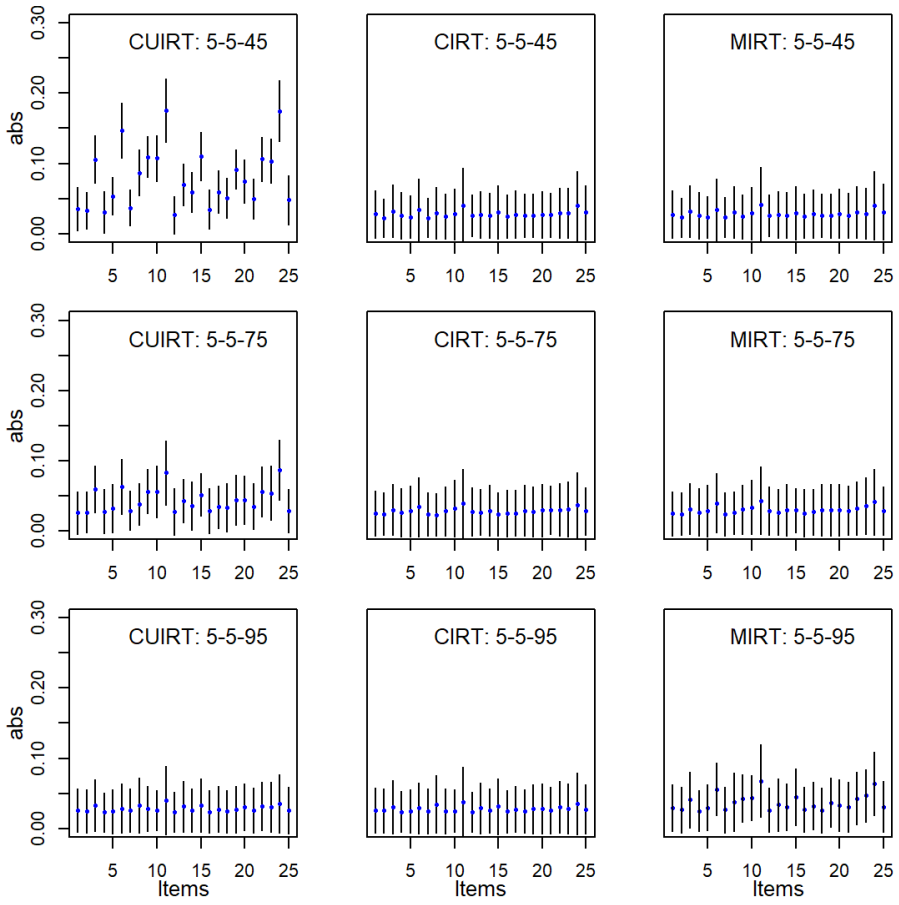


Figure E.5
Item Difficulty Absolute Bias for the 5 Domain, 10 Items per Domain Tests

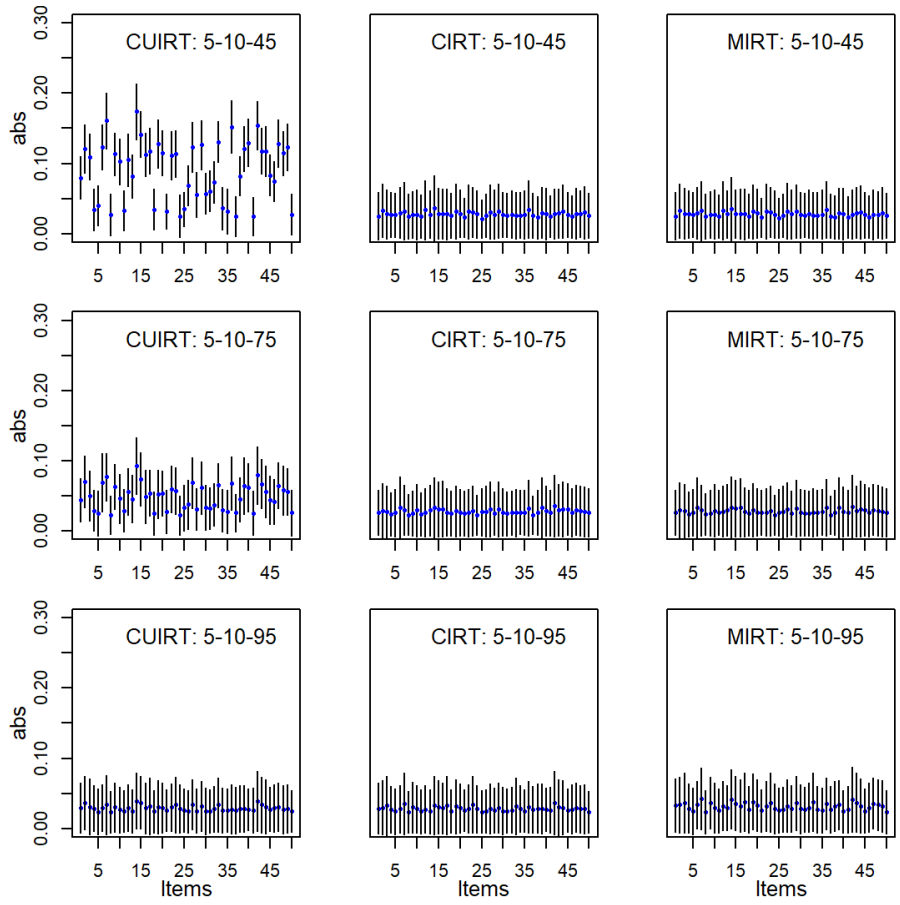
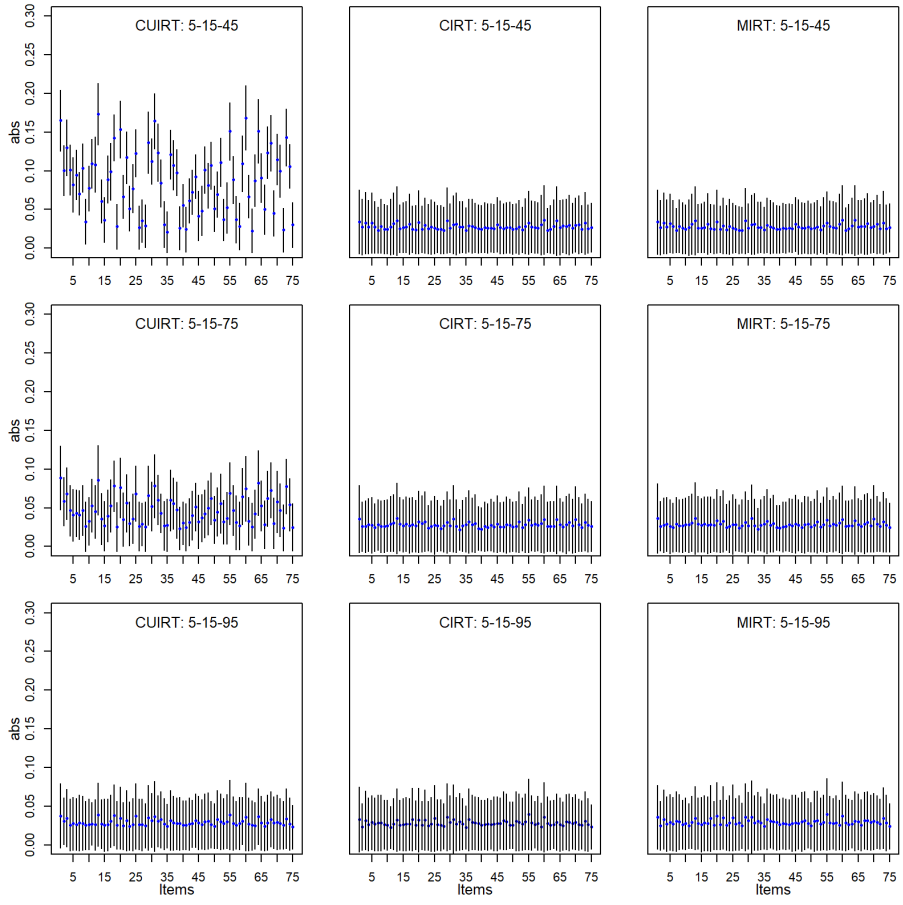


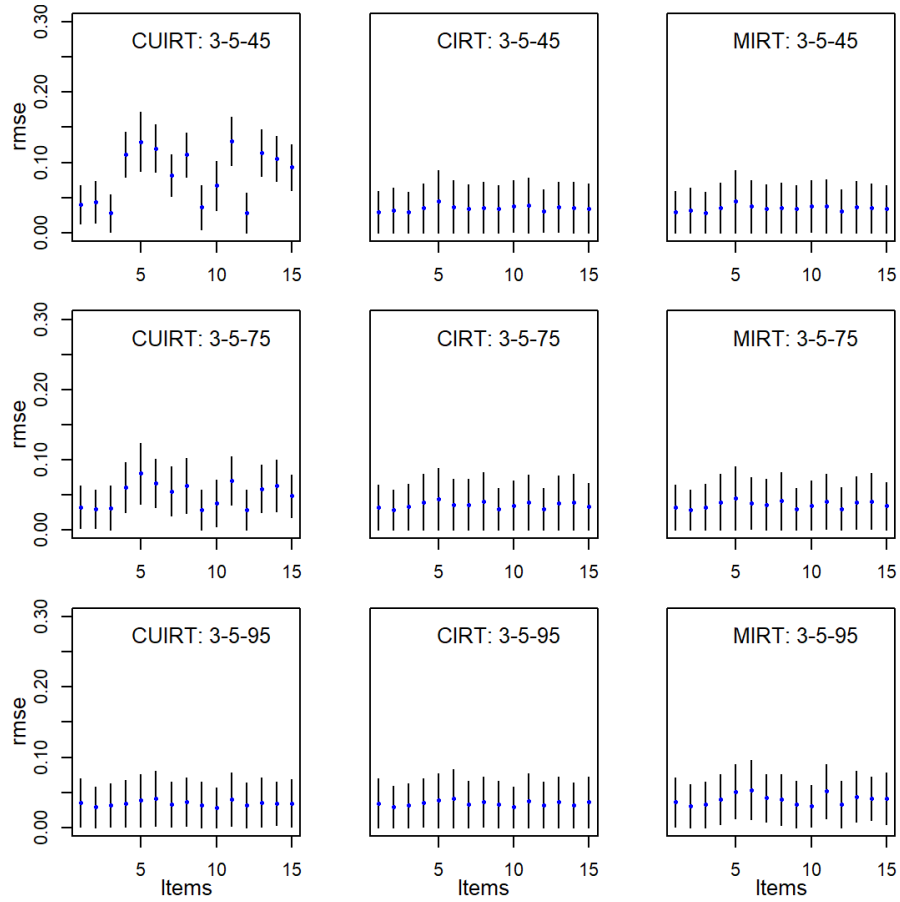
Figure E.6

Item Difficulty Absolute Bias for the 5 Domain, 15 Items per Domain Tests



E.3 RMSE: Three-Subdomain Test Conditions

Figure E.7
Item Difficulty RMSE for the 3 Domain, 5 Items per Domain Tests



E.4 RMSE: Five-Subdomain Test Conditions

Figure E.8

Item Difficulty RMSE for the 3 Domain, 10 Items per Domain Tests

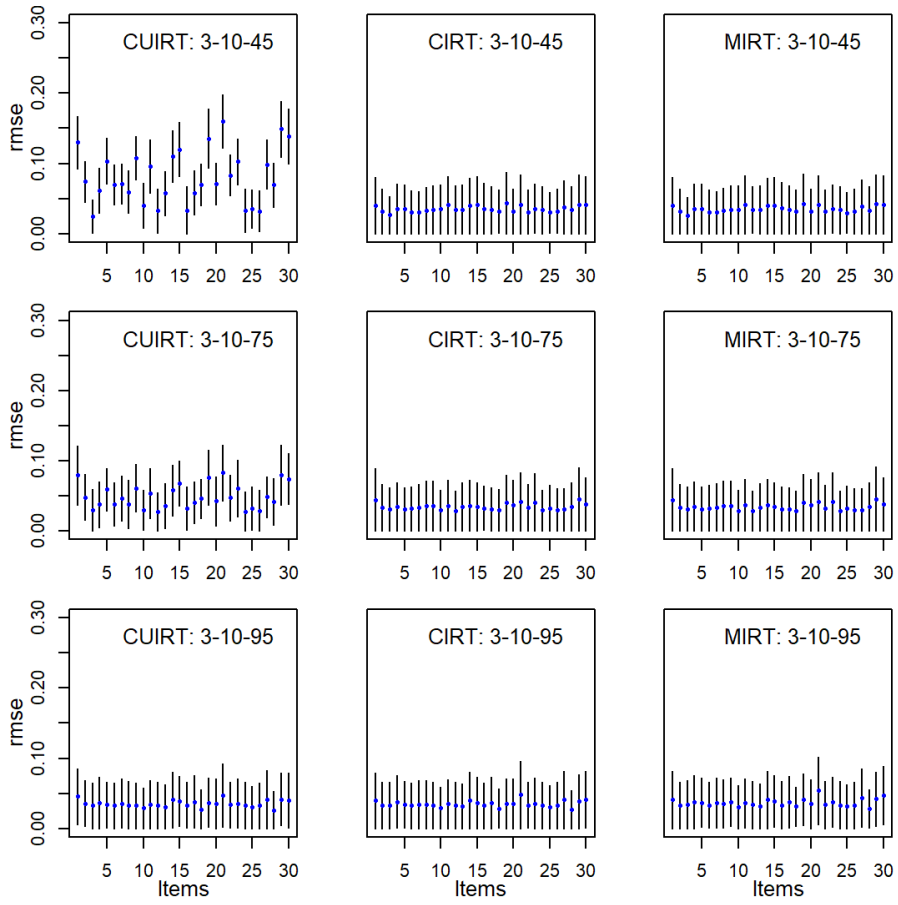


Figure E.9

Item Difficulty RMSE for the 3 Domain, 15 Items per Domain Tests

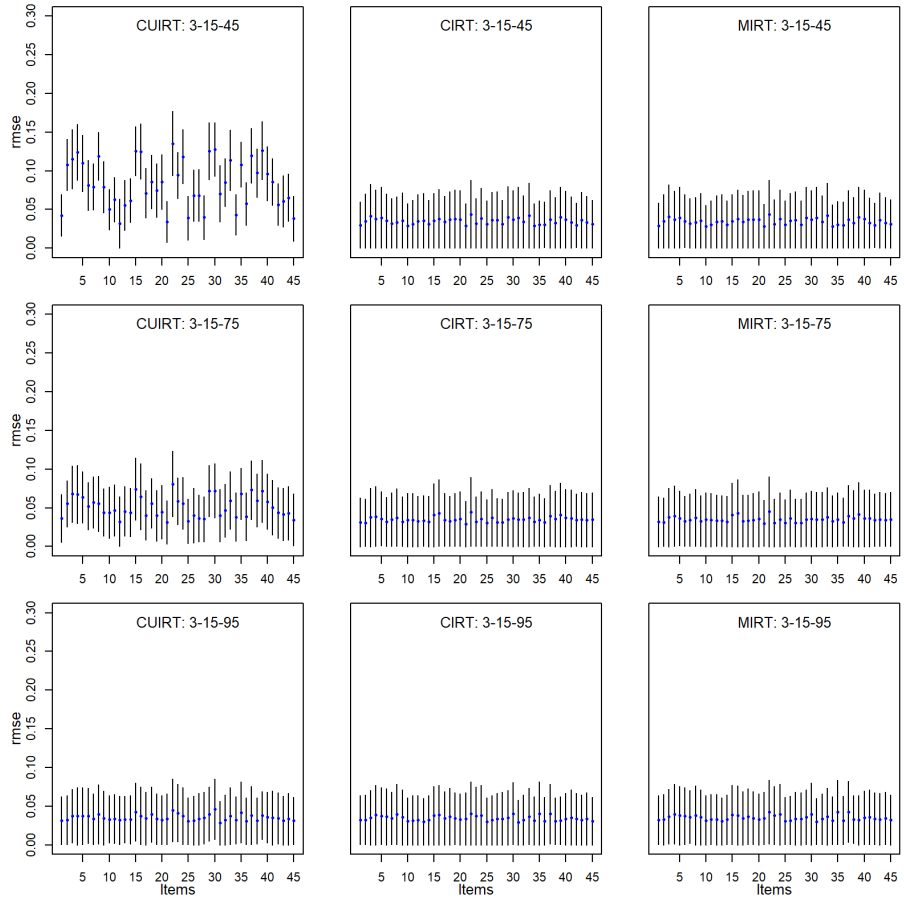


Figure E.10

Item Difficulty RMSE for the 5 Domain, 5 Items per Domain Tests

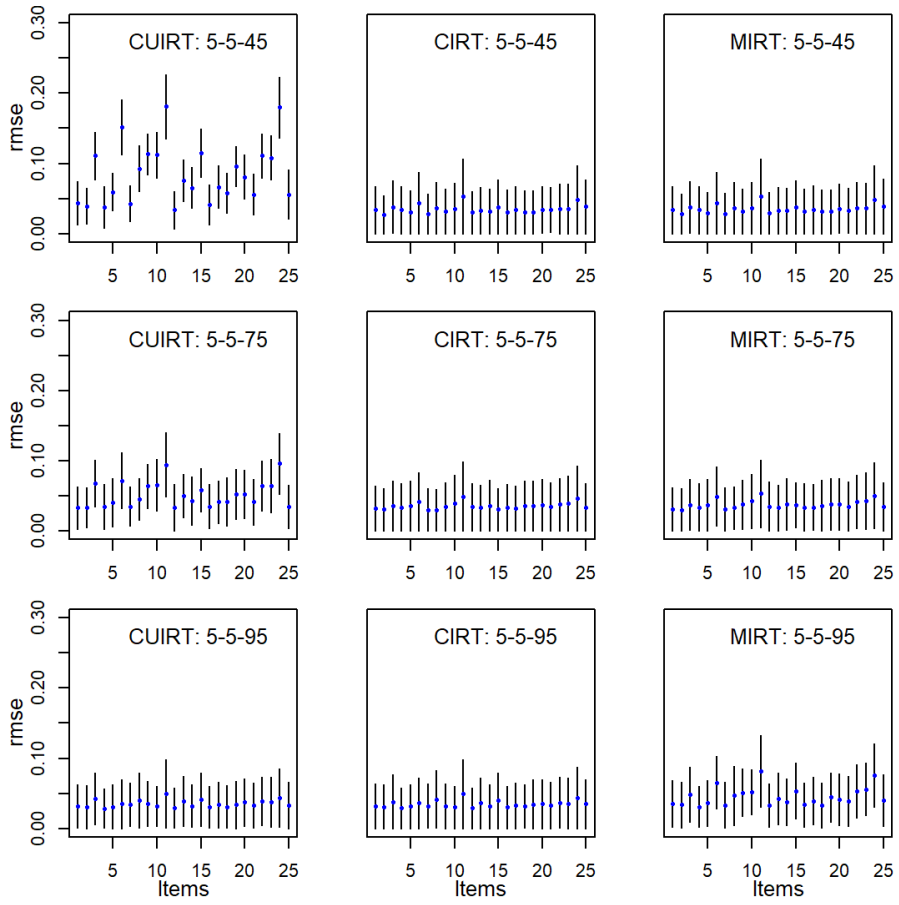


Figure E.11
Item Difficulty RMSE for the 5 Domain, 10 Items per Domain Tests

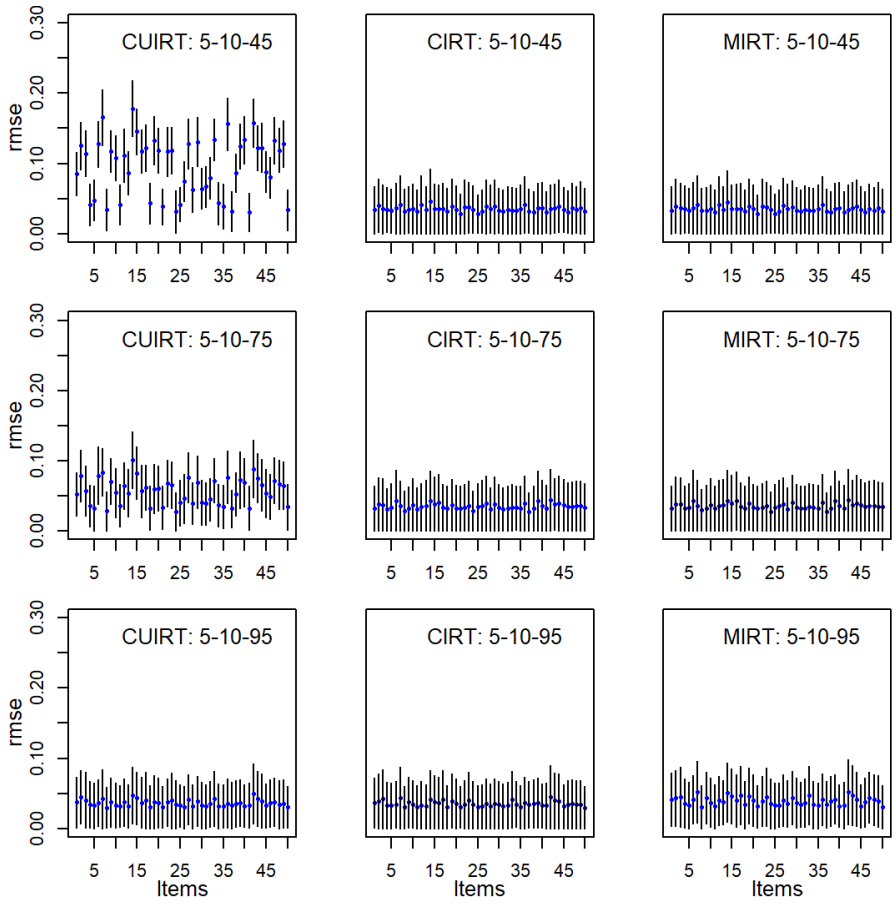
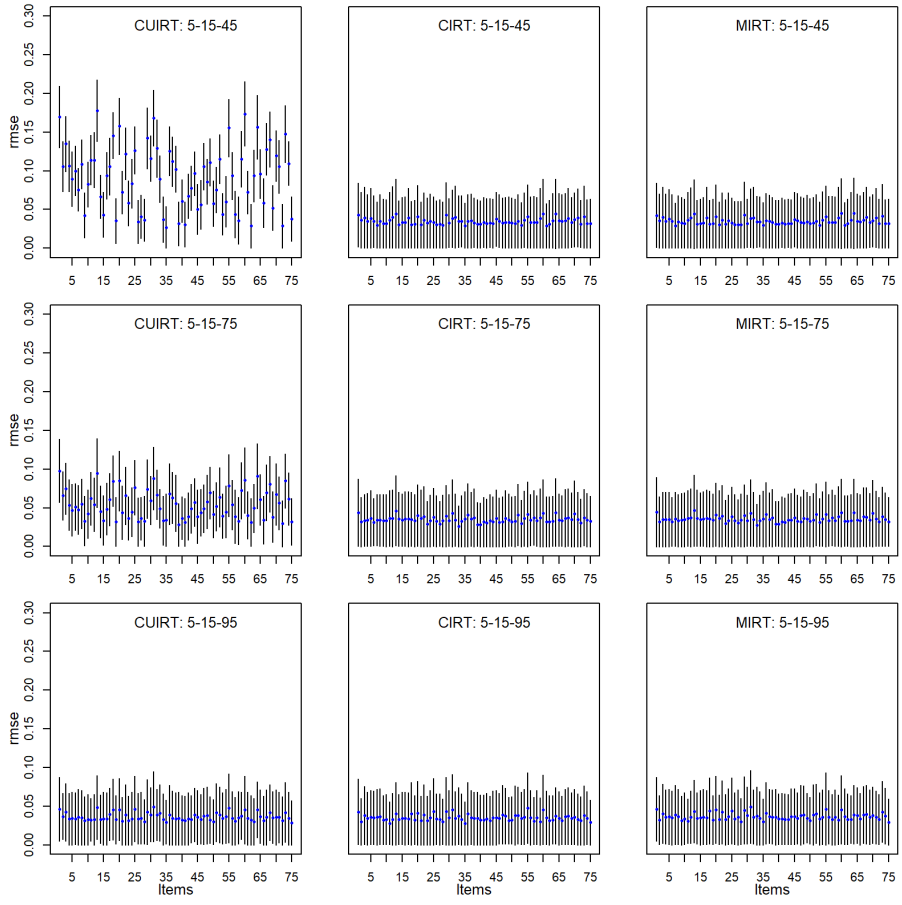


Figure E.12

Item Difficulty RMSE for the 5 Domain, 15 Items per Domain Tests



Appendix F

Study 1 Item Parameter ABS and RMSE: Multiple Groups

Figure F.1

Item Difficulty Absolute Bias for the 3 Domain, 5 Items per Domain Tests

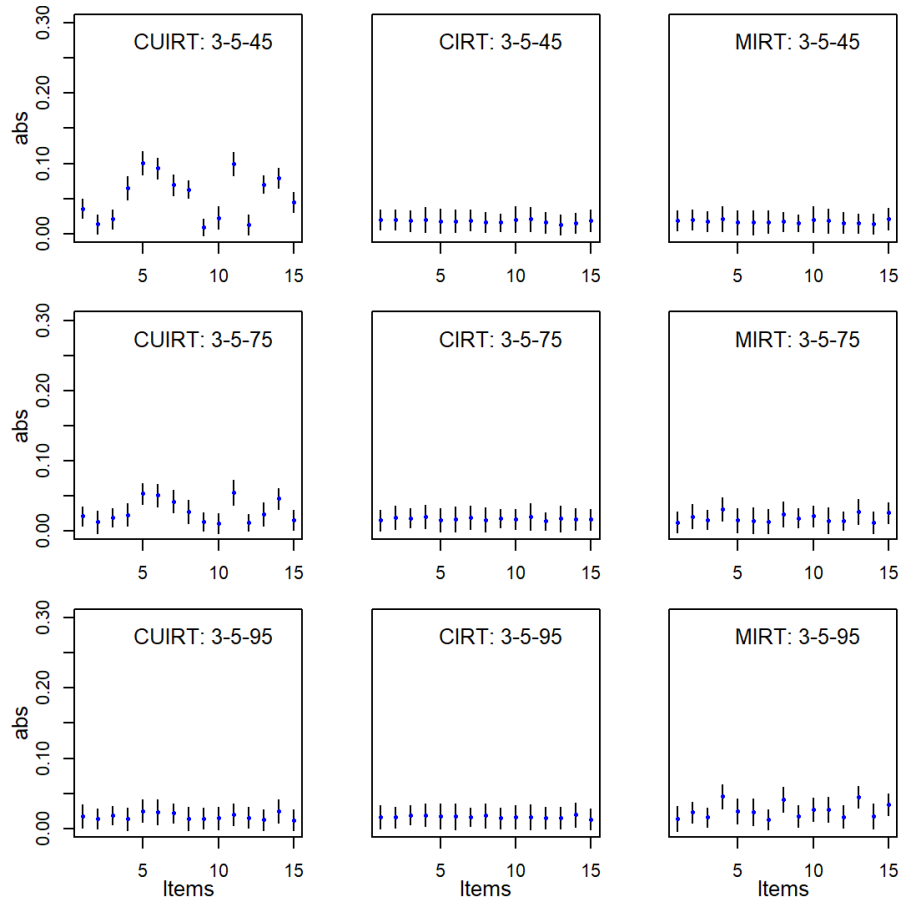


Figure F.2

Item Difficulty Absolute Bias for the 3 Domain, 10 Items per Domain Tests

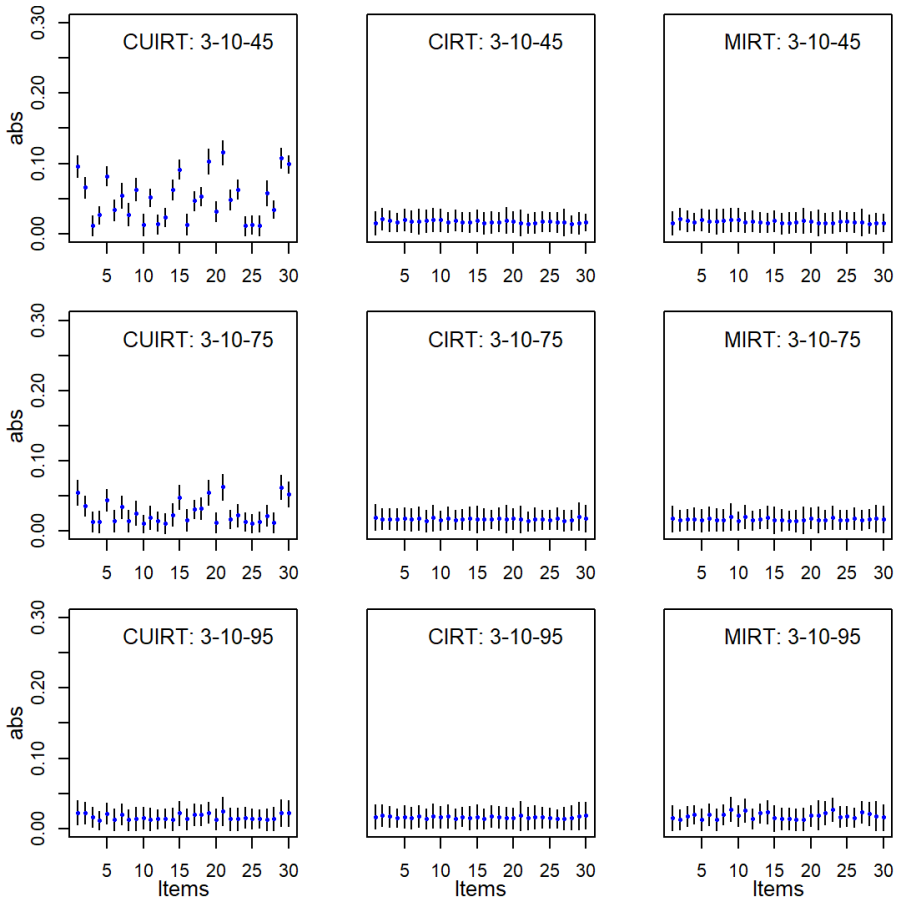


Figure F.3

Item Difficulty Absolute Bias for the 3 Domain, 15 Items per Domain Tests

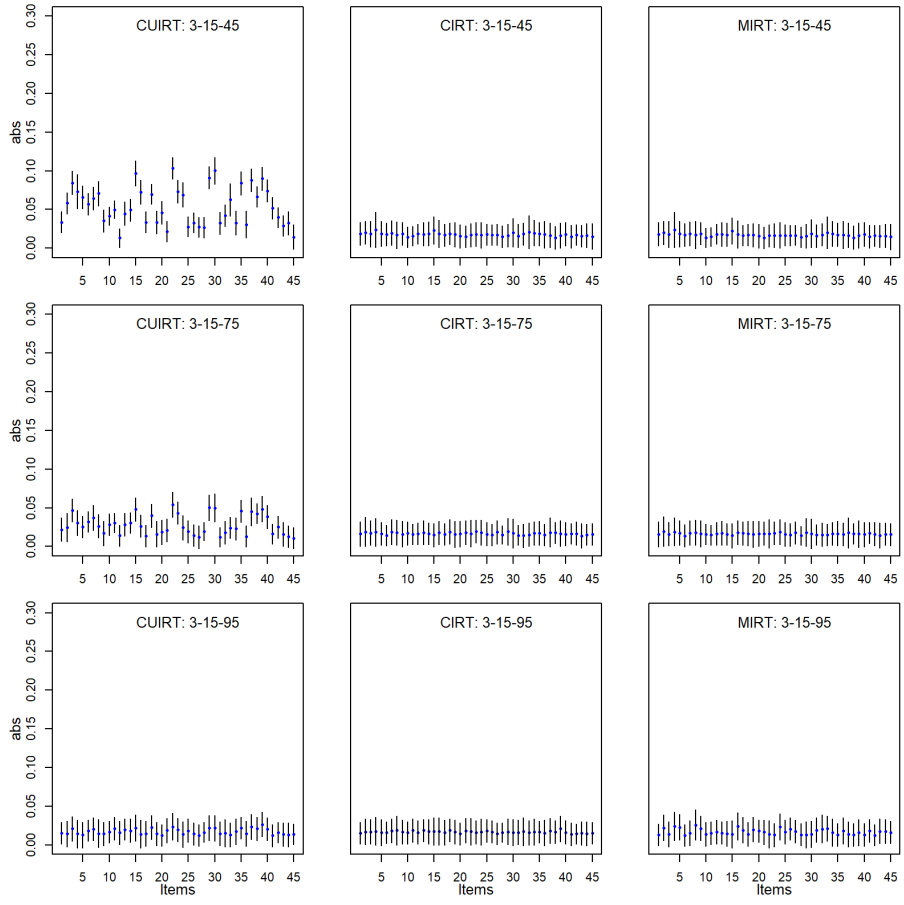


Figure F.4

Item Difficulty Absolute Bias for the 5 Domain, 5 Items per Domain Tests

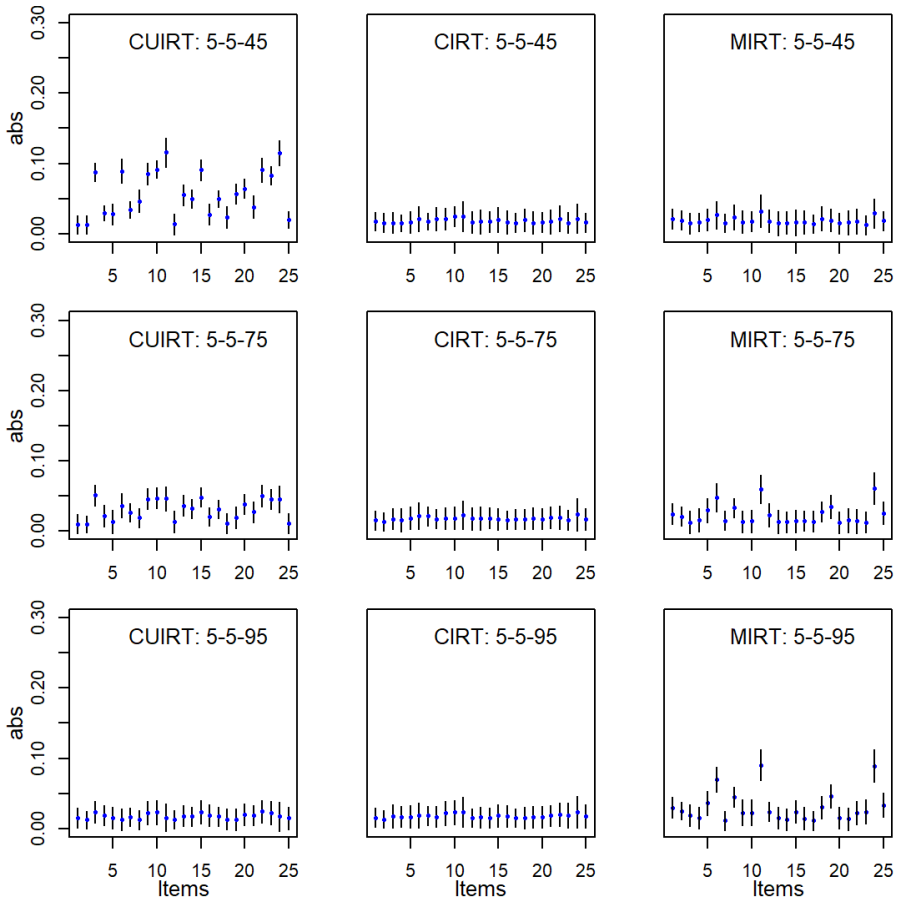


Figure F.5
Item Difficulty Absolute Bias for the 5 Domain, 10 Items per Domain Tests

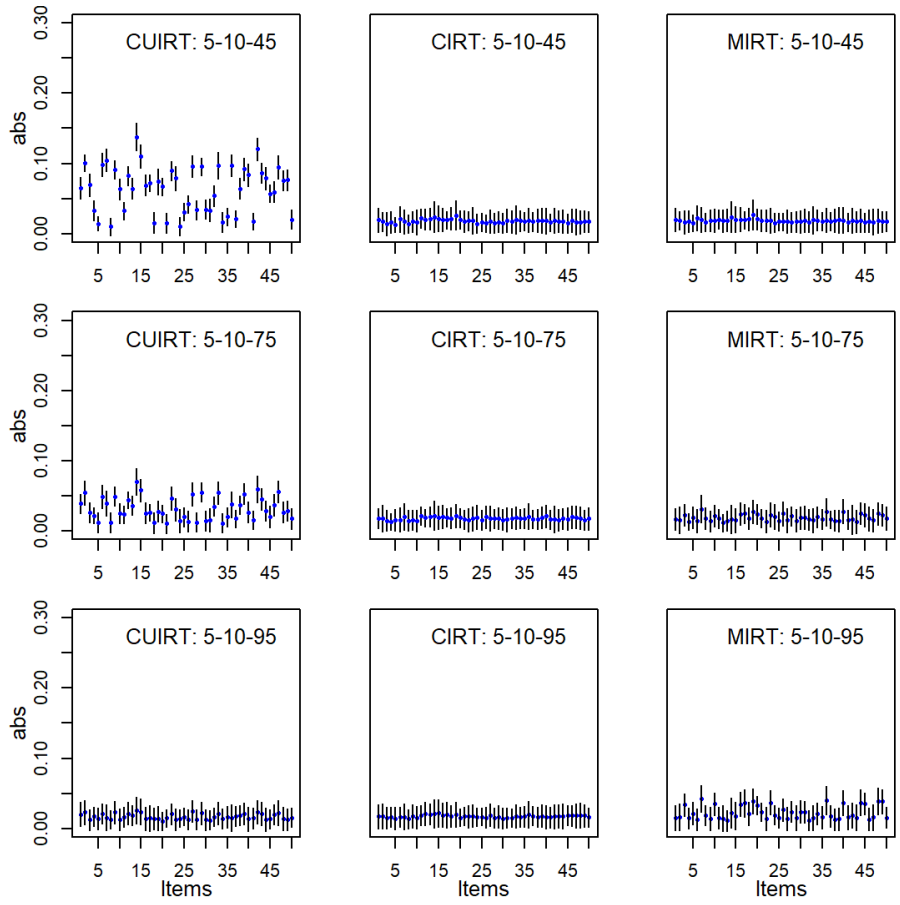
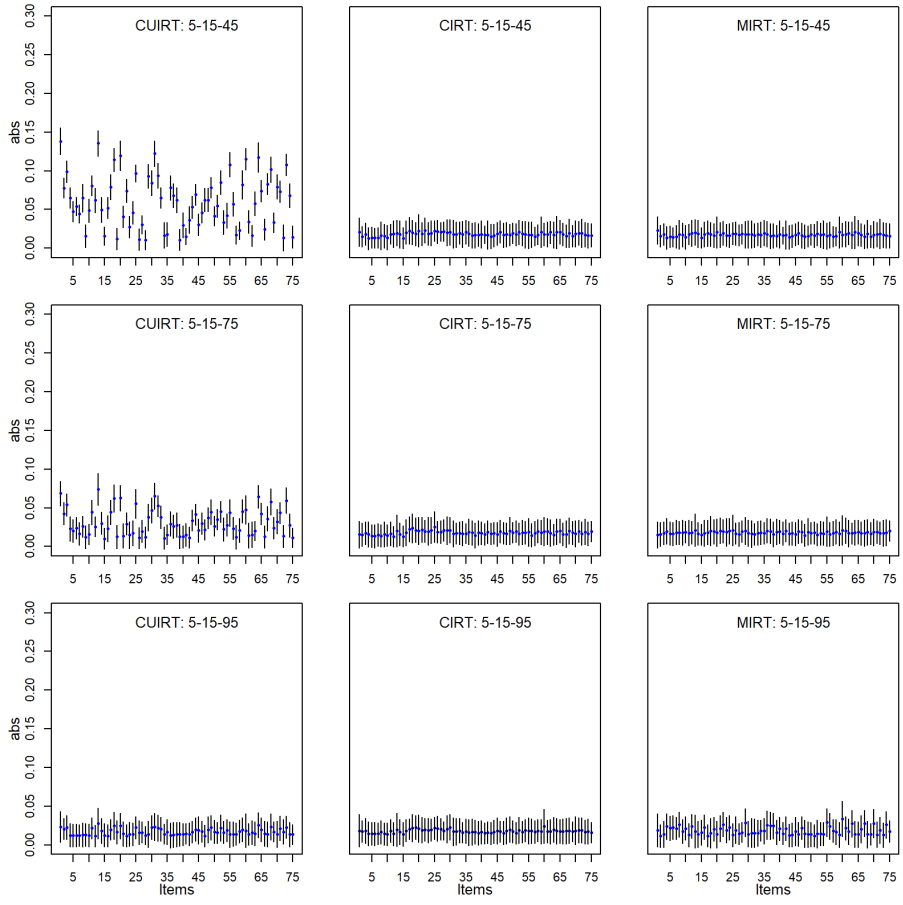


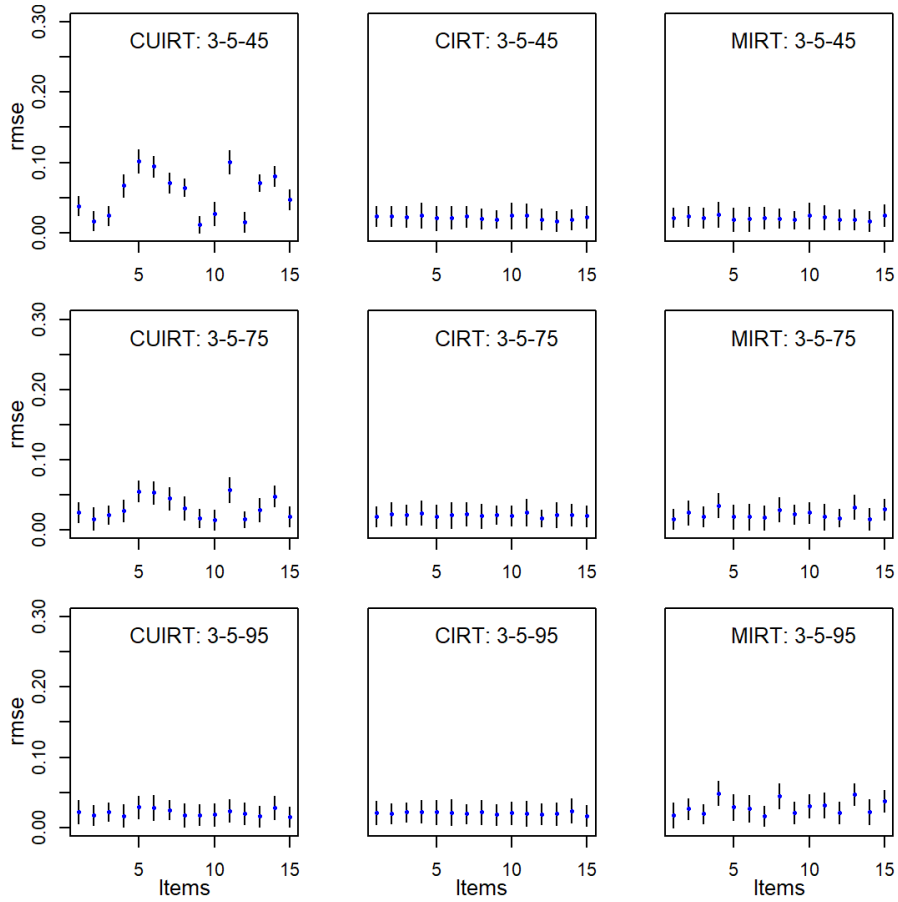
Figure F.6

Item Difficulty Absolute Bias for the 5 Domain, 15 Items per Domain Tests



F.3 RMSE: Three-Subdomain Test Conditions

Figure F.7
Item Difficulty RMSE for the 3 Domain, 5 Items per Domain Tests



F.4 RMSE: Five-Subdomain Test Conditions

Figure F.8

Item Difficulty RMSE for the 3 Domain, 10 Items per Domain Tests

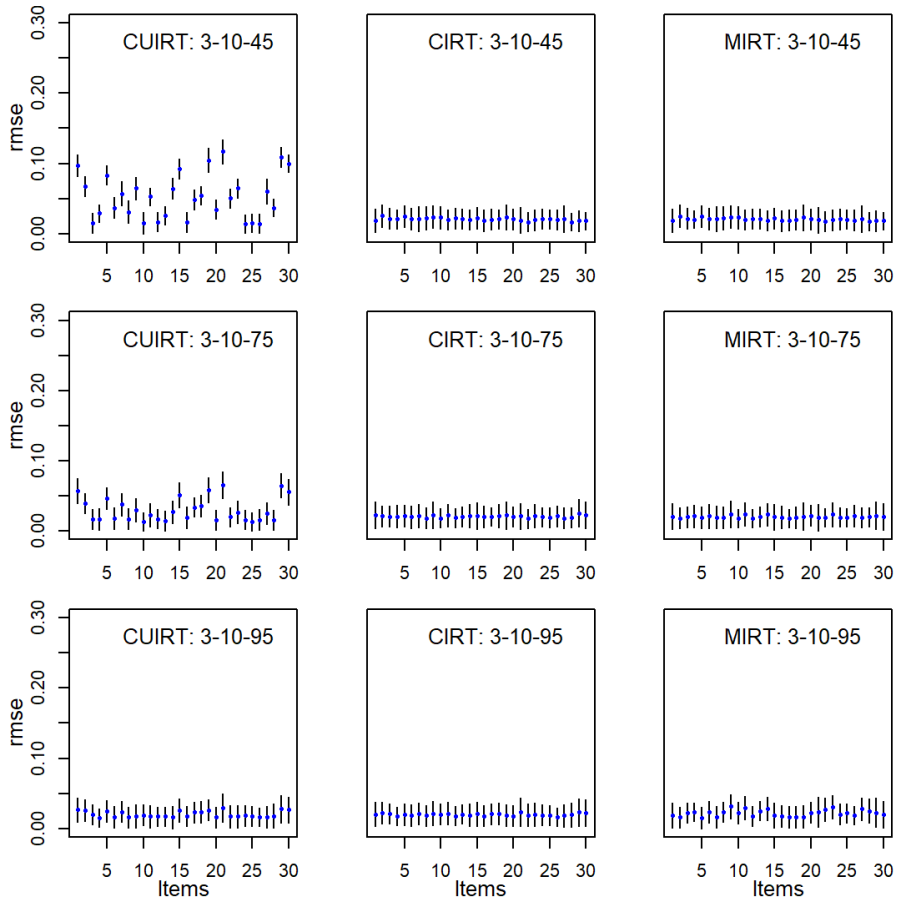


Figure F.9

Item Difficulty RMSE for the 3 Domain, 15 Items per Domain Tests

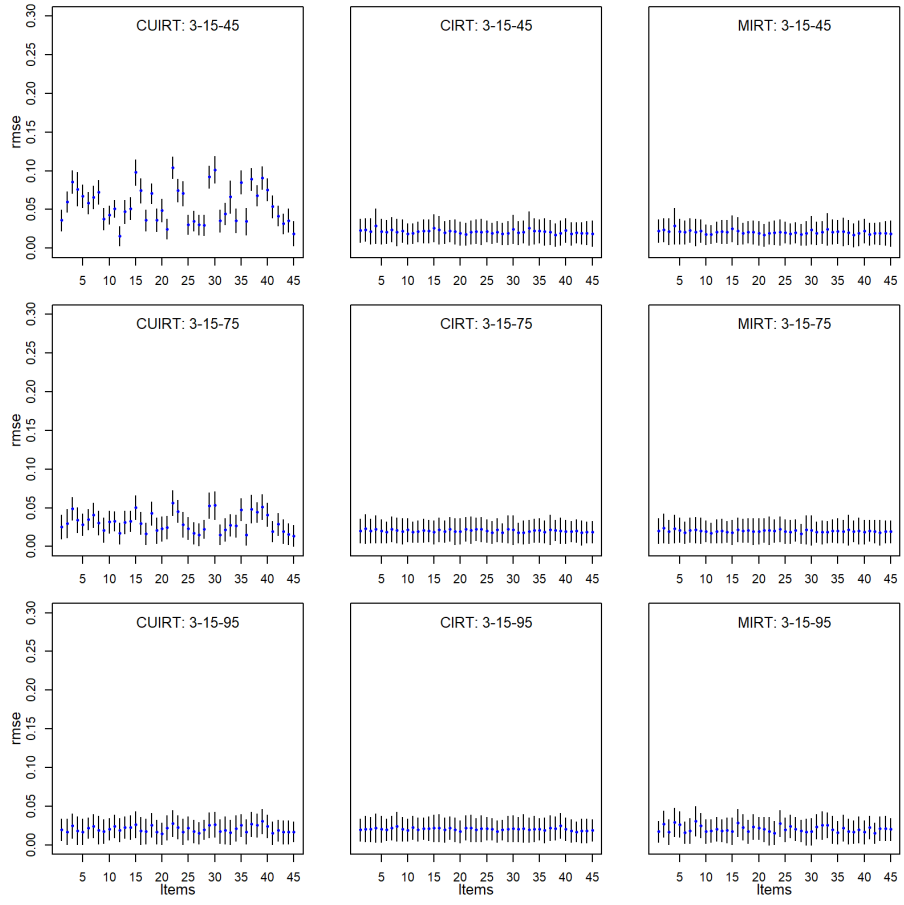


Figure F.10

Item Difficulty RMSE for the 5 Domain, 5 Items per Domain Tests

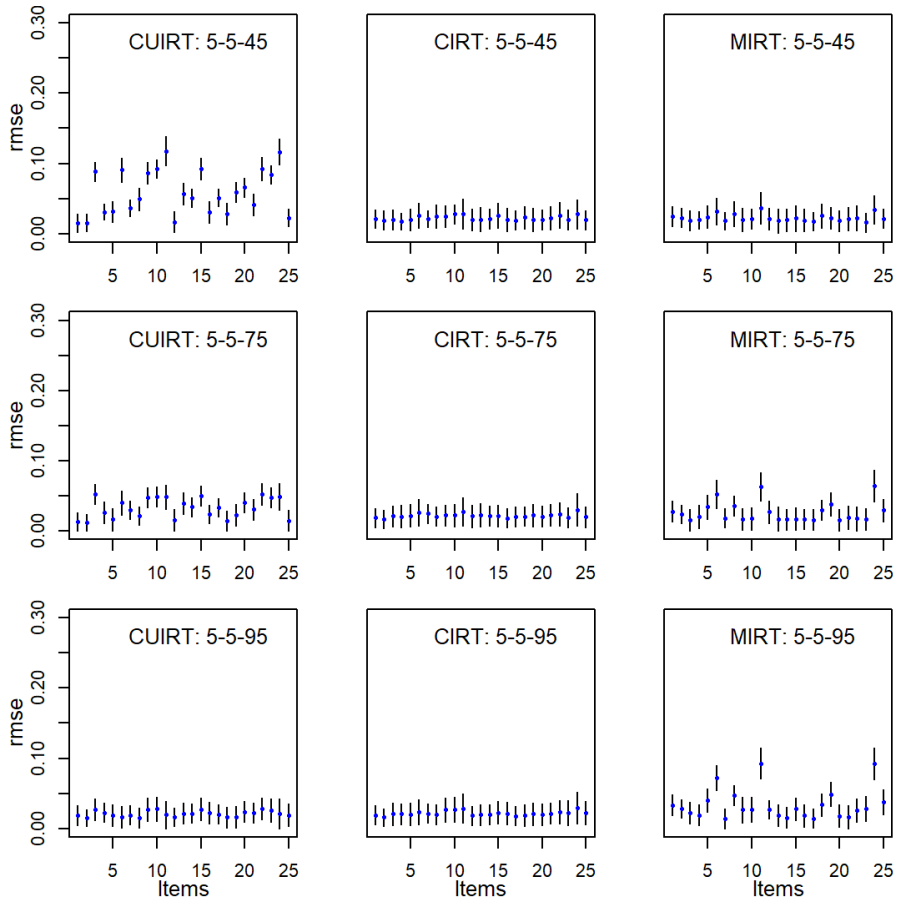


Figure F.11
Item Difficulty RMSE for the 5 Domain, 10 Items per Domain Tests

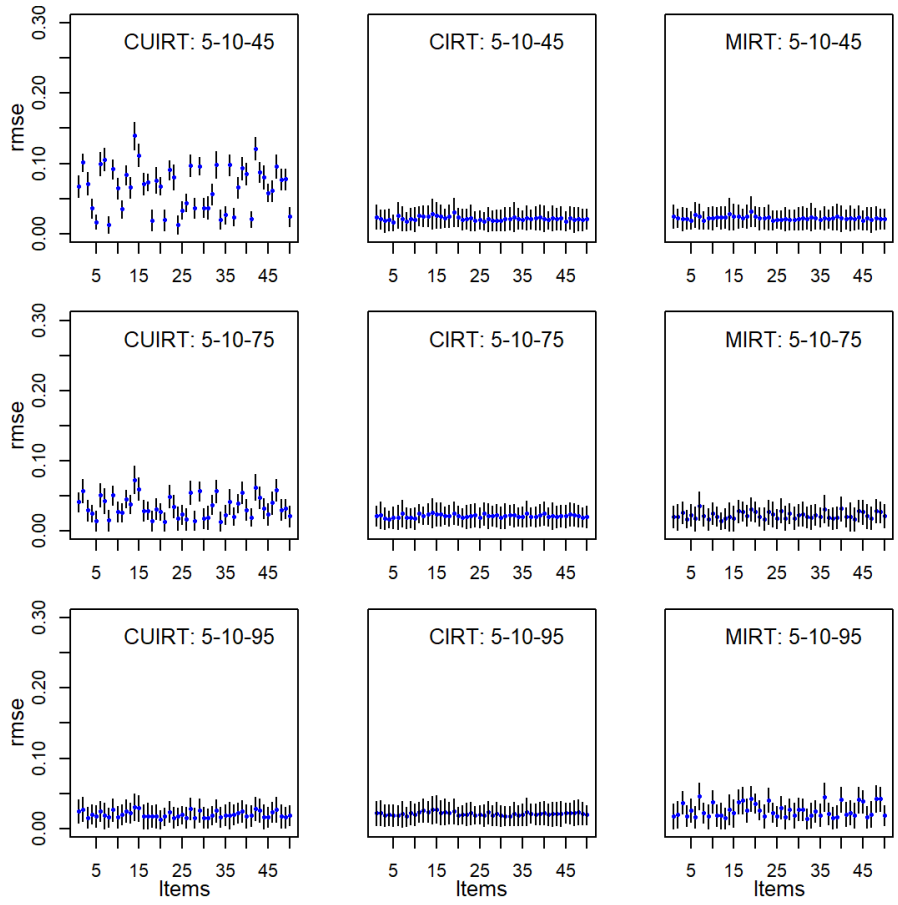
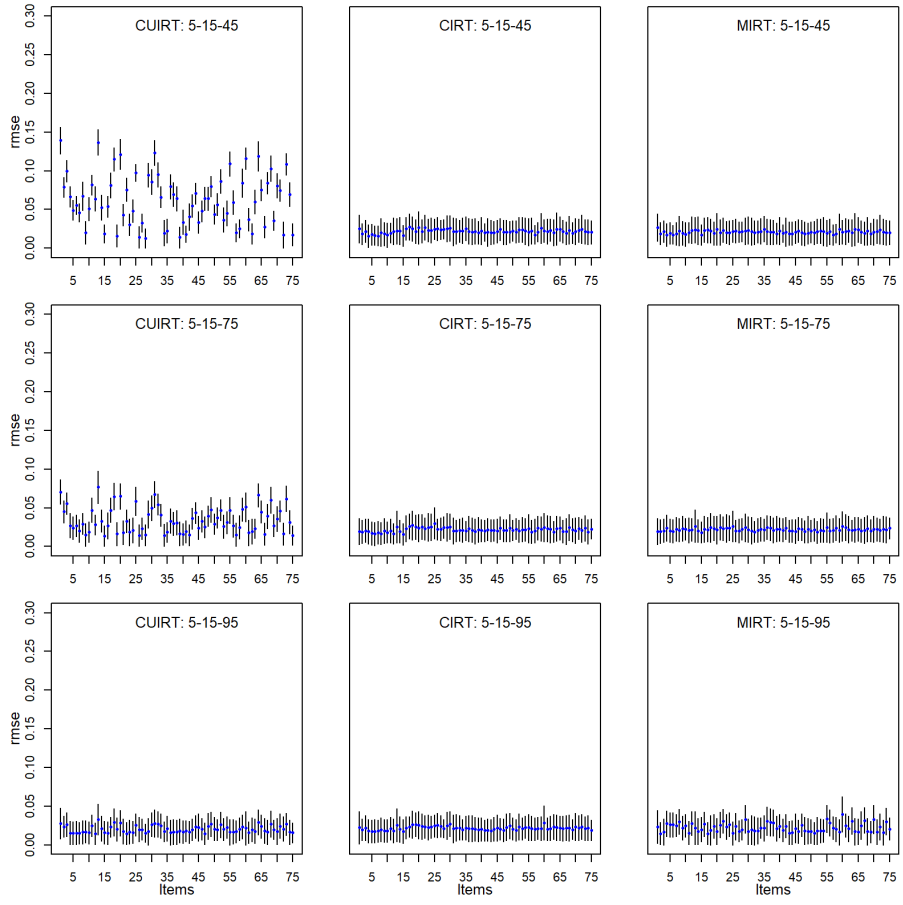


Figure F.12

Item Difficulty RMSE for the 5 Domain, 15 Items per Domain Tests



Appendix G

Study 2 Item Parameter ABS and RMSE: Single Groups

G.1 ABS

G.1.1 Item Discrimination

Figure G.1

Absolute Bias of a-Parameter for the 3 Domain, 40 Items per Domain Tests

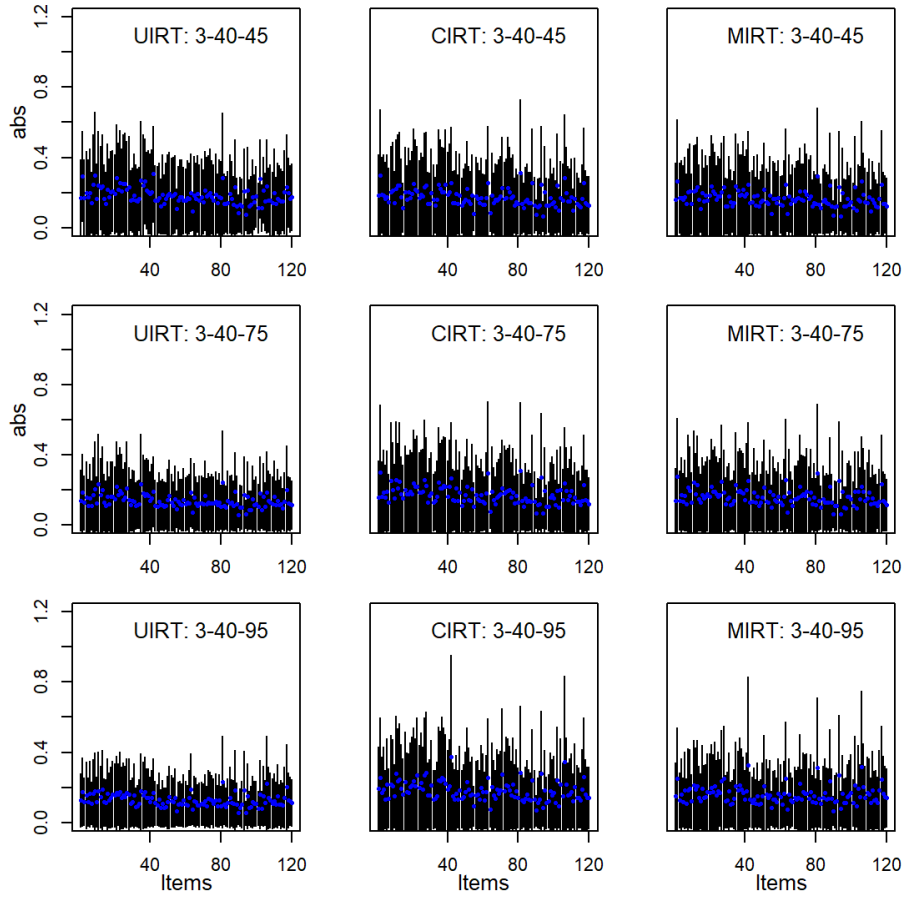


Figure G.2

Absolute Bias of a-Parameter for the 3 Domain, 60 Items per Domain Tests

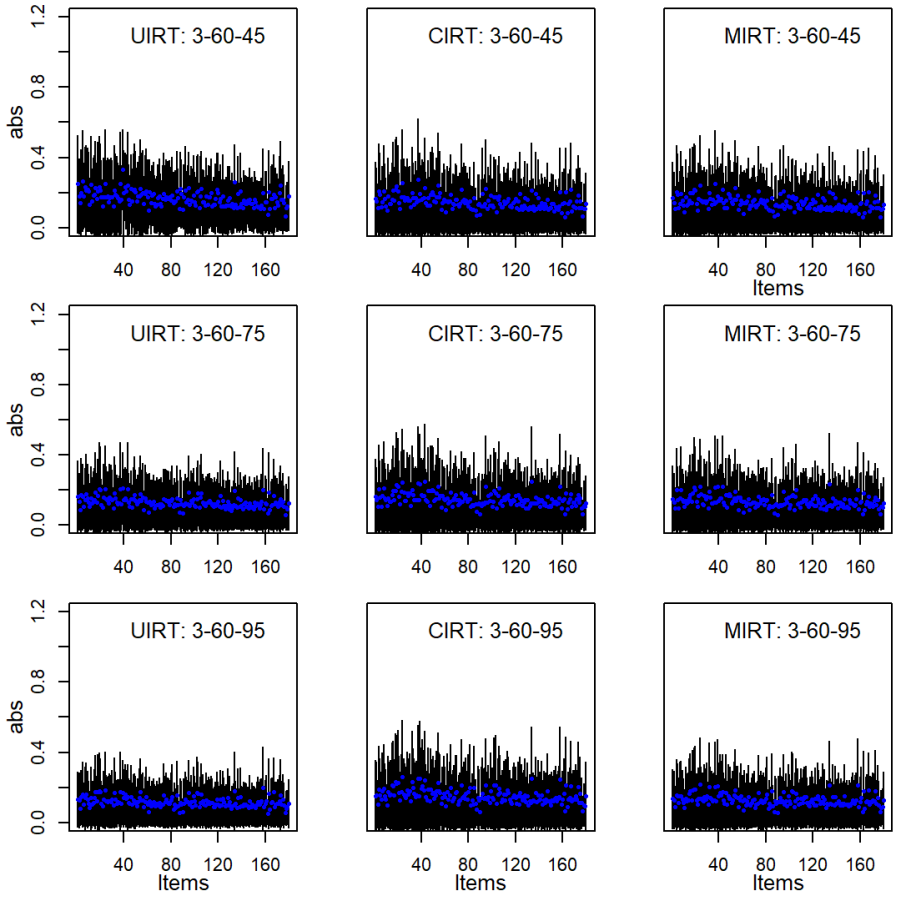


Figure G.3

Absolute Bias of a-Parameter for the 4 Domain, 40 Items per Domain Tests

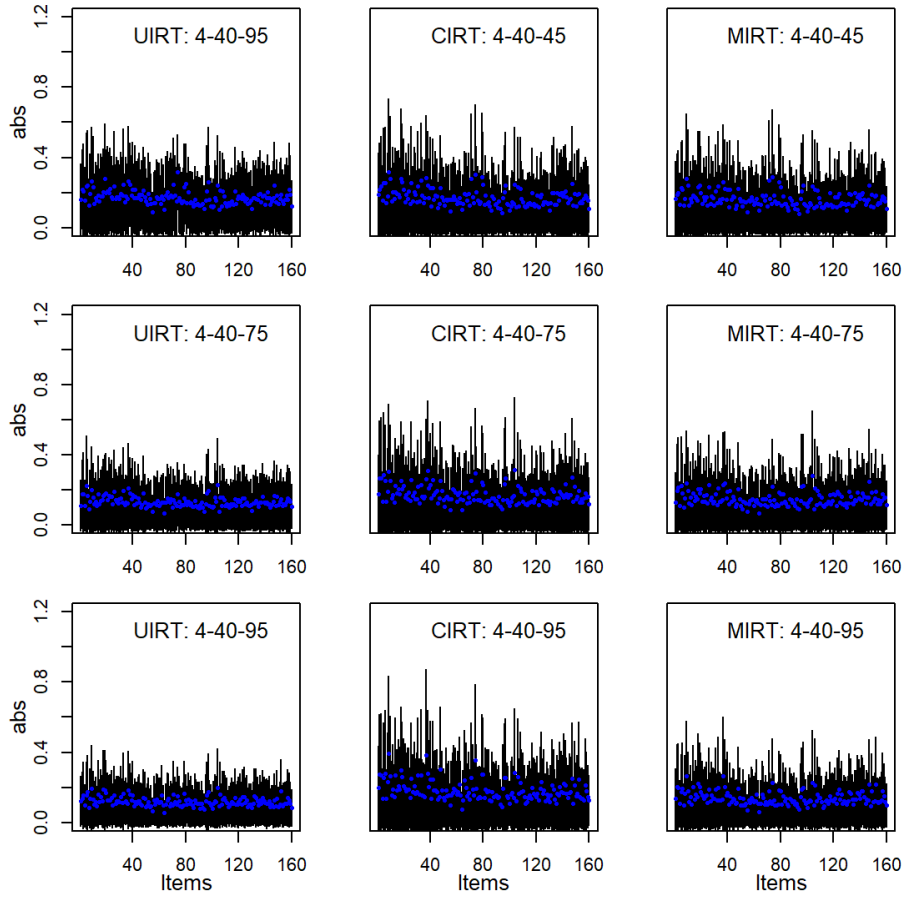
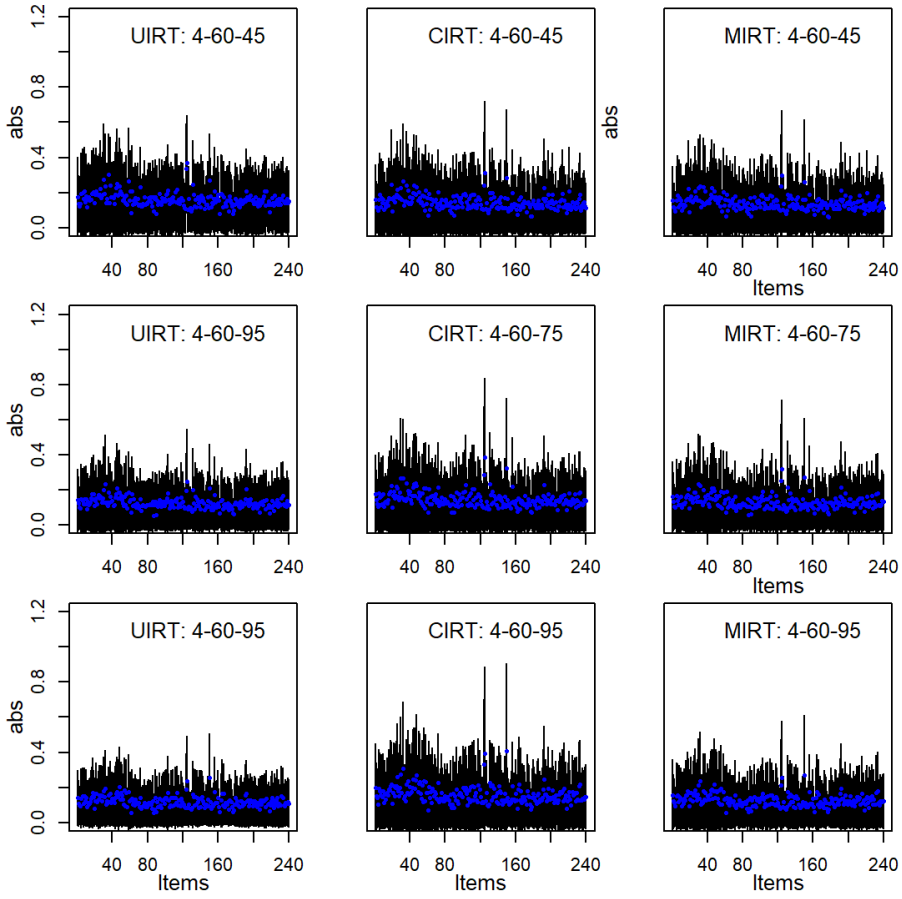


Figure G.4

Absolute Bias of a-Parameter for the 4 Domain, 60 Items per Domain Tests



G.1.2 Item Difficulty

Figure G.5
Absolute Bias of b-Parameter for the 3 Domain, 40 Items per Domain Tests

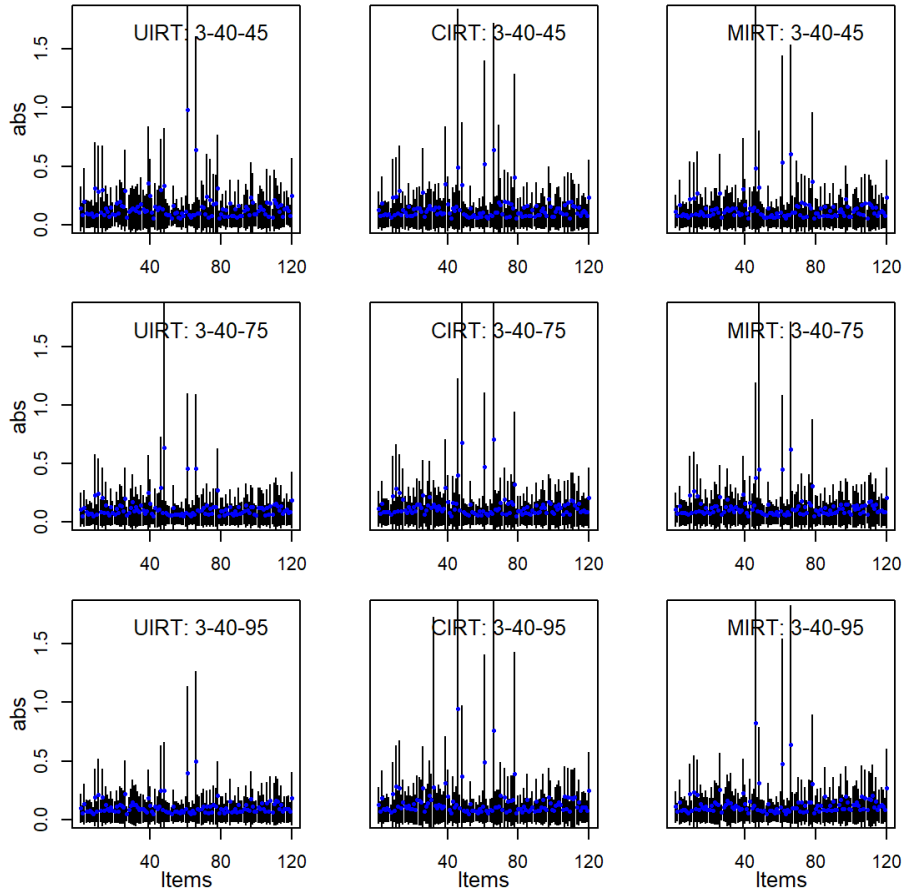


Figure G.6

Absolute Bias of b-Parameter for the 3 Domain, 60 Items per Domain Tests

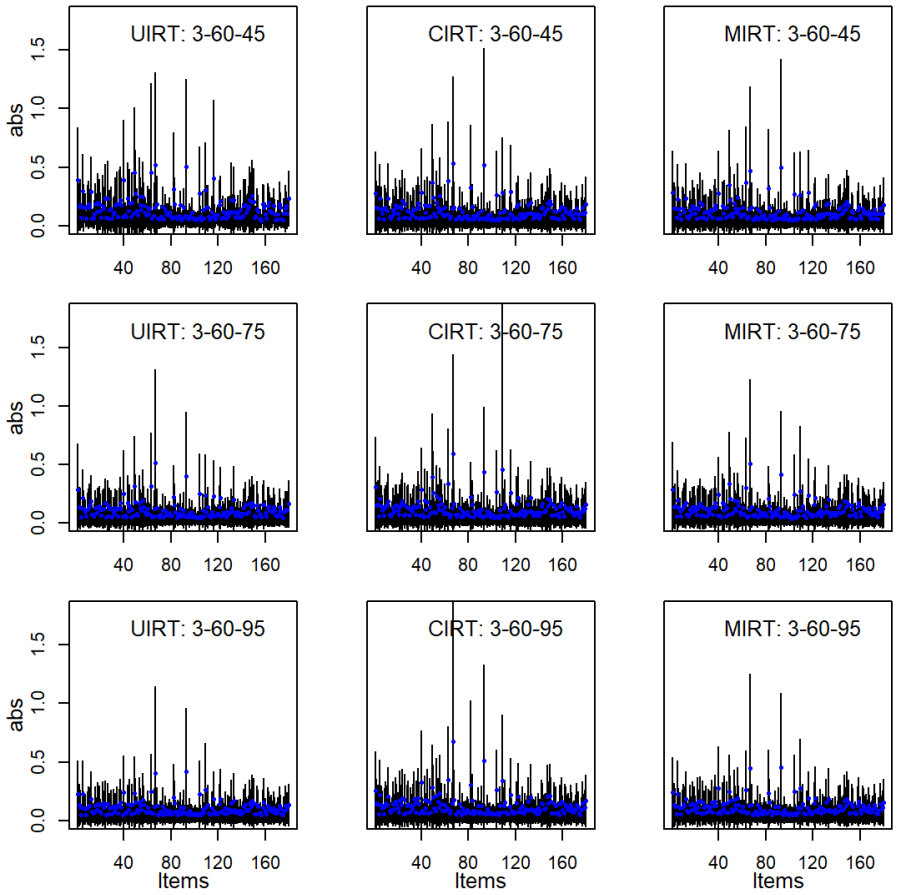


Figure G.7

Absolute Bias of b-Parameter for the 4 Domain, 40 Items per Domain Tests

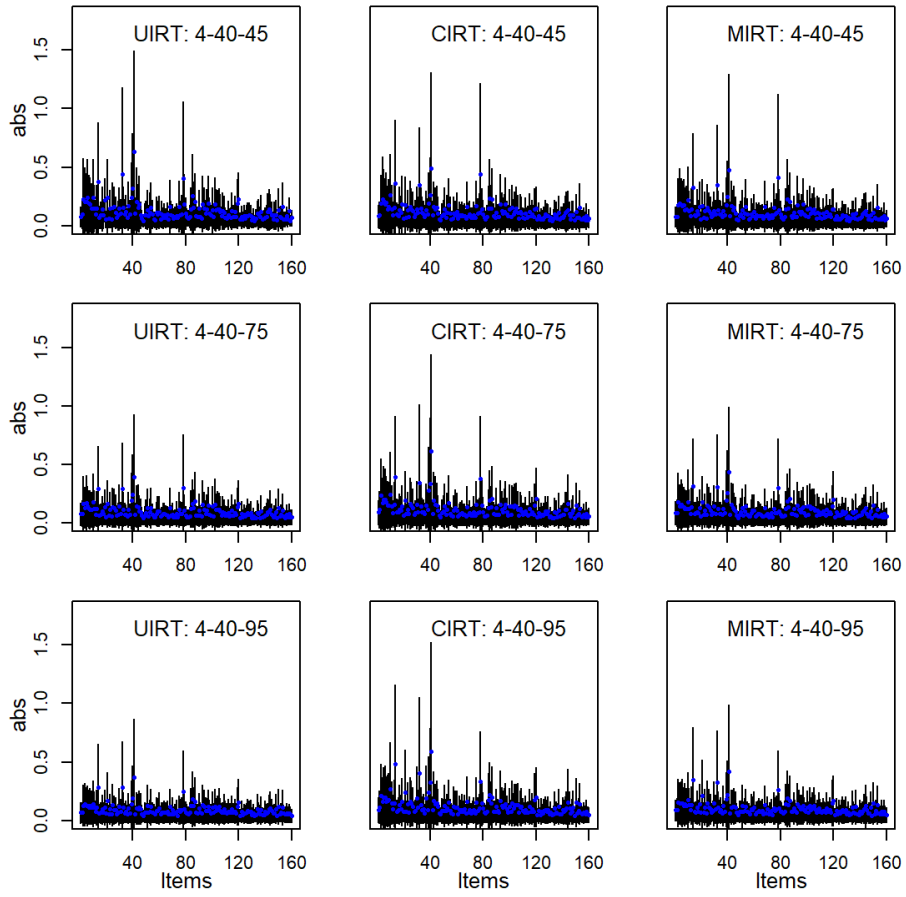
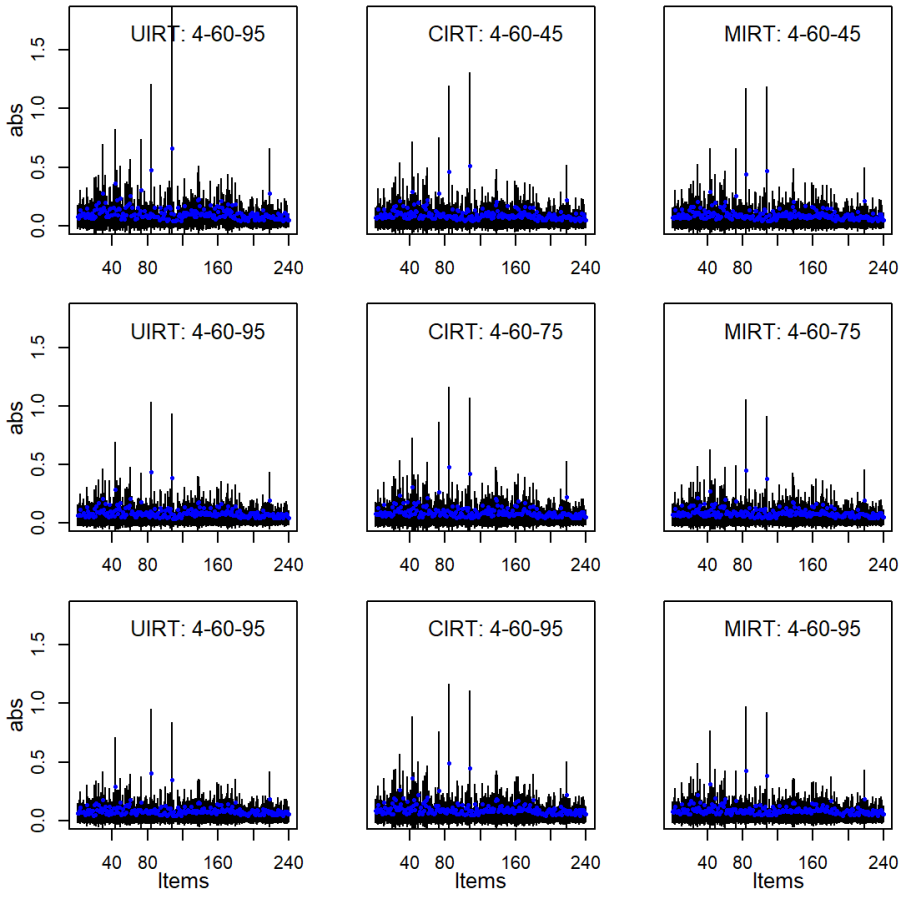


Figure G.8

Absolute Bias of b-Parameter for the 4 Domain, 60 Items per Domain Tests



G.1.3 Item threshold $d1$

Figure G.9
Absolute Bias of d1-Parameter for the 3 Domain, 40 Items per Domain Tests

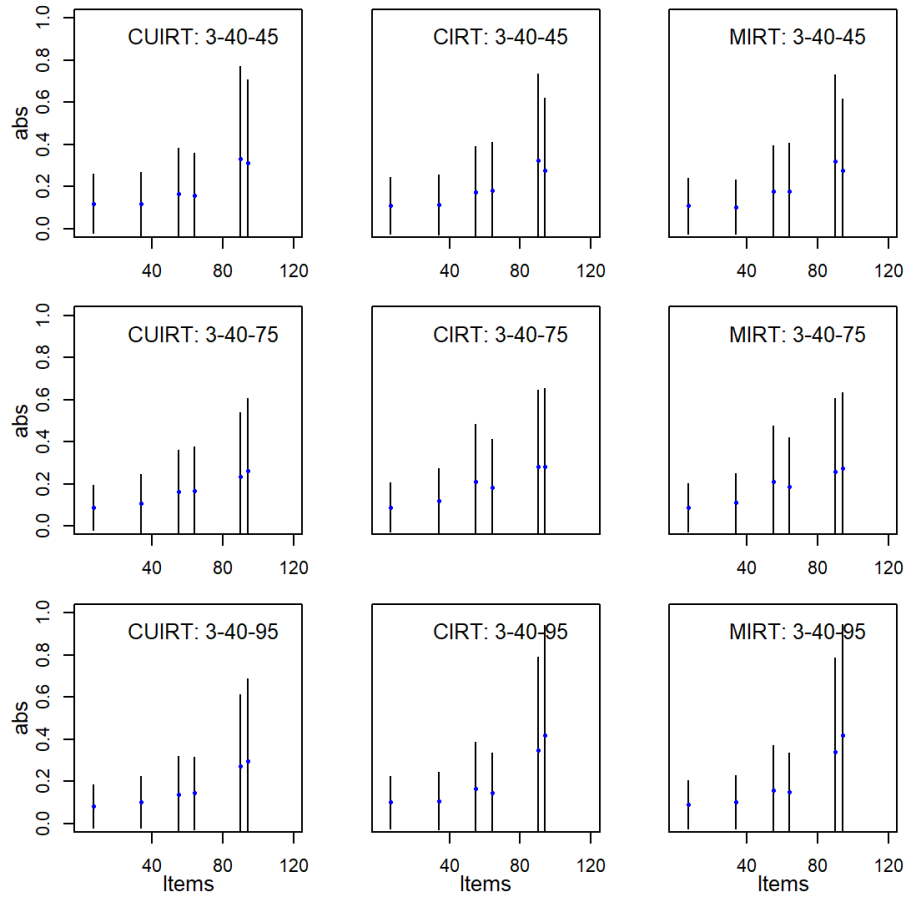


Figure G.10

Absolute Bias of d1-Parameter for the 3 Domain, 60 Items per Domain Tests

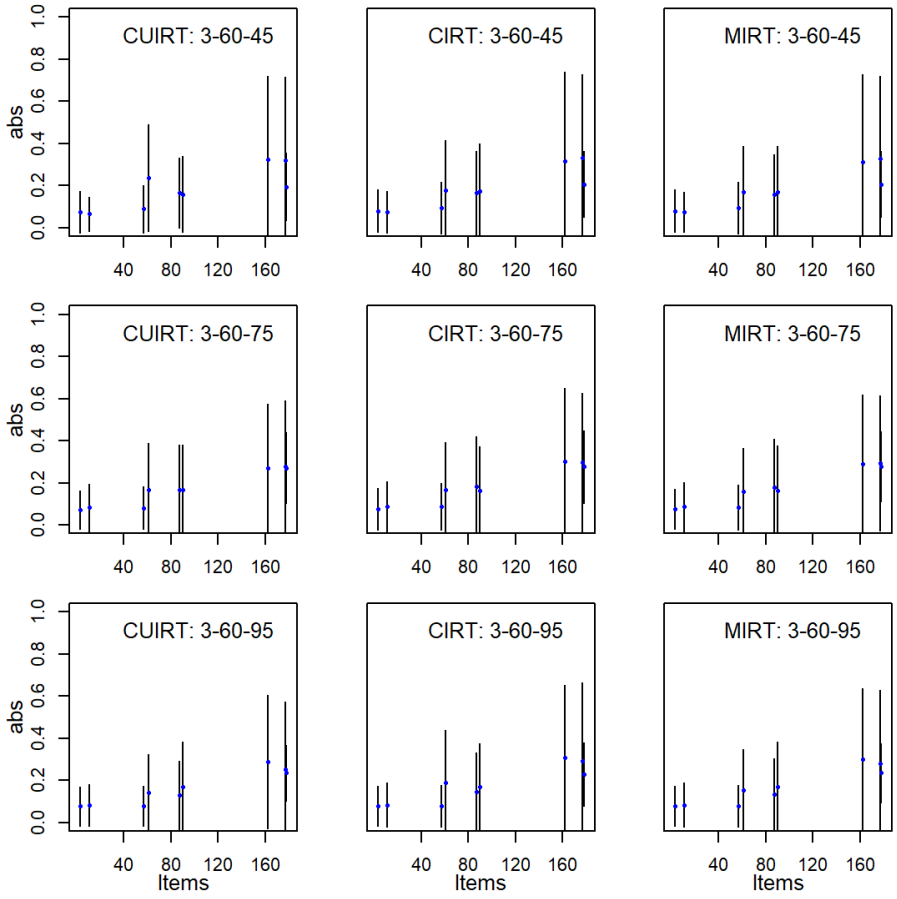


Figure G.11

Absolute Bias of d1-Parameter for the 4 Domain, 40 Items per Domain Tests

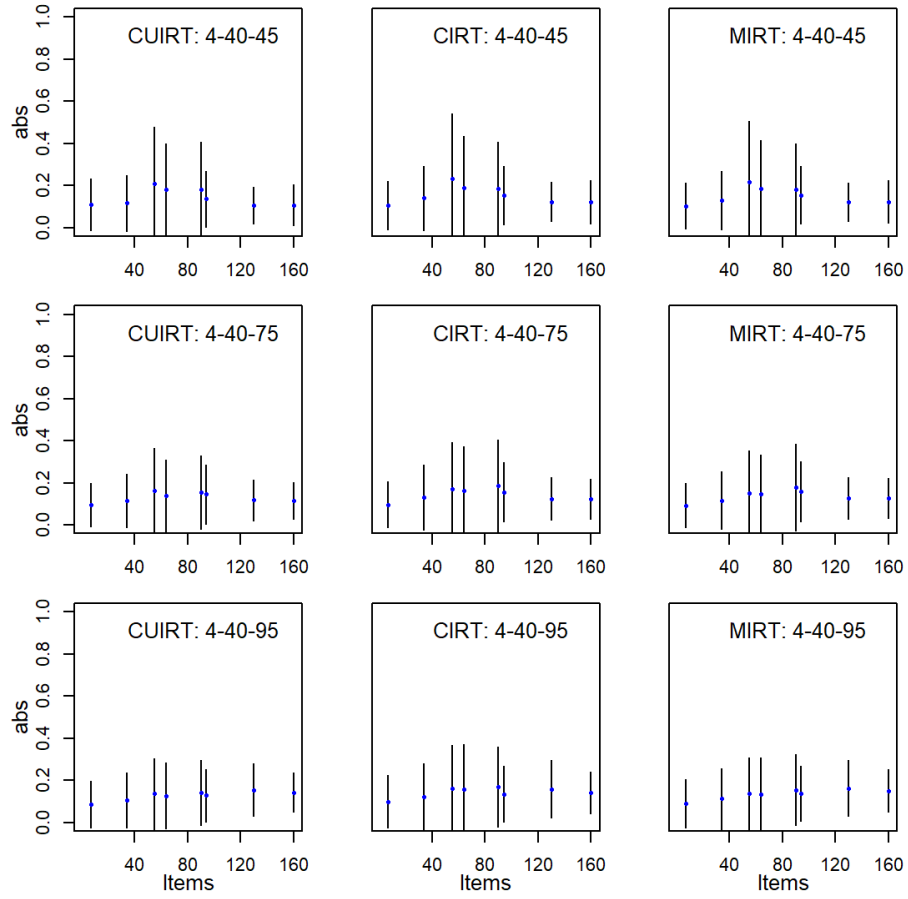
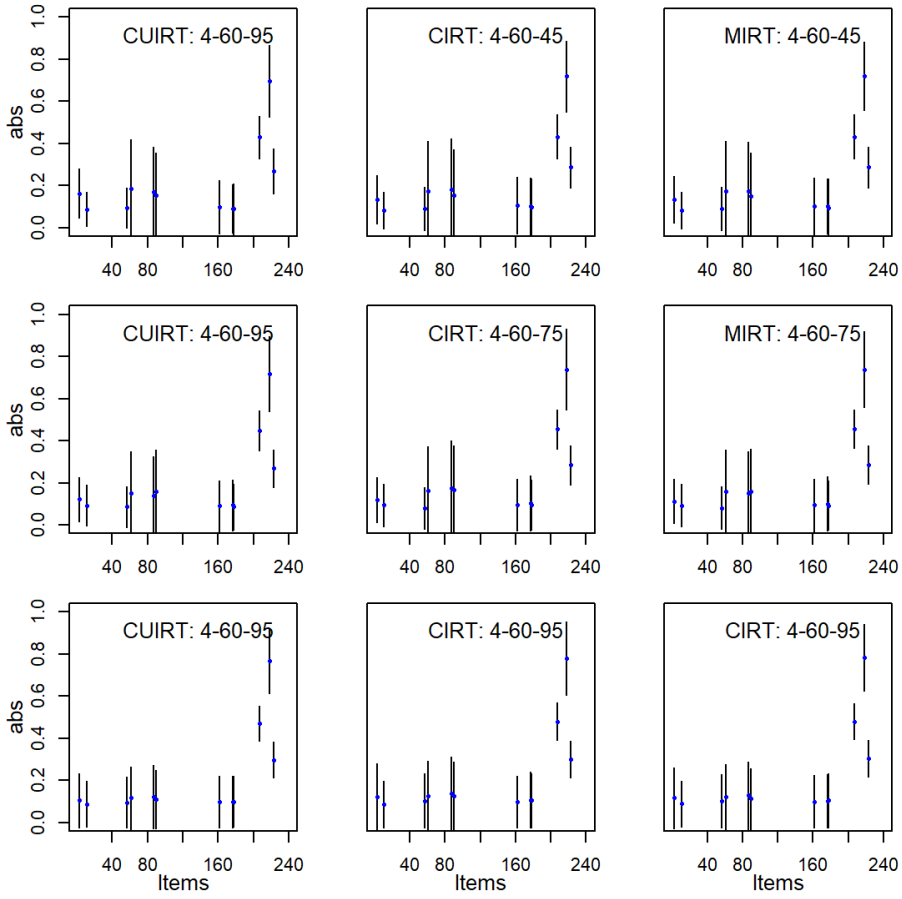


Figure G.12

Absolute Bias of d1-Parameter for the 4 Domain, 60 Items per Domain Tests



G.1.4 Item threshold $d2$

Figure G.13
Absolute Bias of d2-Parameter for the 3 Domain, 40 Items per Domain Tests

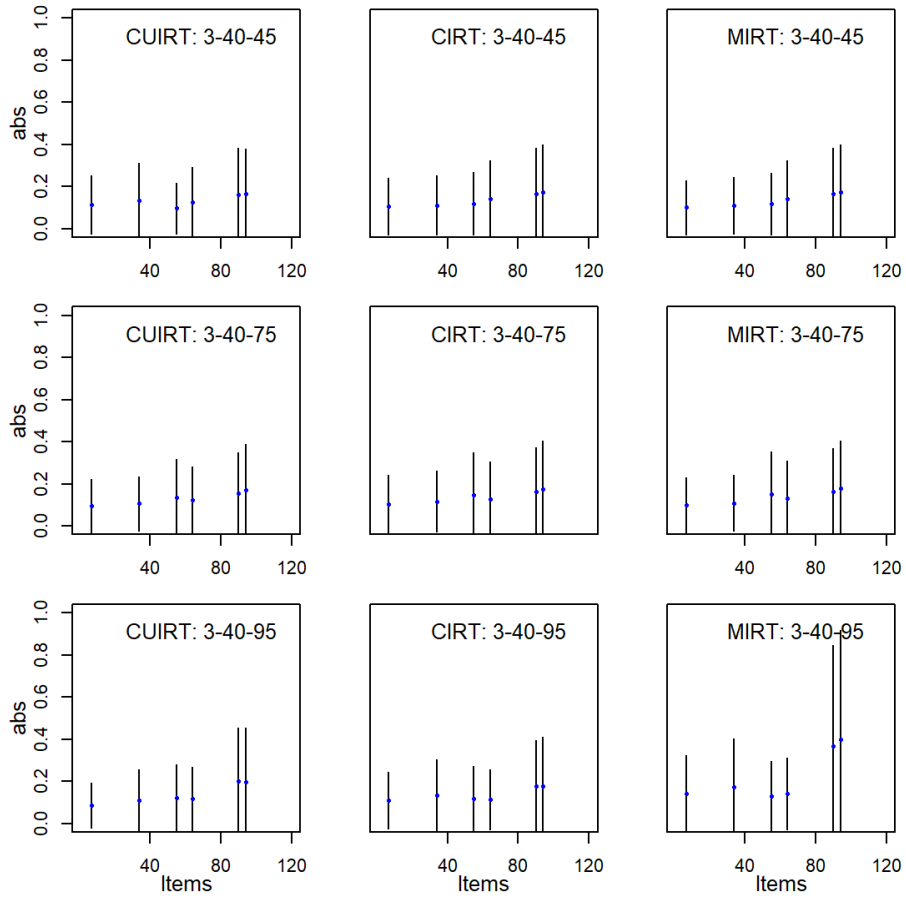


Figure G.14

Absolute Bias of d2-Parameter for the 3 Domain, 60 Items per Domain Tests

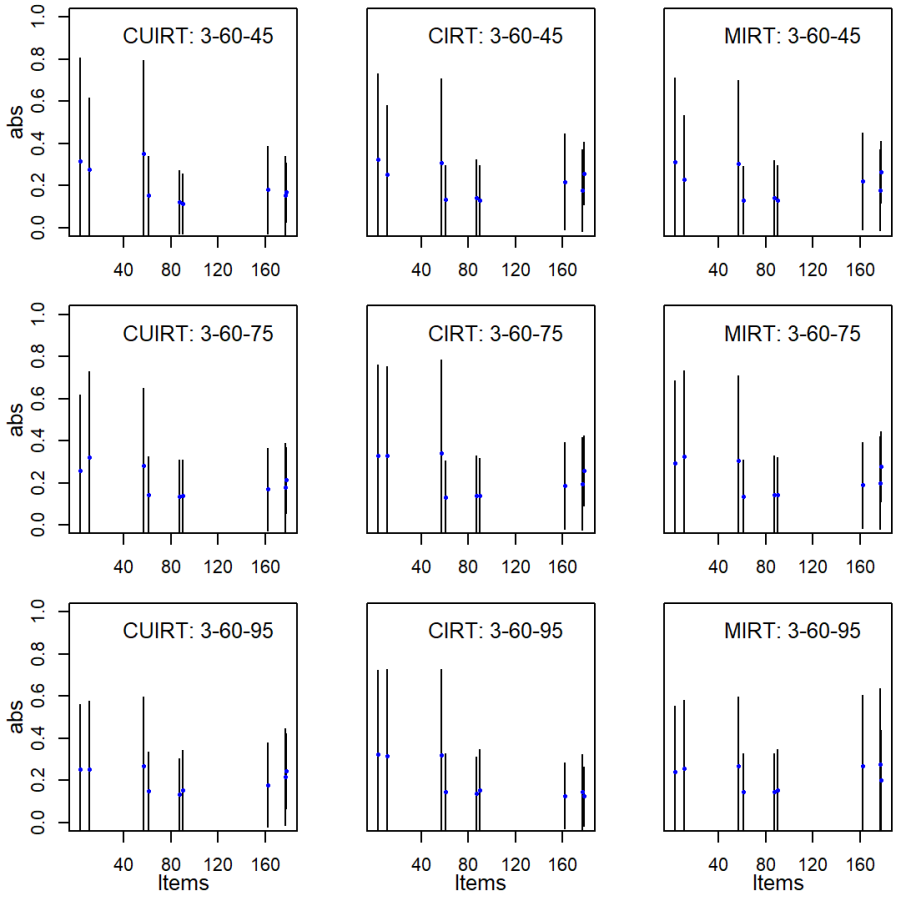


Figure G.15

Absolute Bias of d2-Parameter for the 4 Domain, 40 Items per Domain Tests

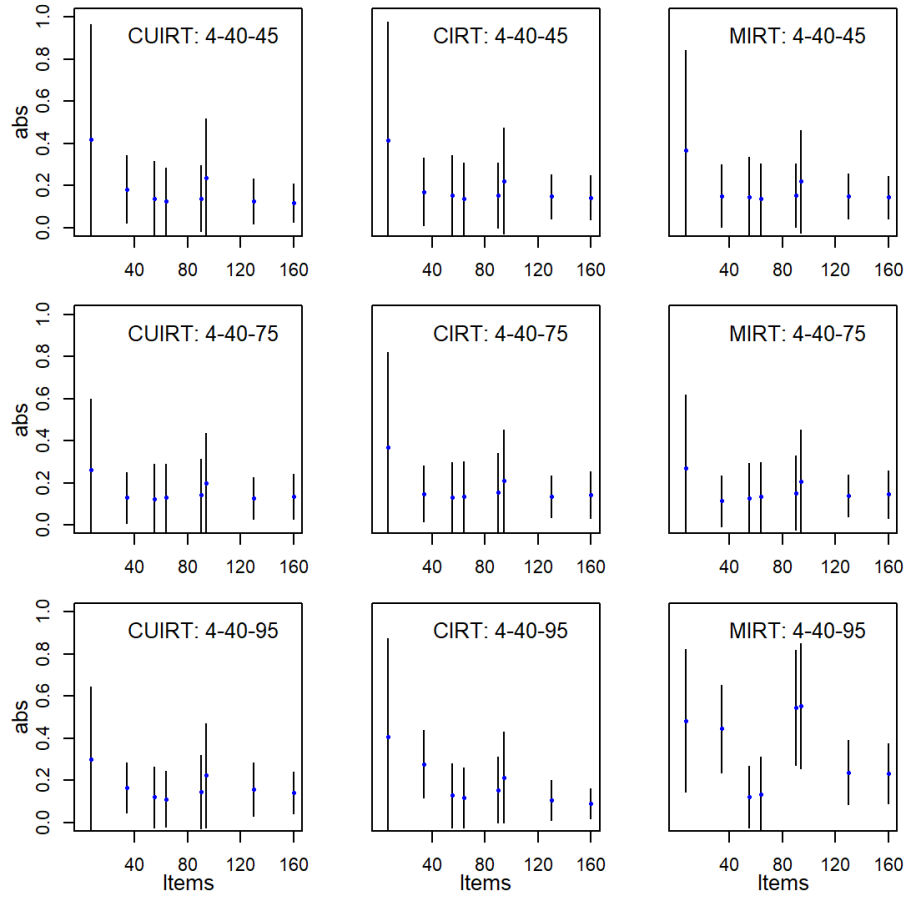
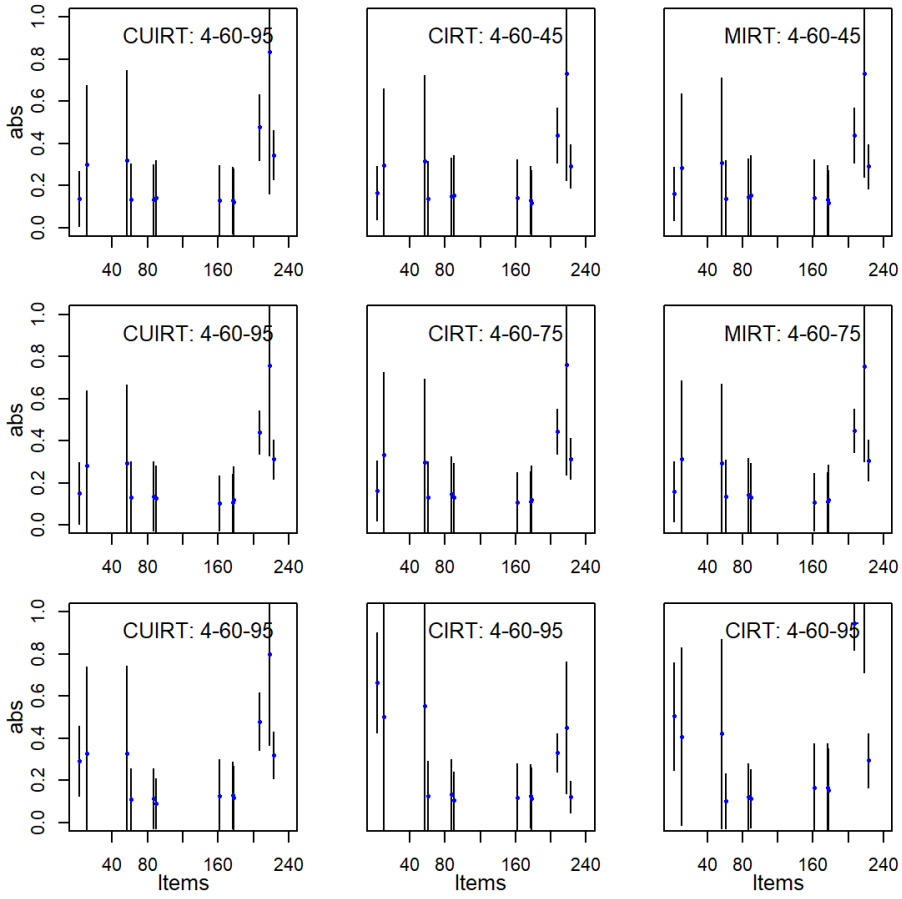


Figure G.16

Absolute Bias of d2-Parameter for the 4 Domain, 60 Items per Domain Tests



G.2 RMSE

G.2.1 Item Discrimination

Figure G.17
RMSE of a-Parameter for the 3 Domain, 40 Items per Domain Tests

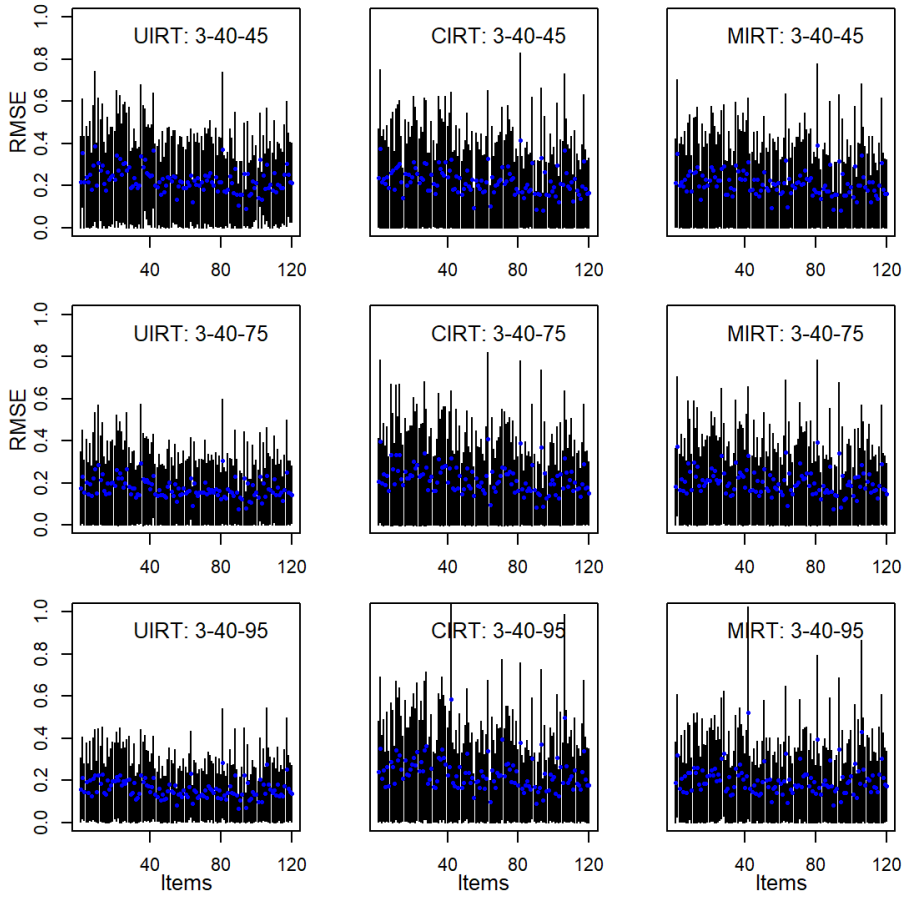


Figure G.18
RMSE of a-Parameter for the 3 Domain, 60 Items per Domain Tests

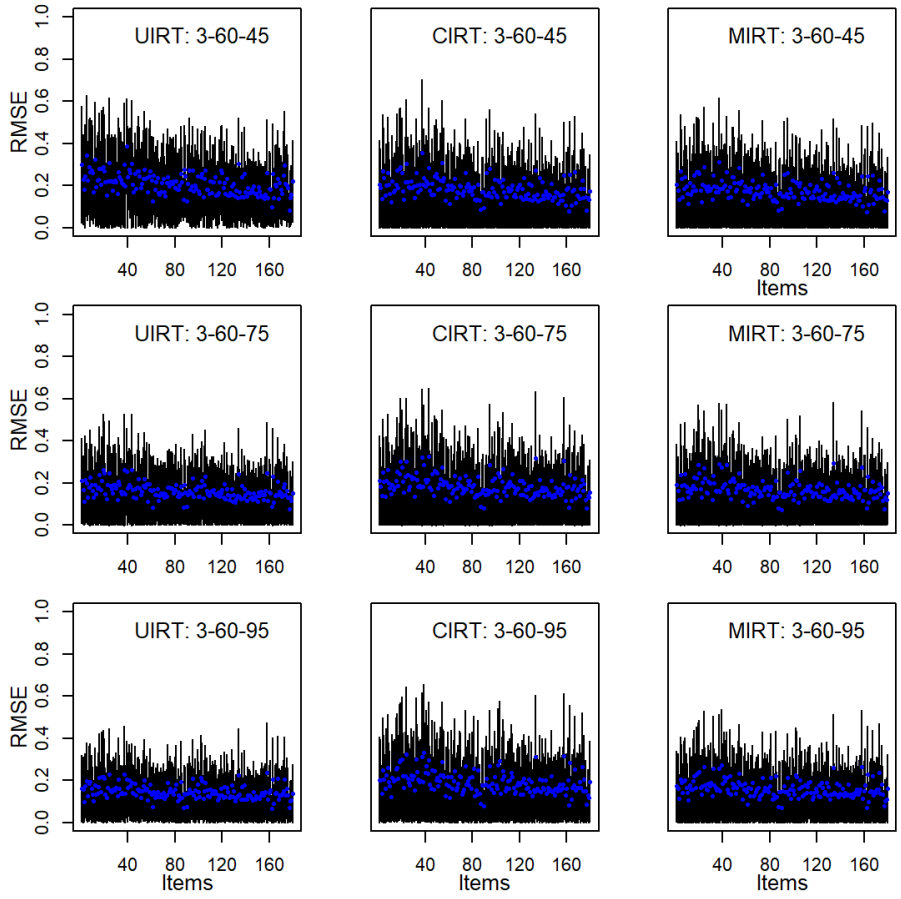


Figure G.19

RMSE of a-Parameter for the 4 Domain, 40 Items per Domain Tests

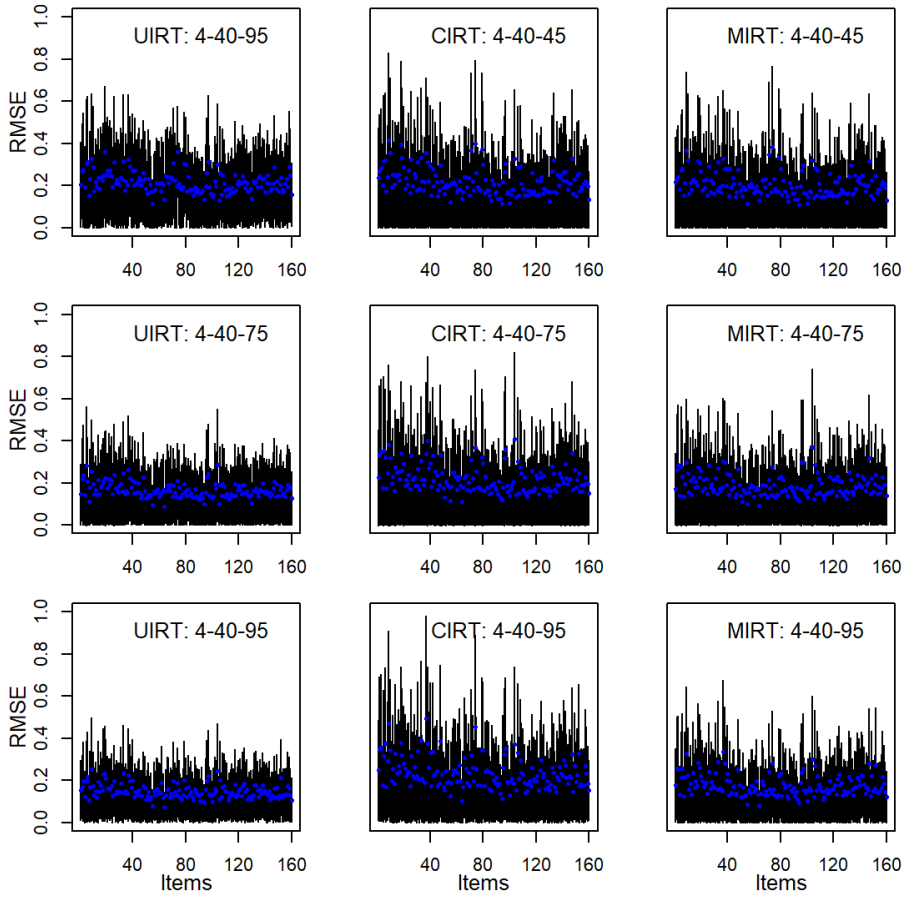
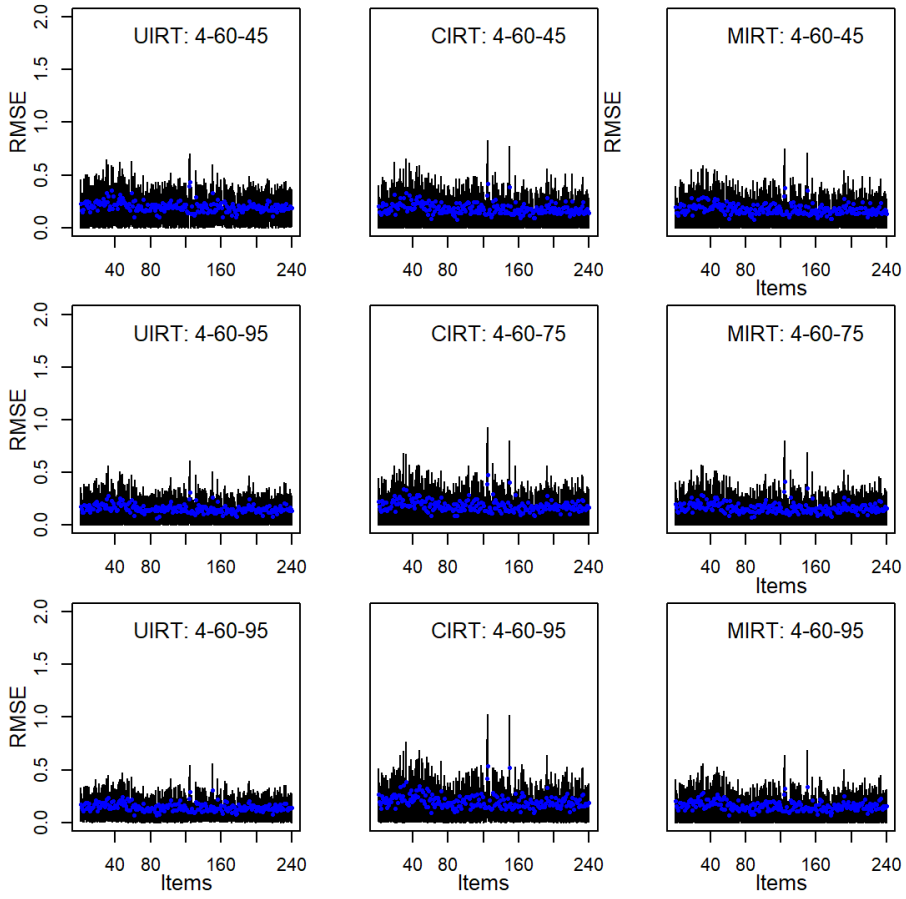


Figure G.20

RMSE of a-Parameter for the 4 Domain, 60 Items per Domain Tests



G.2.2 Item Difficulty

Figure G.21
RMSE of b-Parameter for the 3 Domain, 40 Items per Domain Tests

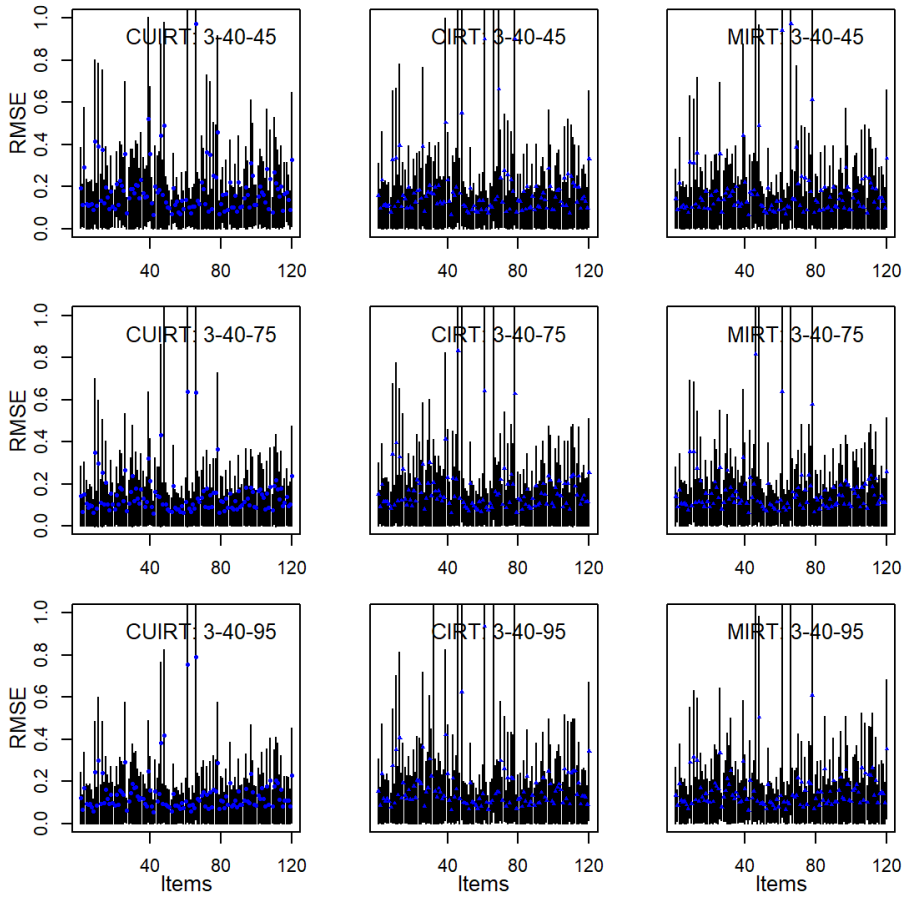


Figure G.22

RMSE of b-Parameter for the 3 Domain, 60 Items per Domain Tests

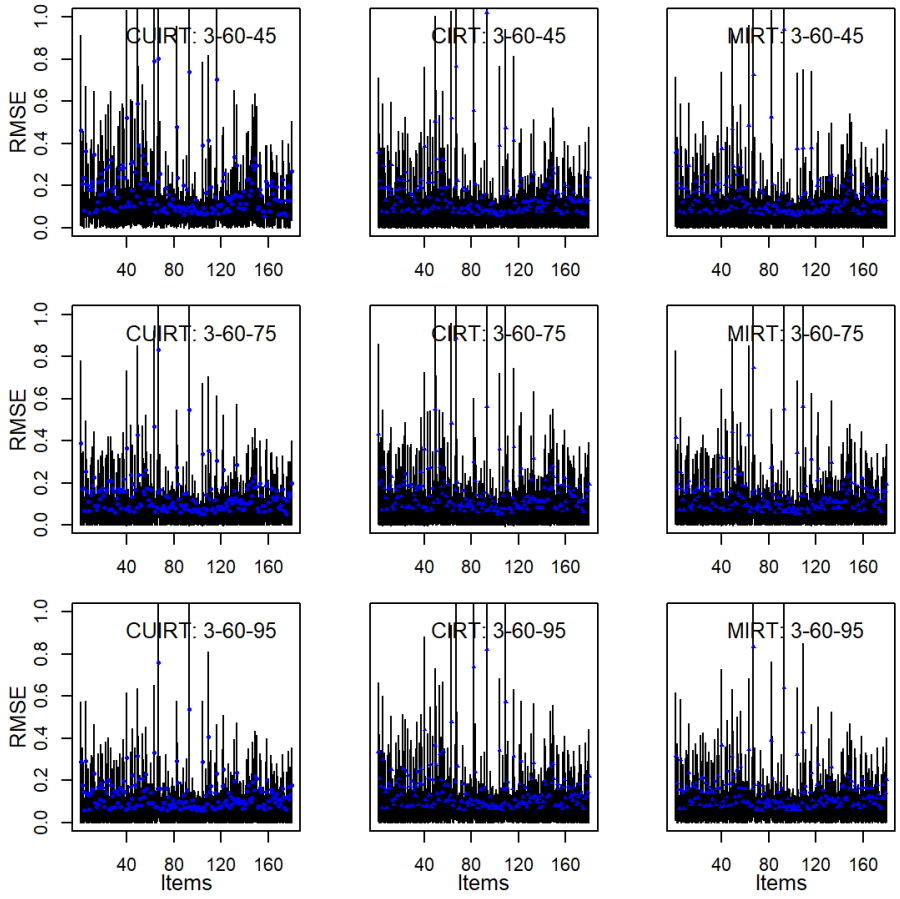


Figure G.23

RMSE of b-Parameter for the 4 Domain, 40 Items per Domain Tests

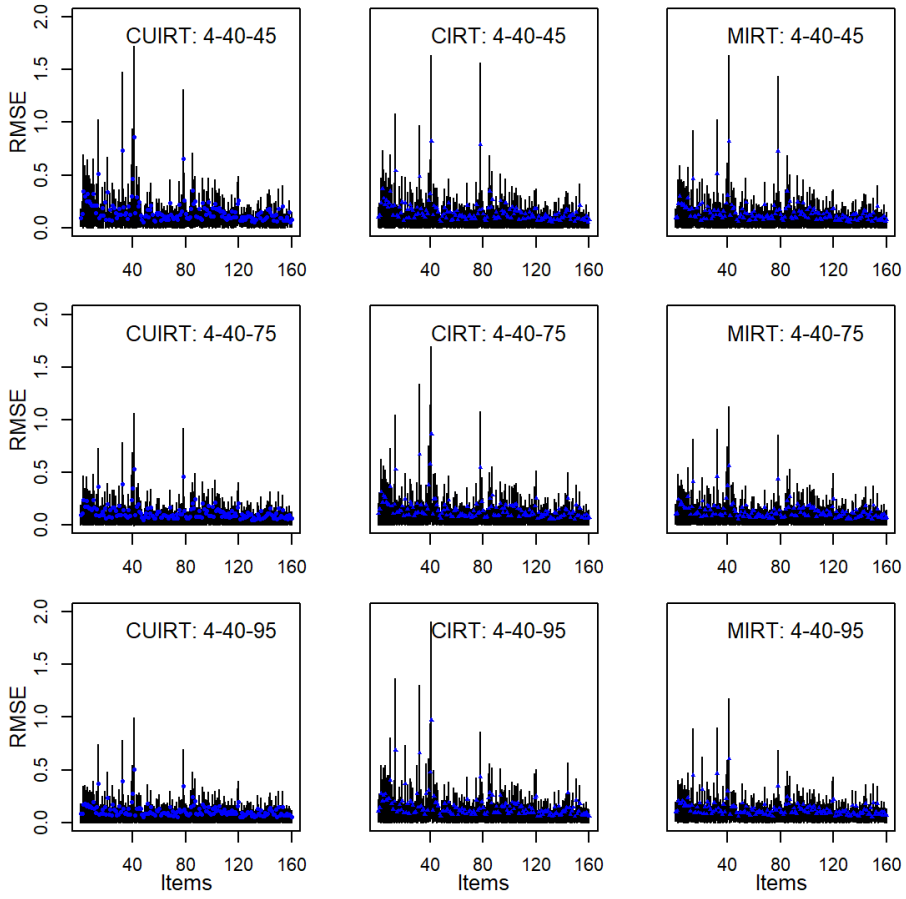
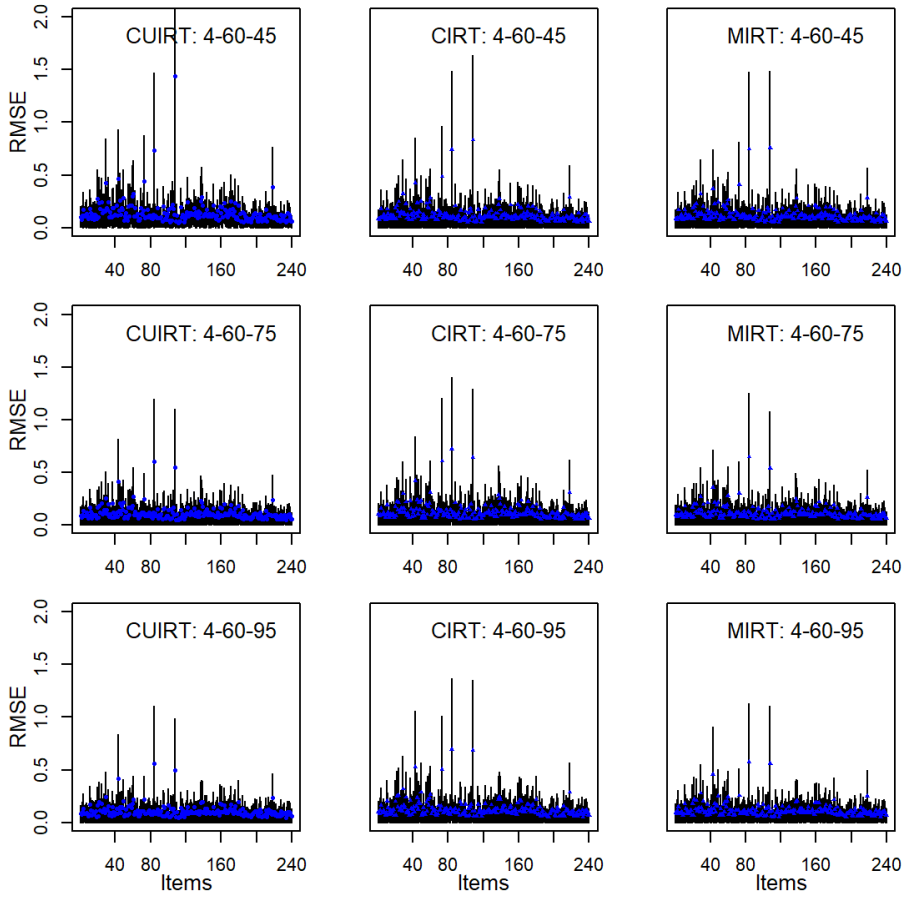


Figure G.24

RMSE of b-Parameter for the 4 Domain, 60 Items per Domain Tests



G.2.3 Item threshold $d1$

Figure G.25
RMSE of d1-Parameter for the 3 Domain, 40 Items per Domain Tests

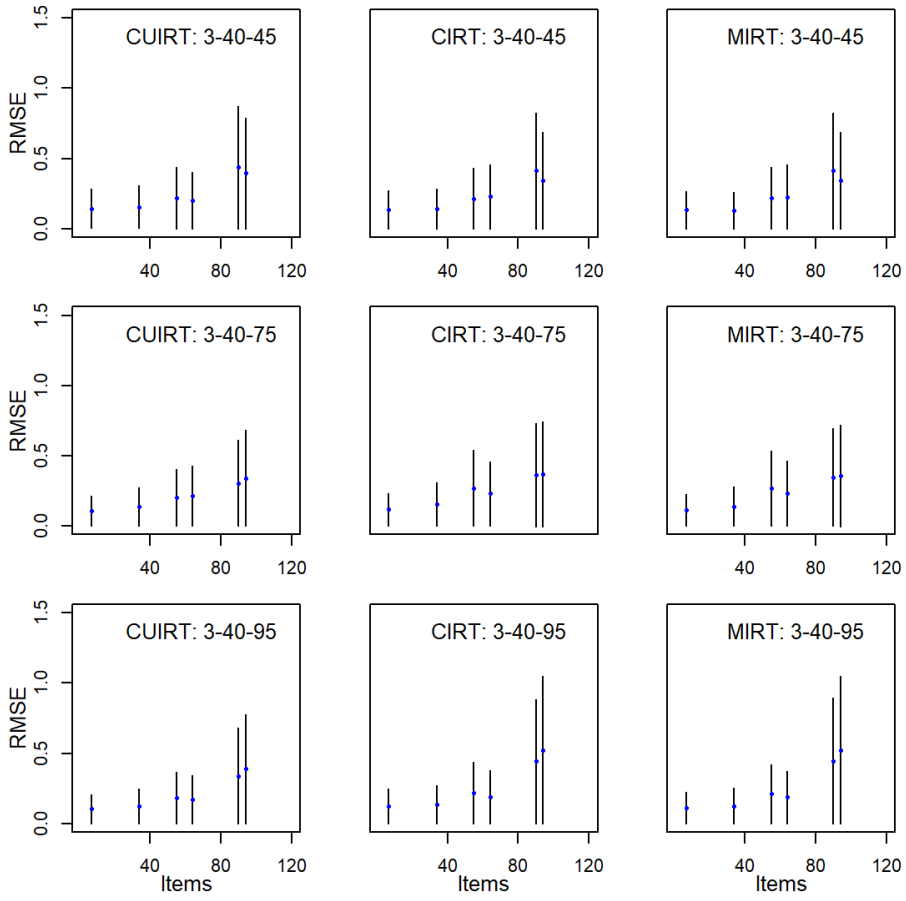


Figure G.26

RMSE of d1-Parameter for the 3 Domain, 60 Items per Domain Tests

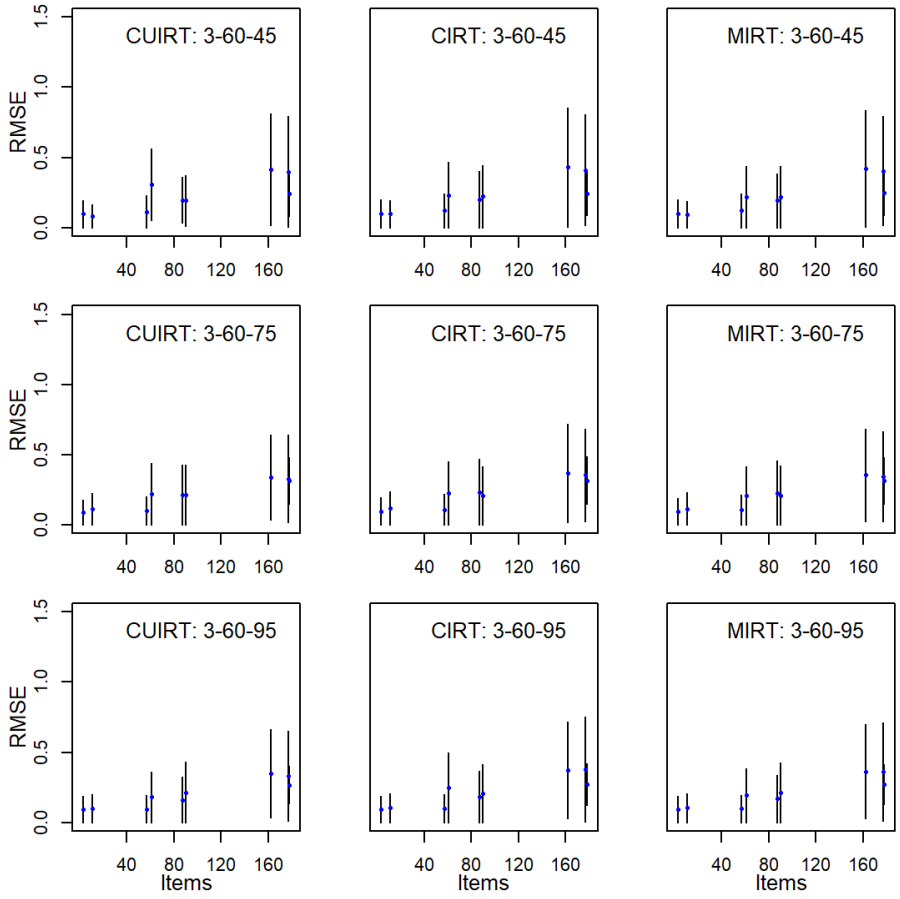


Figure G.27
RMSE of d1-Parameter for the 4 Domain, 40 Items per Domain Tests

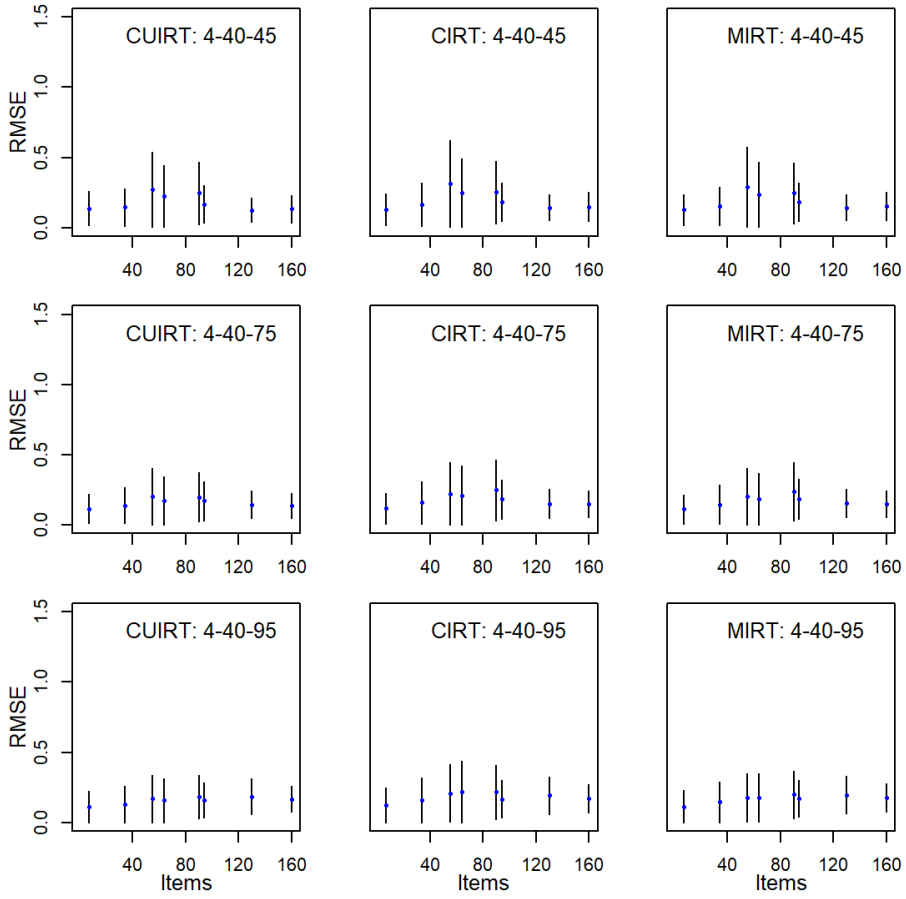
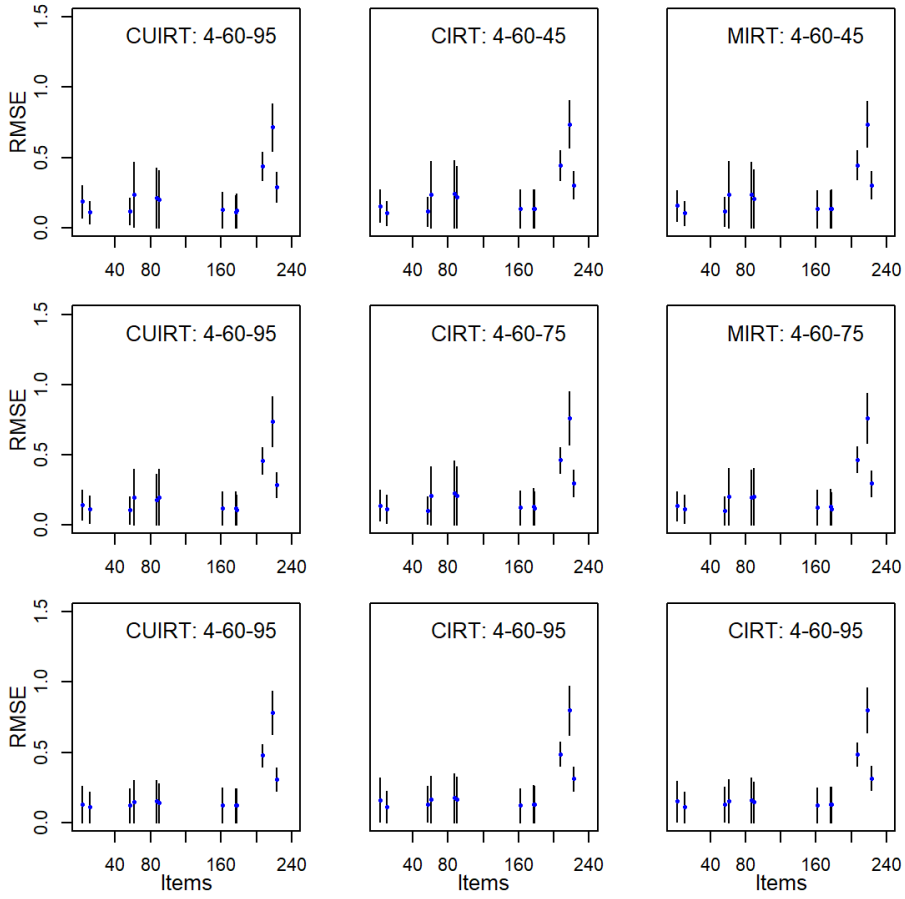


Figure G.28

RMSE of d1-Parameter for the 4 Domain, 60 Items per Domain Tests



G.2.4 Item threshold $d2$

Figure G.29
RMSE of d2-Parameter for the 3 Domain, 40 Items per Domain Tests

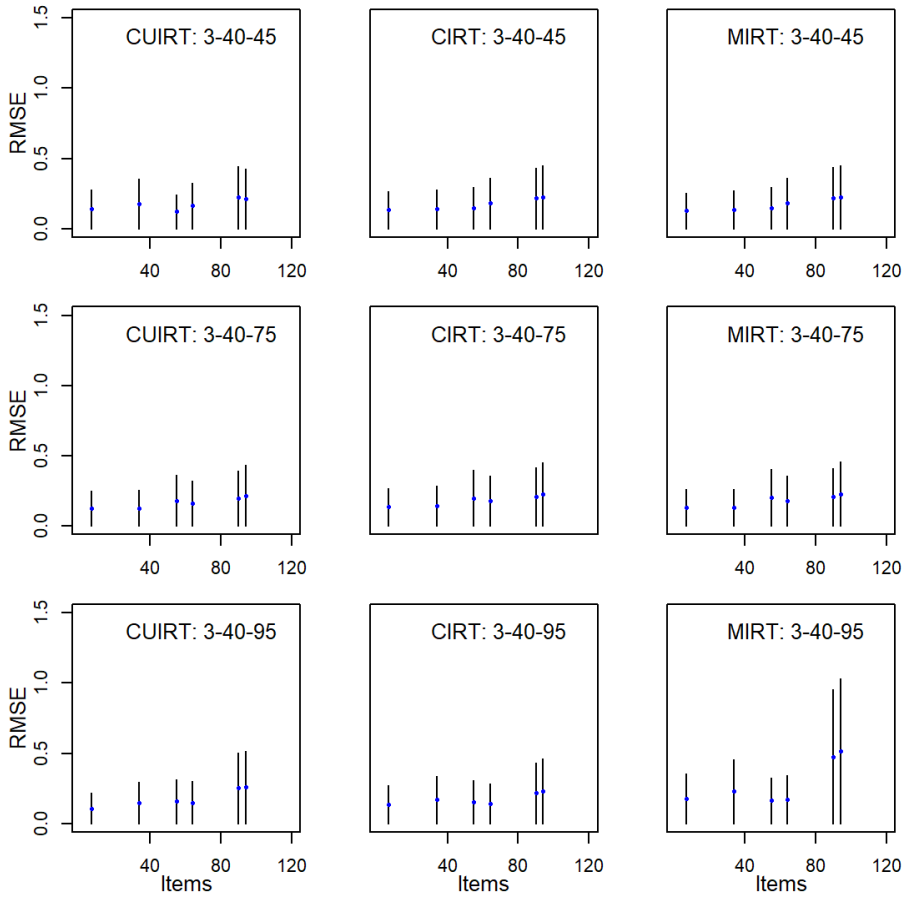


Figure G.30

RMSE of d2-Parameter for the 3 Domain, 60 Items per Domain Tests

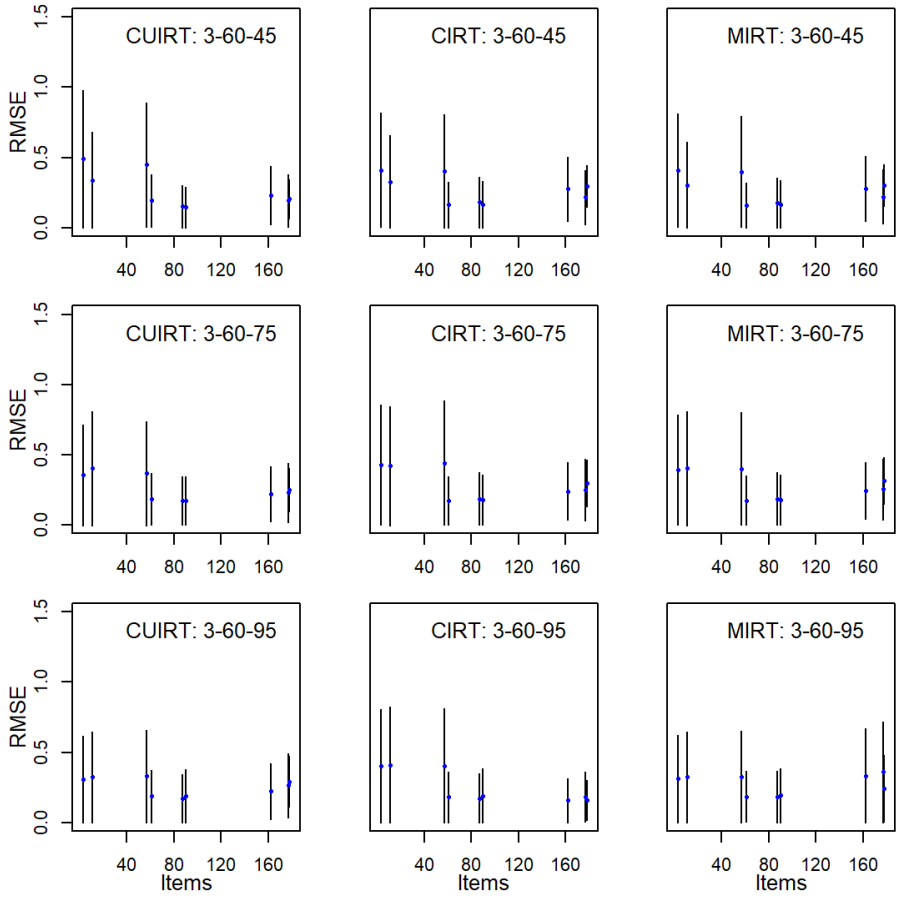


Figure G.31
RMSE of d2-Parameter for the 4 Domain, 40 Items per Domain Tests

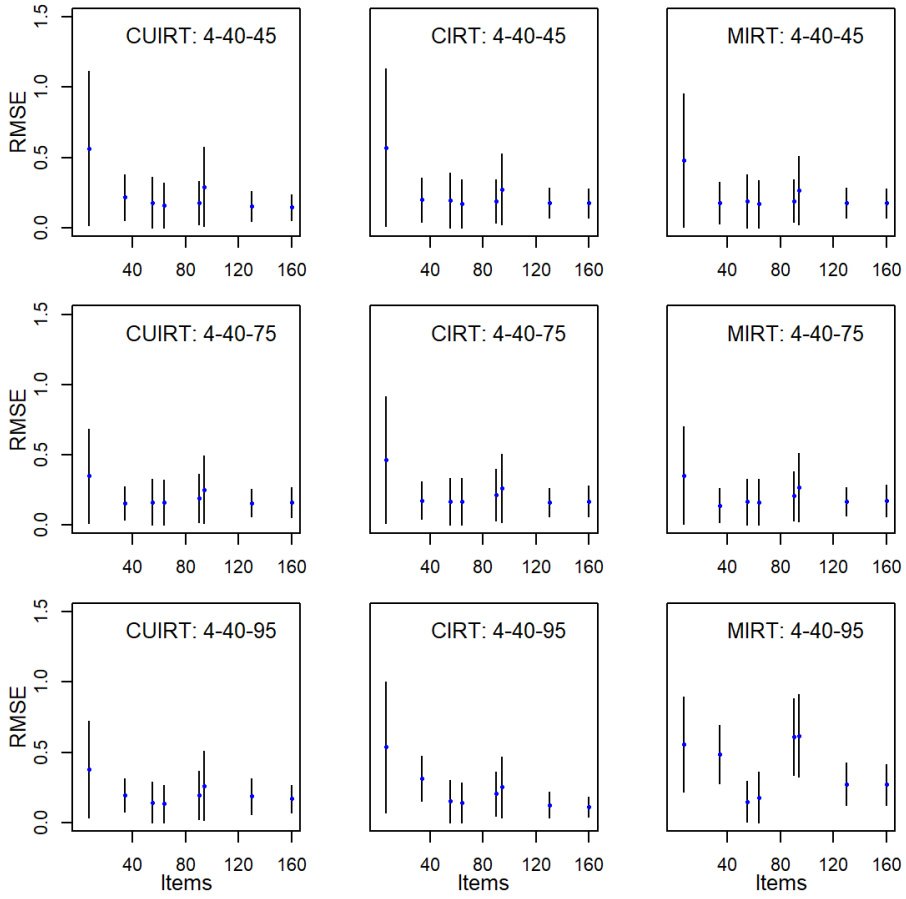
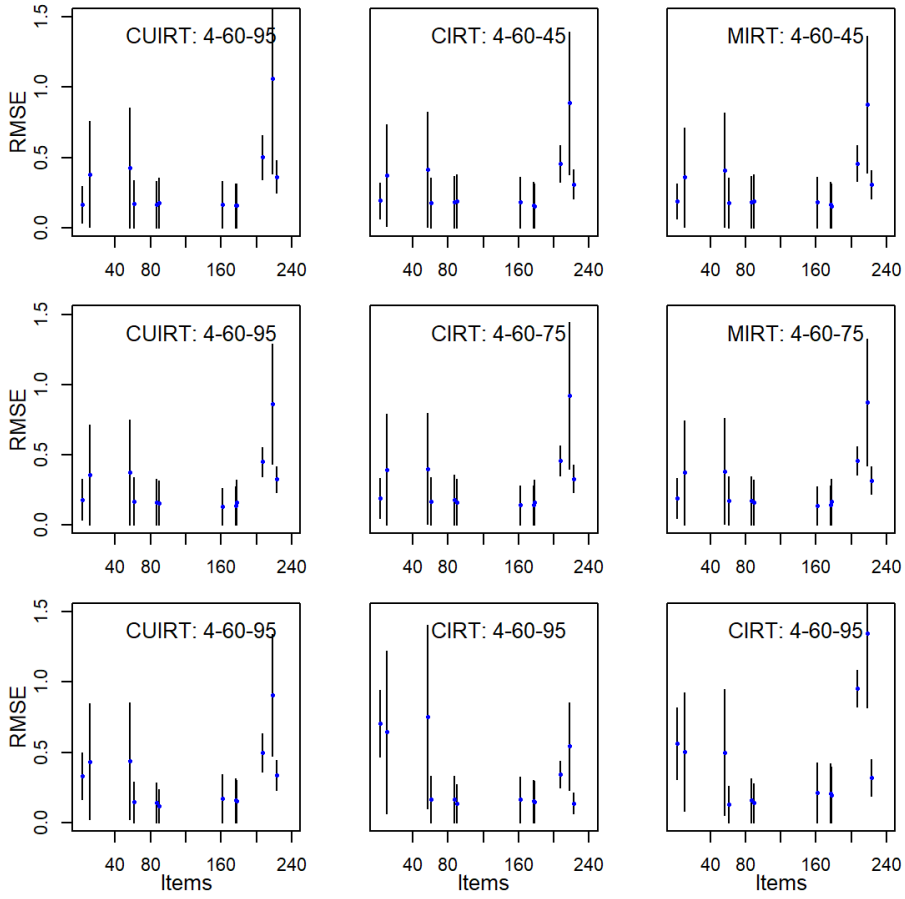


Figure G.32

RMSE of d2-Parameter for the 4 Domain, 60 Items per Domain Tests



Appendix H

Study 2 Item Parameter ABS and RMSE: Multiple Groups

H.1 ABS

H.1.1 Item Discrimination

Figure H.1

Absolute Bias of a-Parameter for the 3 Domain, 40 Items per Domain Tests

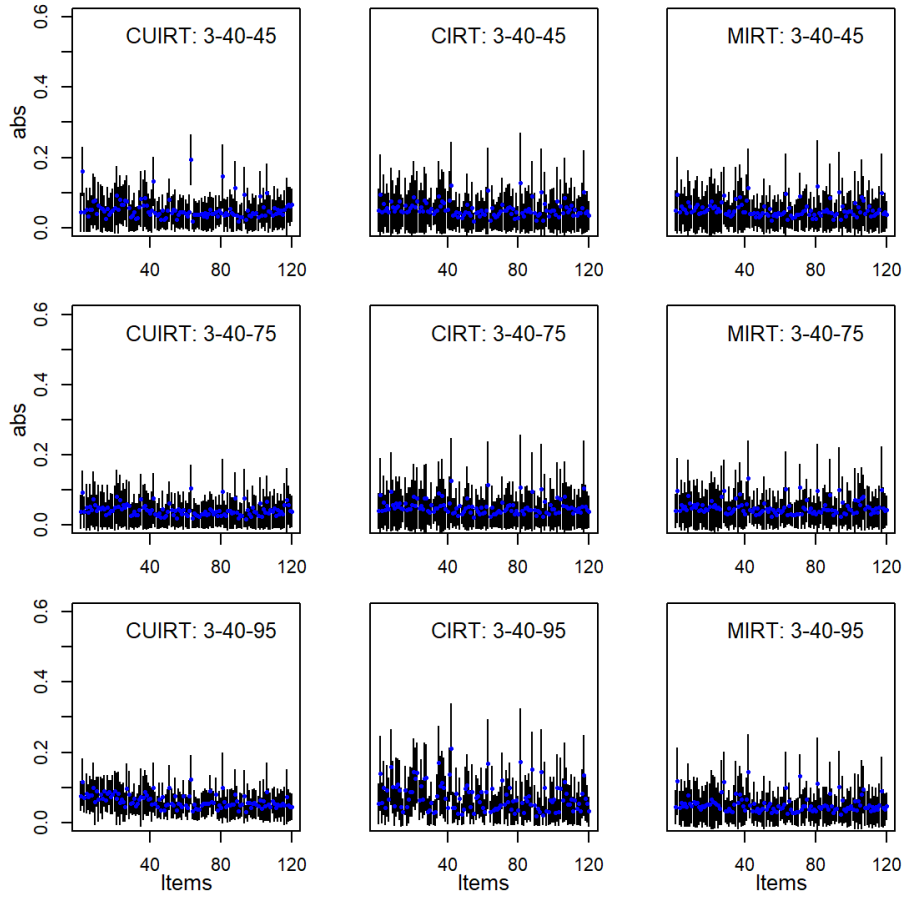


Figure H.2

Absolute Bias of a-Parameter for the 3 Domain, 60 Items per Domain Tests

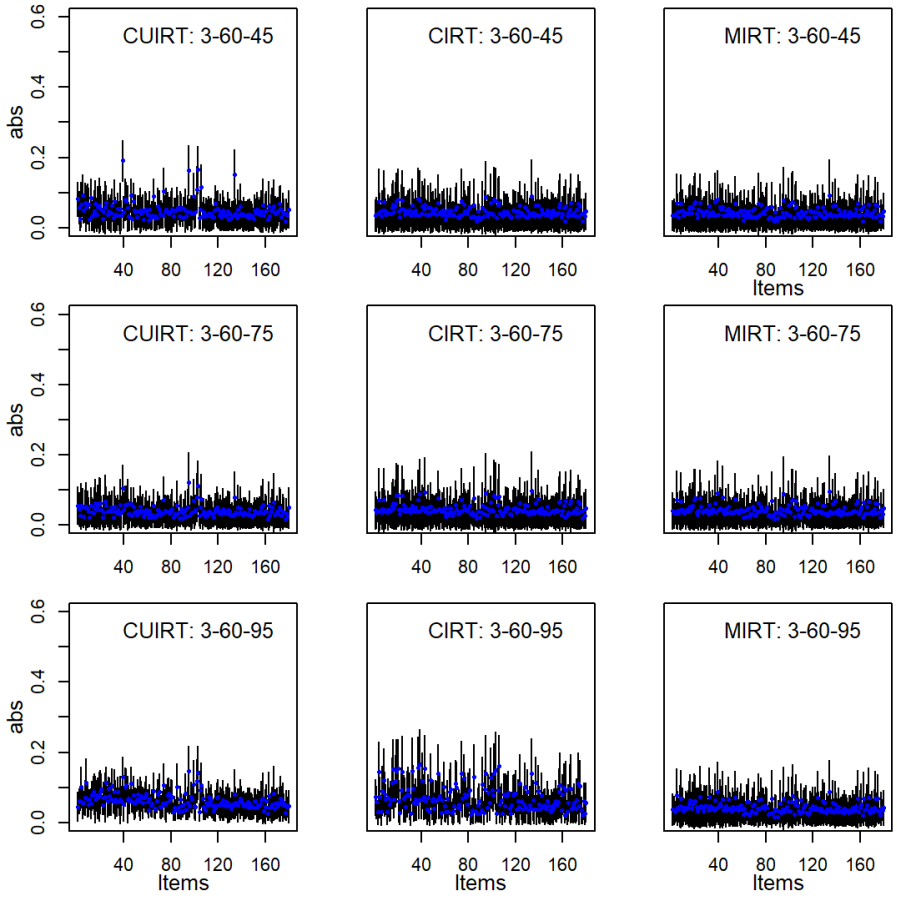


Figure H.3

Absolute Bias of a-Parameter for the 4 Domain, 40 Items per Domain Tests

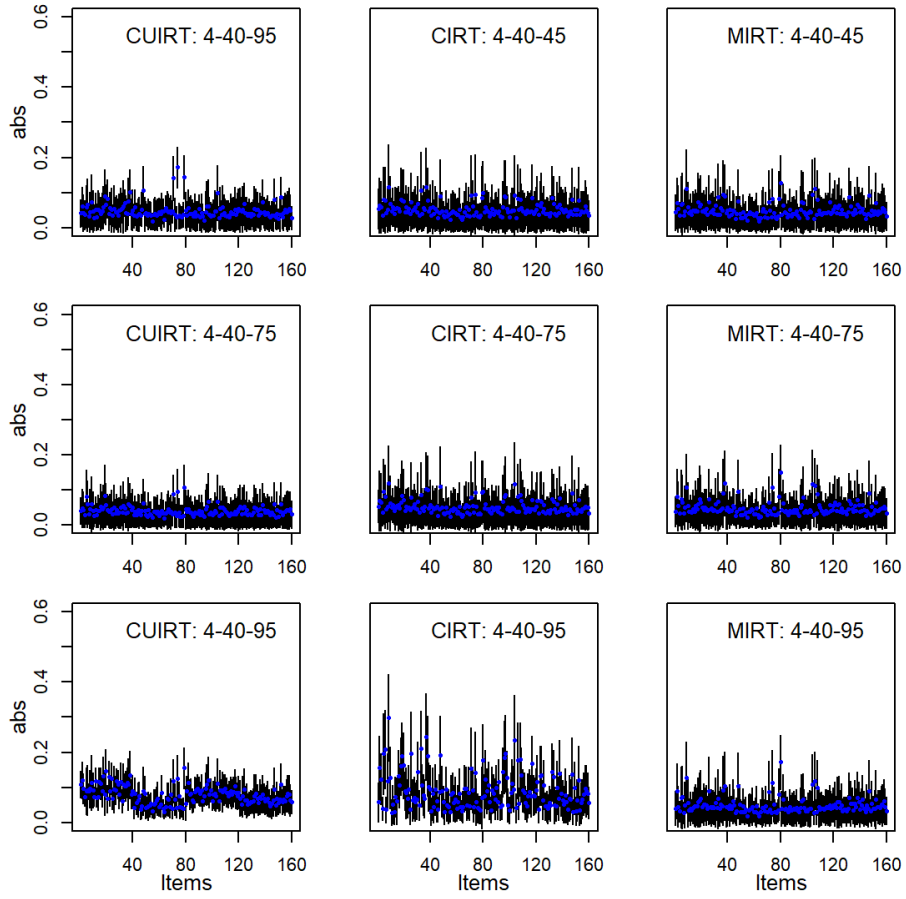
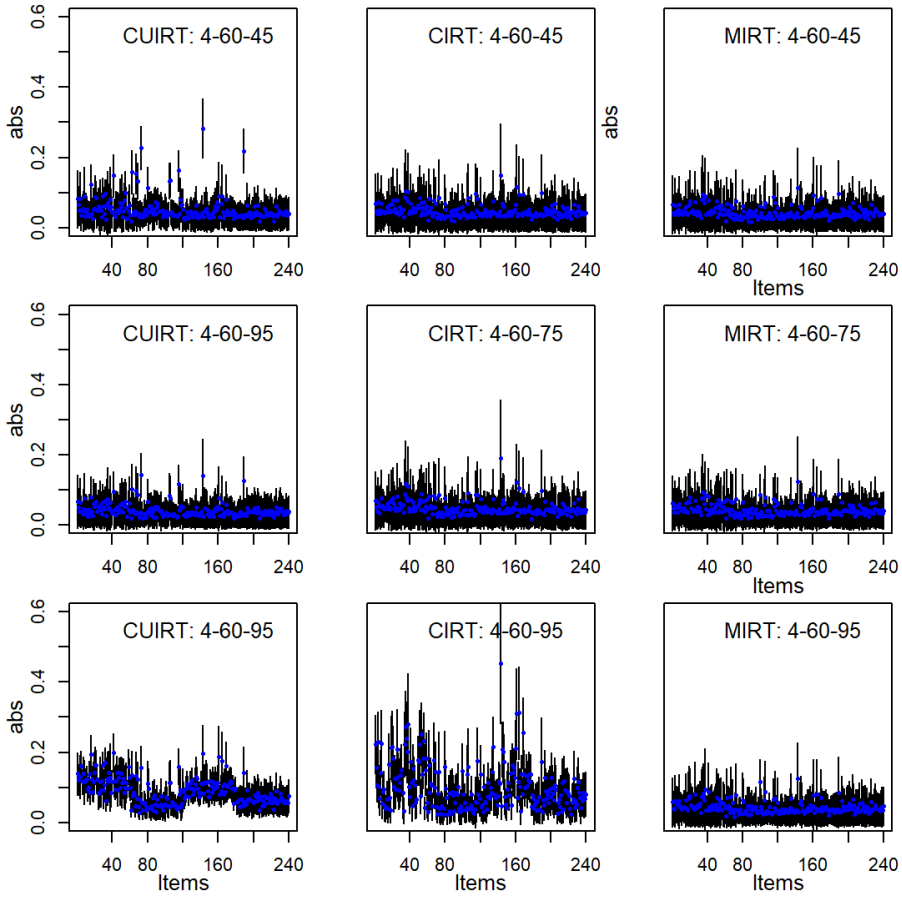


Figure H.4

Absolute Bias of a-Parameter for the 4 Domain, 60 Items per Domain Tests



H.1.2 Item Difficulty

Figure H.5
Absolute Bias of b-Parameter for the 3 Domain, 40 Items per Domain Tests

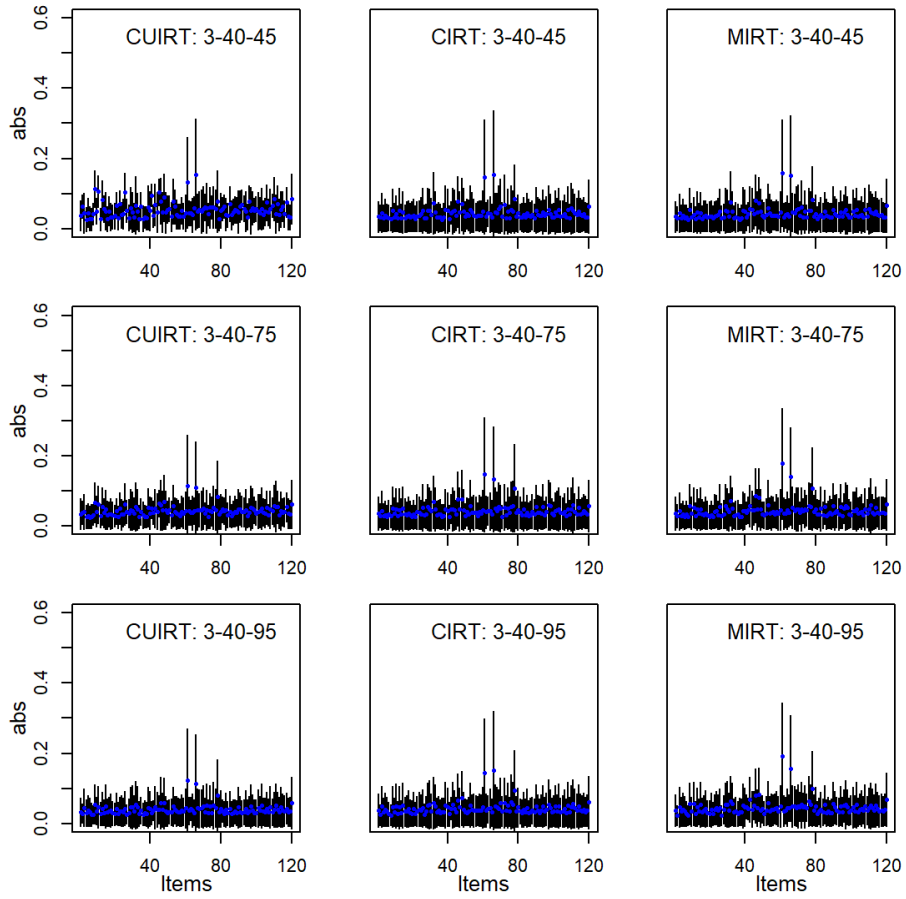


Figure H.6

Absolute Bias of b-Parameter for the 3 Domain, 60 Items per Domain Tests

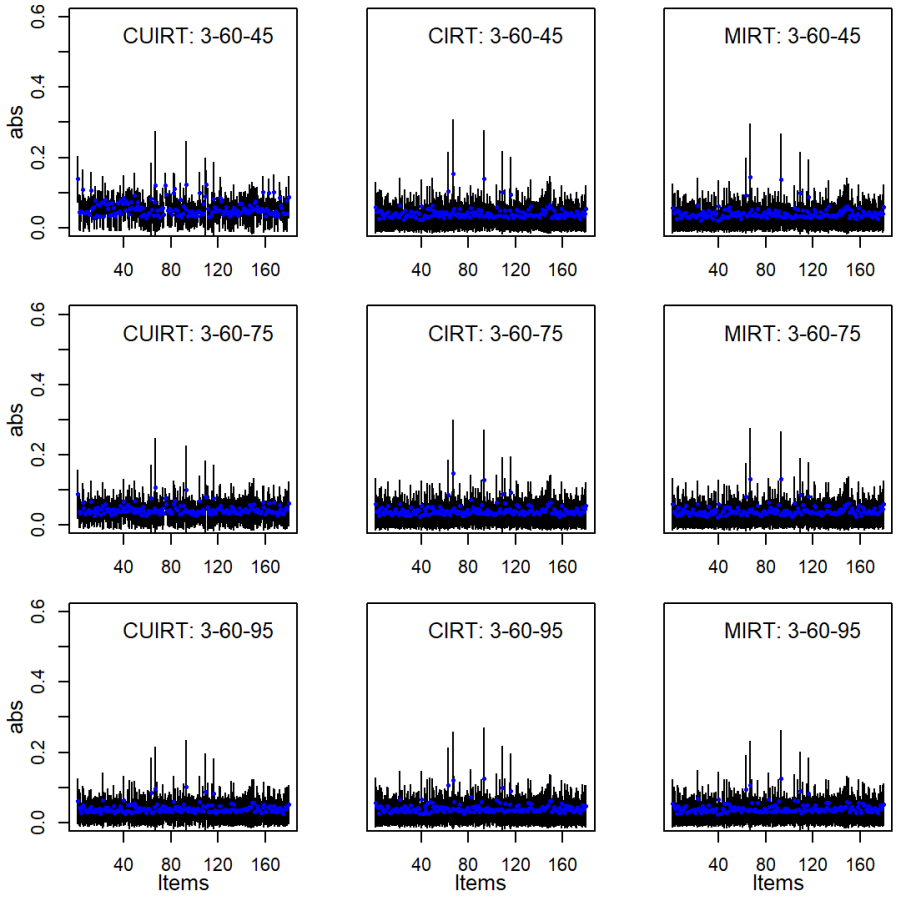


Figure H.7

Absolute Bias of b-Parameter for the 4 Domain, 40 Items per Domain Tests

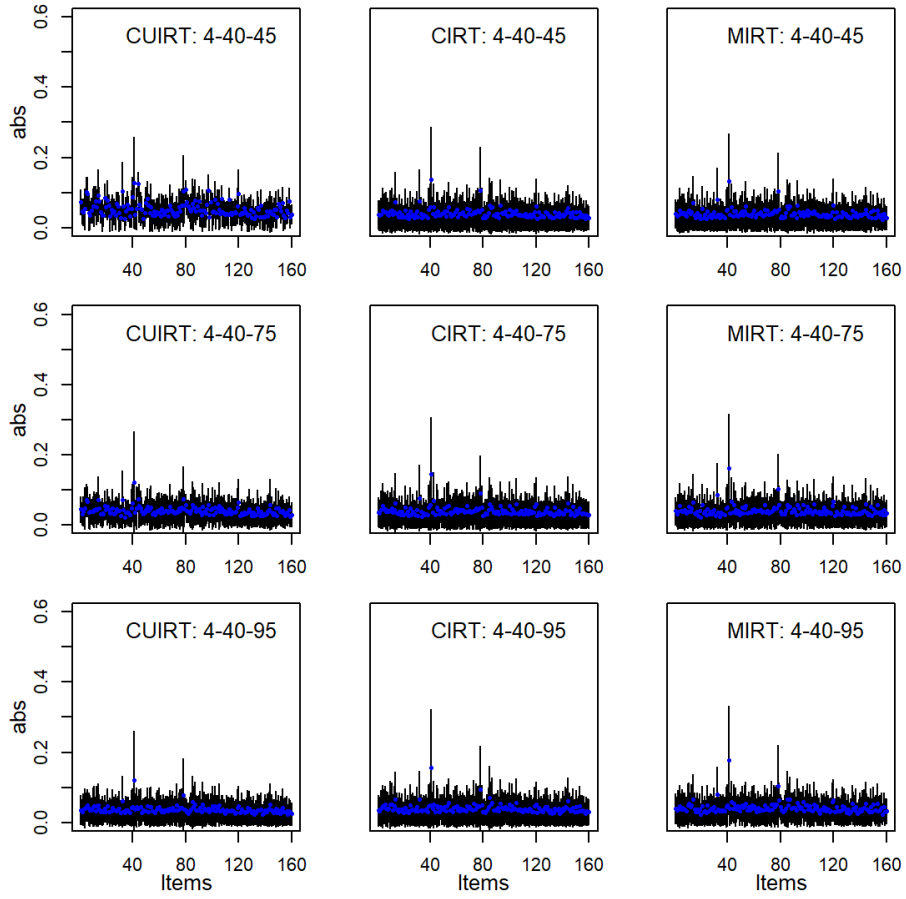
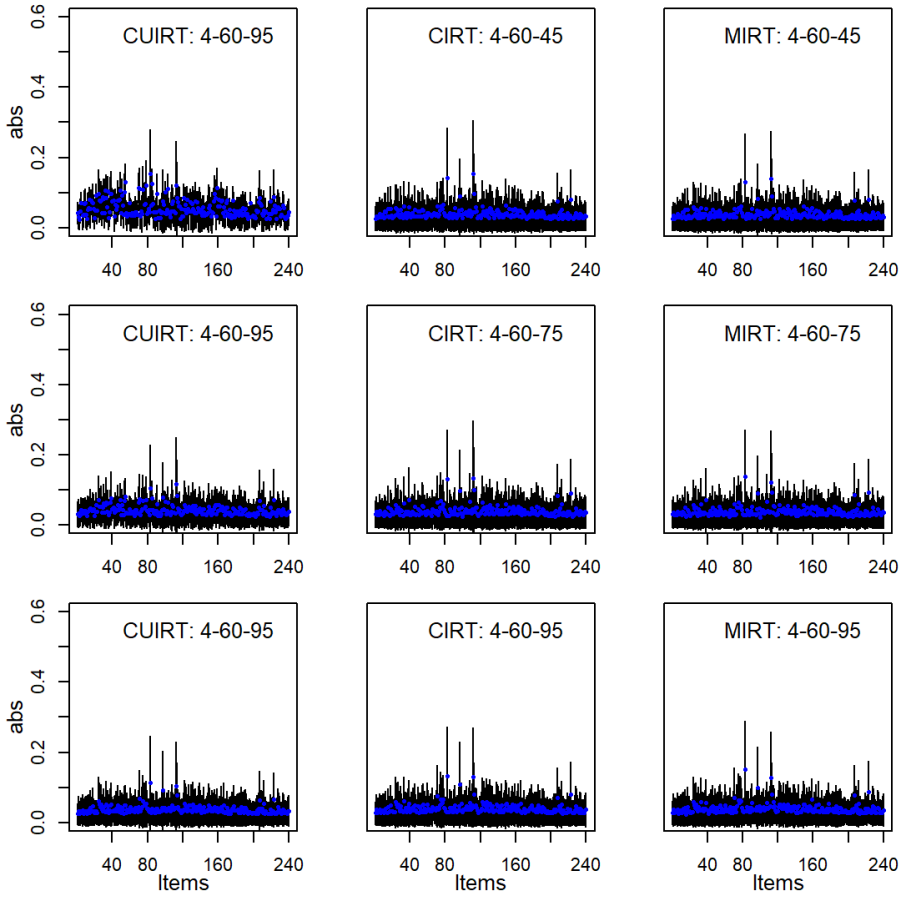


Figure H.8

Absolute Bias of b-Parameter for the 4 Domain, 60 Items per Domain Tests



H.1.3 Item threshold $d1$

Figure H.9
Absolute Bias of d1-Parameter for the 3 Domain, 40 Items per Domain Tests

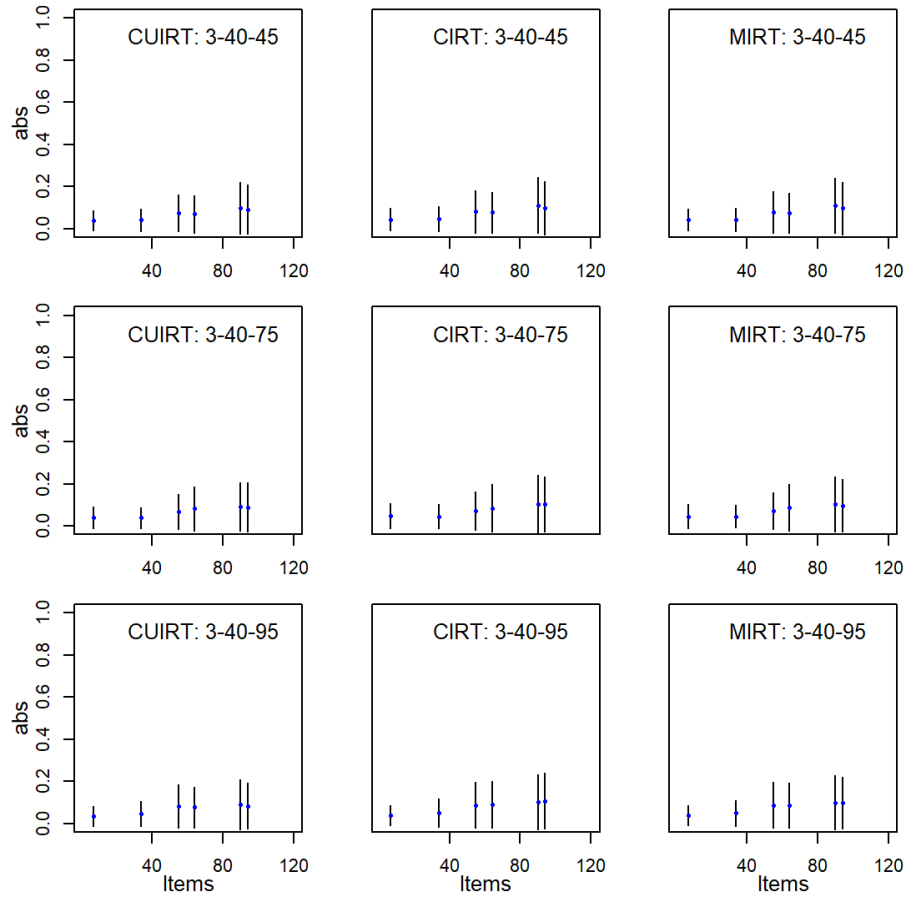


Figure H.10

Absolute Bias of d1-Parameter for the 3 Domain, 60 Items per Domain Tests

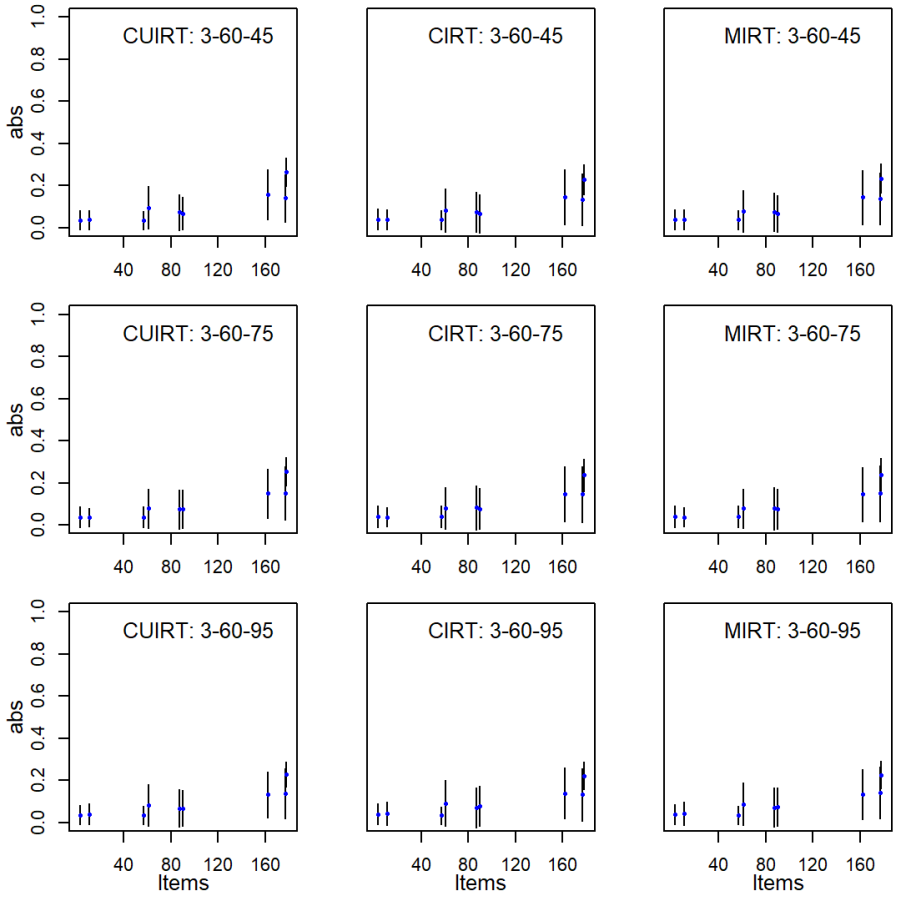


Figure H.11

Absolute Bias of d1-Parameter for the 4 Domain, 40 Items per Domain Tests

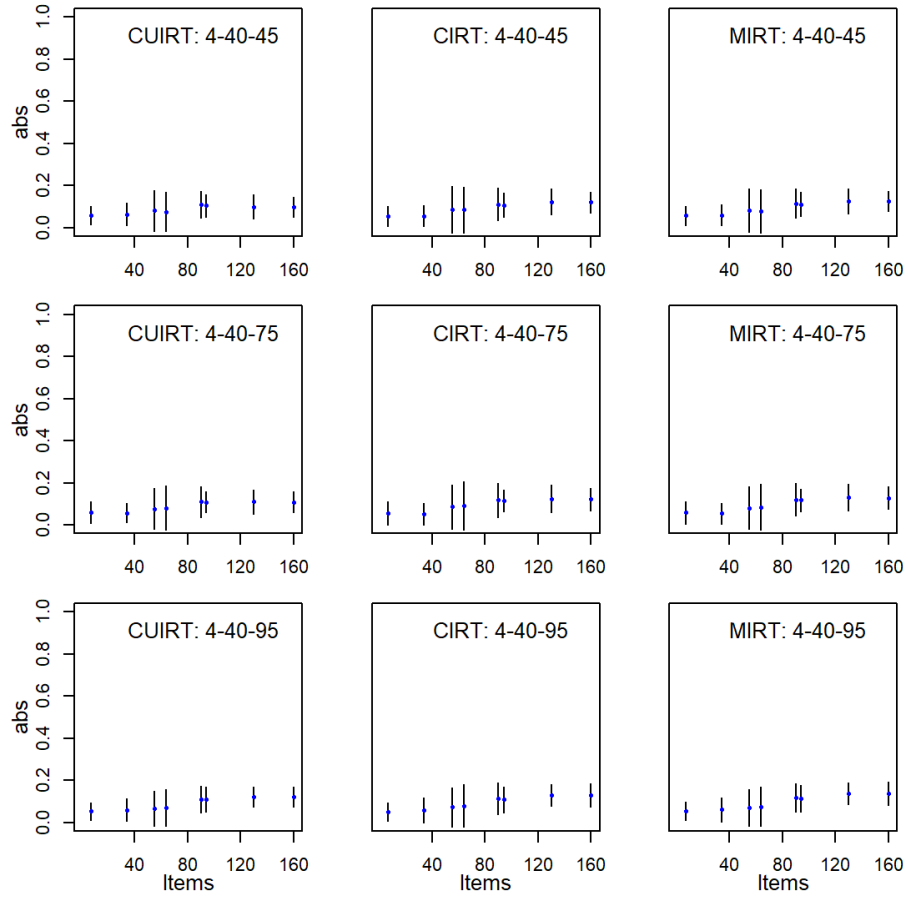
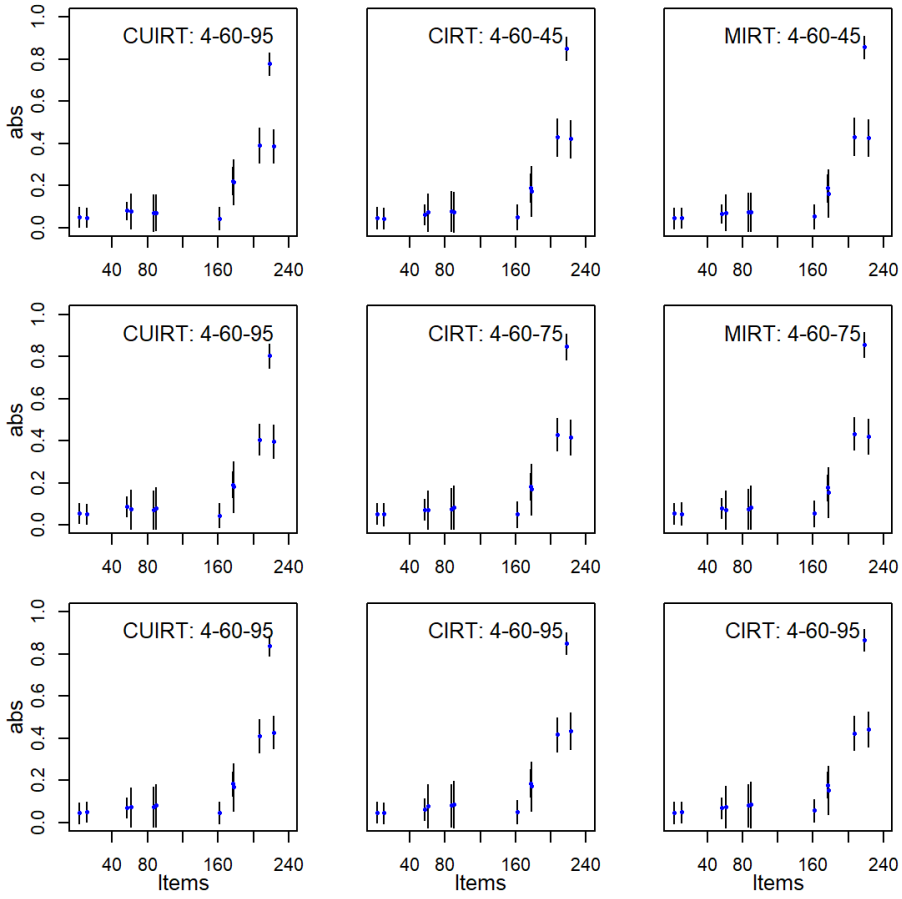


Figure H.12

Absolute Bias of d1-Parameter for the 4 Domain, 60 Items per Domain Tests



H.1.4 Item threshold $d2$

Figure H.13
Absolute Bias of d2-Parameter for the 3 Domain, 40 Items per Domain Tests

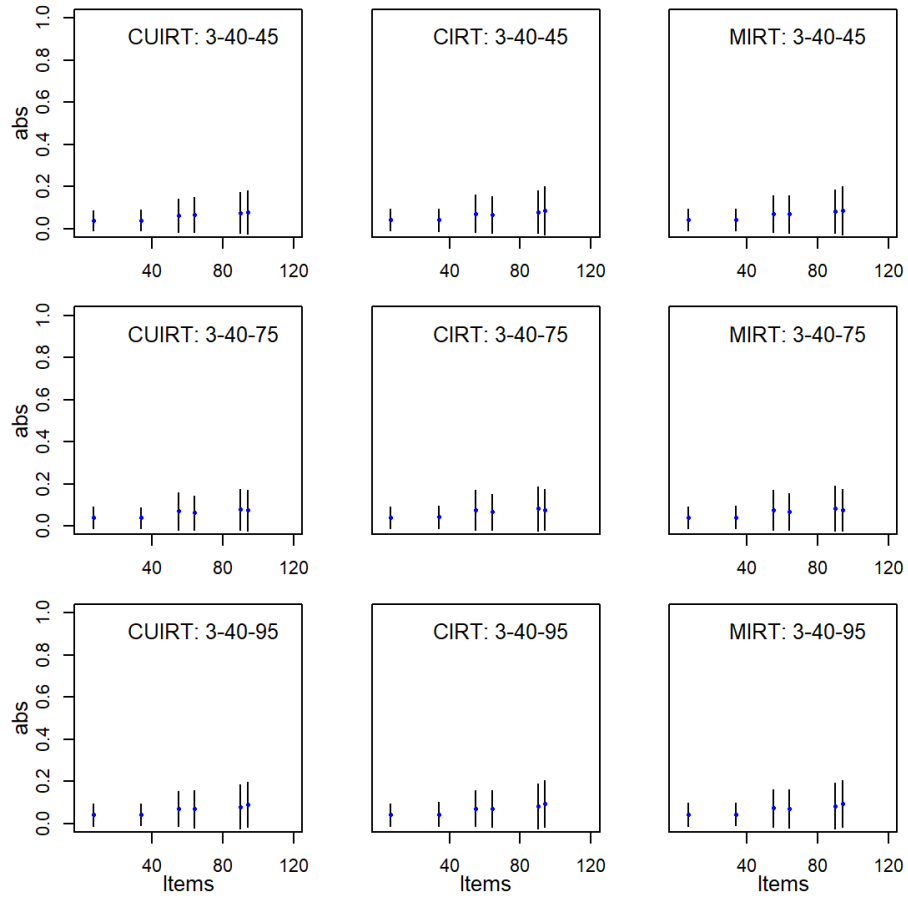


Figure H.14

Absolute Bias of d2-Parameter for the 3 Domain, 60 Items per Domain Tests

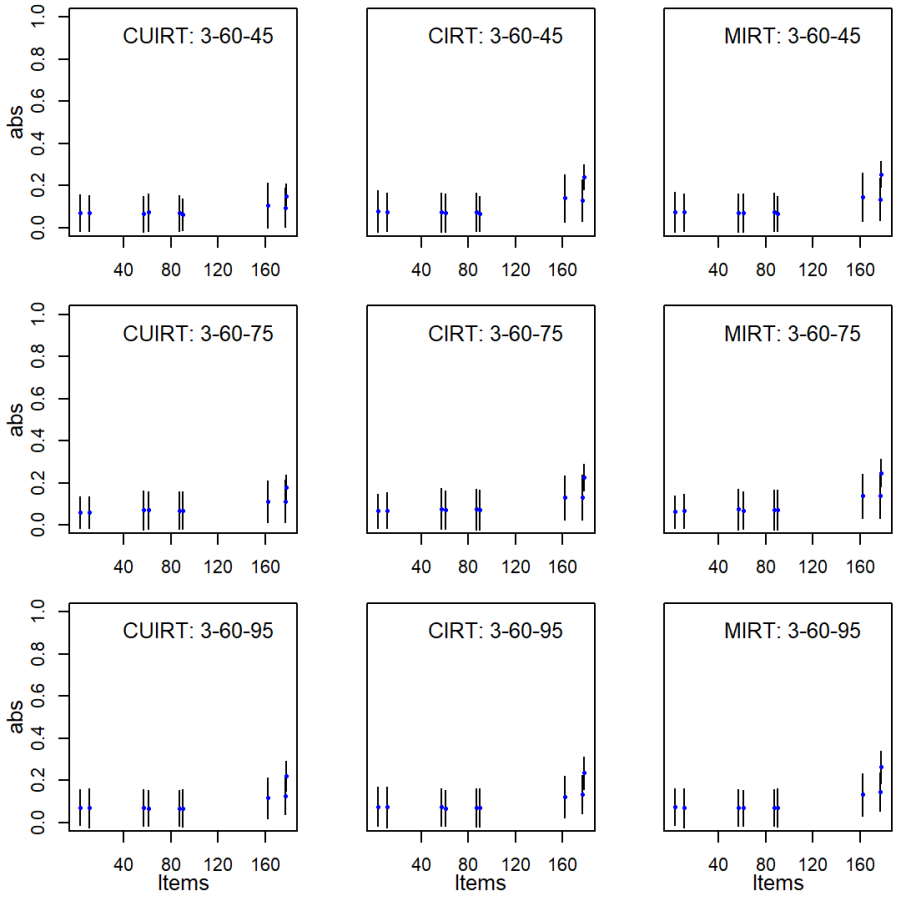


Figure H.15

Absolute Bias of d2-Parameter for the 4 Domain, 40 Items per Domain Tests

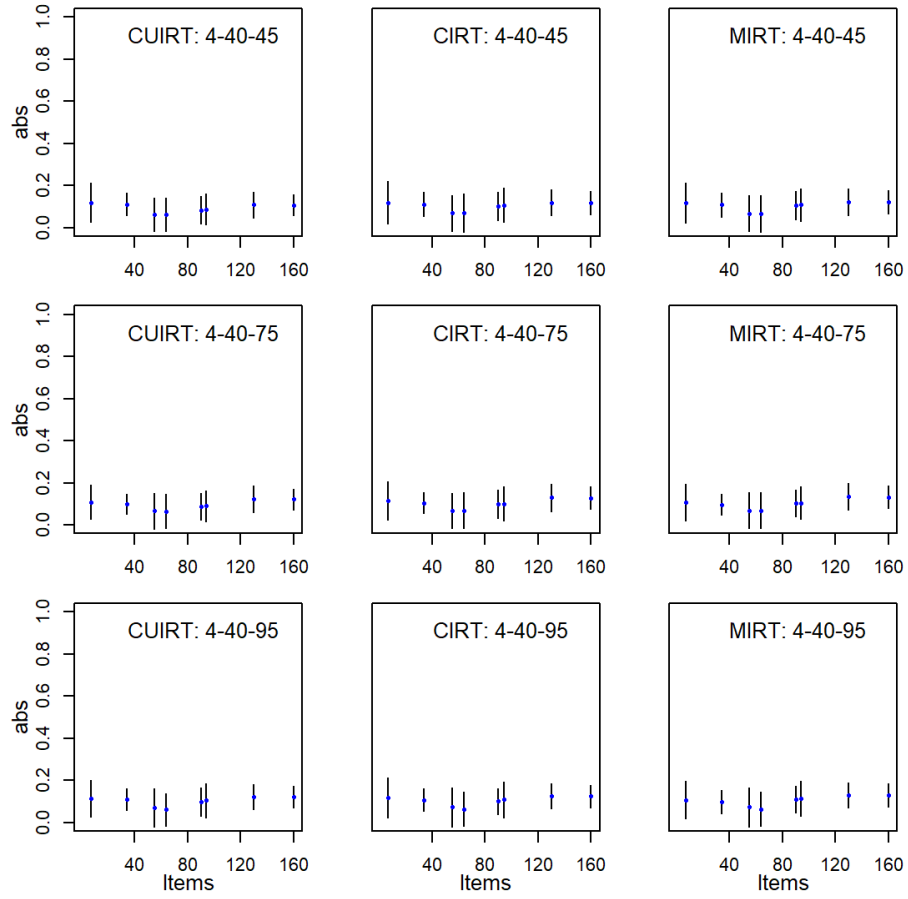
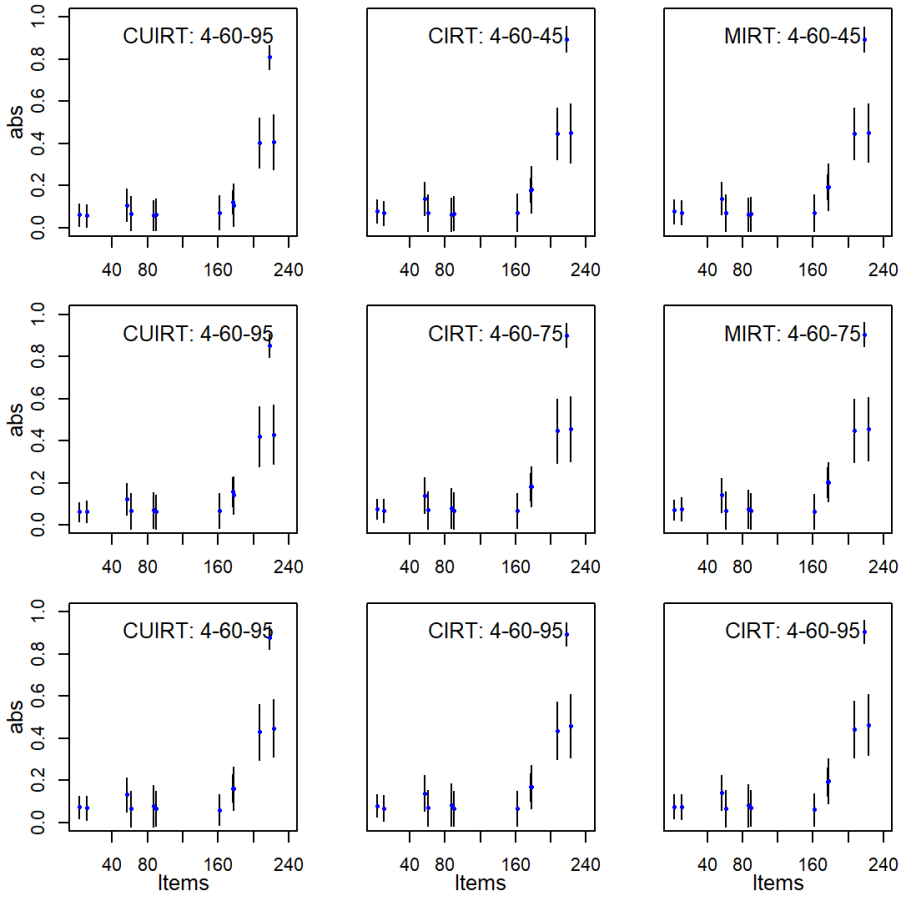


Figure H.16

Absolute Bias of d2-Parameter for the 4 Domain, 60 Items per Domain Tests



H.2 RMSE

H.2.1 Item Discrimination

Figure H.17
RMSE of a-Parameter for the 3 Domain, 40 Items per Domain Tests

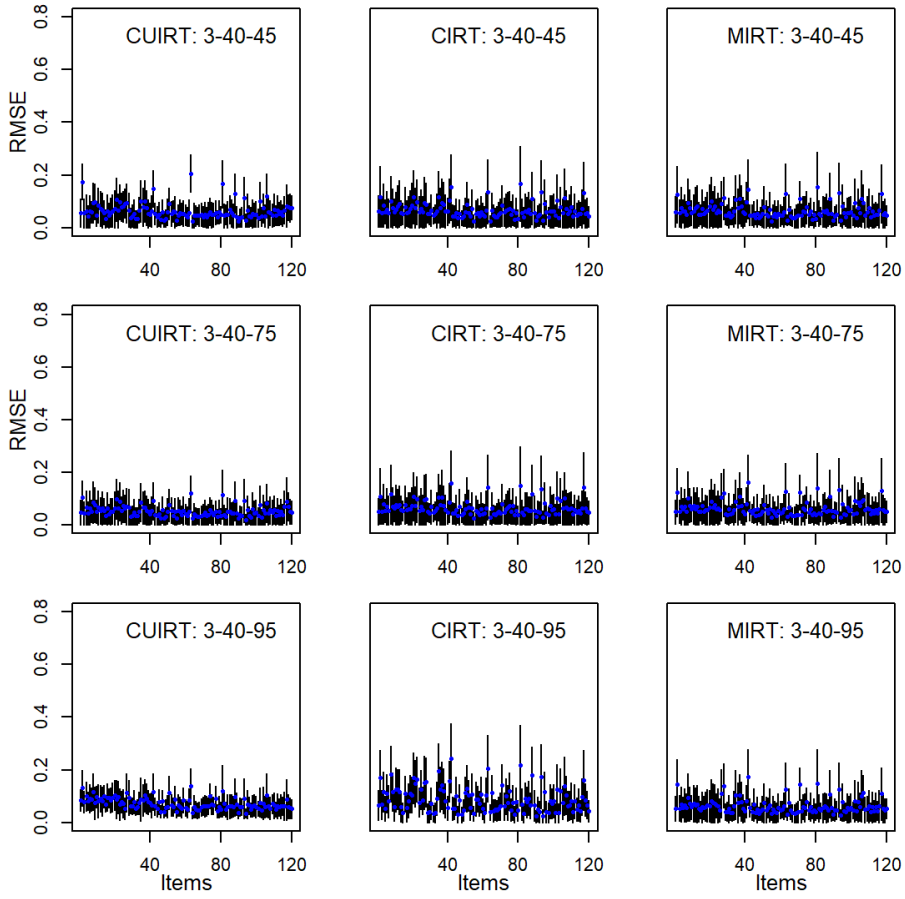


Figure H.18

RMSE of a-Parameter for the 3 Domain, 60 Items per Domain Tests

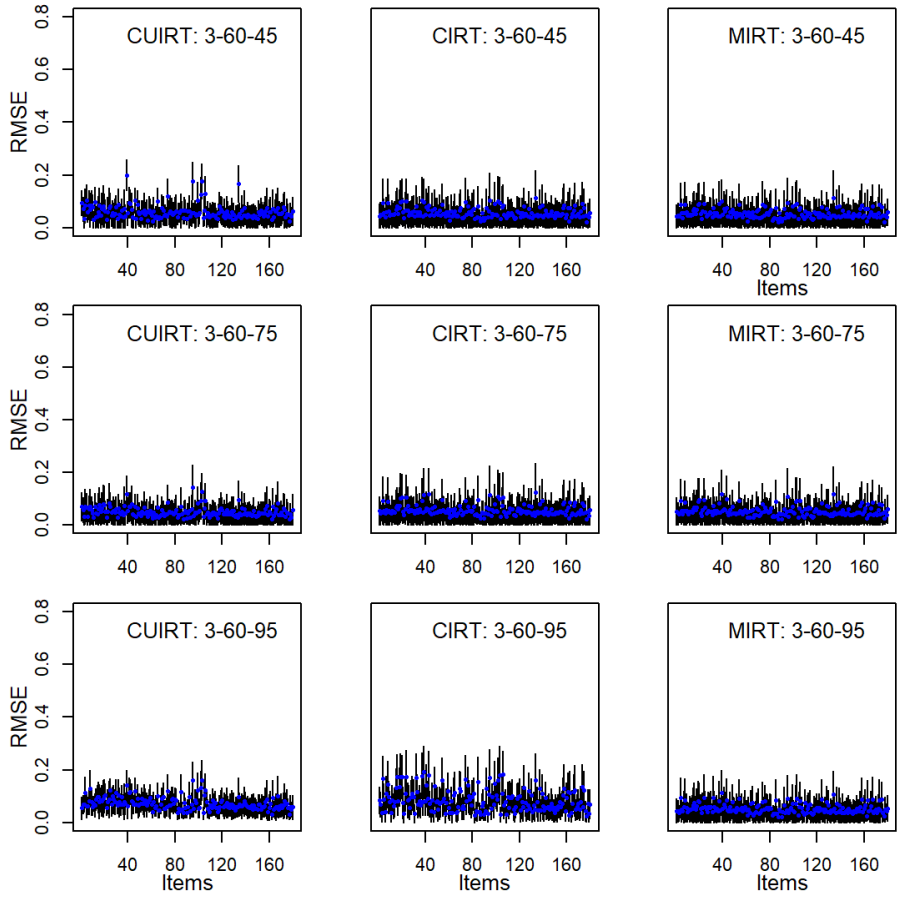


Figure H.19

RMSE of a-Parameter for the 4 Domain, 40 Items per Domain Tests

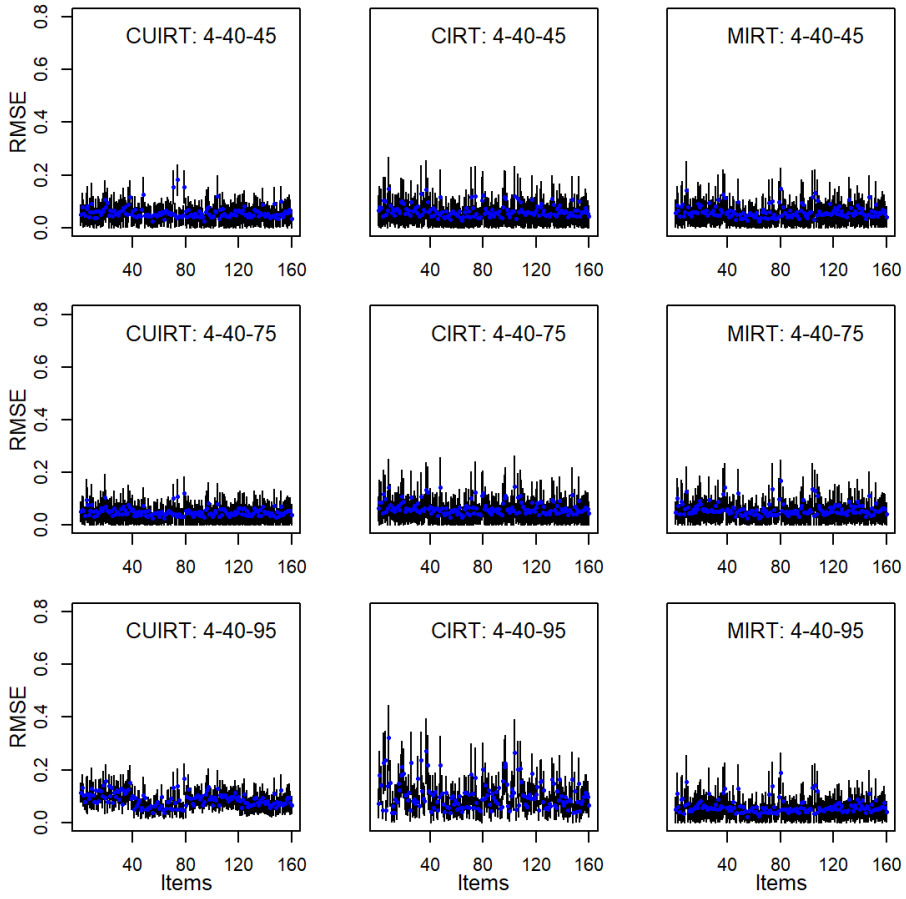
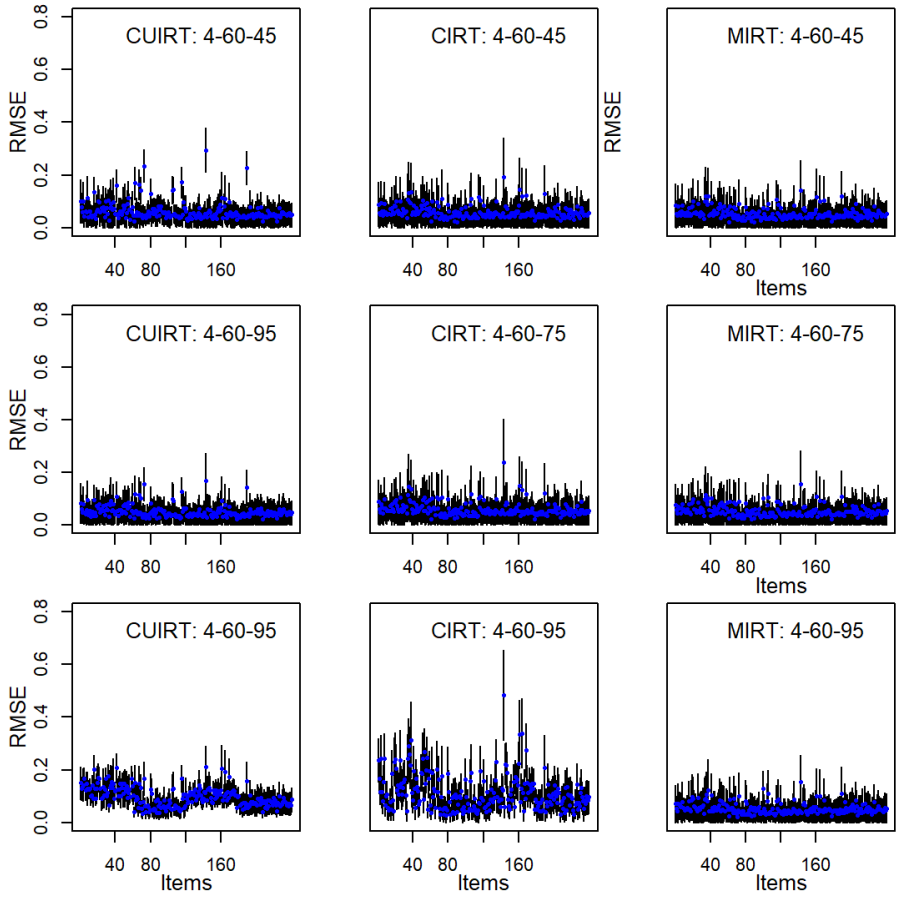


Figure H.20

RMSE of a-Parameter for the 4 Domain, 60 Items per Domain Tests



H.2.2 Item Difficulty

Figure H.21
RMSE of b-Parameter for the 3 Domain, 40 Items per Domain Tests

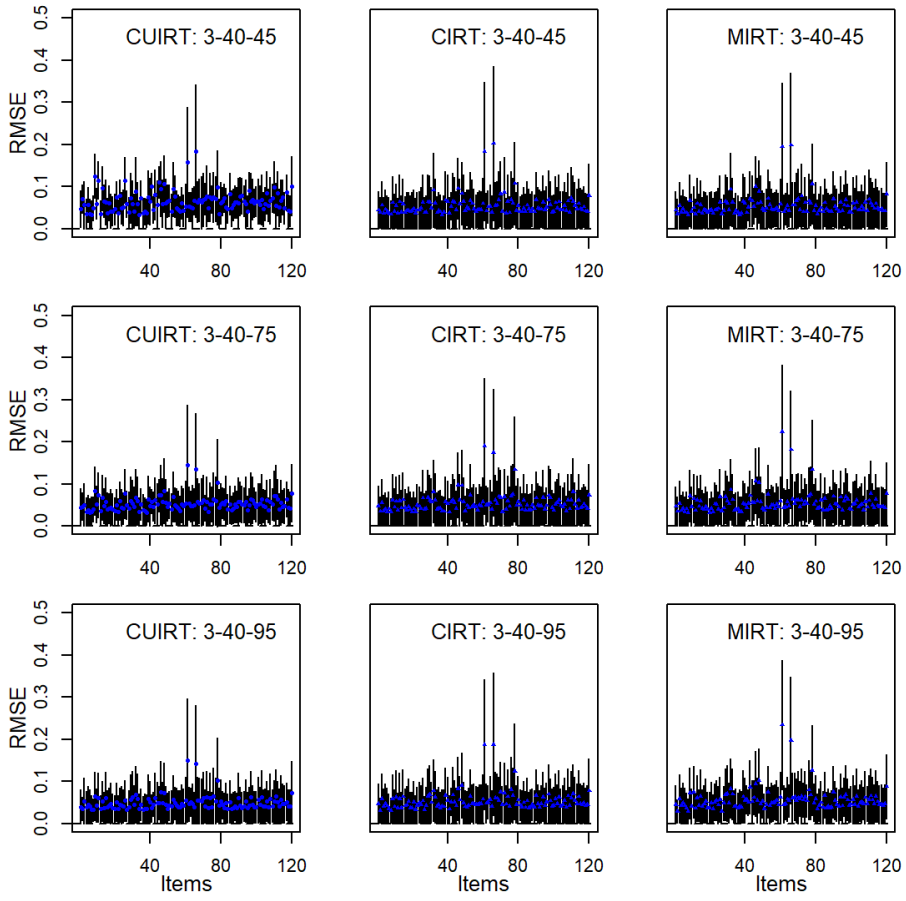


Figure H.22

RMSE of b-Parameter for the 3 Domain, 60 Items per Domain Tests

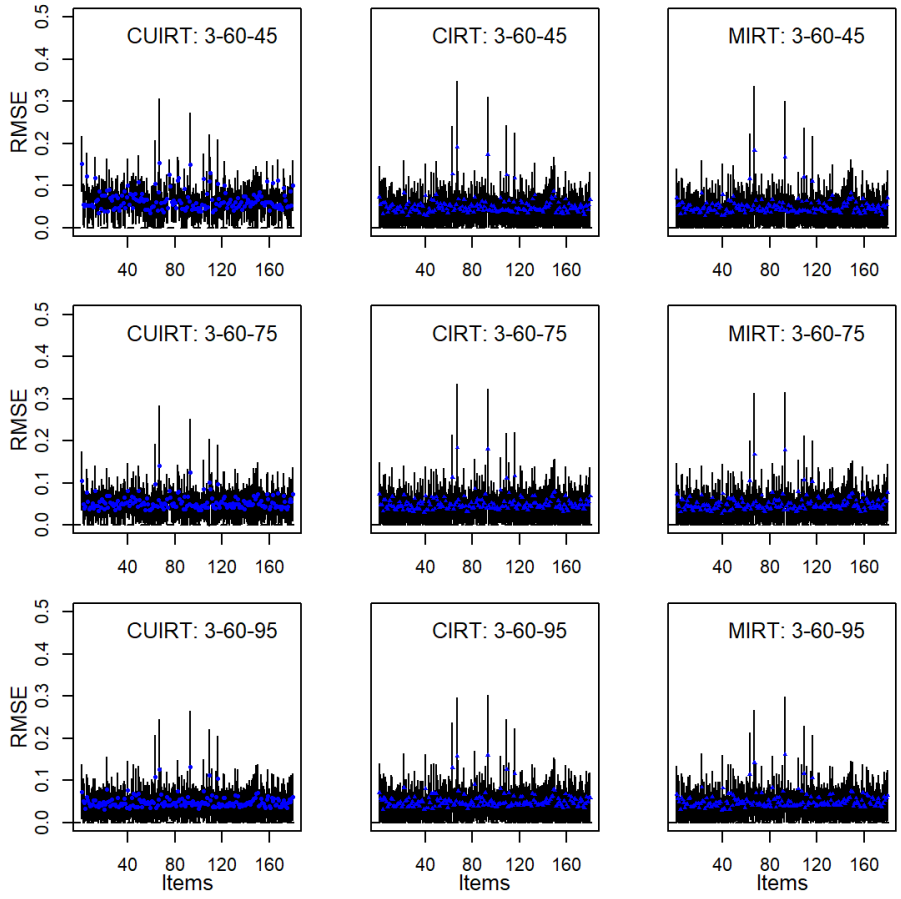


Figure H.23

RMSE of b-Parameter for the 4 Domain, 40 Items per Domain Tests

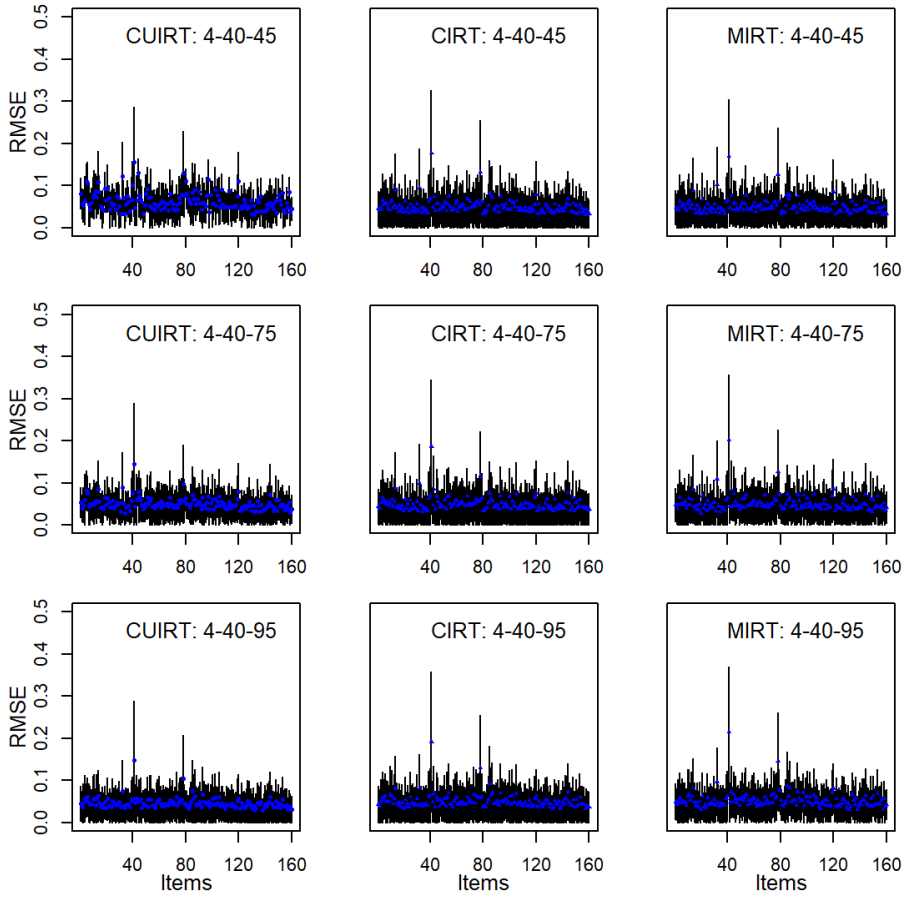
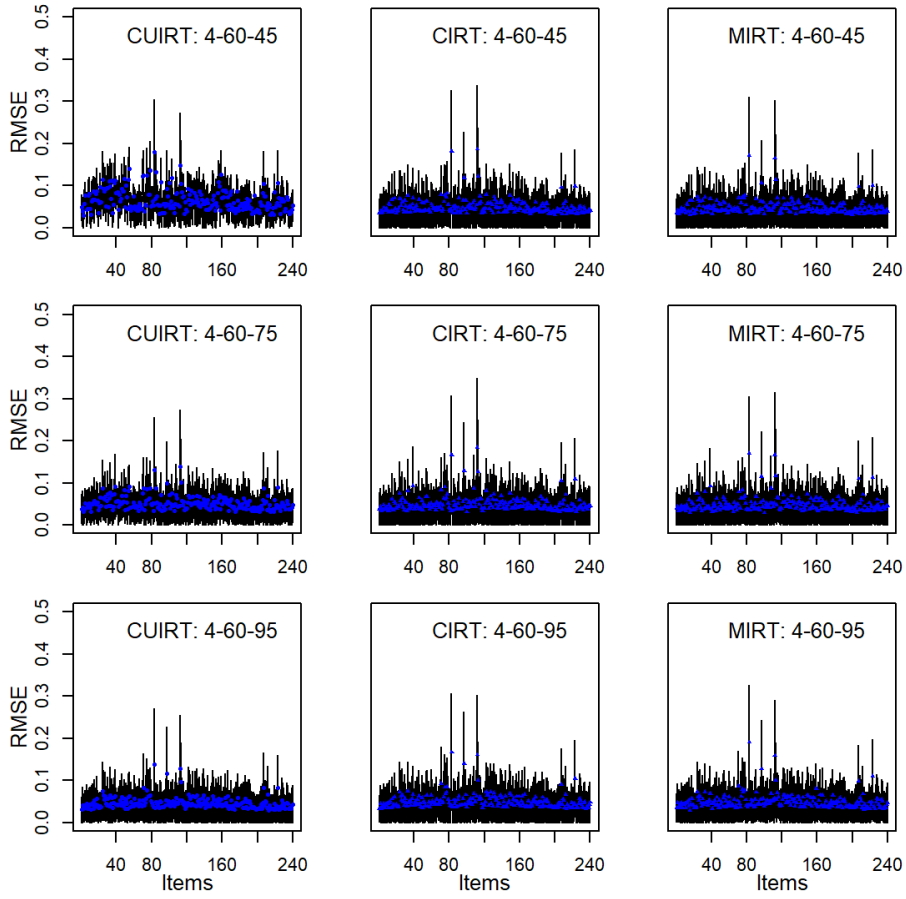


Figure H.24

RMSE of b-Parameter for the 4 Domain, 60 Items per Domain Tests



H.2.3 Item threshold $d1$

Figure H.25
RMSE of d1-Parameter for the 3 Domain, 40 Items per Domain Tests

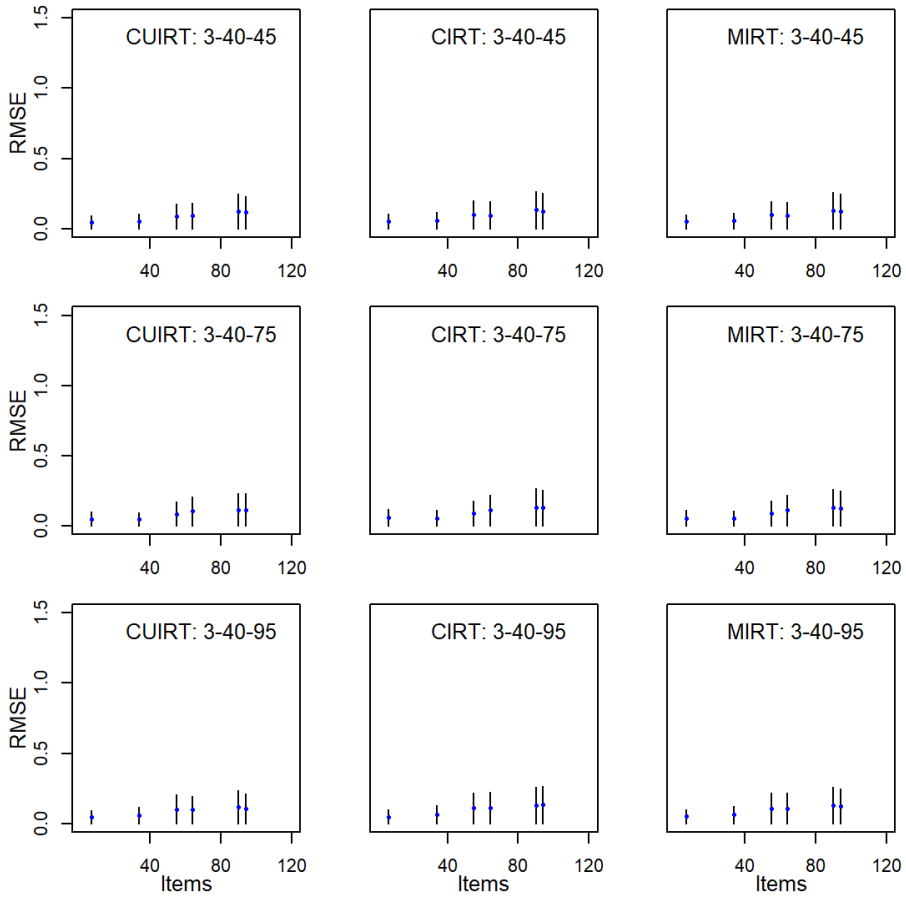


Figure H.26

RMSE of d1-Parameter for the 3 Domain, 60 Items per Domain Tests

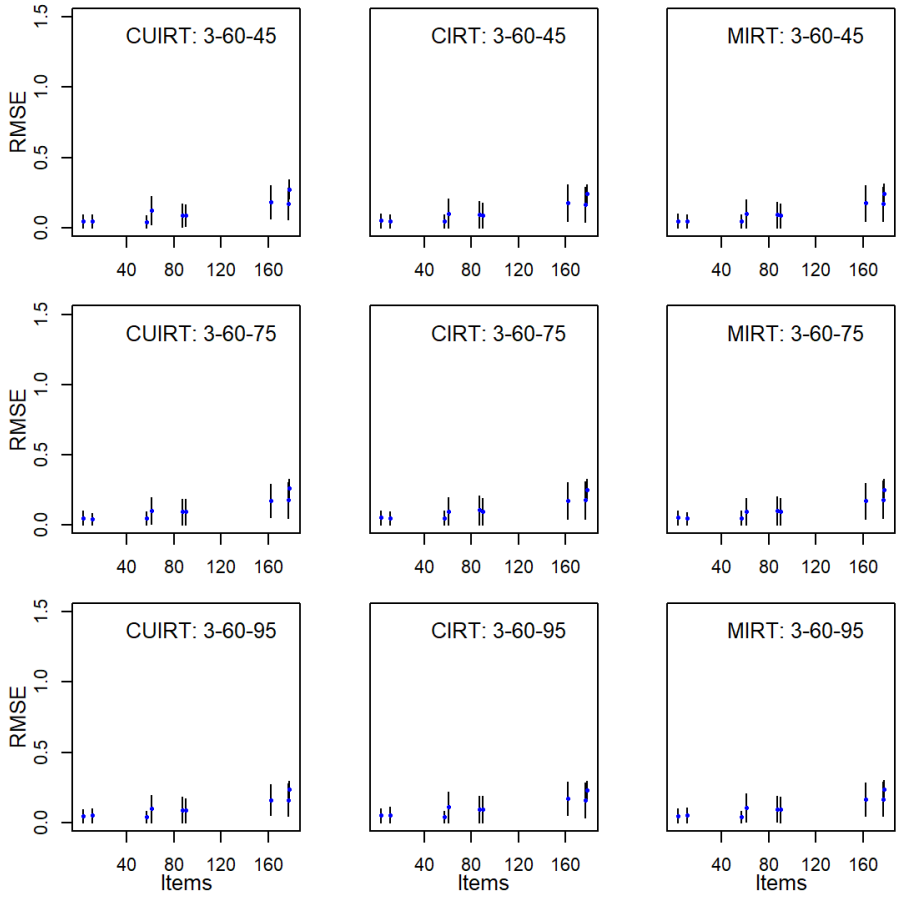


Figure H.27
RMSE of d1-Parameter for the 4 Domain, 40 Items per Domain Tests

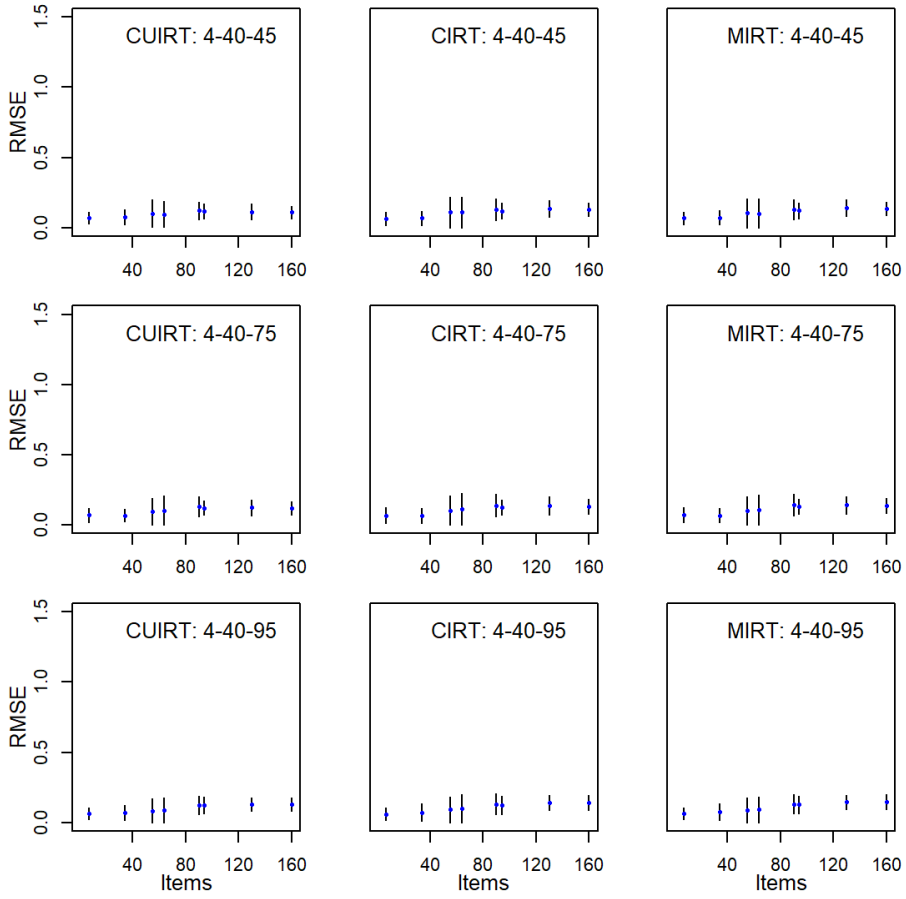
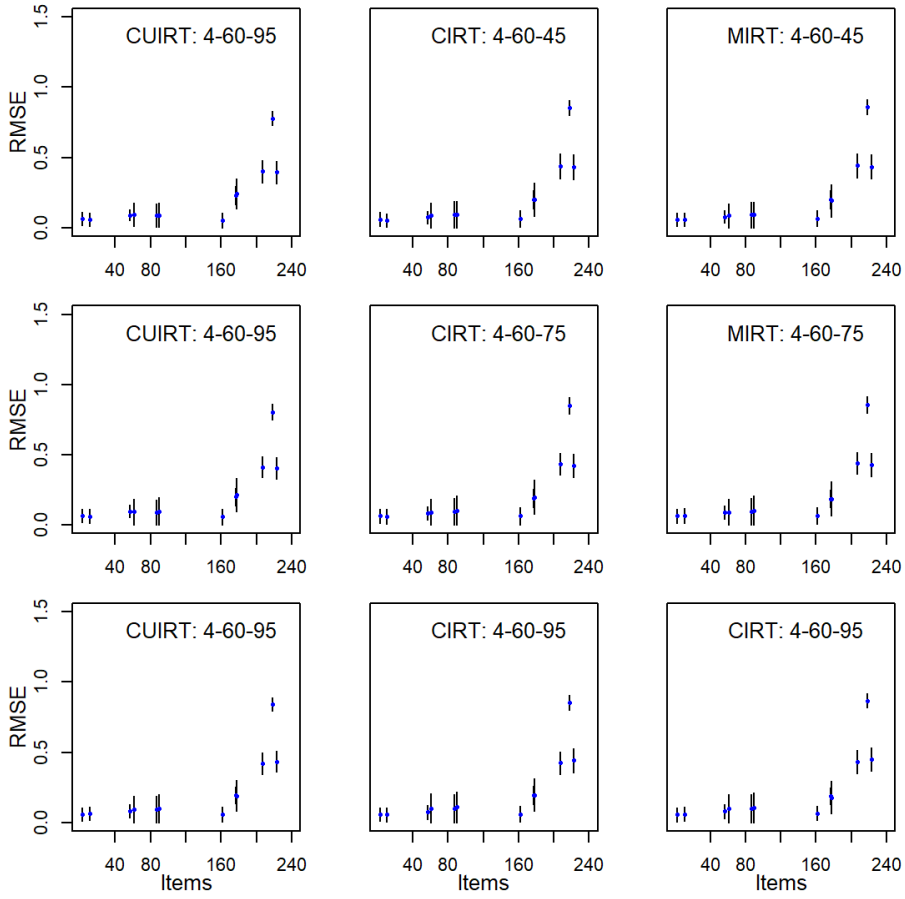


Figure H.28

RMSE of d1-Parameter for the 4 Domain, 60 Items per Domain Tests



H.2.4 Item threshold $d2$

Figure H.29
RMSE of d2-Parameter for the 3 Domain, 40 Items per Domain Tests

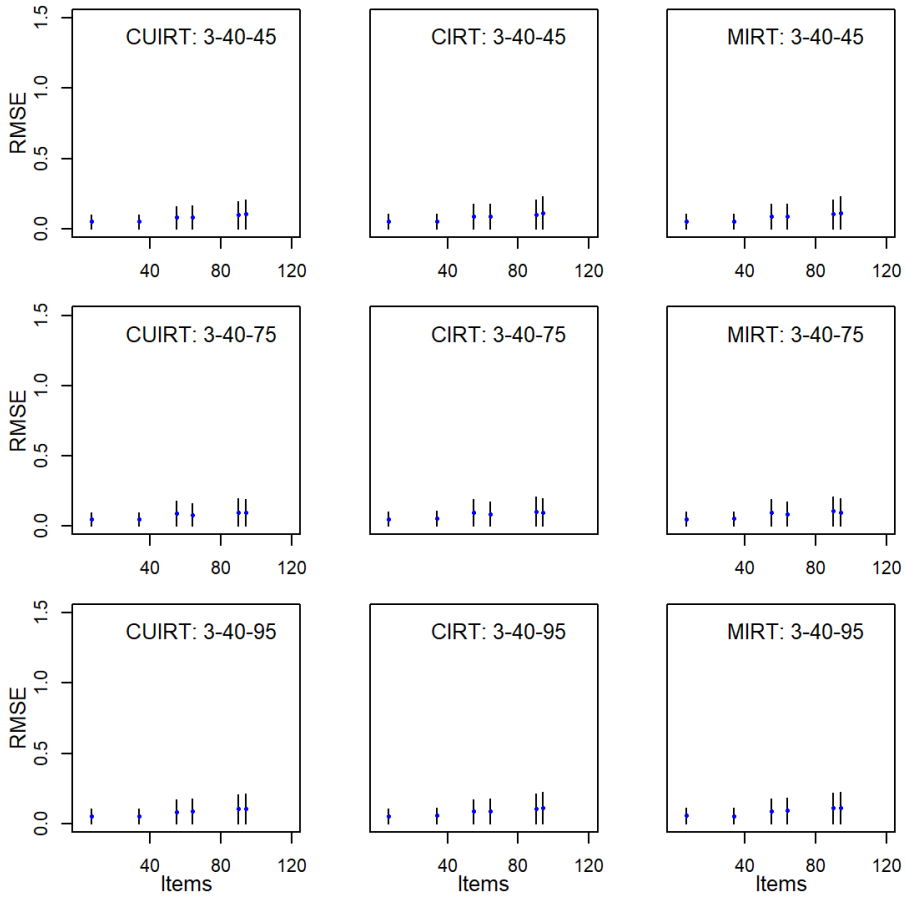


Figure H.30

RMSE of d2-Parameter for the 3 Domain, 60 Items per Domain Tests

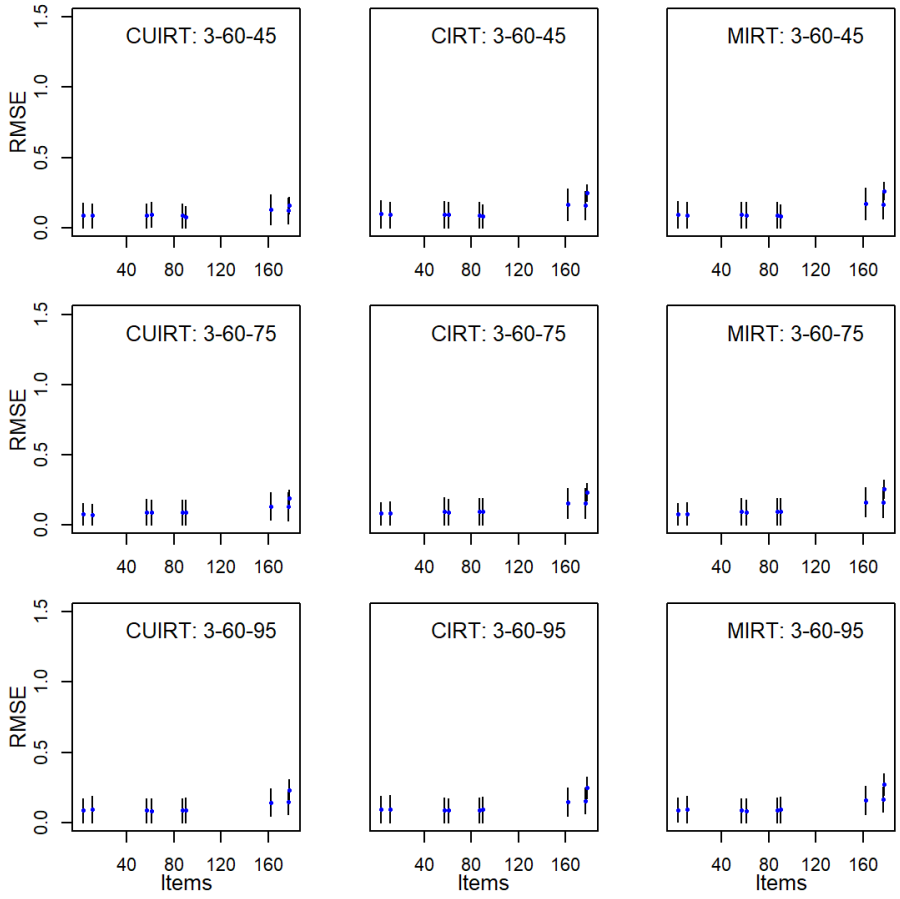


Figure H.31
RMSE of d2-Parameter for the 4 Domain, 40 Items per Domain Tests

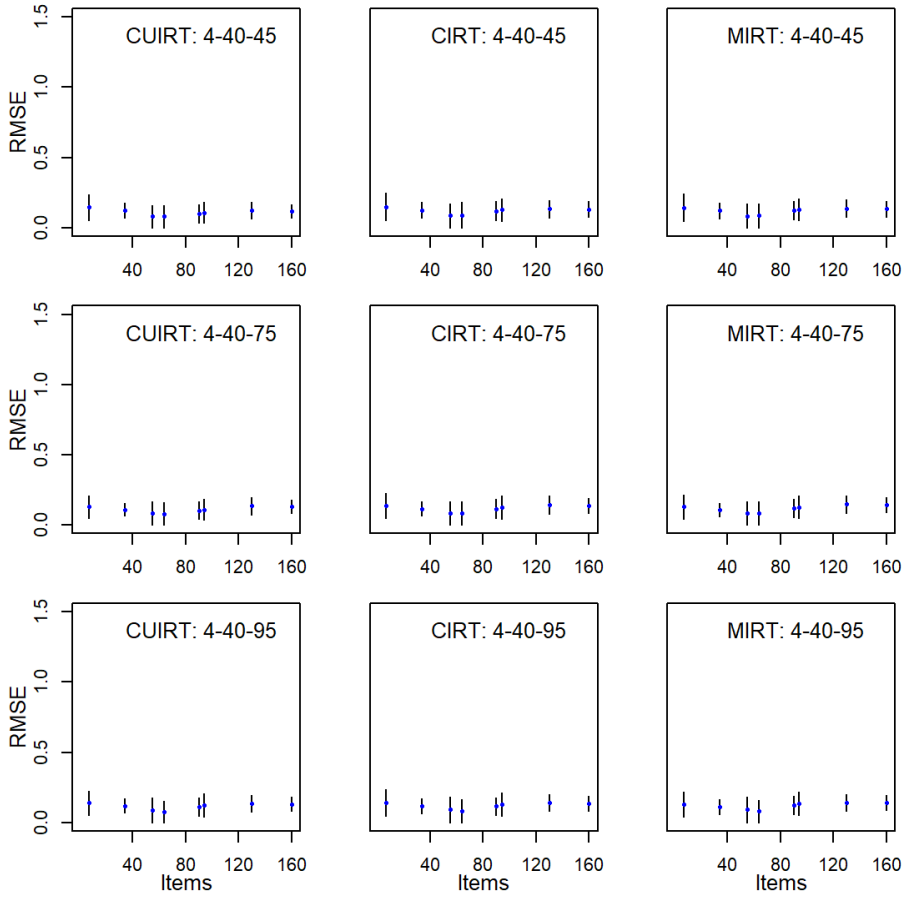
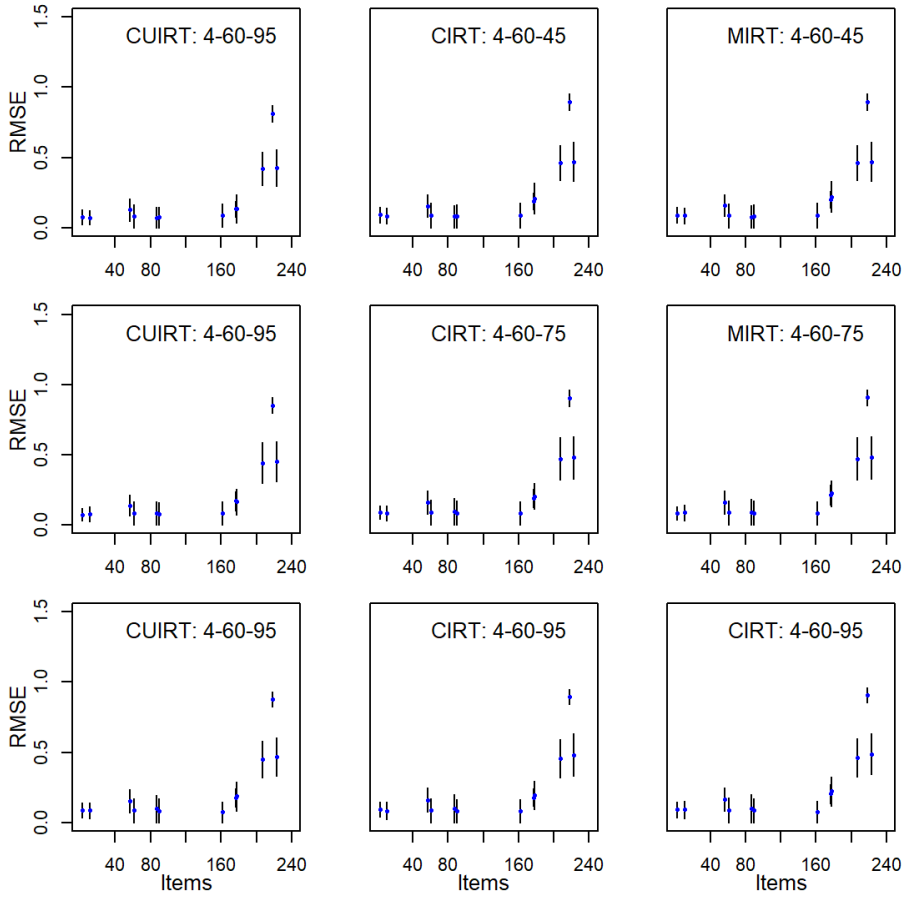


Figure H.32

RMSE of d2-Parameter for the 4 Domain, 60 Items per Domain Tests



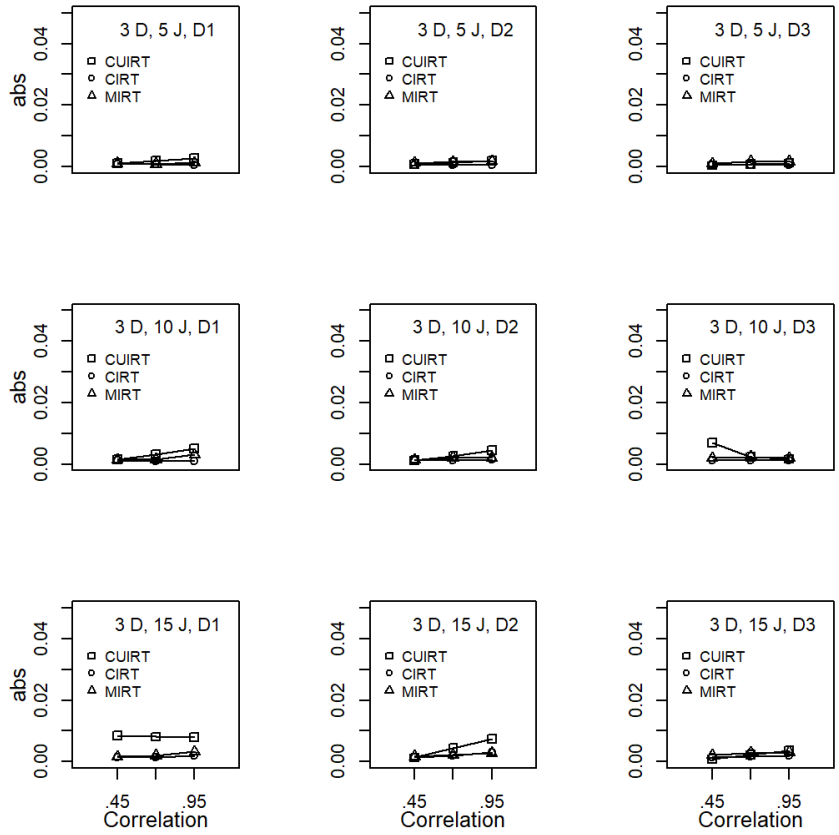
Appendix I

Study 1 Score Parameter ABS and RMSE: Single Groups

I.1 ABS

Figure I.1

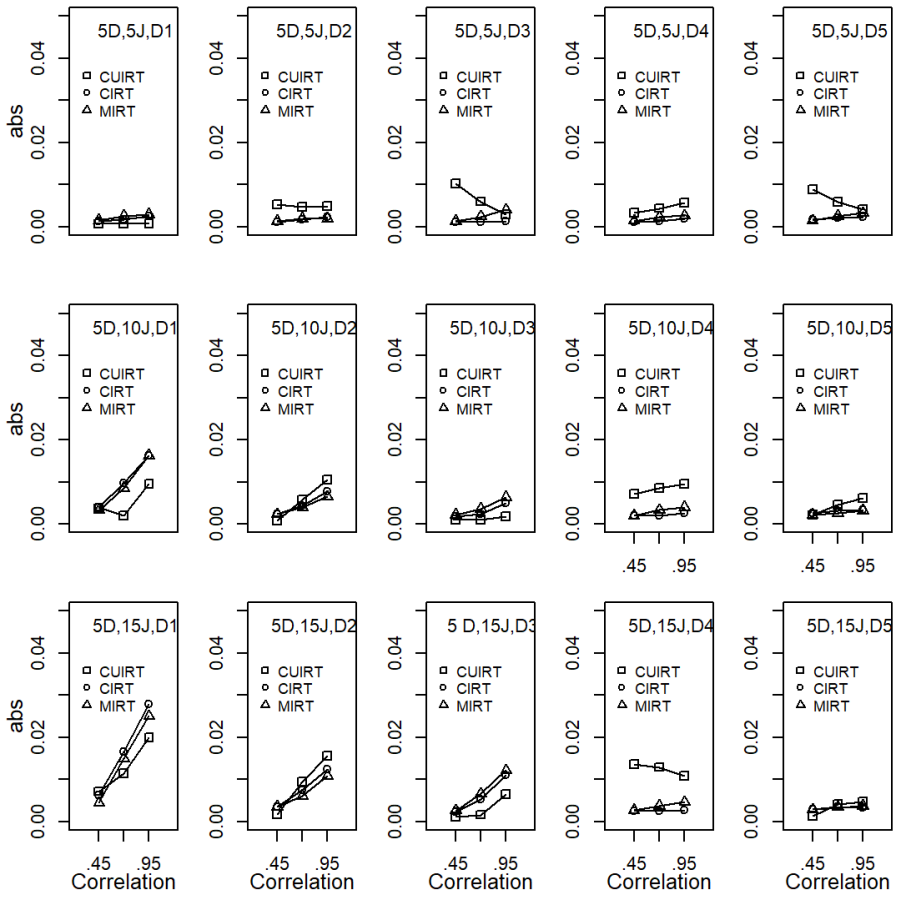
Subscale Score ABS for the 3 Subdomain Tests



Note. D = number of domains; J = number of items; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

Figure I.2

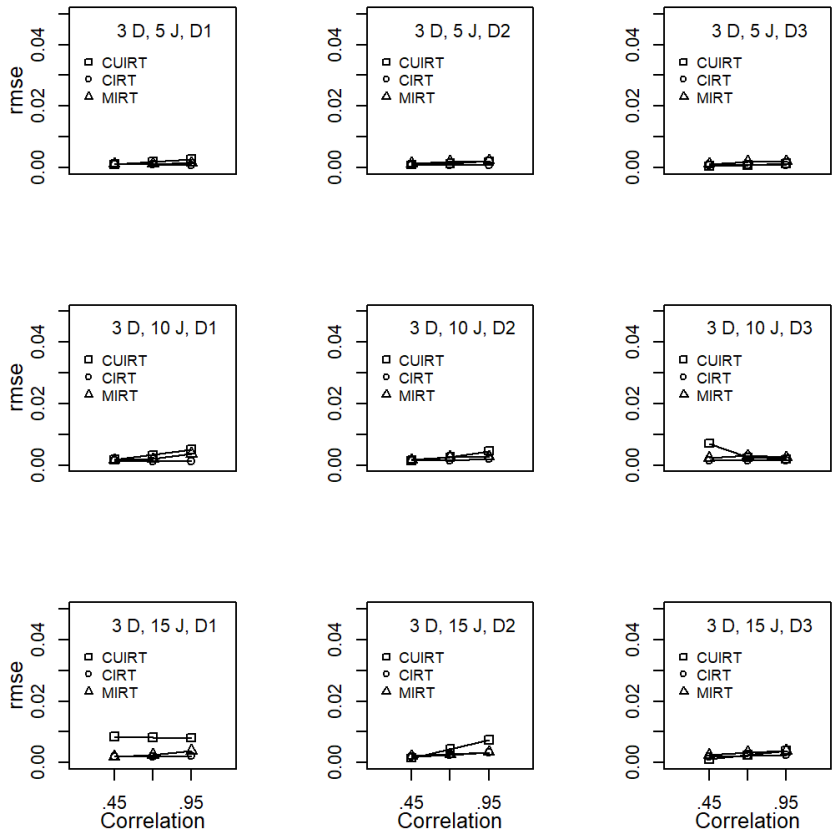
Subscale Score ABS for the 5 Subdomain Tests



Note. D = number of domains; J = number of items; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4; $D5$ = domain 5.

I.2 RMSE

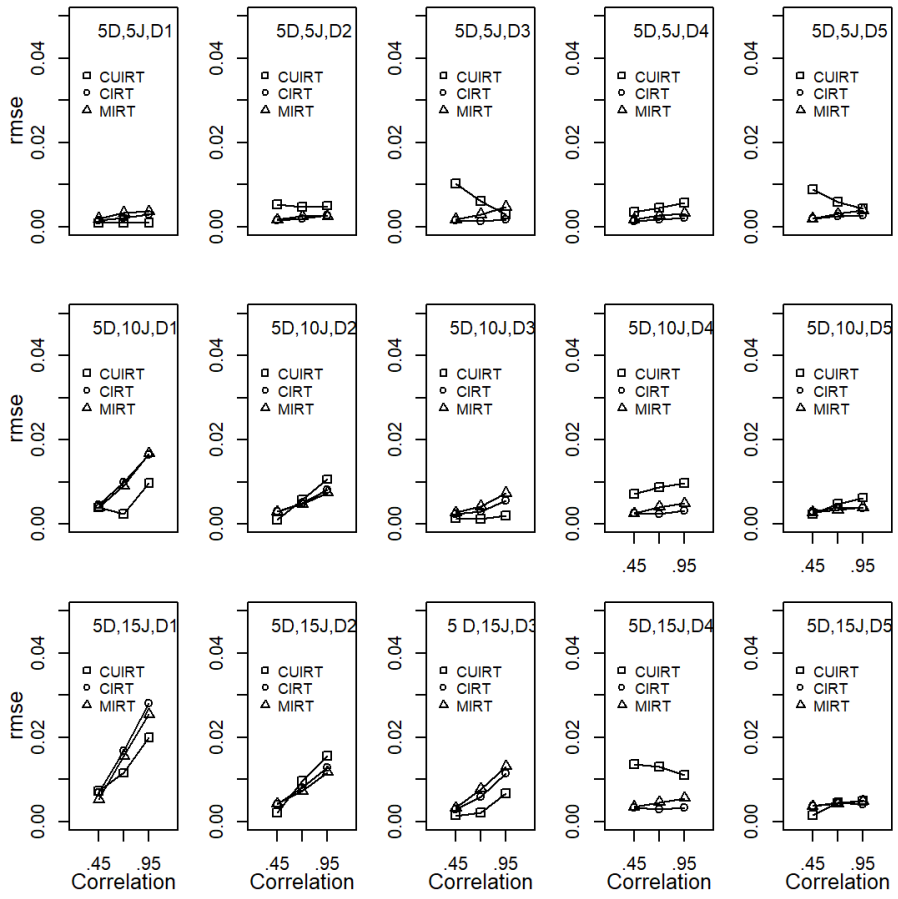
Figure I.3
Subscale Score RMSE for the 3 Subdomain Tests



Note. D = number of domains; J = number of items; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

Figure I.4

Subscale Score RMSE for the 5 Subdomain Tests



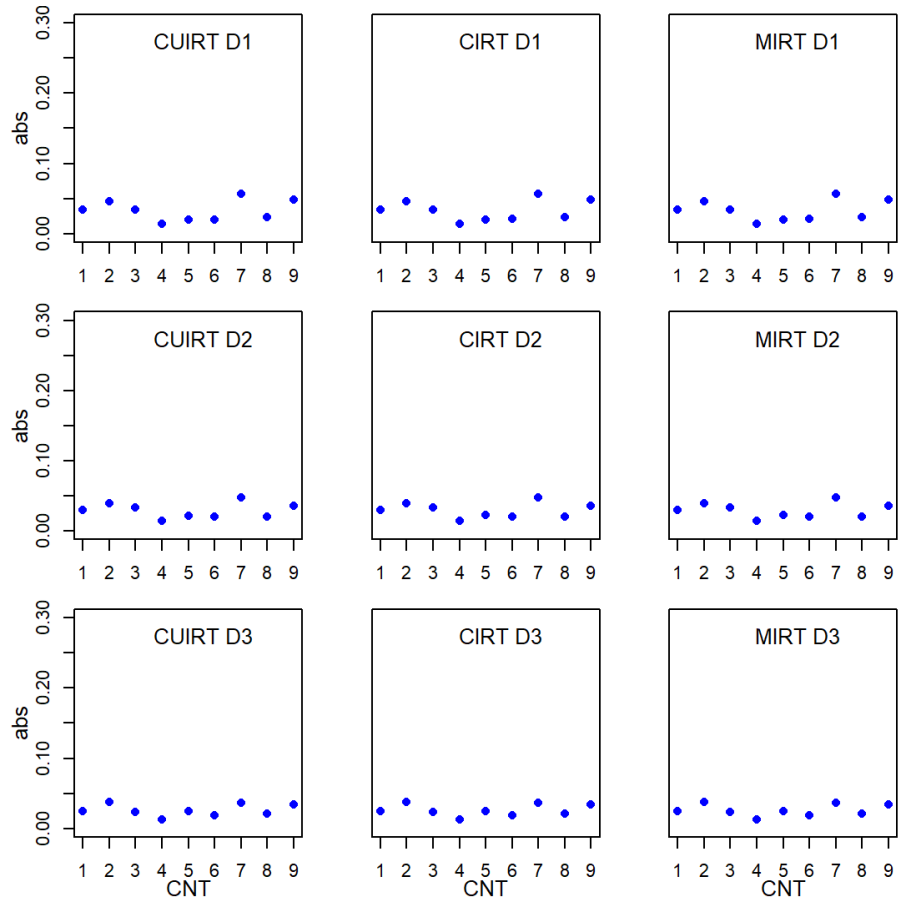
Note. D = number of domains; J = number of items; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4; $D5$ = domain 5.

Appendix J

Study 1 Score Parameter ABS and RMSE: Multiple Groups

J.1 ABS

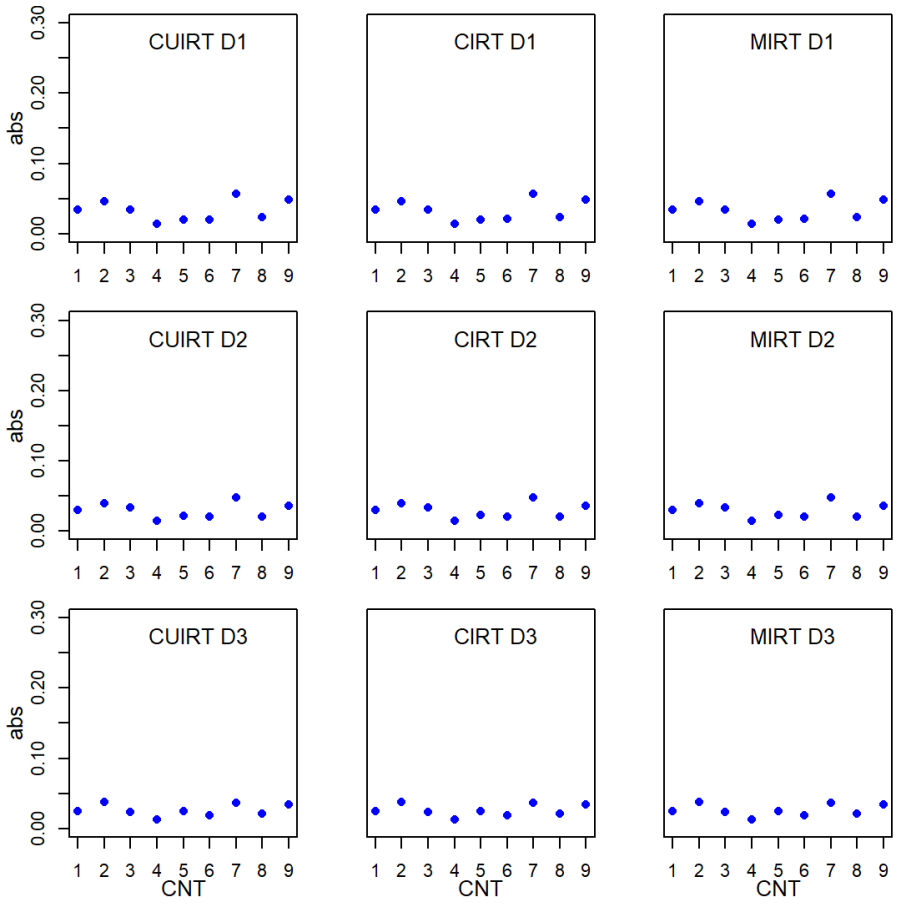
Figure J.1
Subscale score ABS for the 3-Domain, 5-item, .45 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

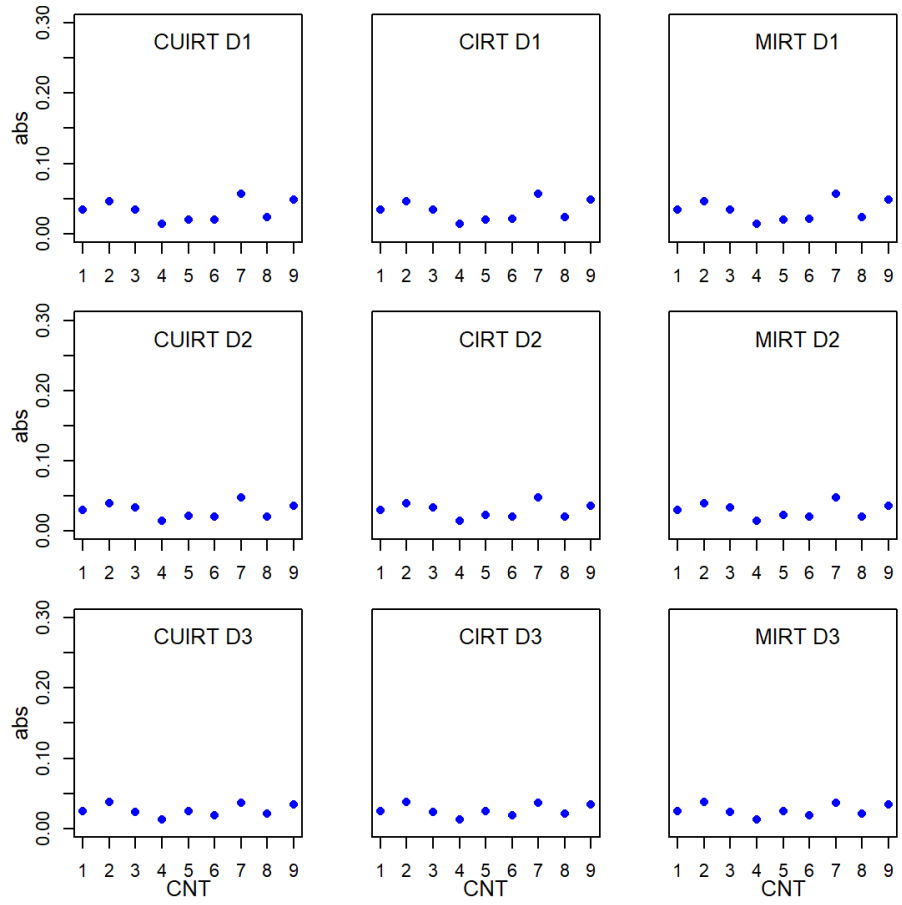
Figure J.2

Subscale score ABS for the 3-Domain, 5-item, .75 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

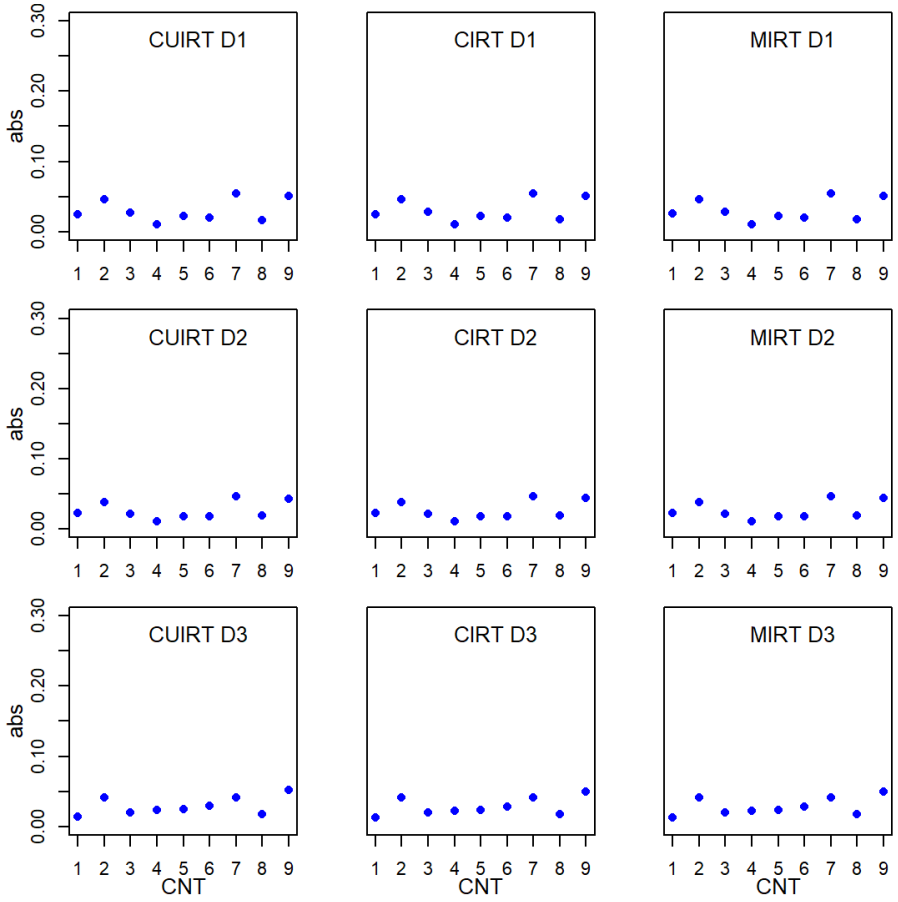
Figure J.3
Subscale score ABS for the 3-Domain, 5-item, .95 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

Figure J.4

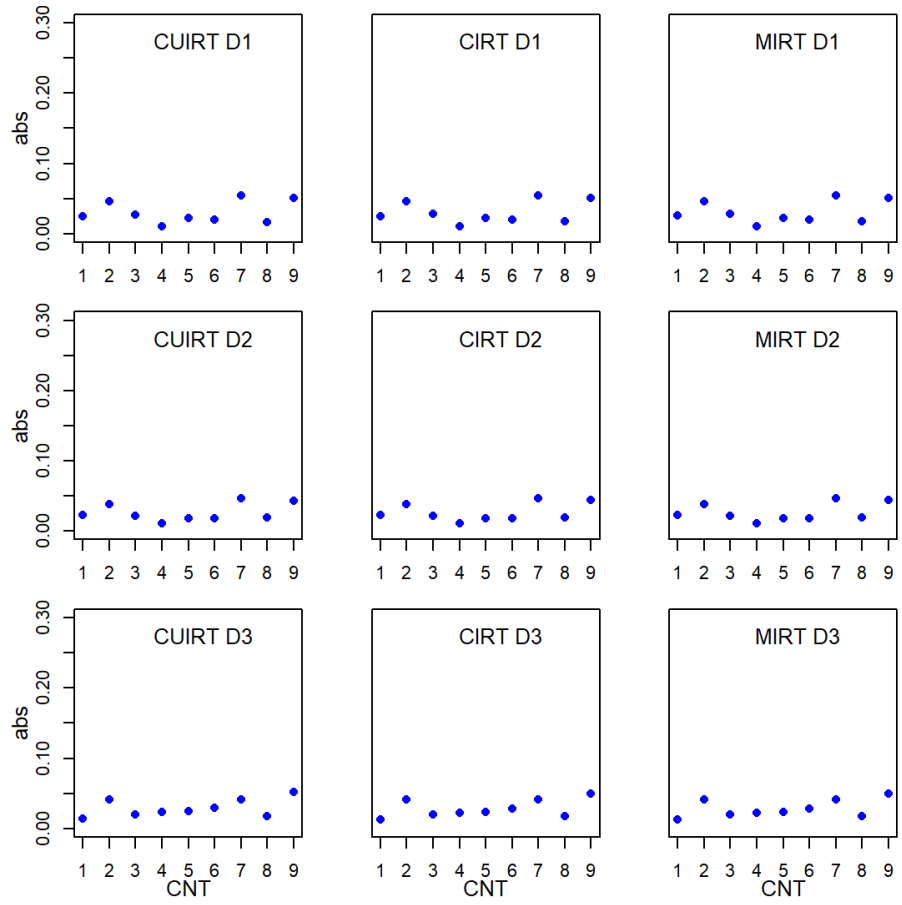
Subscale score ABS for the 3-Domain, 10-item, .45 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

Figure J.5

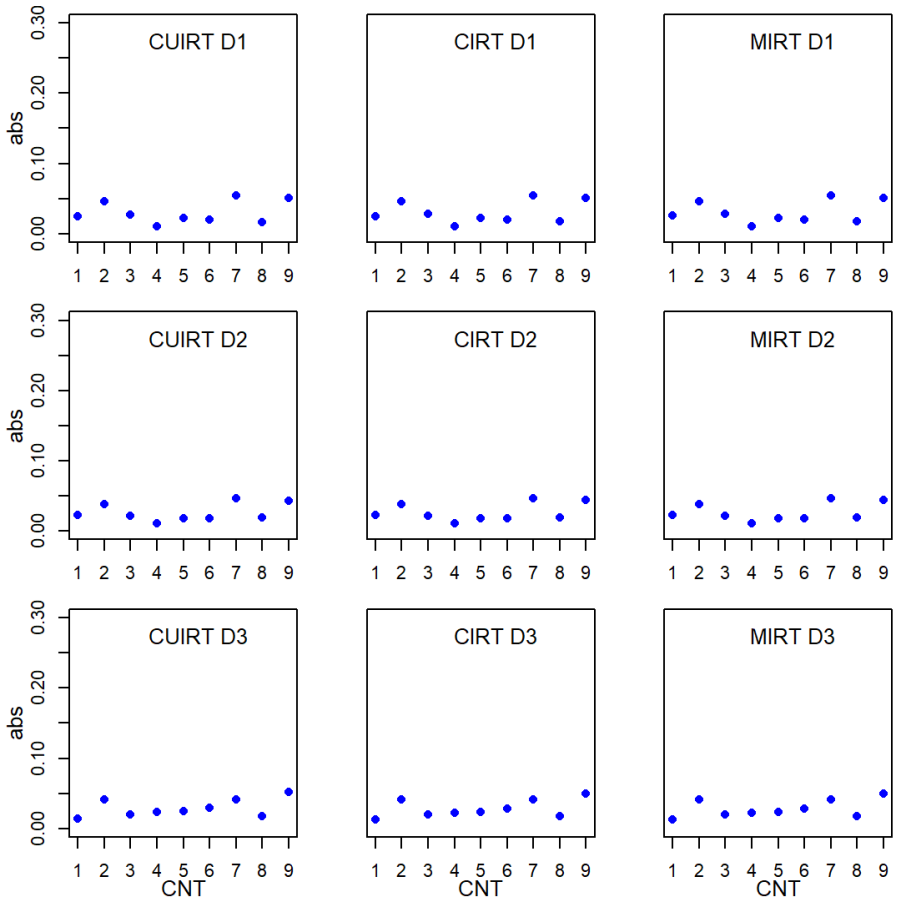
Subscale score ABS for the 3-Domain, 10-item, .75 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

Figure J.6

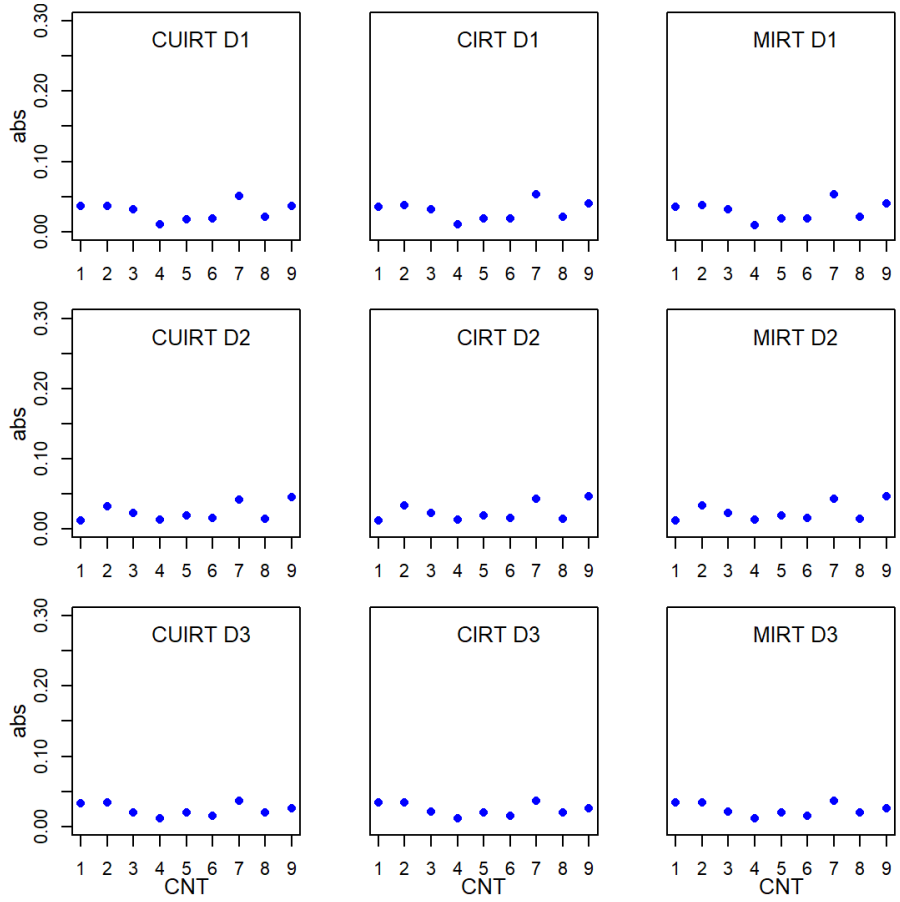
Subscale score ABS for the 3-Domain, 10-item, .95 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

Figure J.7

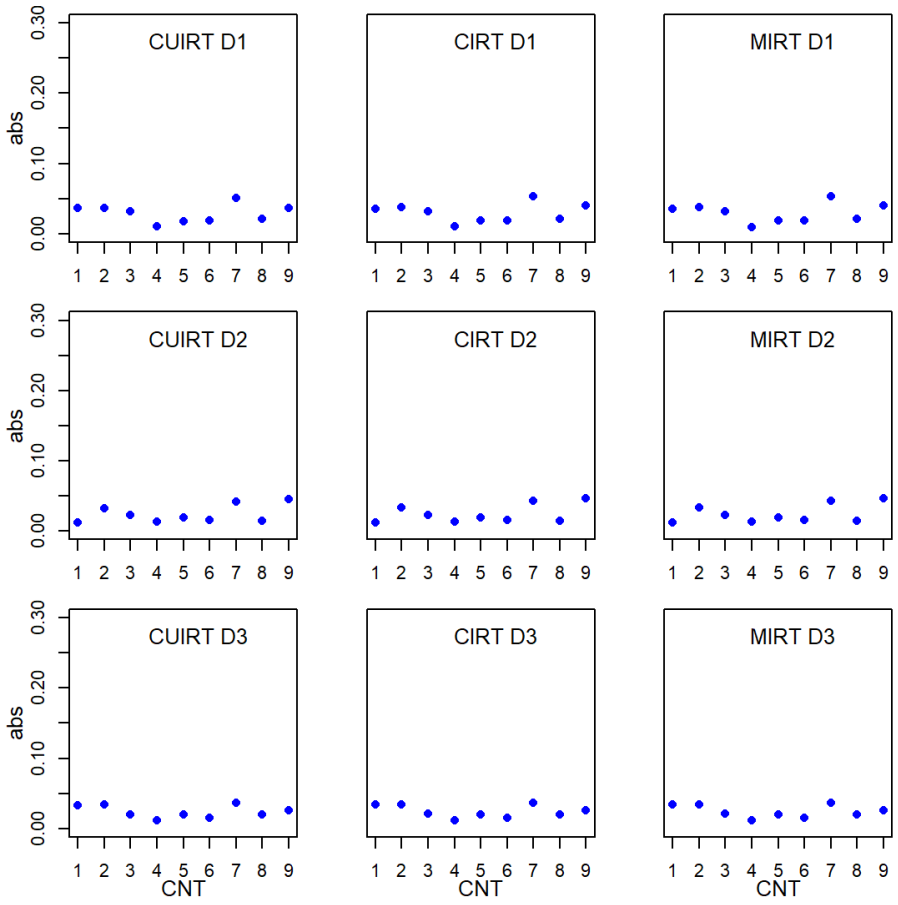
Subscale score ABS for the 3-Domain, 15-item, .45 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

Figure J.8

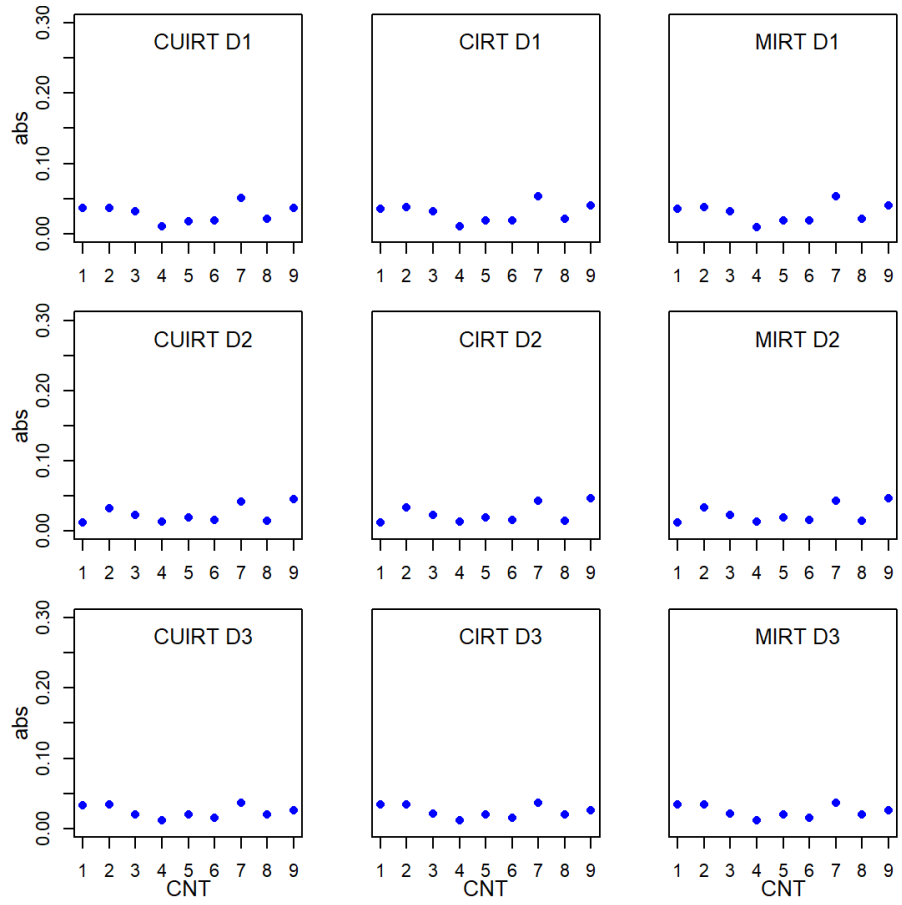
Subscale score ABS for the 3-Domain, 15-item, .75 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

Figure J.9

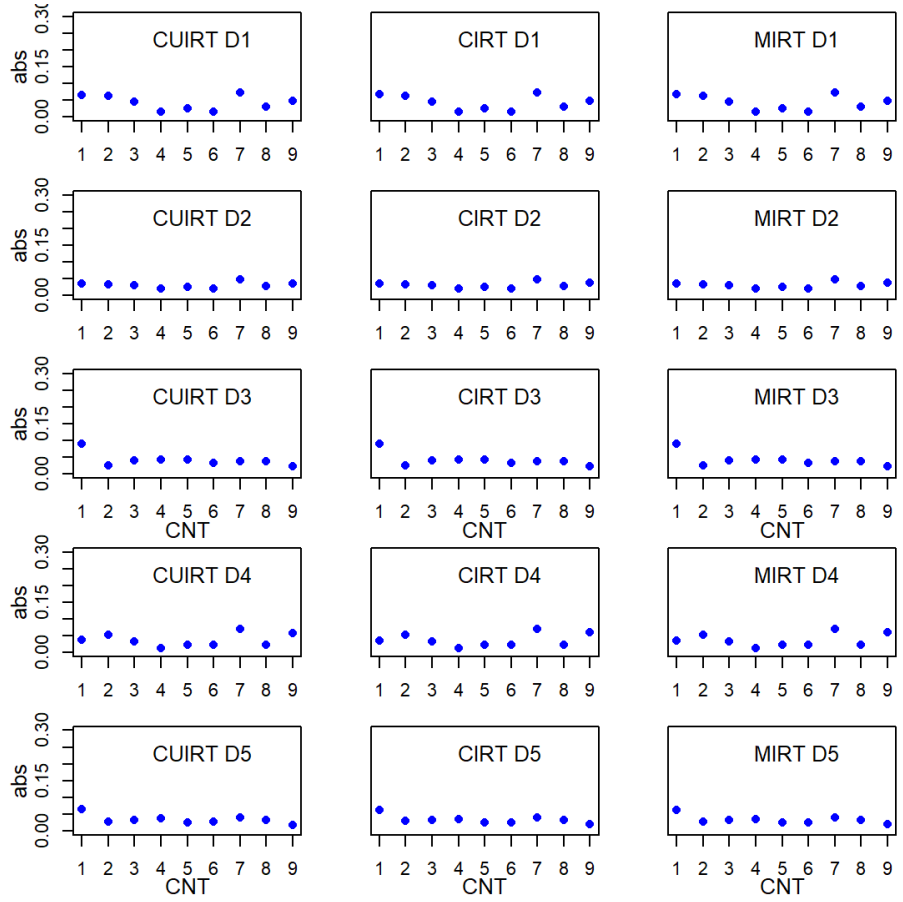
Subscale score ABS for the 3-Domain, 15-item, .95 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

Figure J.10

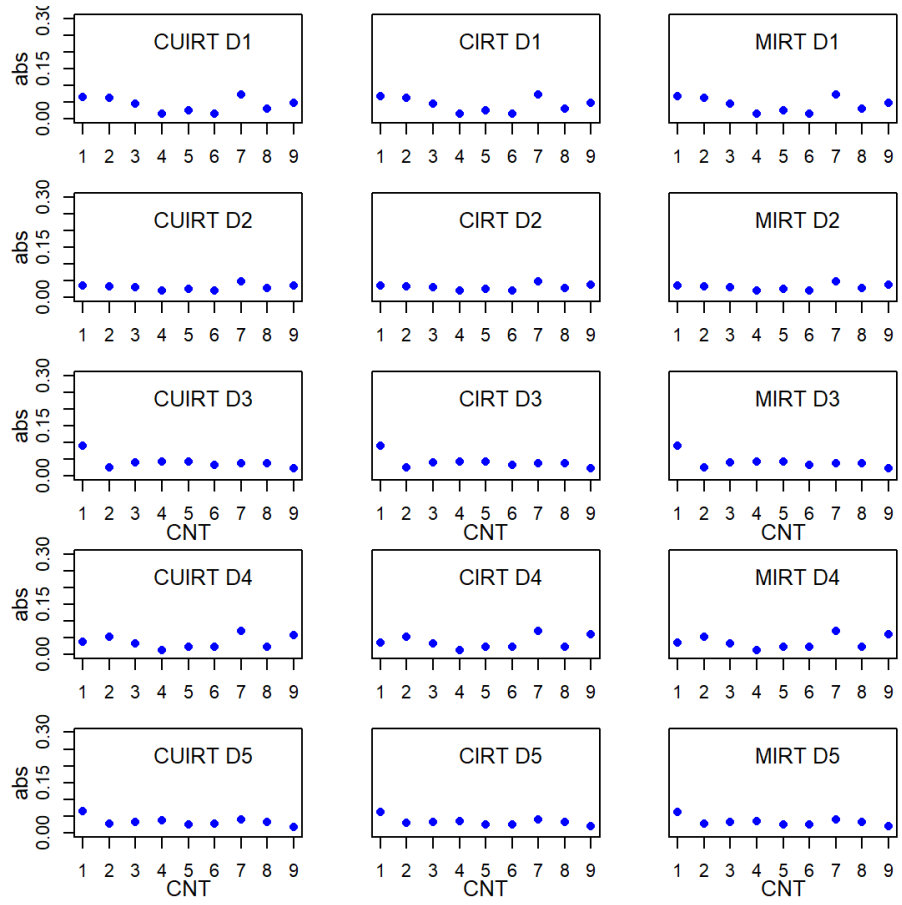
Subscale score ABS for the 5-Domain, 5-item, .45 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4; $D5$ = domain 5.

Figure J.11

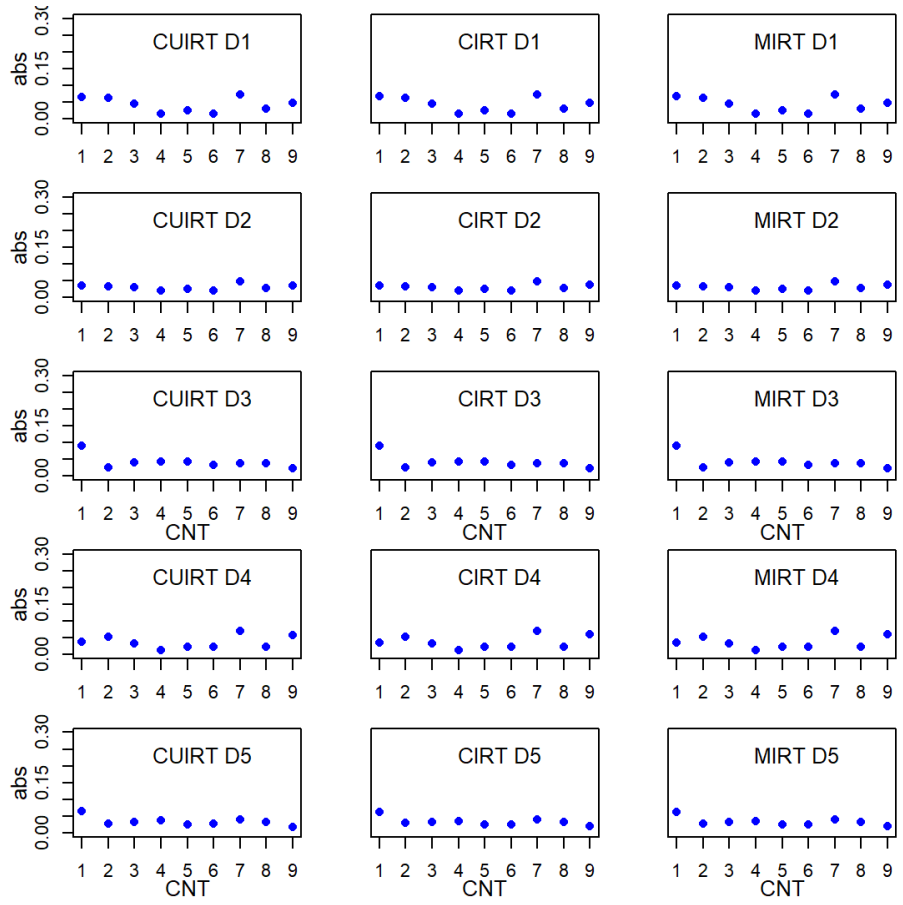
Subscale score ABS for the 5-Domain, 5-item, .75 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4; $D5$ = domain 5.

Figure J.12

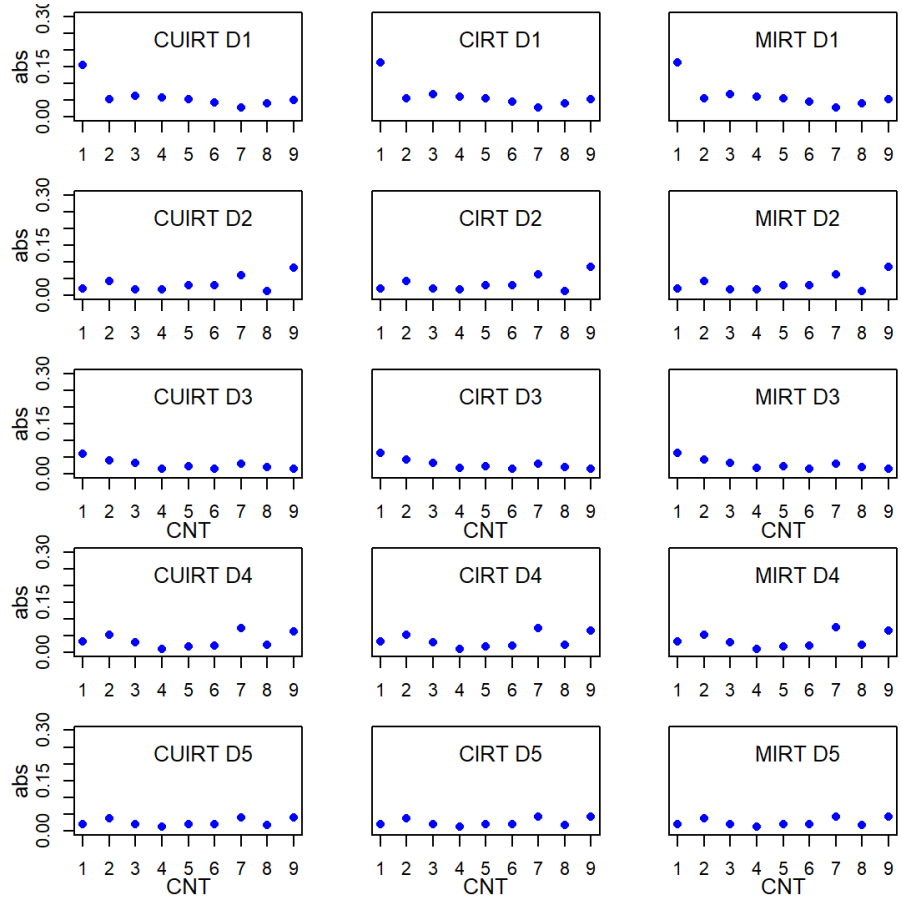
Subscale score ABS for the 5-Domain, 5-item, .95 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4; $D5$ = domain 5.

Figure J.13

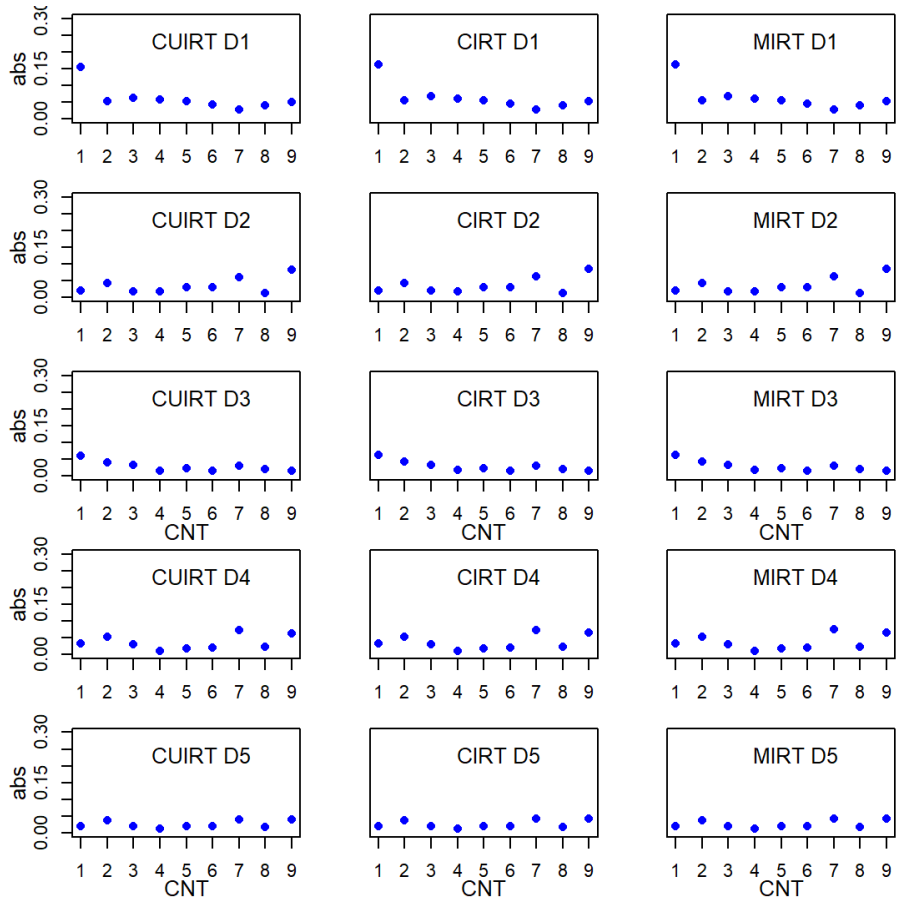
Subscale score ABS for the 5-Domain, 10-item, .45 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4; $D5$ = domain 5.

Figure J.14

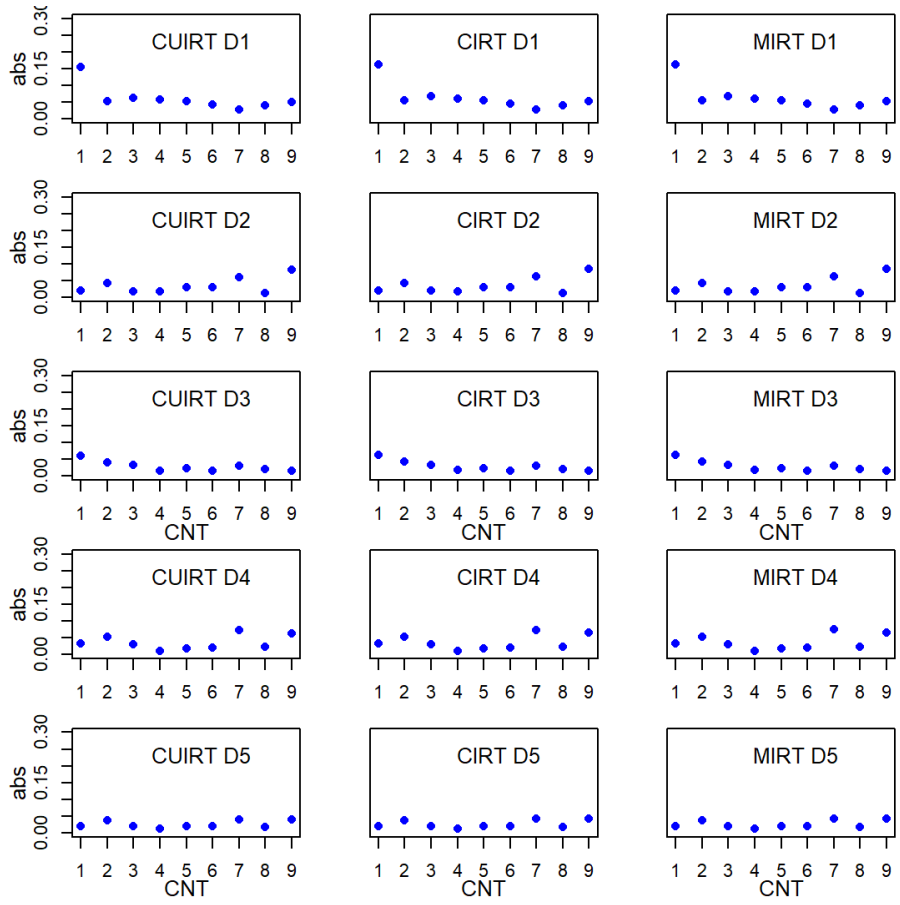
Subscale score ABS for the 5-Domain, 10-item, .75 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4; $D5$ = domain 5.

Figure J.15

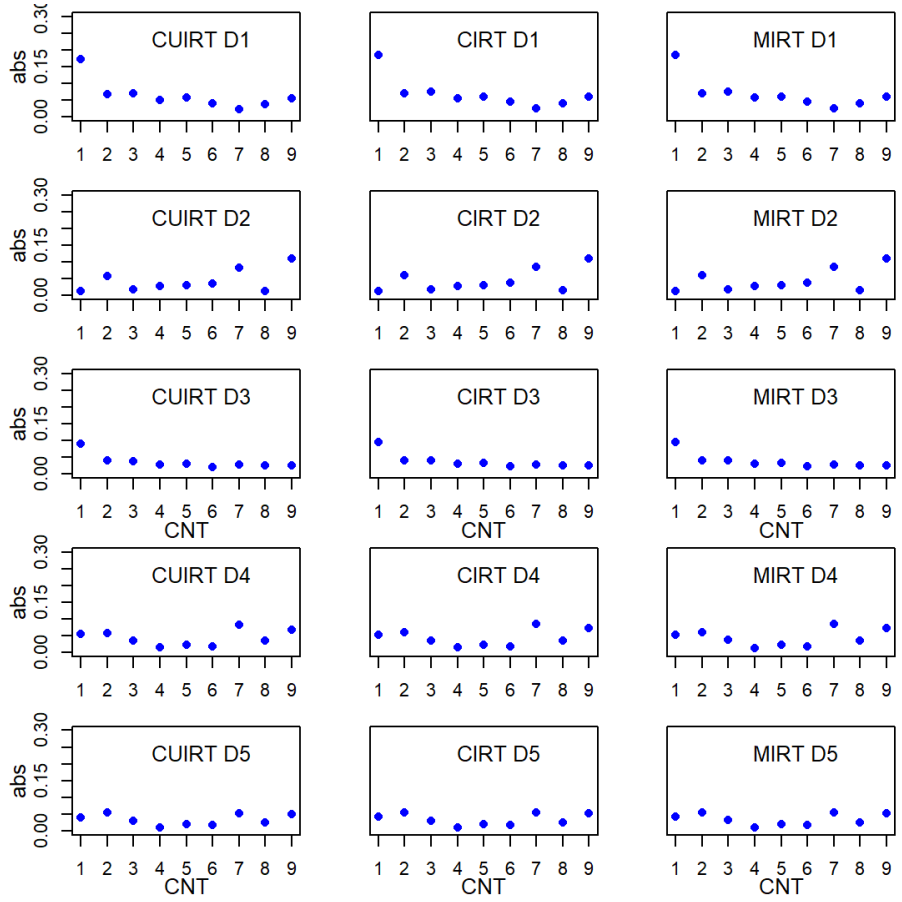
Subscale score ABS for the 5-Domain, 10-item, .95 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4; $D5$ = domain 5.

Figure J.16

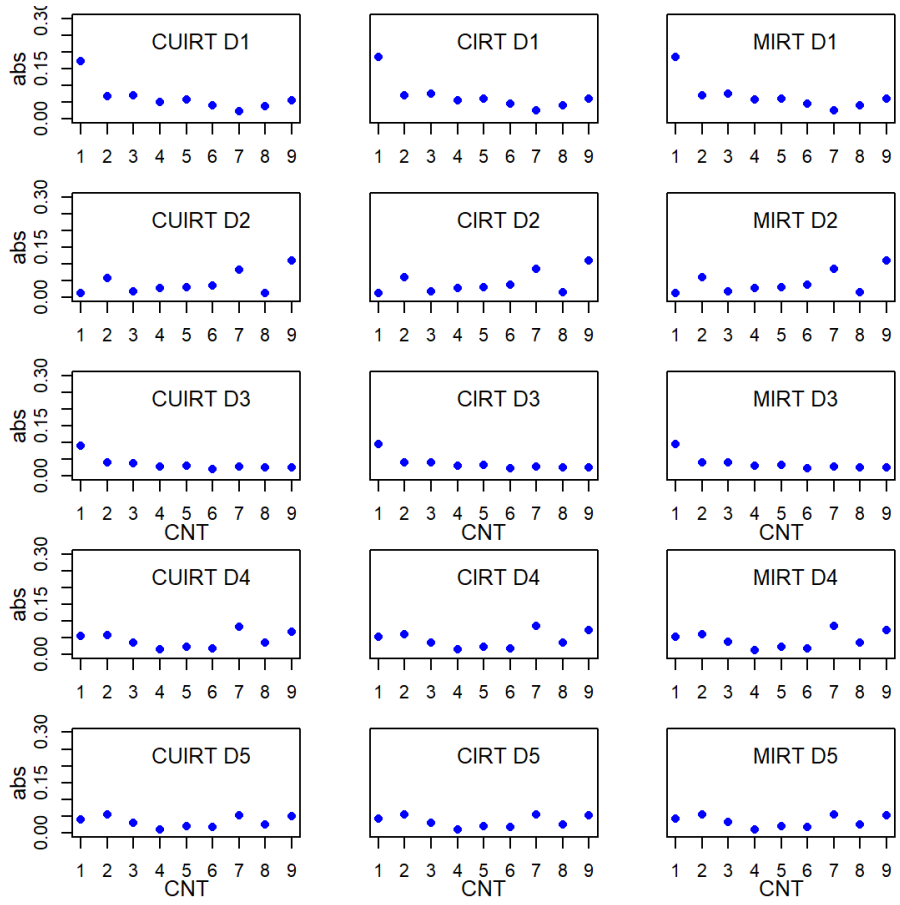
Subscale score ABS for the 5-Domain, 15-item, .45 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4; $D5$ = domain 5.

Figure J.17

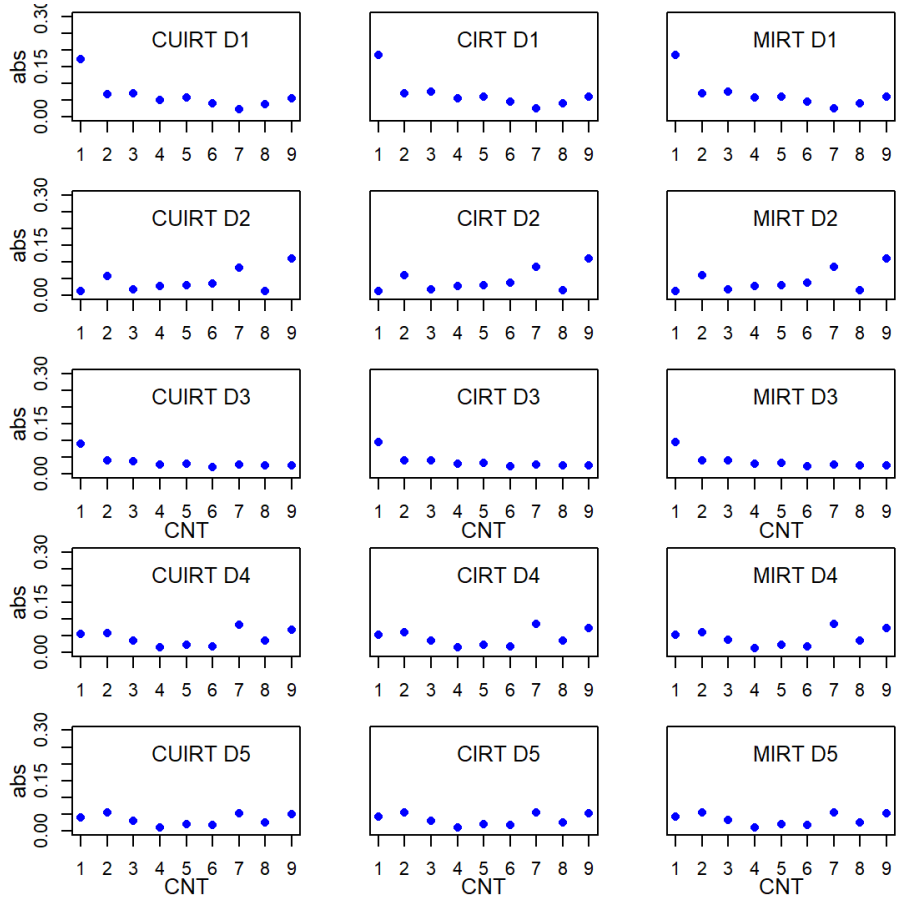
Subscale score ABS for the 5-Domain, 15-item, .75 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4; $D5$ = domain 5.

Figure J.18

Subscale score ABS for the 5-Domain, 15-item, .95 Correlation Subdomain Tests

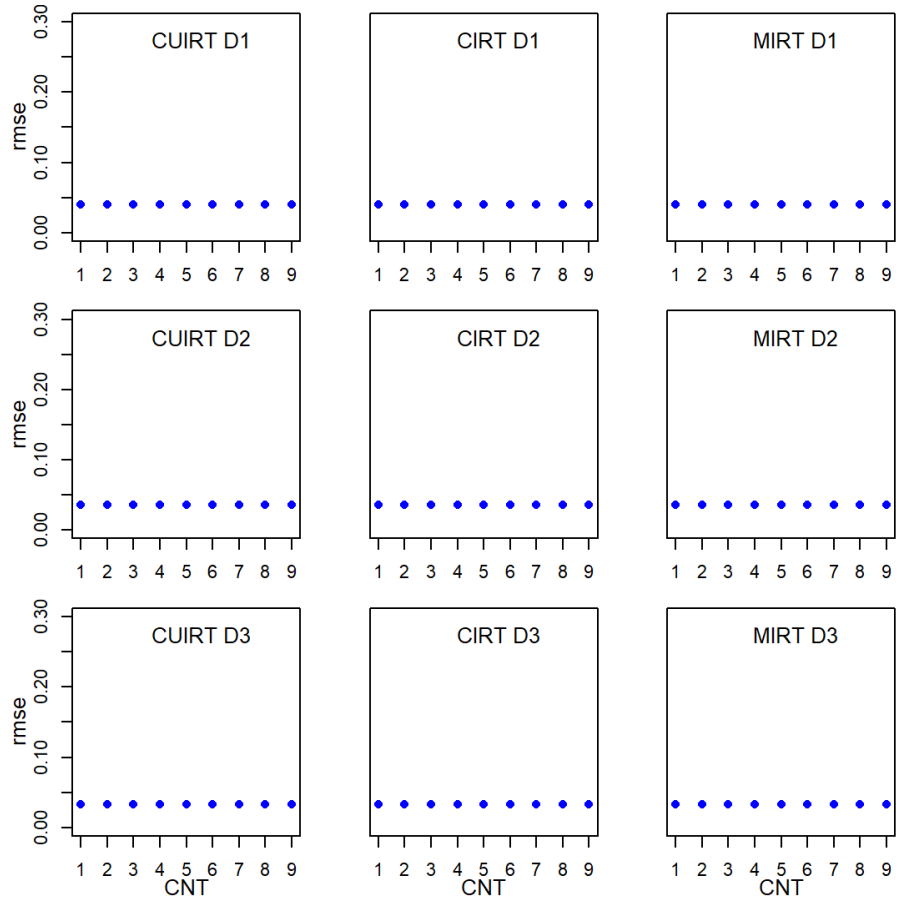


Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4; $D5$ = domain 5.

J.2 RMSE

Figure J.19

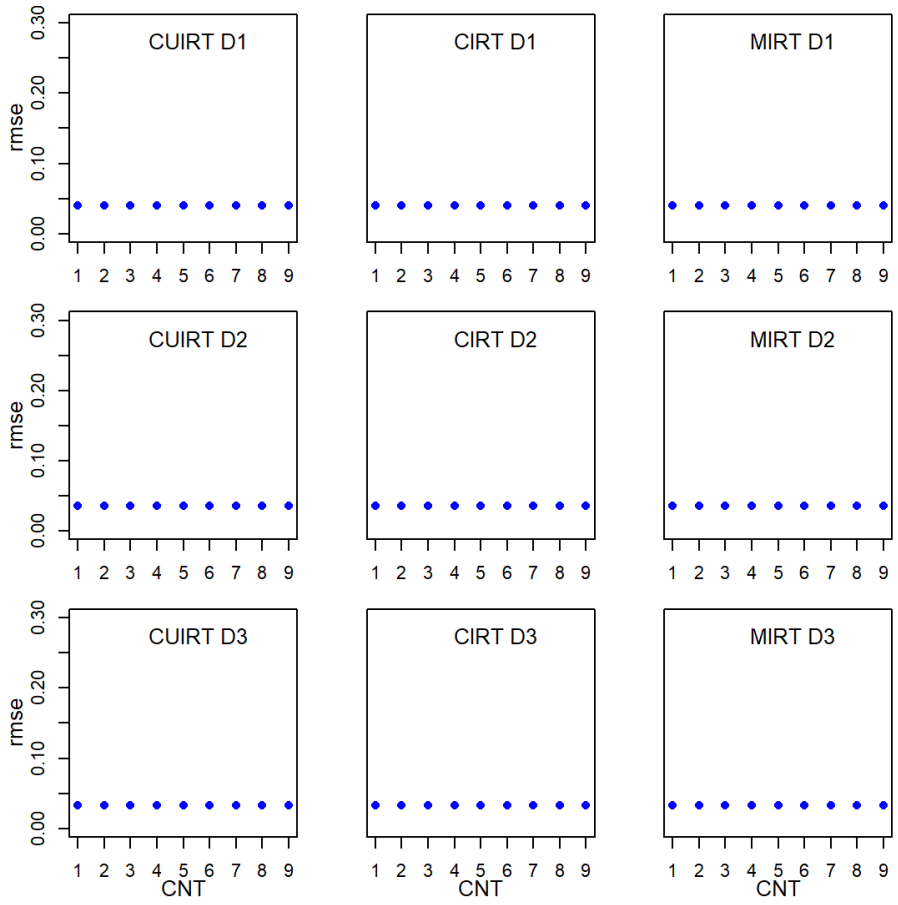
Subscale score RMSE for the 3-Domain, 5-item, .45 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

Figure J.20

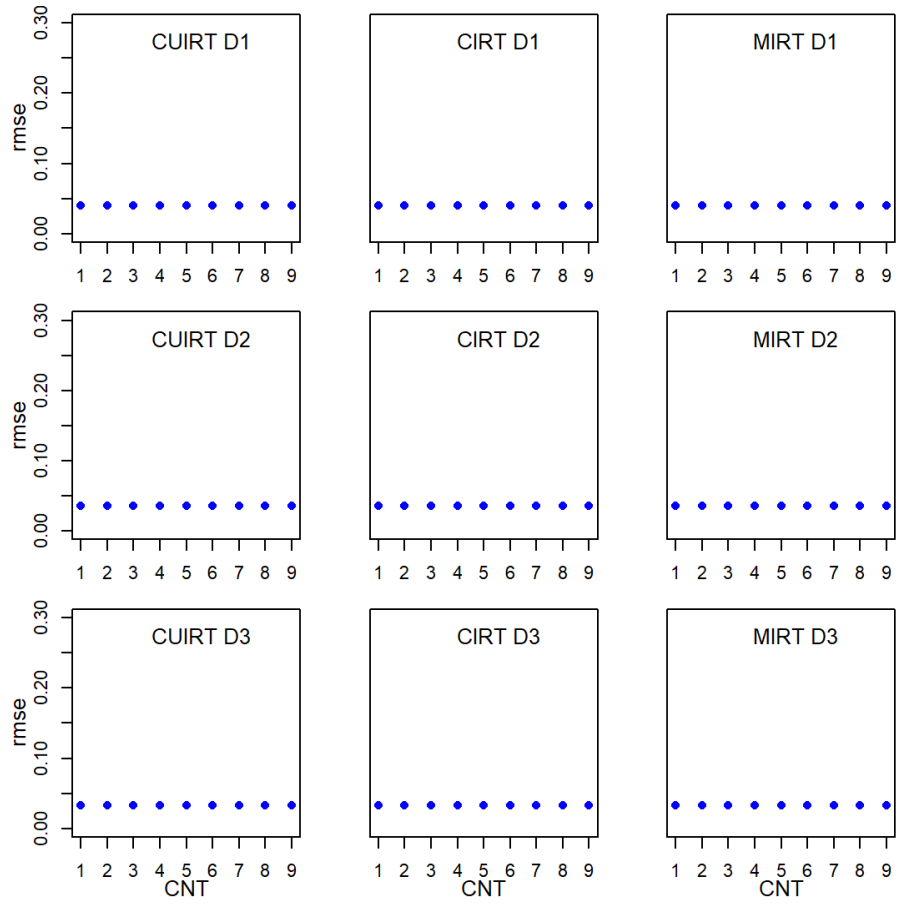
Subscale score RMSE for the 3-Domain, 5-item, .75 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

Figure J.21

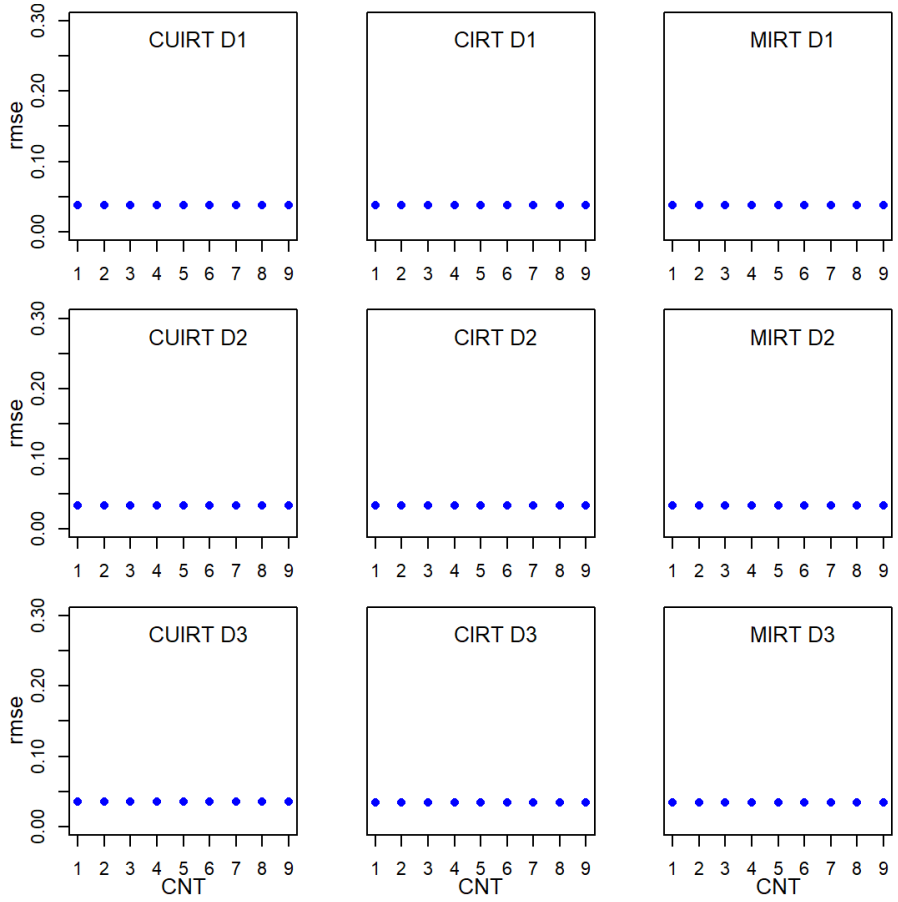
Subscale score RMSE for the 3-Domain, 5-item, .95 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

Figure J.22

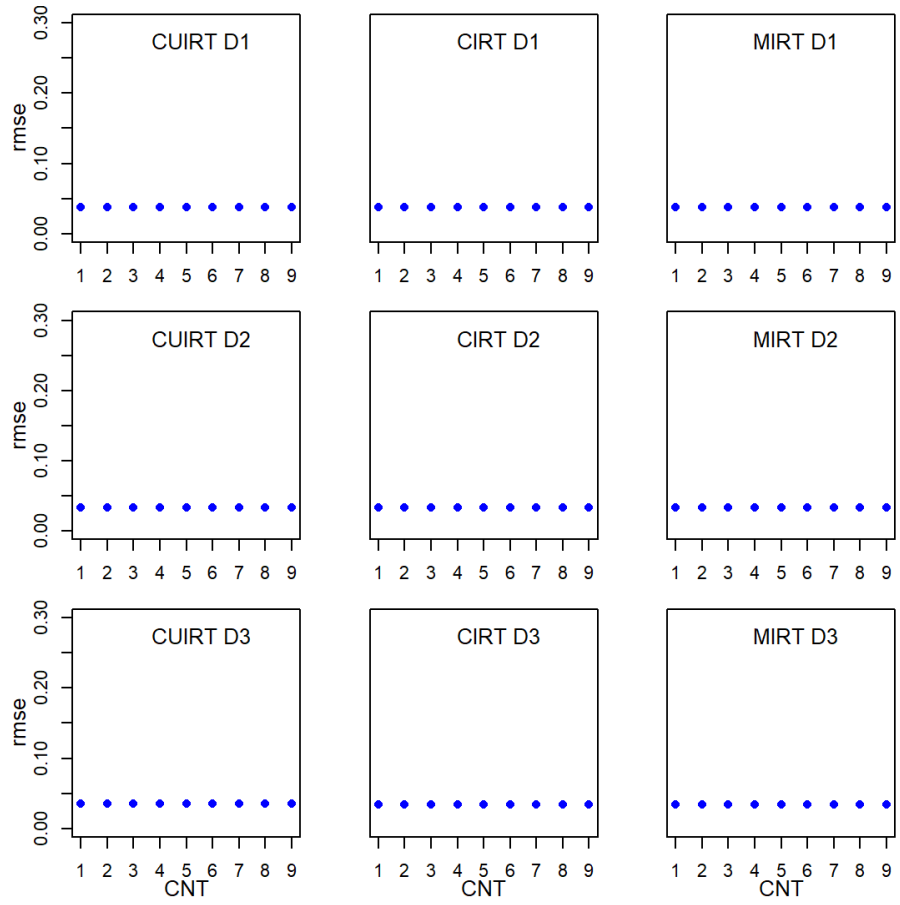
Subscale score RMSE for the 3-Domain, 10-item, .45 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

Figure J.23

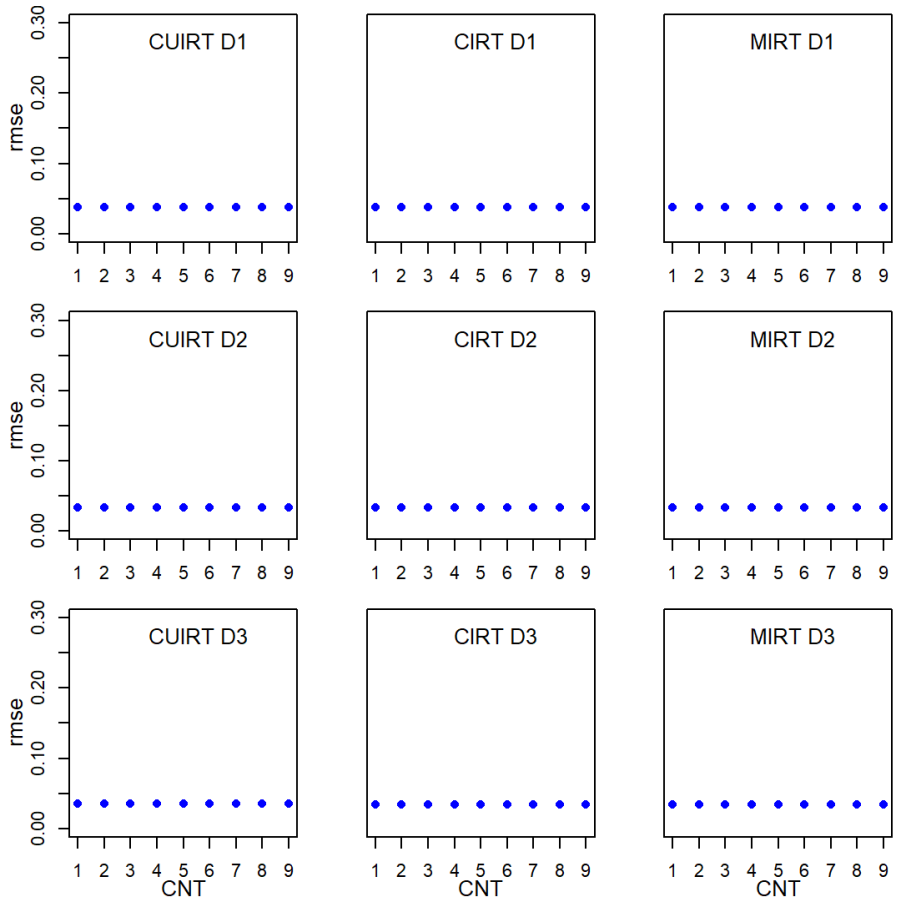
Subscale score RMSE for the 3-Domain, 10-item, .75 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

Figure J.24

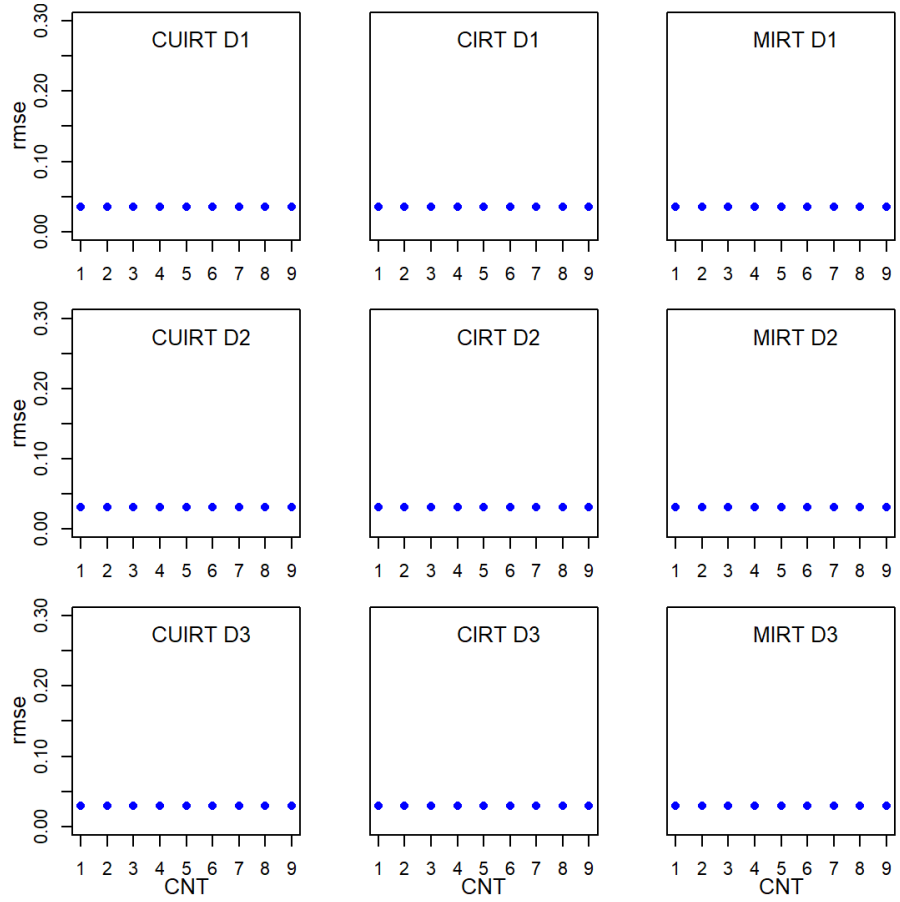
Subscale score RMSE for the 3-Domain, 10-item, .95 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

Figure J.25

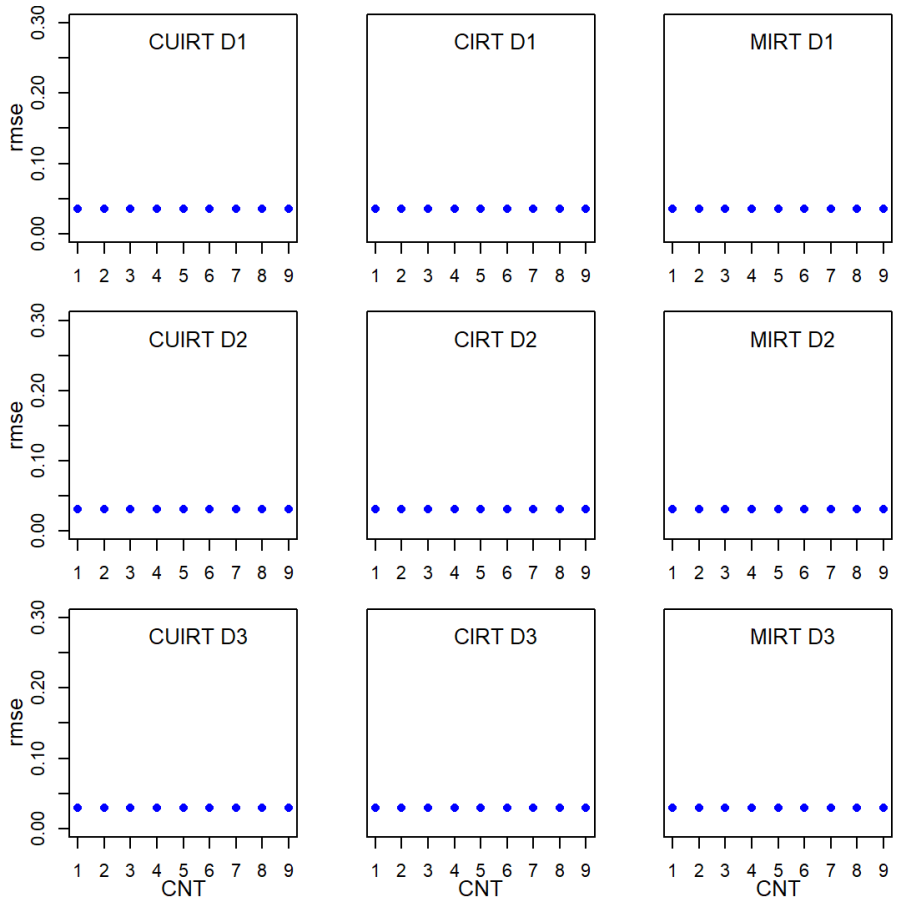
Subscale score RMSE for the 3-Domain, 15-item, .45 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

Figure J.26

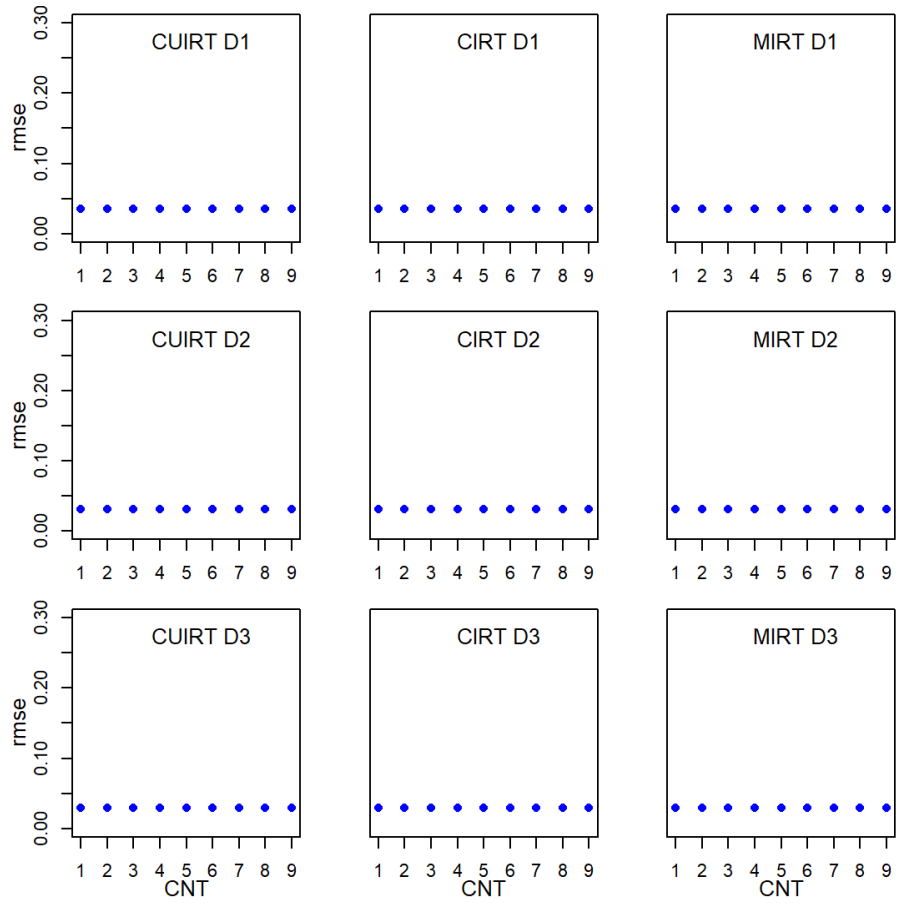
Subscale score RMSE for the 3-Domain, 15-item, .75 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

Figure J.27

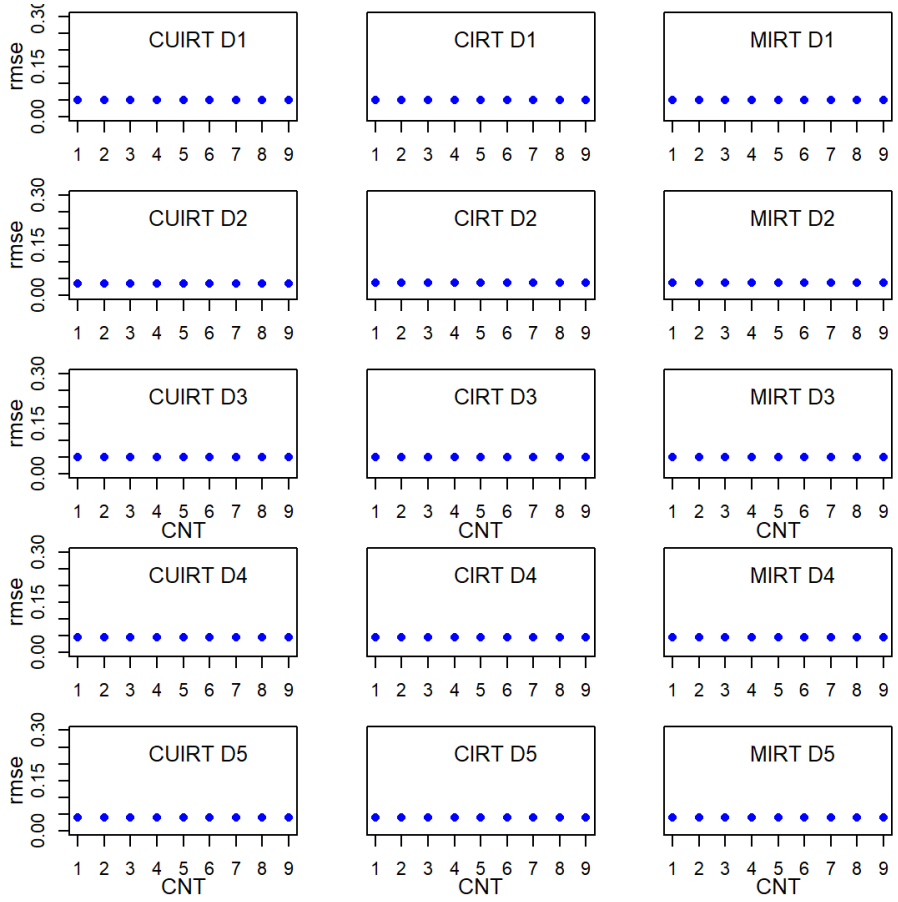
Subscale score RMSE for the 3-Domain, 15-item, .95 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

Figure J.28

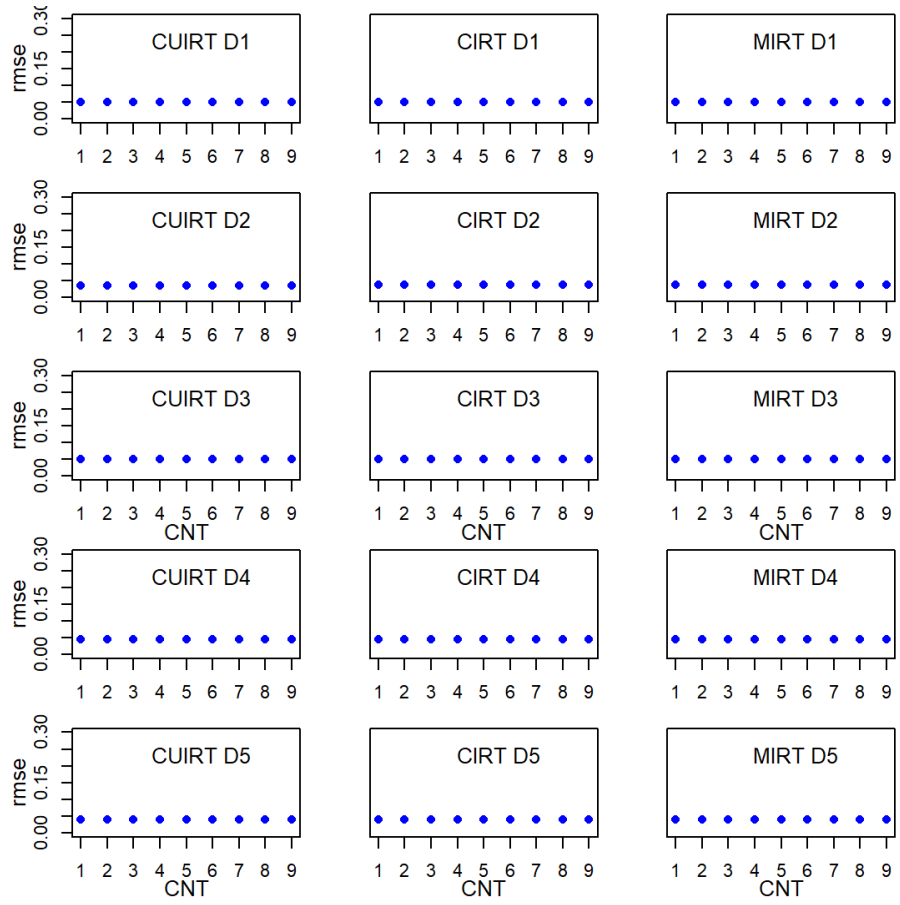
Subscale score RMSE for the 5-Domain, 5-item, .45 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4; $D5$ = domain 5.

Figure J.29

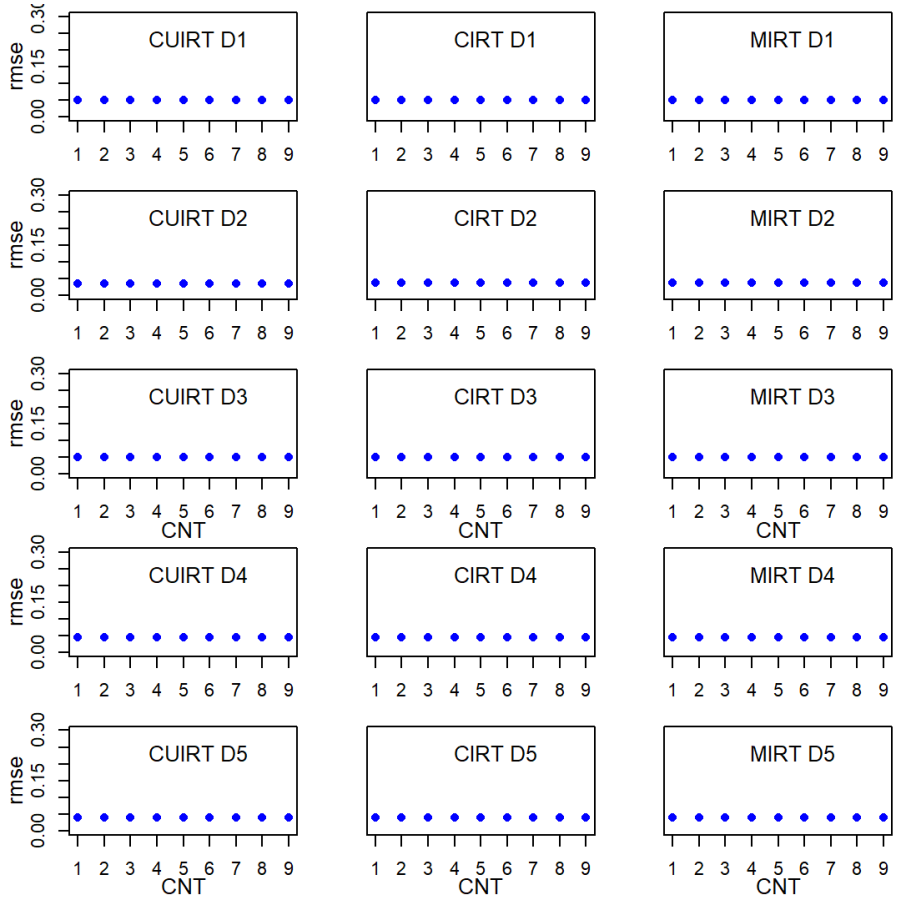
Subscale score RMSE for the 5-Domain, 5-item, .75 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4; $D5$ = domain 5.

Figure J.30

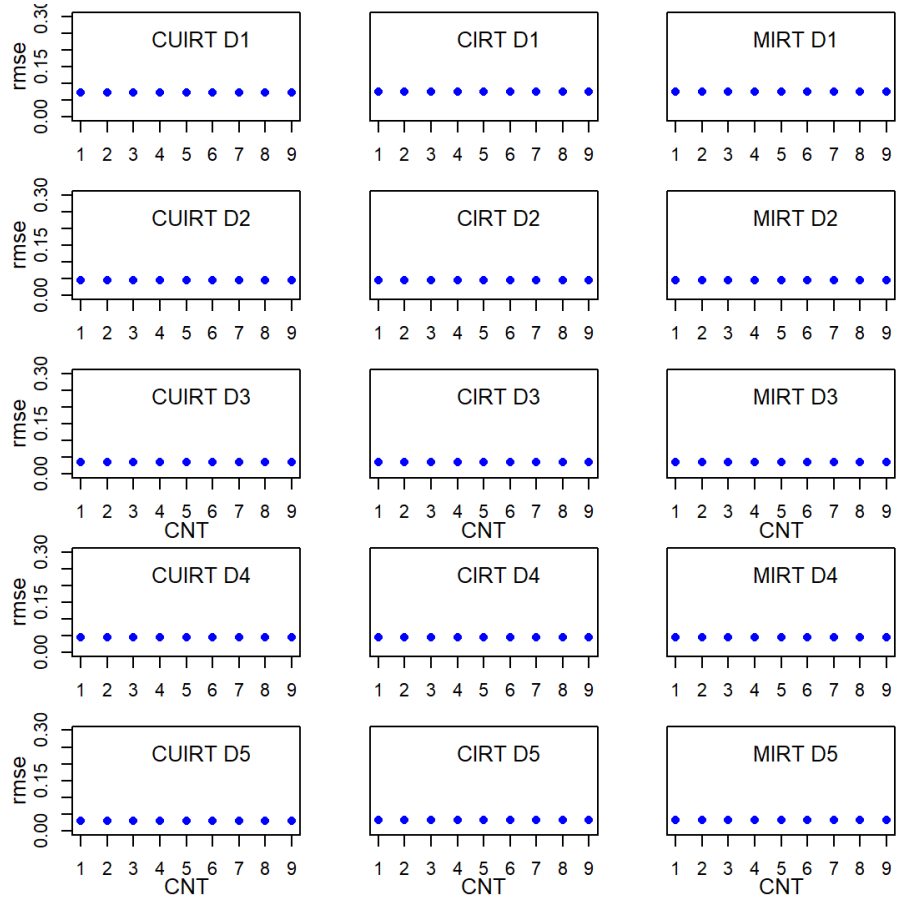
Subscale score RMSE for the 5-Domain, 5-item, .95 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4; $D5$ = domain 5.

Figure J.31

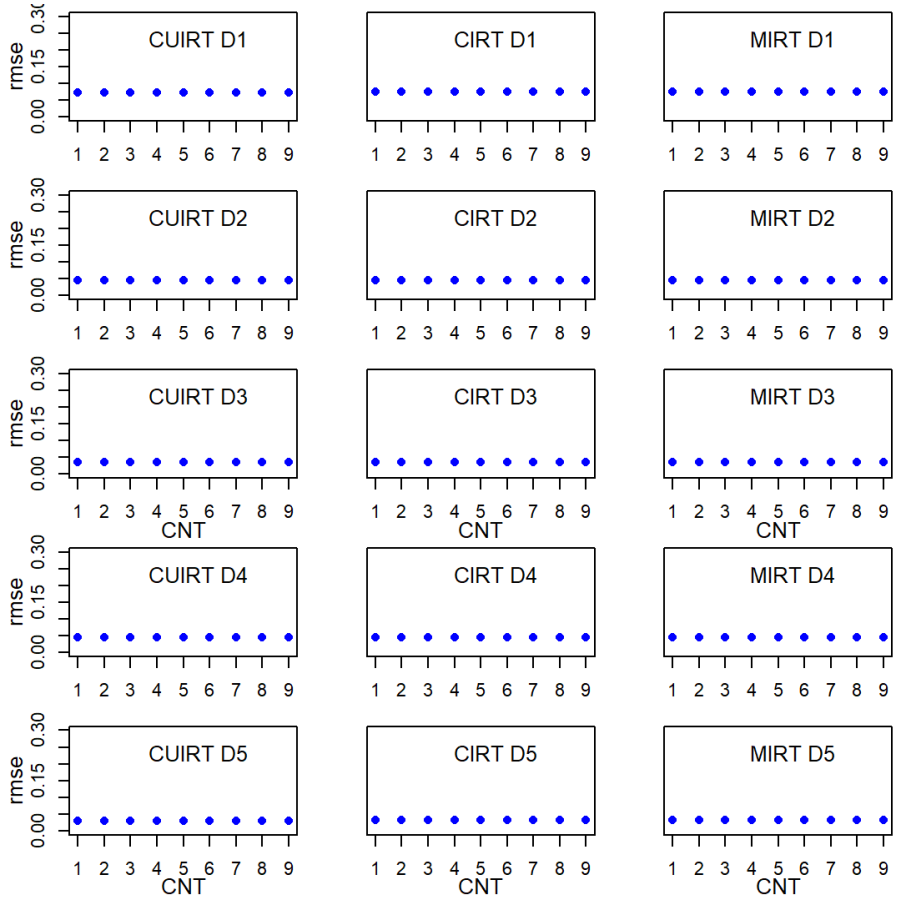
Subscale score RMSE for the 5-Domain, 10-item, .45 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4; $D5$ = domain 5.

Figure J.32

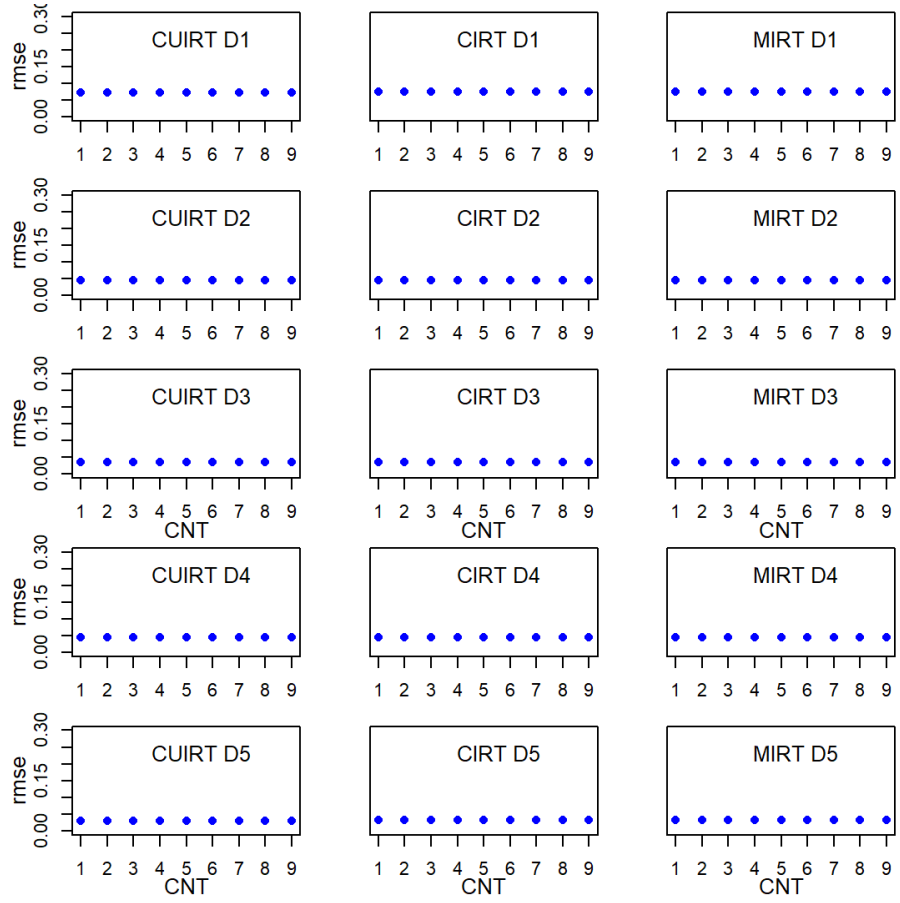
Subscale score RMSE for the 5-Domain, 10-item, .75 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4; $D5$ = domain 5.

Figure J.33

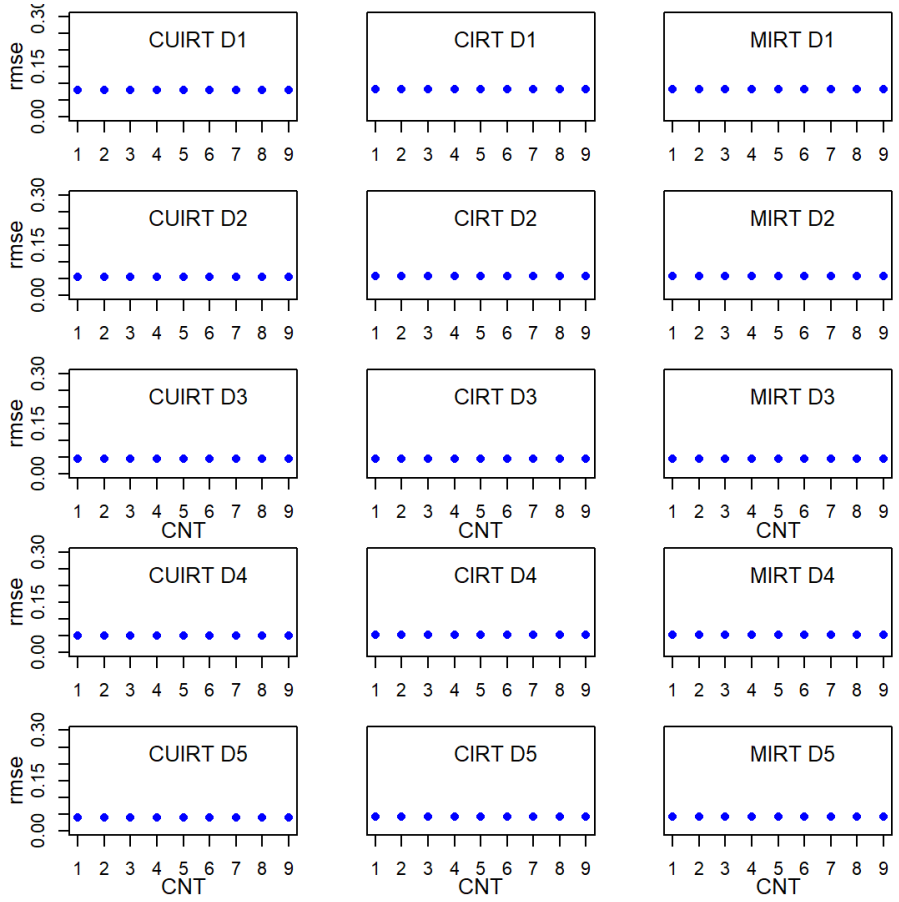
Subscale score RMSE for the 5-Domain, 10-item, .95 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4; $D5$ = domain 5.

Figure J.34

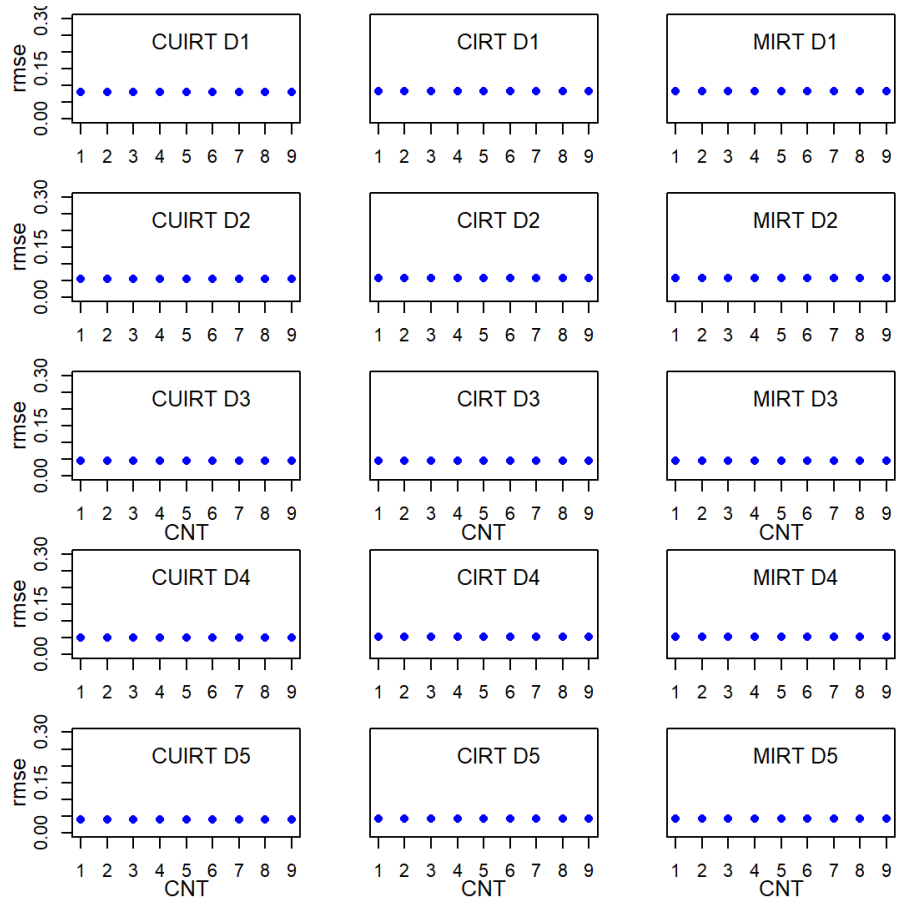
Subscale score RMSE for the 5-Domain, 15-item, .45 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4; $D5$ = domain 5.

Figure J.35

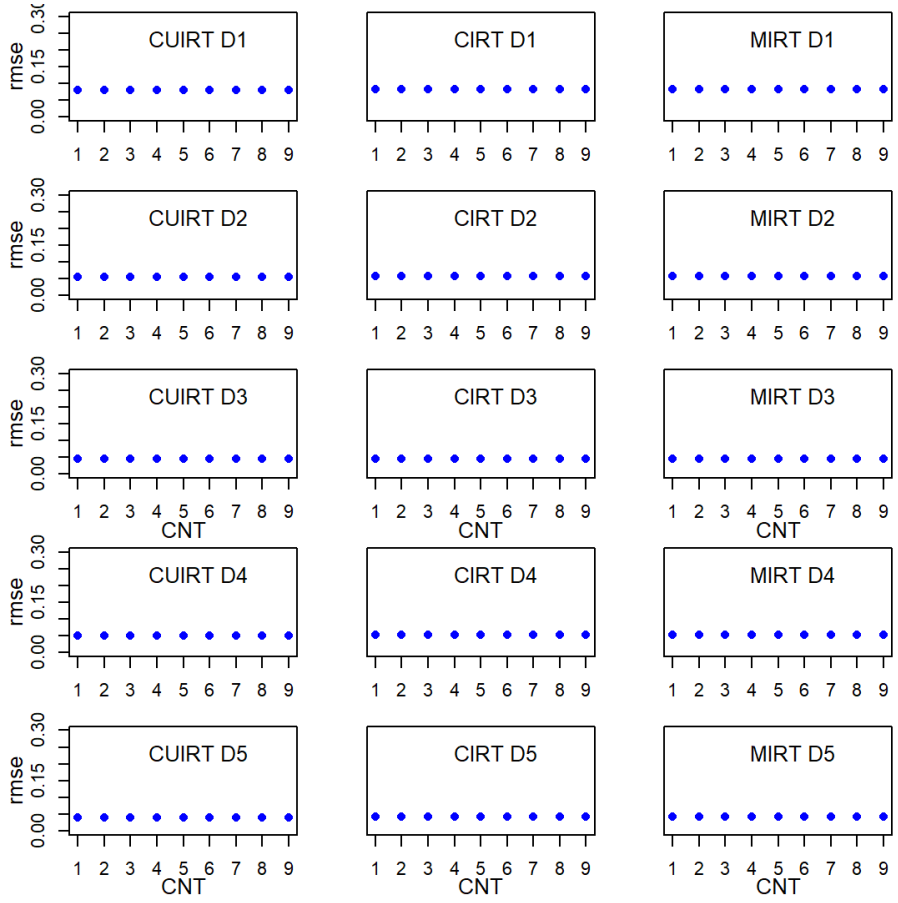
Subscale score RMSE for the 5-Domain, 15-item, .75 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4; $D5$ = domain 5.

Figure J.36

Subscale score RMSE for the 5-Domain, 15-item, .95 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4; $D5$ = domain 5.

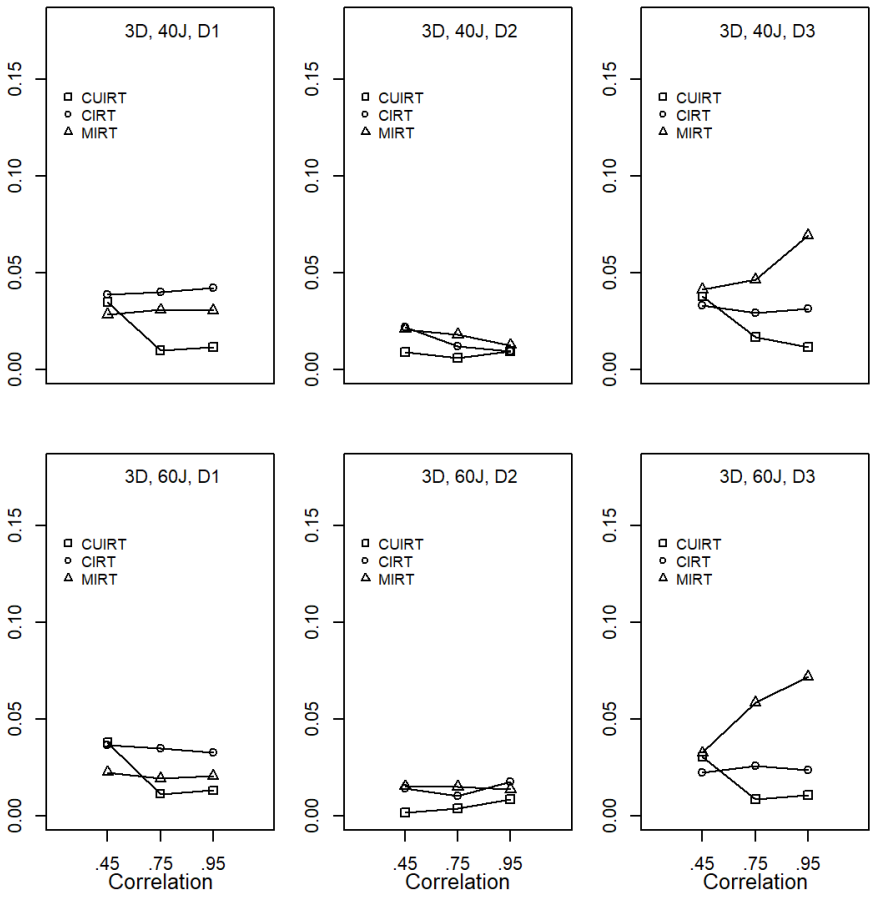
Appendix K

Study 2 Score Parameter ABS and RMSE: Single Groups

K.1 ABS

Figure K.1

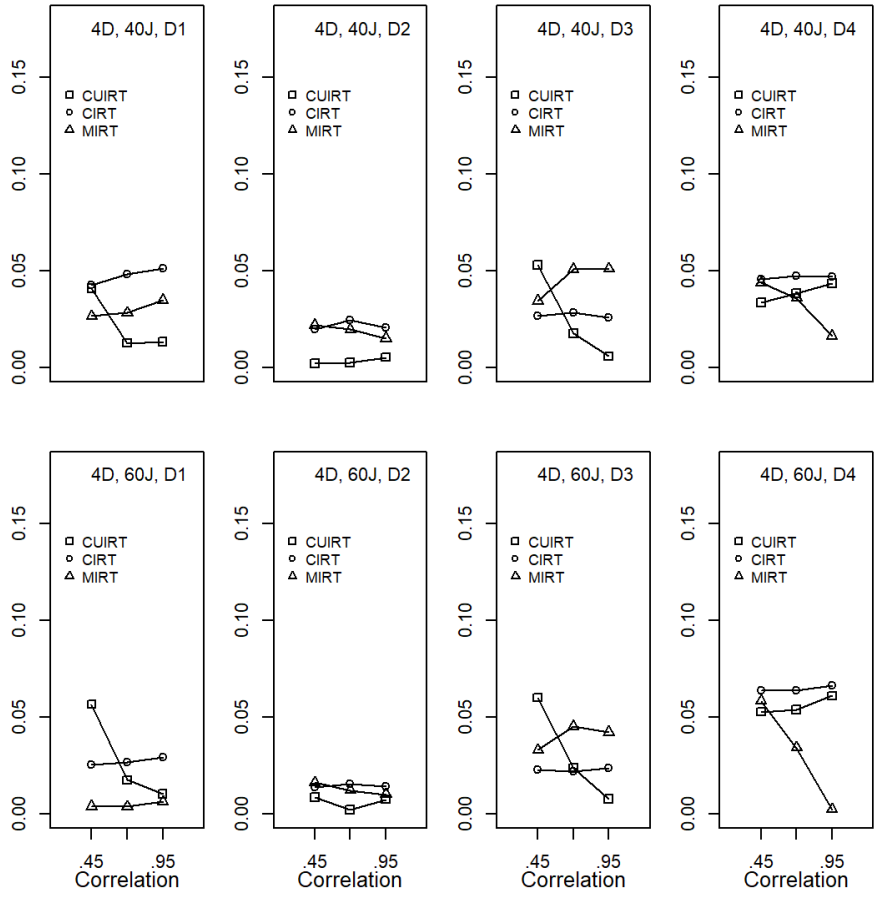
Subscale score ABS for the 3 Subdomain Tests



Note. D = number of domains; J = number of items; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

Figure K.2

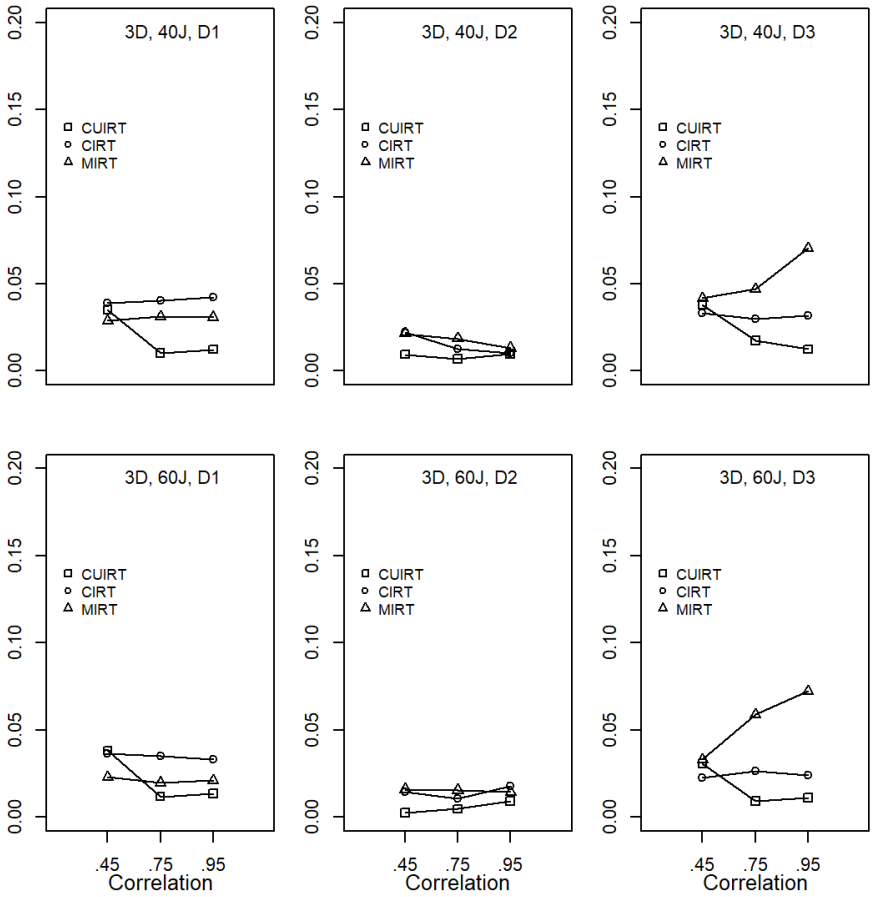
Subscale score ABS for the 4 Subdomain Tests



Note. D = number of domains; J = number of items; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4.

K.2 RMSE

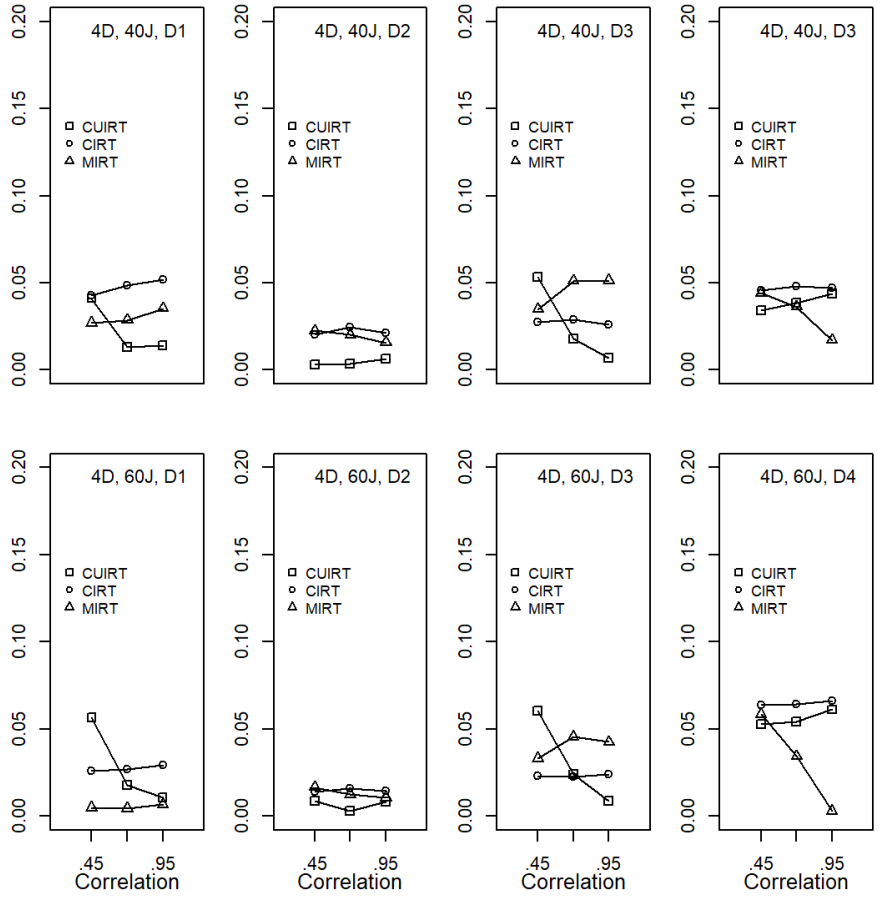
Figure K.3
Subscale score RMSE for the 3 Subdomain Tests



Note. D = number of domains; J = number of items; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

Figure K.4

Subscale score RMSE for the 4 Subdomain Tests



Note. D = number of domains; J = number of items; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4.

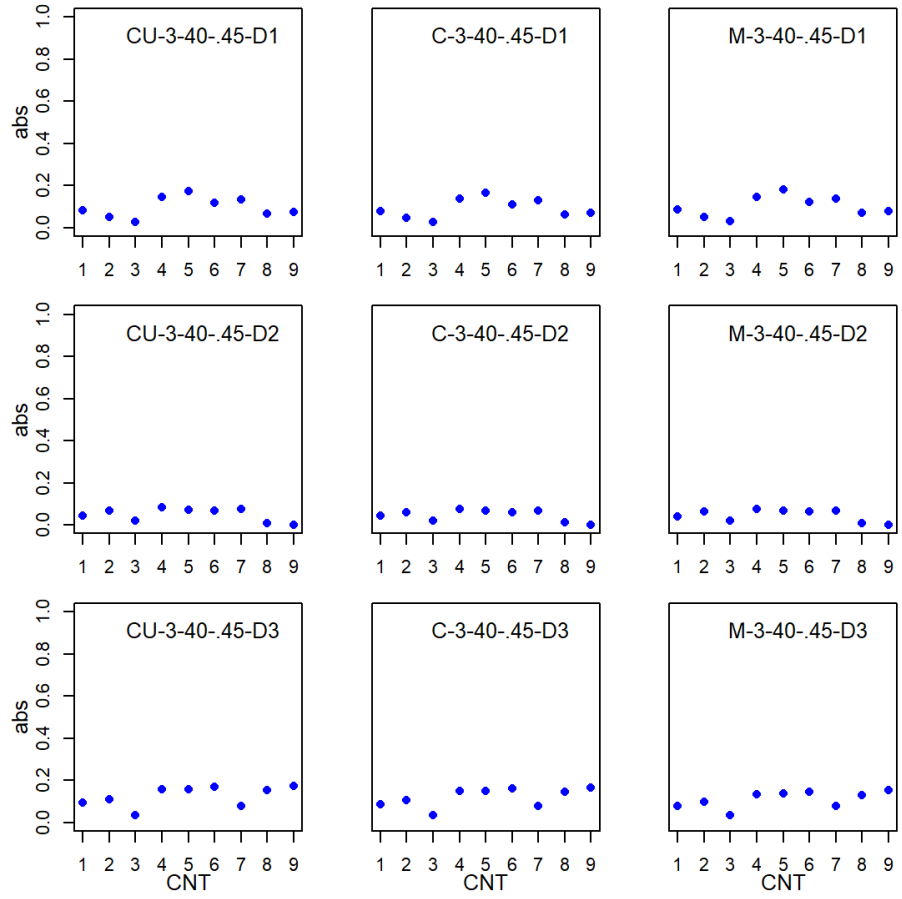
Appendix L

Study 2 Score Parameter ABS and RMSE: Multiple Groups

L.1 ABS

Figure L.1

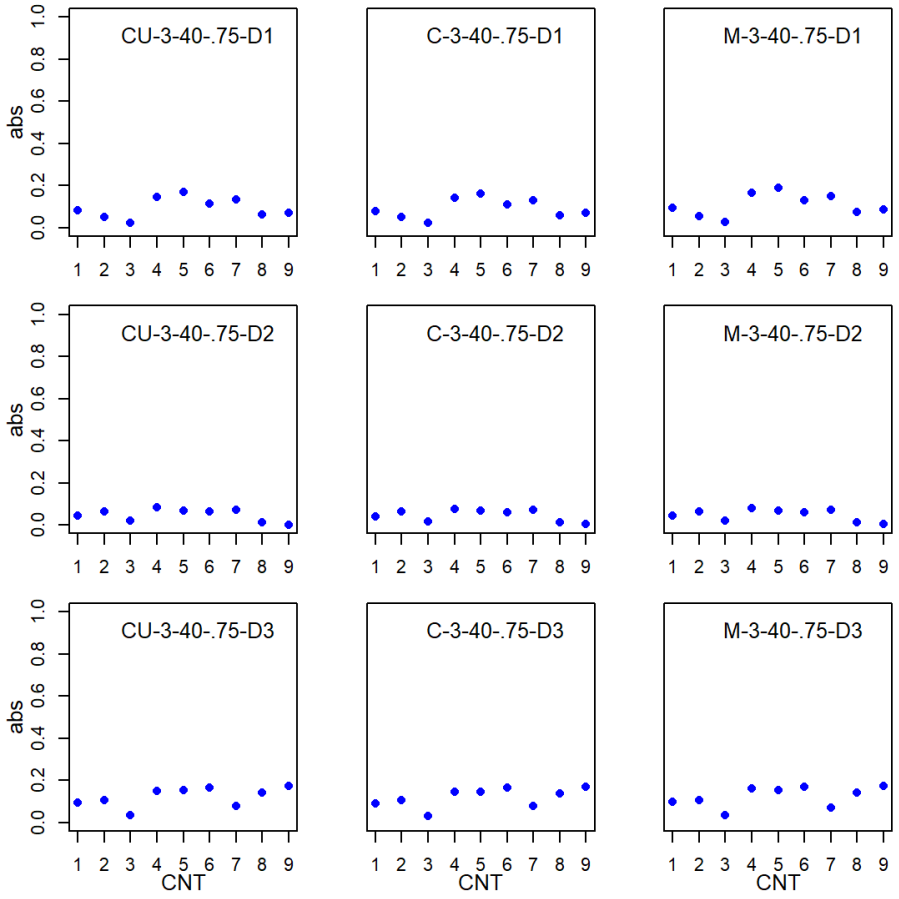
Subscale Score ABS for the 3-Subdomain, 40-Item, .45 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

Figure L.2

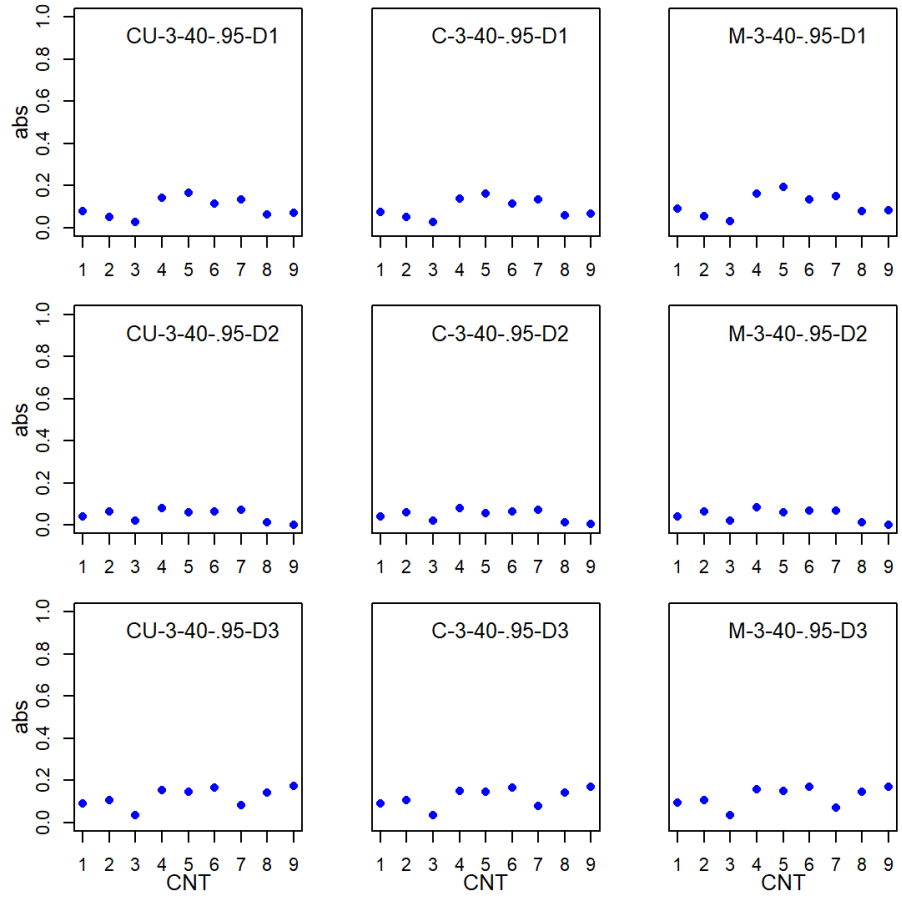
Subscale Score ABS for the 3-Subdomain, 40-Item, .75 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

Figure L.3

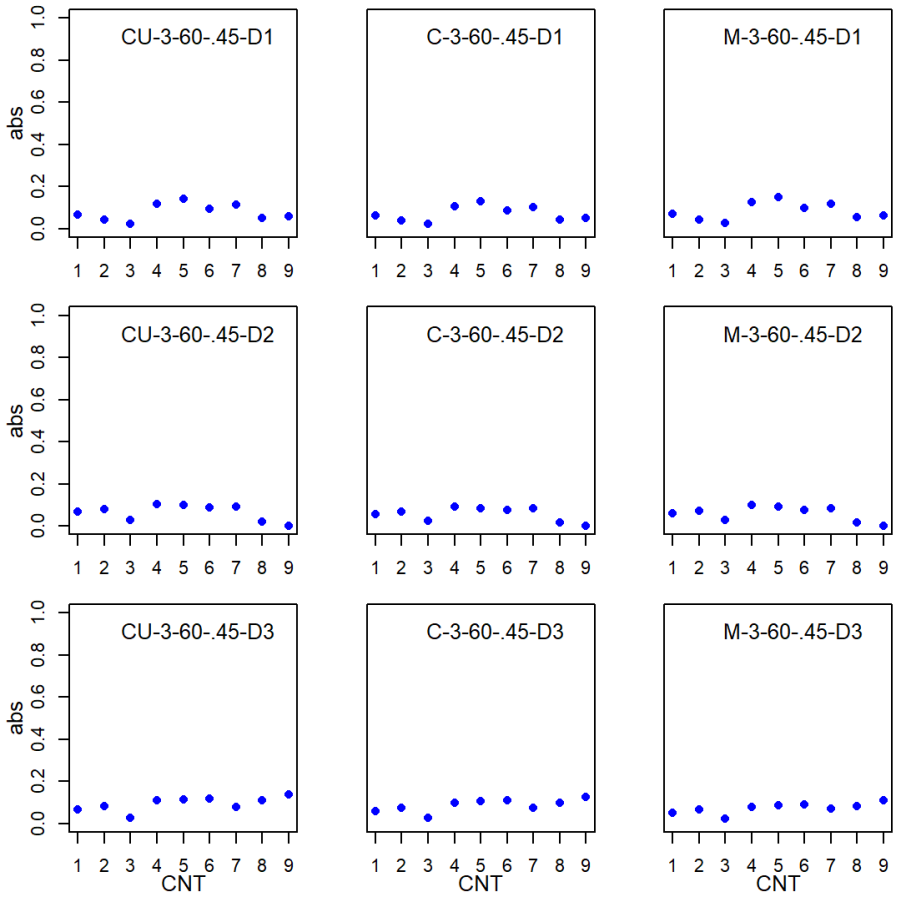
Subscale Score ABS for the 3-Subdomain, 40-Item, .95 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

Figure L.4

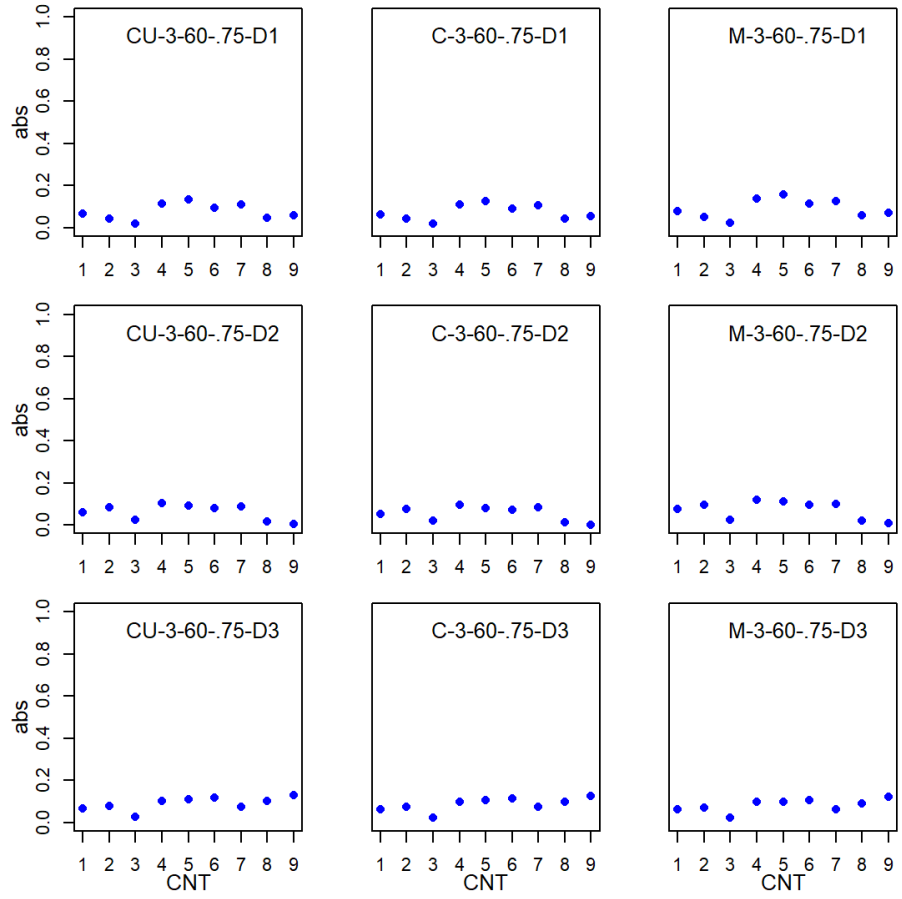
Subscale Score ABS for the 3-Subdomain, 60-Item, .45 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

Figure L.5

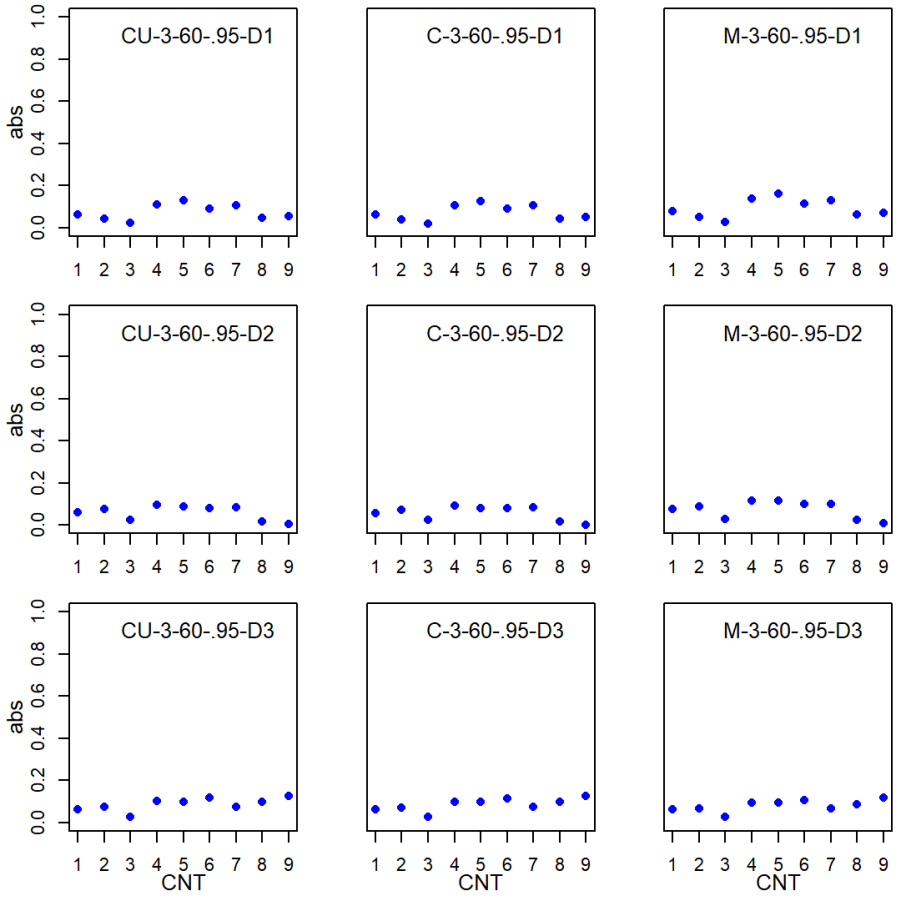
Subscale Score ABS for the 3-Subdomain, 60-Item, .75 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

Figure L.6

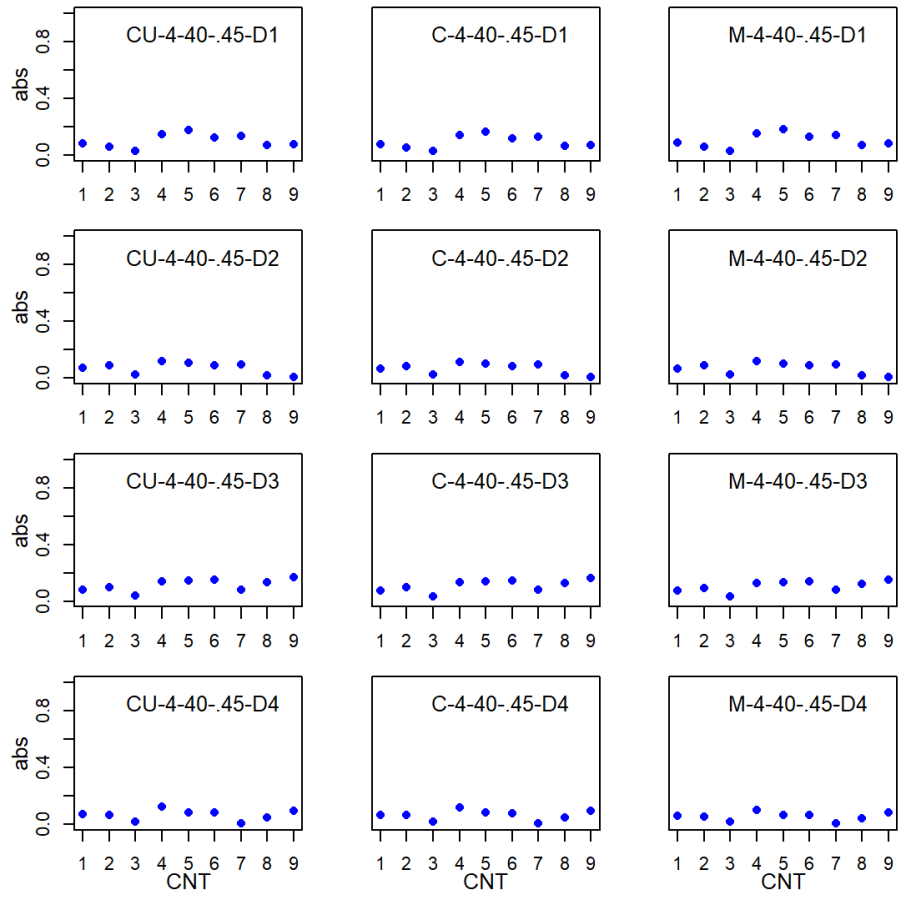
Subscale Score ABS for the 3-Subdomain, 60-Item, .95 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

Figure L.7

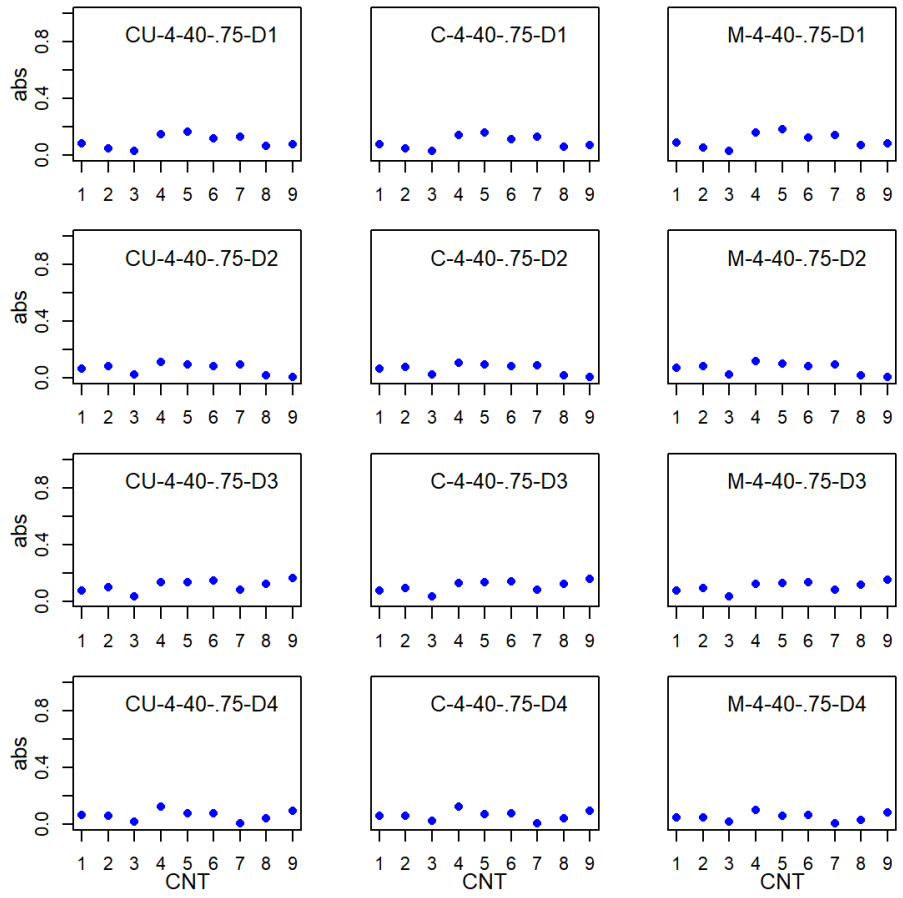
Subscale Score ABS for the 4-Subdomain, 40-Item, .45 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4.

Figure L.8

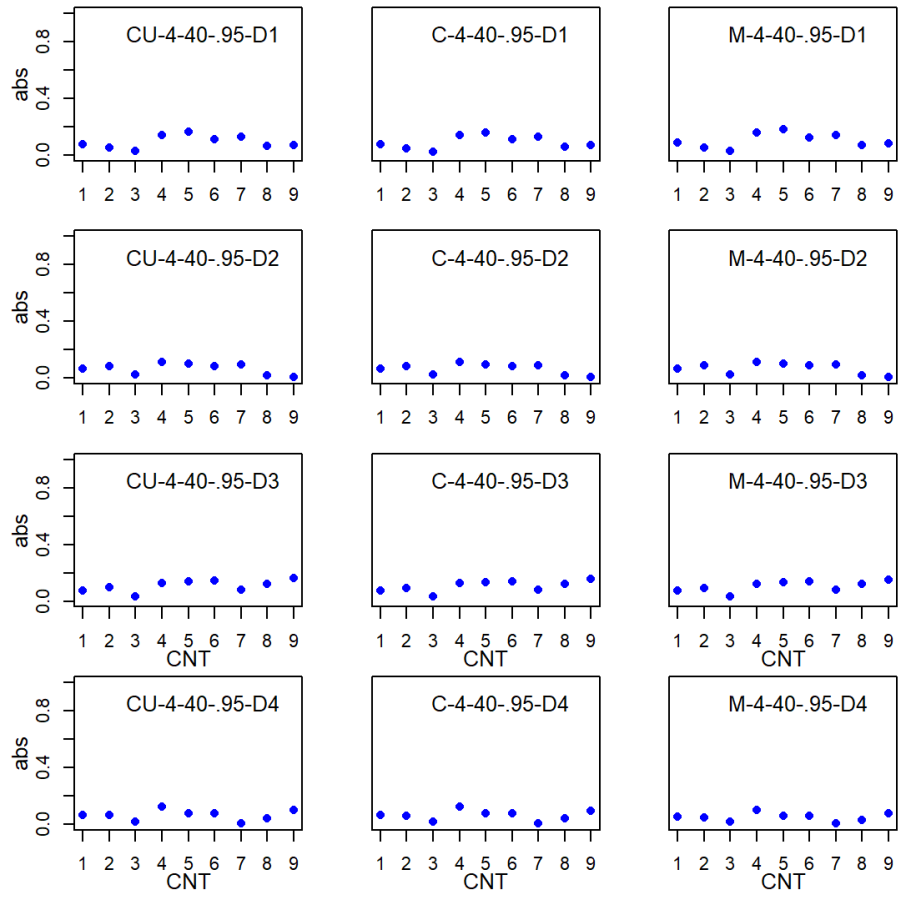
Subscale Score ABS for the 4-Subdomain, 40-Item, .75 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4.

Figure L.9

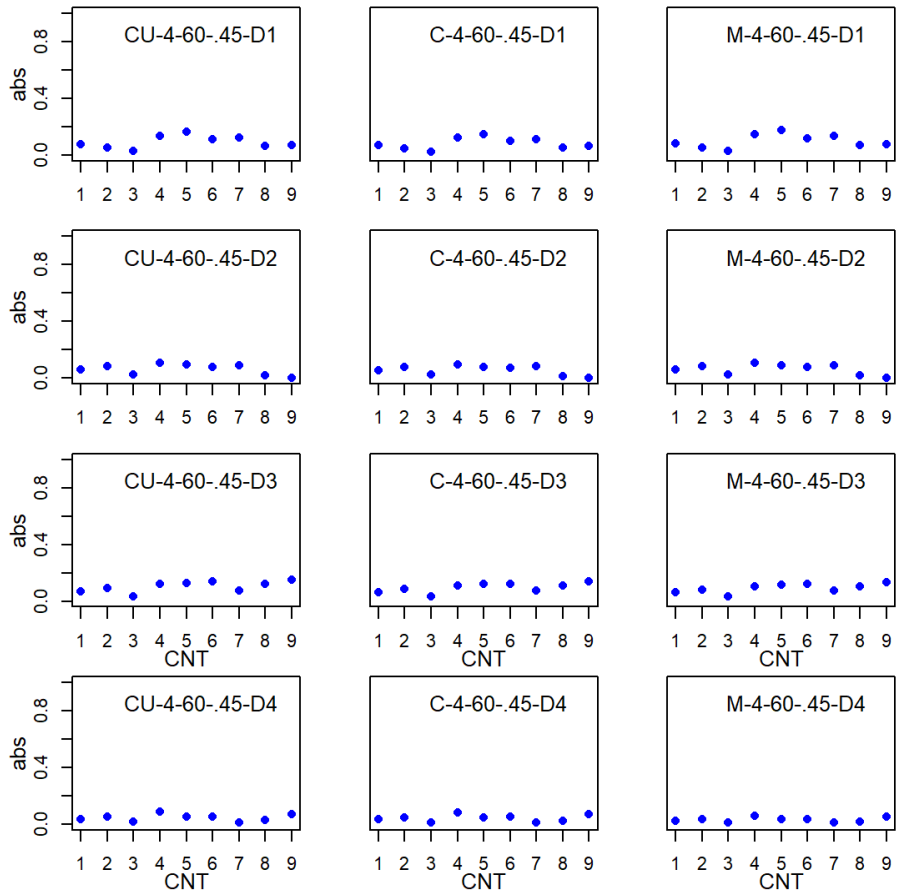
Subscale Score ABS for the 4-Subdomain, 40-Item, .95 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4.

Figure L.10

Subscale Score ABS for the 4-Subdomain, 60-Item, .45 Correlation Subdomain Tests

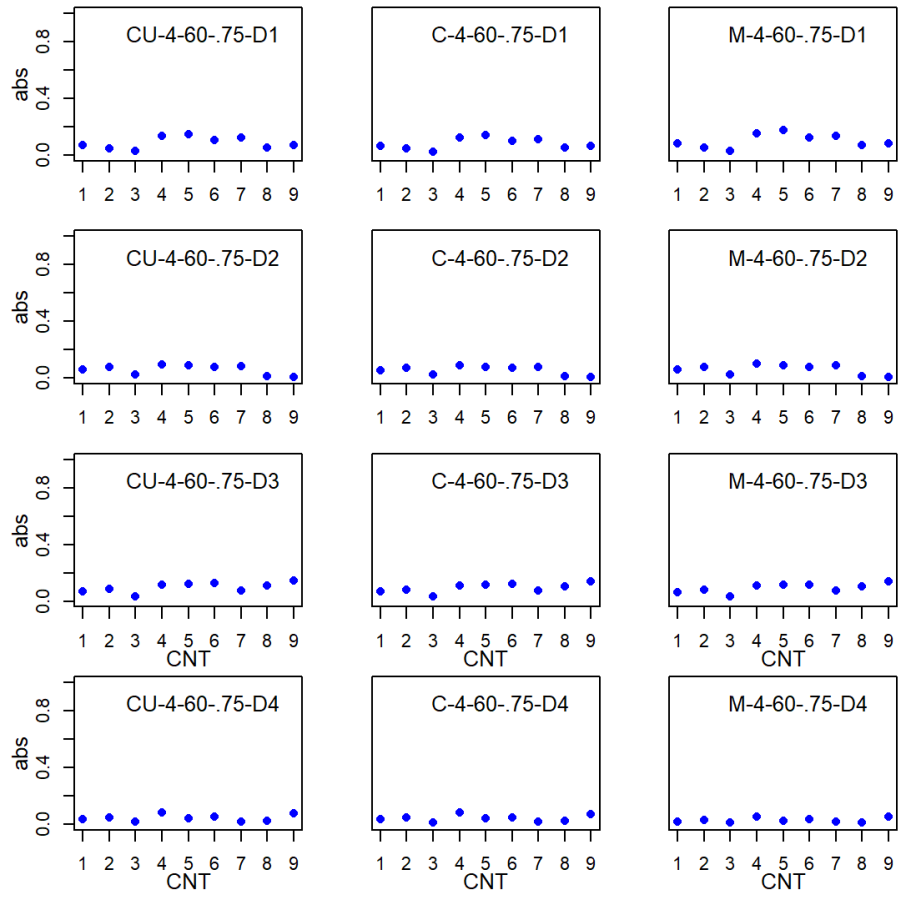


Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4.

L.2 RMSE

Figure L.11

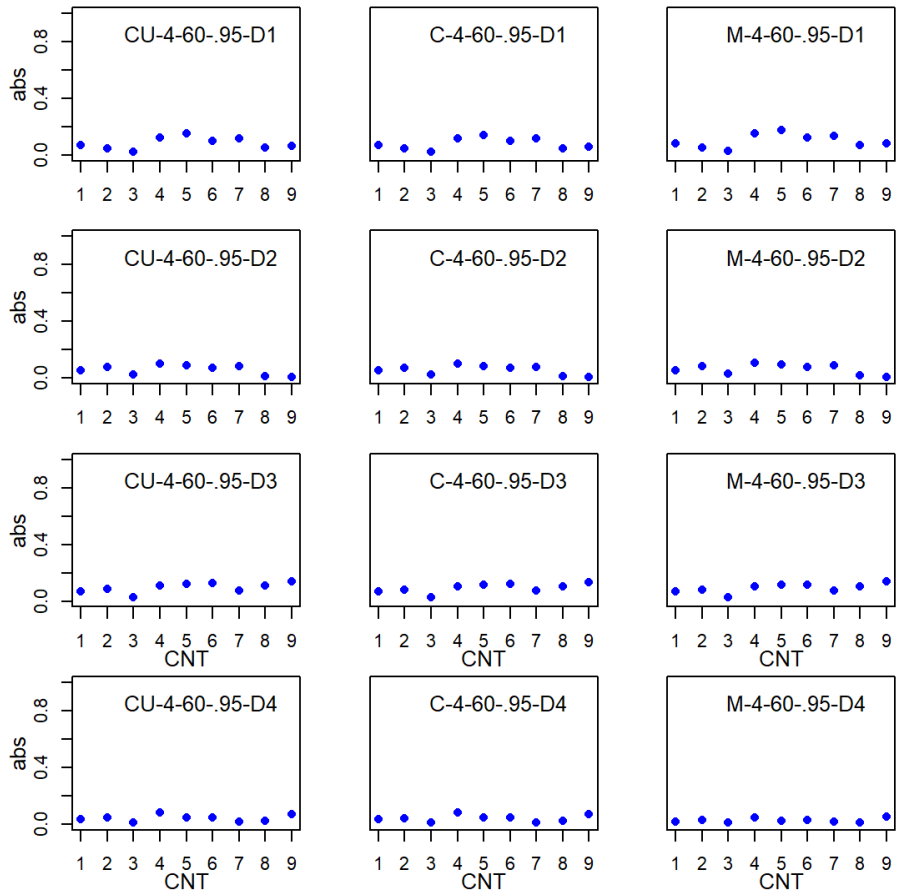
Subscale Score ABS for the 4-Subdomain, 60-Item, .75 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4.

Figure L.12

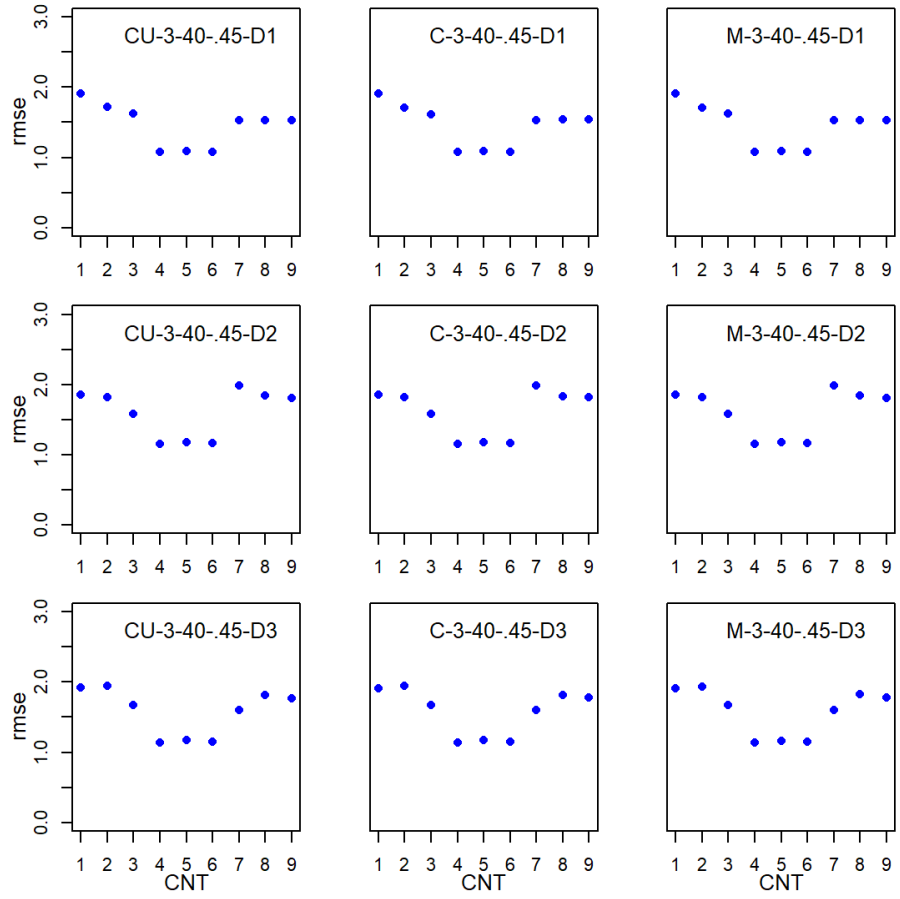
Subscale Score ABS for the 4-Subdomain, 60-Item, .95 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4.

Figure L.13

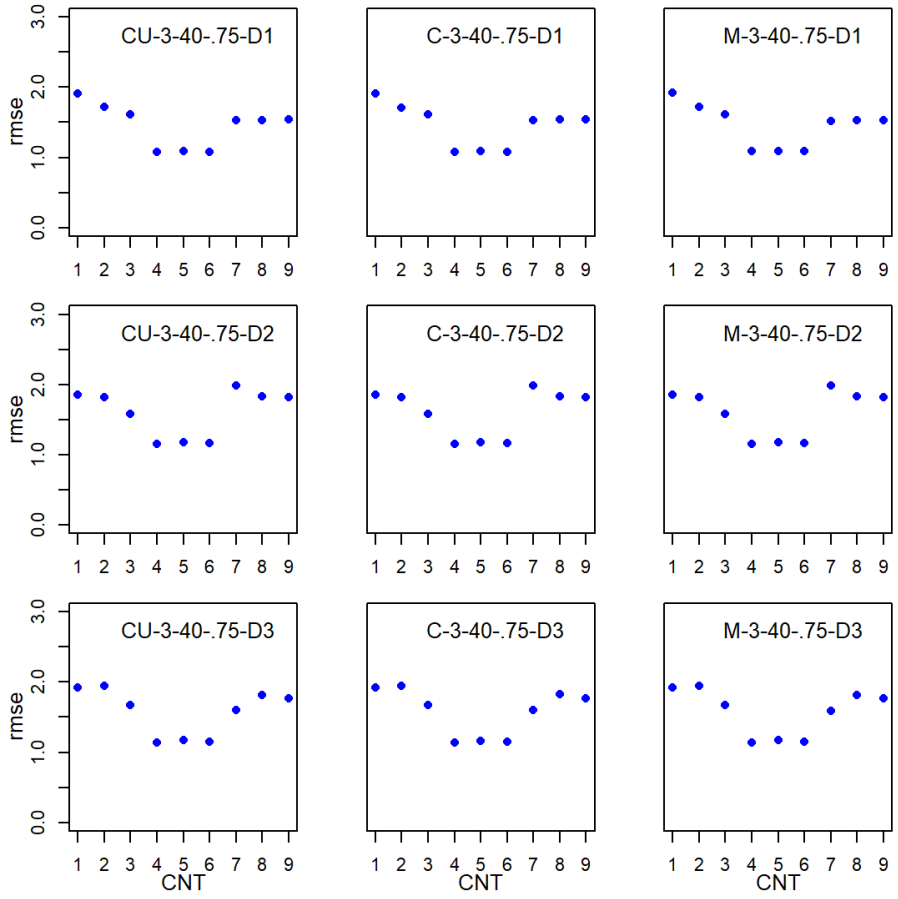
Subscale Score RMSE for the 3-Subdomain, 40-Item, .45 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

Figure L.14

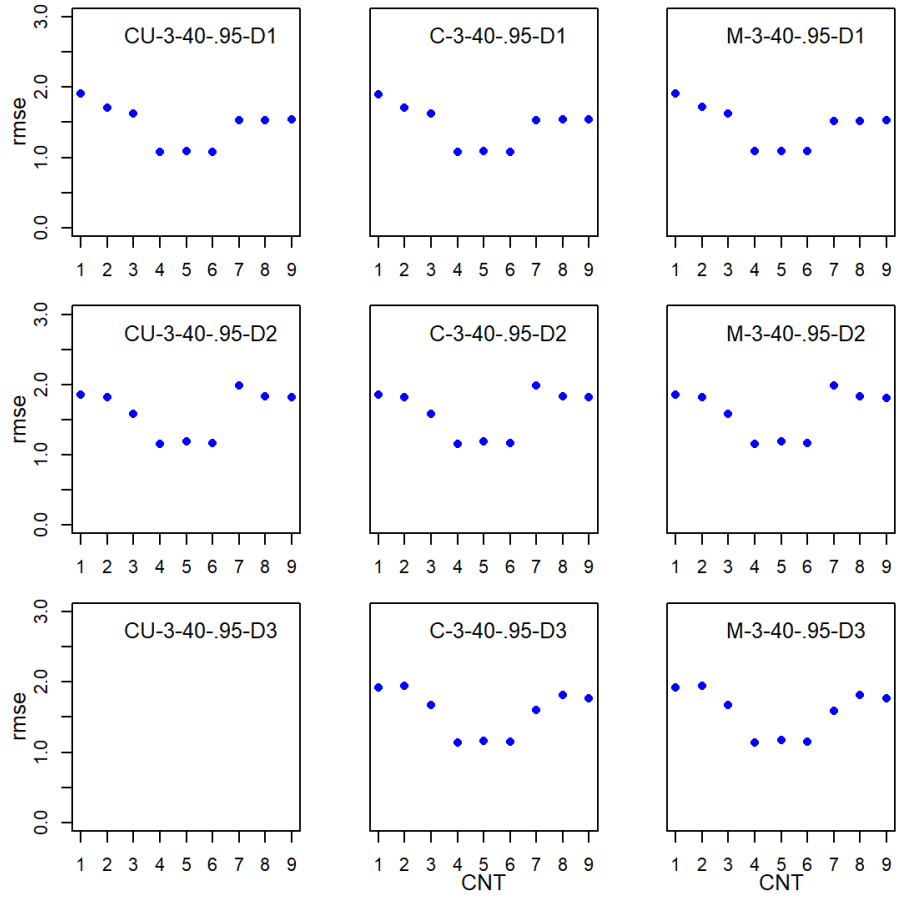
Subscale Score RMSE for the 3-Subdomain, 40-Item, .75 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

Figure L.15

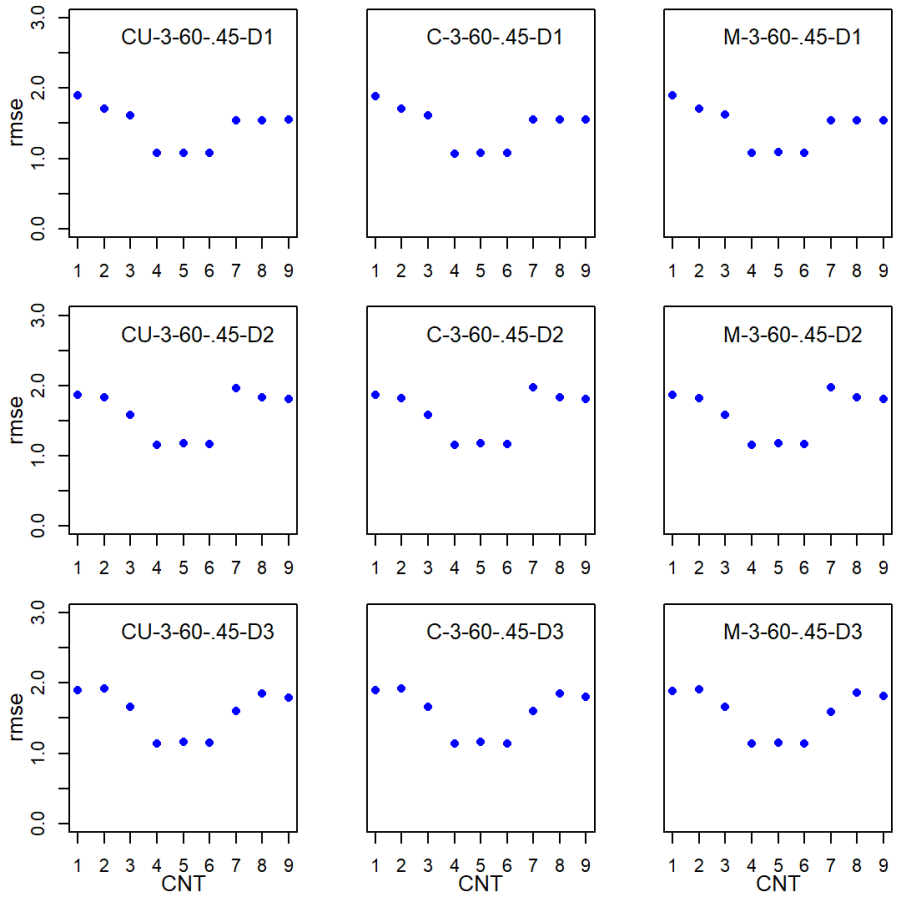
Subscale Score RMSE for the 3-Subdomain, 40-Item, .95 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

Figure L.16

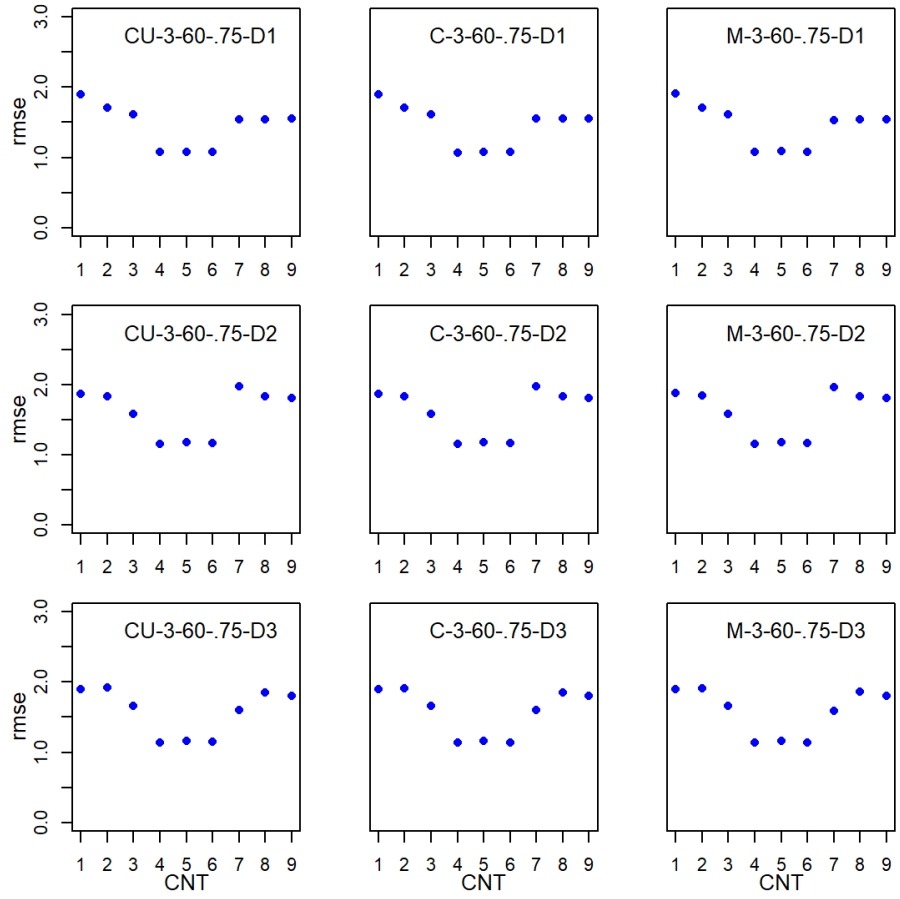
Subscale Score RMSE for the 3-Subdomain, 60-Item, .45 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

Figure L.17

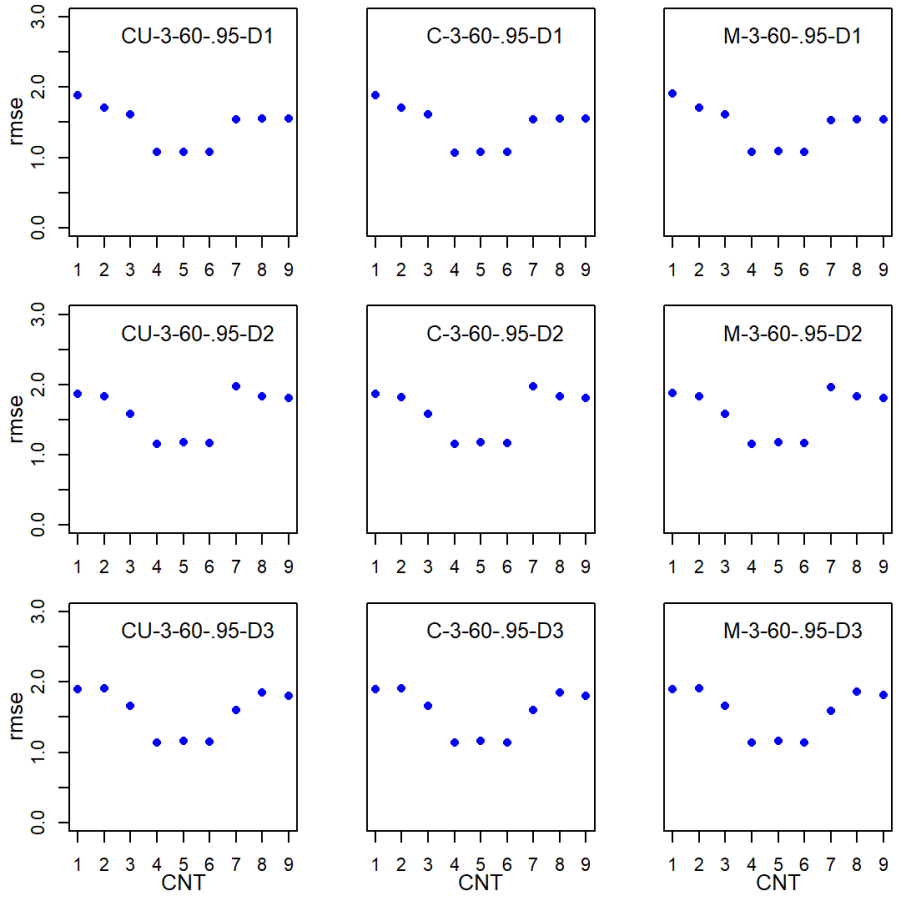
Subscale Score RMSE for the 3-Subdomain, 60-Item, .75 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

Figure L.18

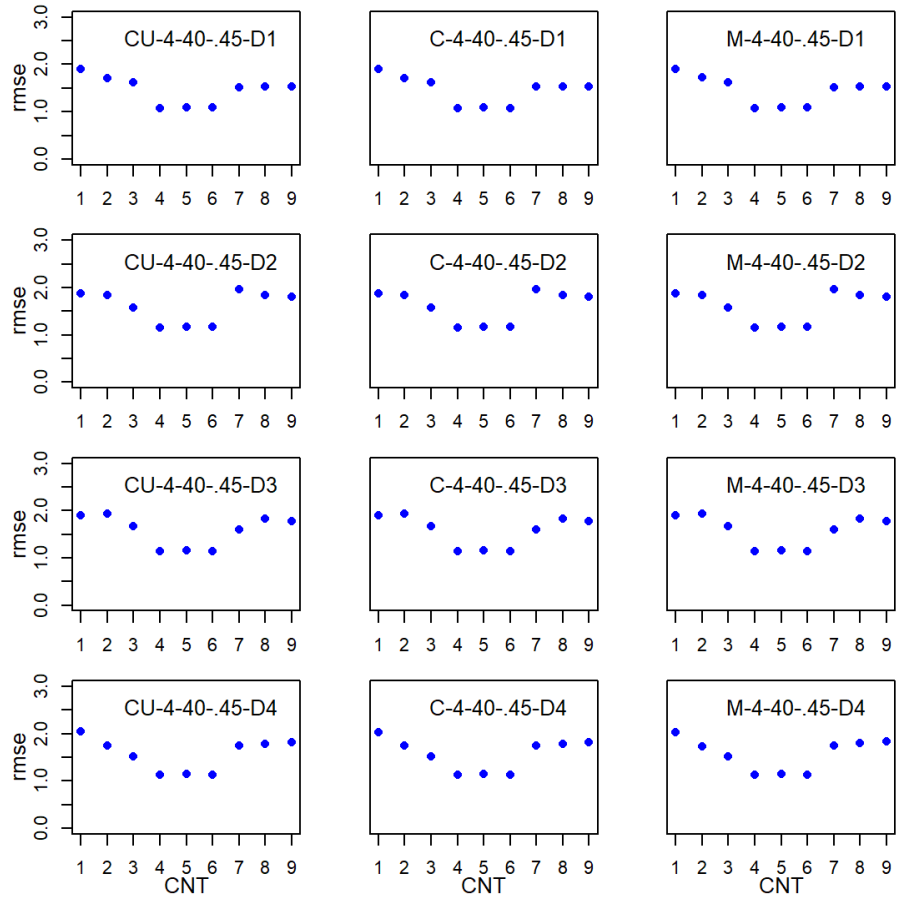
Subscale Score RMSE for the 3-Subdomain, 60-Item, .95 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3.

Figure L.19

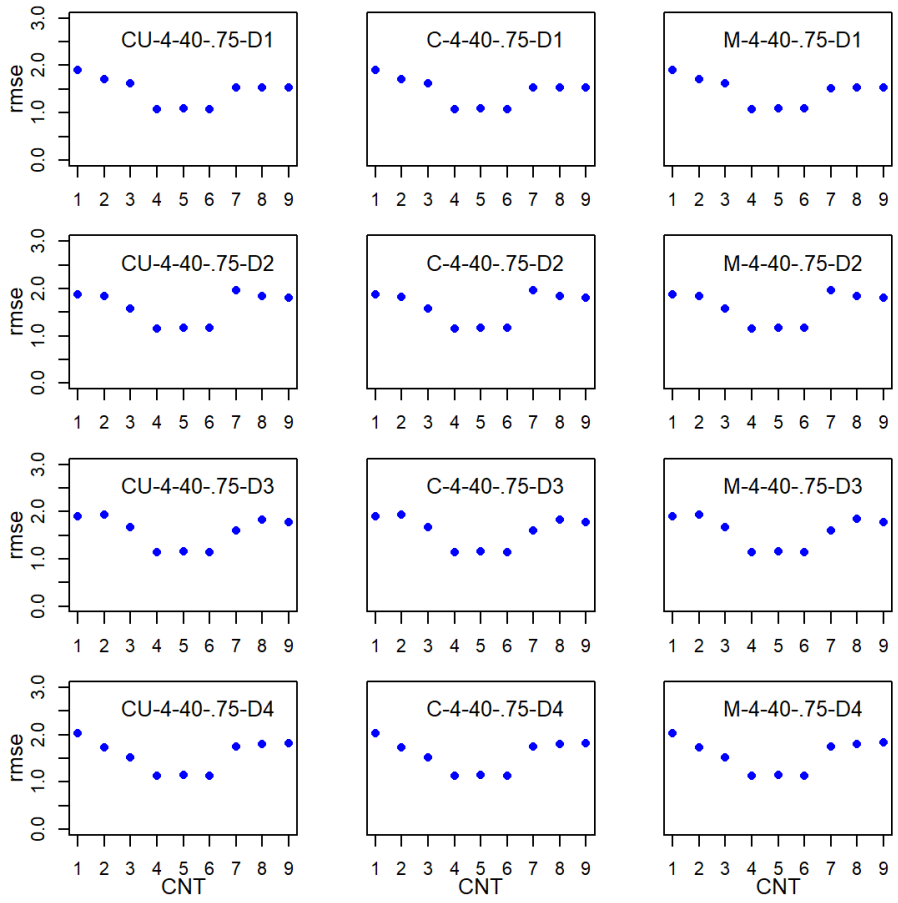
Subscale Score RMSE for the 4-Subdomain, 40-Item, .45 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4.

Figure L.20

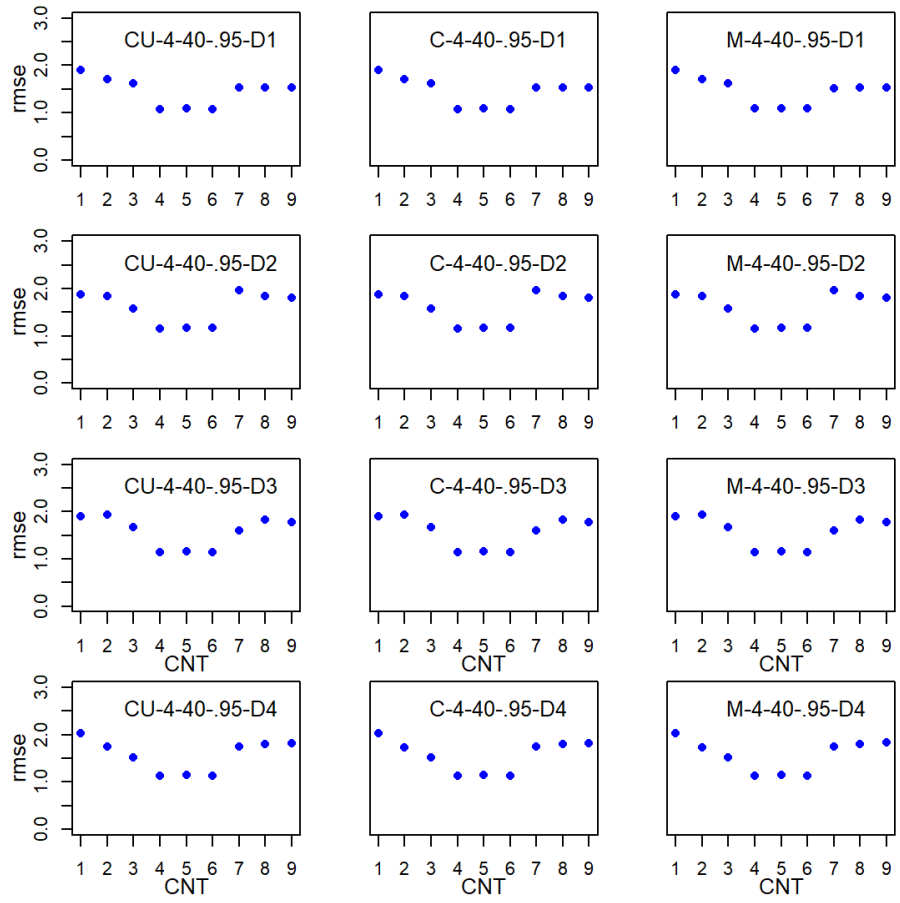
Subscale Score RMSE for the 4-Subdomain, 40-Item, .75 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4.

Figure L.21

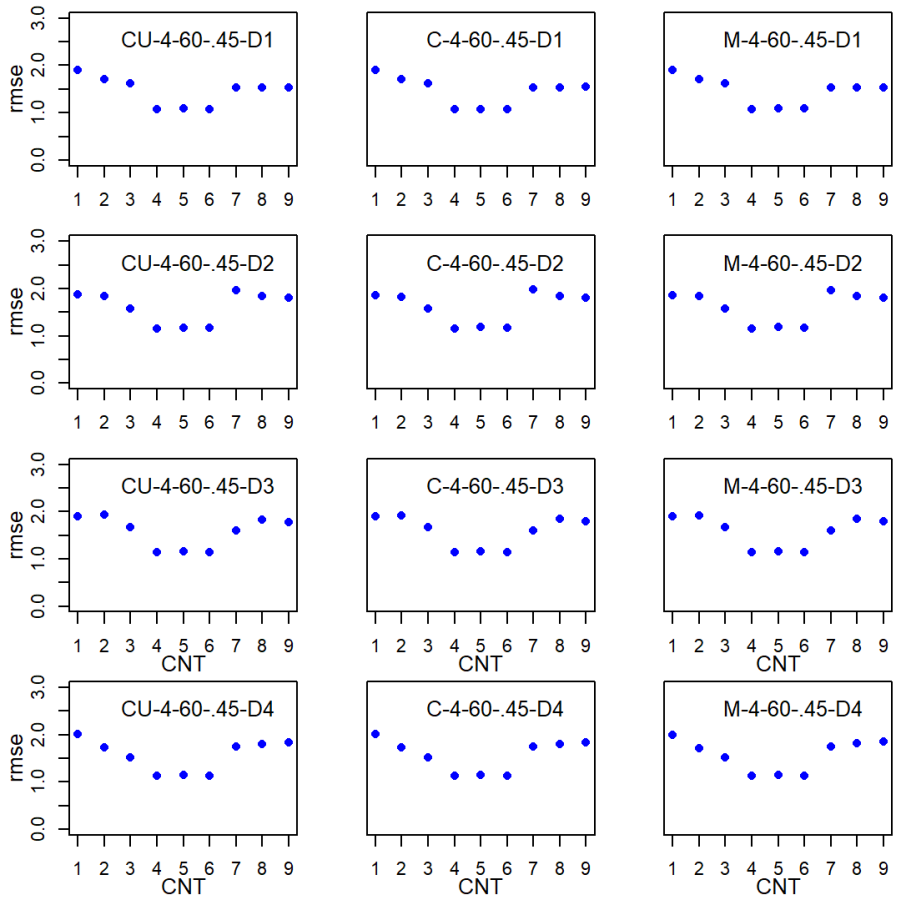
Subscale Score RMSE for the 4-Subdomain, 40-Item, .95 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4.

Figure L.22

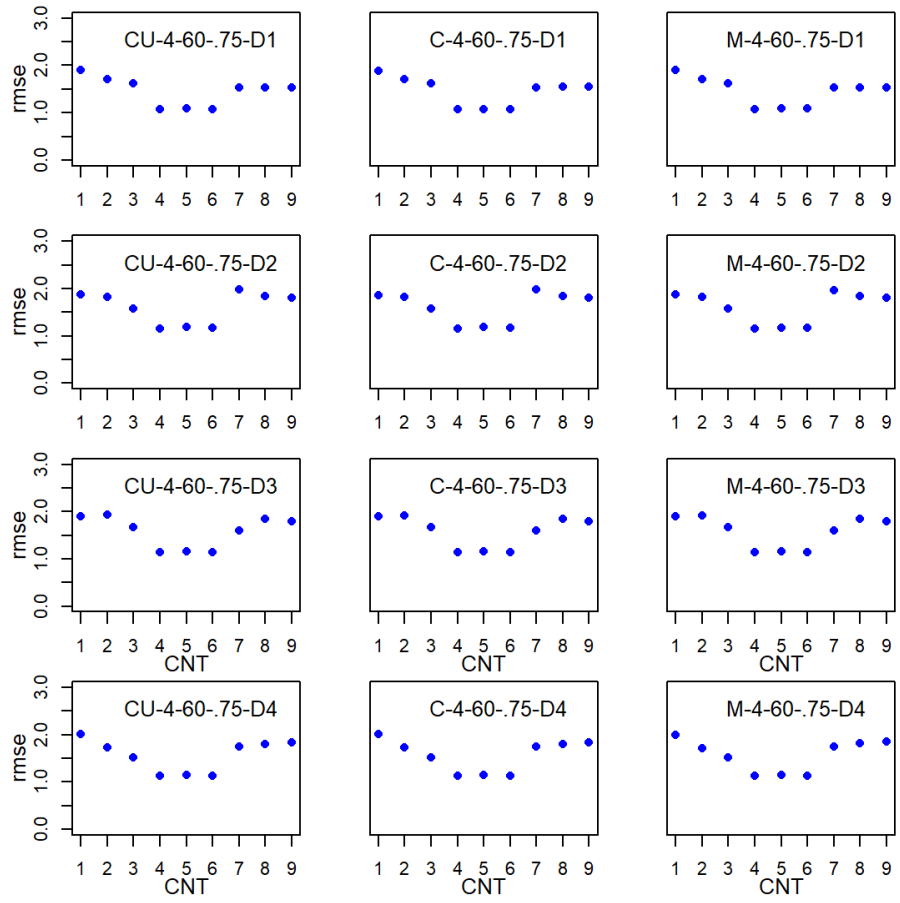
Subscale Score RMSE for the 4-Subdomain, 60-Item, .45 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4.

Figure L.23

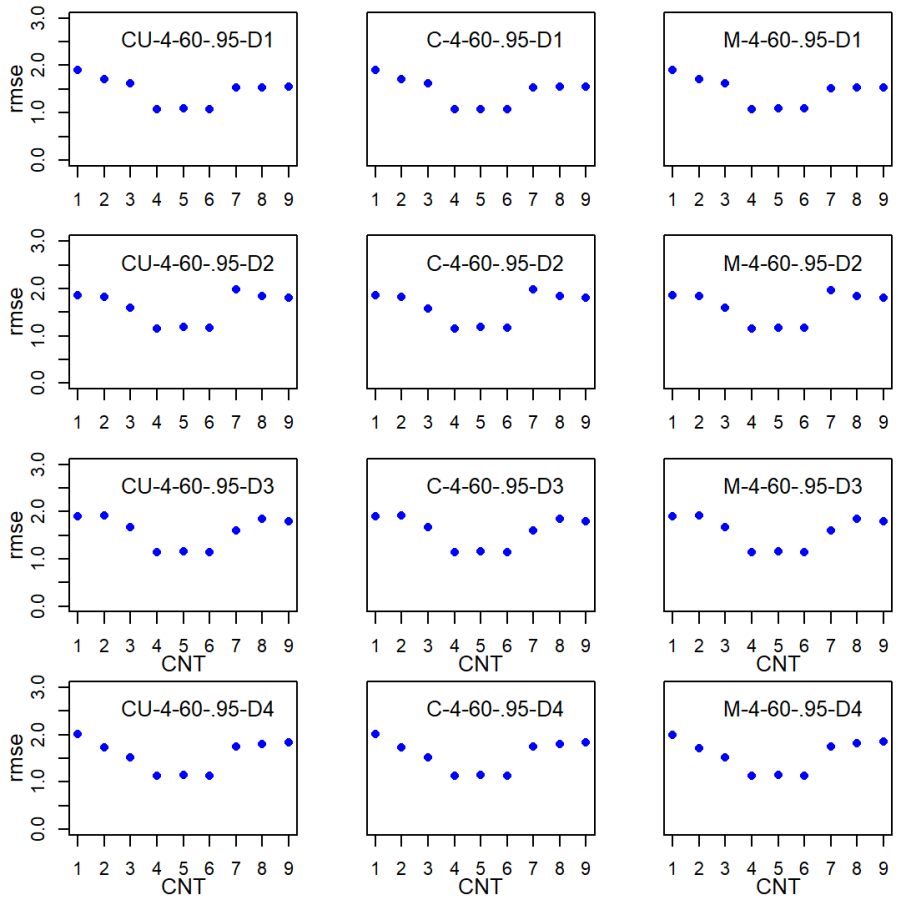
Subscale Score RMSE for the 4-Subdomain, 60-Item, .75 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4.

Figure L.24

Subscale Score RMSE for the 4-Subdomain, 60-Item, .95 Correlation Subdomain Tests



Note. CNT = country; CU = CUIRT; CU_Op = CUIRT-Op; C = CIRT; M = MIRT; $D1$ = domain 1; $D2$ = domain 2; $D3$ = domain 3; $D4$ = domain 4.