THE IMPACT OF MISSING DATA ON THE DICHOTOMOUS MIXTURE IRT MODELS

by

SUNBOK LEE

(Under the direction of Seock-Ho Kim and Allan S. Cohen)

Abstract

A mixture IRT model introduces latent classes into the model and is used when researchers want to model the heterogeneity of a population. It is designed to separate a population of individuals into qualitatively or quantitatively different subgroups or classes. Even though the mixture IRT model has turned out to be useful in many applications, the fundamental concern in using the mixture IRT model is whether the underlying population structure identified by the model really exists or not. The flexibility of the mixture IRT model may produce spurious classes to fit data better. A number of extended mixture IRT models have been developed and applied to various applications in IRT. It is well known that missing data can occur in many contexts and can result in biased parameter estimates. This study investigated whether the missing data could bias parameter estimates and also could produce additional classes in the dichotomous mixture IRT models. A simulation study was conducted to investigate the impact of missing data on the bias, root mean square error (RMSE), and the rate of correct identification of classes in mixture IRT models. The results of simulation study showed that missing data could introduce bias in parameter estimation and spurious classes in class identification.

Index words: Mixture model, item response theory, spurious class, missing data

THE IMPACT OF MISSING DATA ON THE DICHOTOMOUS MIXTURE IRT MODELS

by

SUNBOK LEE

B.A., Sogang University, Seoul, Korea, 2002

M.S., University of Georgia, 2008

M.S., University of Georgia, 2008

A Dissertation Submitted to the Graduate Faculty

of The University of Georgia in Partial Fulfillment

of the

Requirements for the Degree

Doctor of Philosophy

Athens, Georgia

2012

THE IMPACT OF MISSING DATA ON THE DICHOTOMOUS MIXTURE IRT MODELS

by

SUNBOK LEE

Approved:

Major Professor: Seock-Ho Kim and Allan S. Cohen

Committee: Jonathan L. Templin
Gary J. Lautenschlager

Electronic Version Approved:

Maureen Grasso
Dean of the Graduate School
The University of Georgia
August 2012

TABLE OF CONTENTS

Page

LIST OF FIGURES

Page

LIST OF TABLES

Page

Chapter 1

INTRODUCTION

1.1 Statement of the Problem

Researchers are applying a mixture structure to statistical models at an increasing rate. The mixture structure could be adopted by assuming that the population of interest is composed of more than one latent class. More formally, the mixture structure could be introduced to a statistical model by considering the mixture density function, which can be expressed by the sum of component density functions (McLachlan, 2000). By introducing a mixture structure, researchers can explore the heterogeneity of a population or detect a qualitative distinction among the subgroups. For example, the growth mixture model (GMM), (Muthén & Shedden, 1999; Muthén, 2001) was developed to introduce a mixture structure to the traditional latent growth model. GMM could be used to separate a population into subgroups representing distinct growth patterns over time. It has been applied to many contexts of research including reading skill (Parrila, Aunola, Leskinen, Nurmi, & Kirby, 2005), depression (Stoolmiller, Kim, & Capaldi, 2005), and community intervention (Segawa, Ngwe, Li, & Flay, 2005).

Rost (1900) introduced the mixture structure to the Rasch model (Rasch, 1960) in item response theory (IRT). In the traditional Rasch model, the parameters of each item are assumed to be the same across the whole population, which reflects the implicit assumption that the population of interest is homogeneous. In the mixture Rasch model (MRM), however, the parameters of each item are allowed to be different across latent classes, which gives us flexibility to model heterogeneity in the population.

The mixture IRT model has proved to be useful in many applications. It can be used to detect latent groups using different strategies for responding to test items (Bolt, Cohen, &

Wollack, 2001) and identifying items showing differential item functioning (Cohen & Bolt, 2005; Cohen, Gregg, & Deng, 2005). Mixture models are widely used when a single parametric family is unable to provide a satisfactory model for local variations in the observed data such as image processing and medical applications (McLachlan, 2000).

However, with the increasing use of mixture models, there have been concerns about spurious classes from the mixture model. Day (1969) demonstrated the existence of spurious classes by showing that a certain univariate projection of data generated by a 10 dimensional multivariate normal distribution can be identified to have multiple classes. Even though this example is somewhat extreme, one thing suggested by this example is that the classes identified by the mixture model do not always reflect the underlying population heterogeneity. Bauer (2007) and Bauer and Curran (2003) discussed five assumptions of GMM and how the violation of each assumption affects the identification of the correct number of classes. The five GMM assumptions were within-class normality, correct specification of covariance structure, correct specification of linear or non-linear effect of covariates, missing at random (MAR) assumptions for missing data, and independence of sample individuals. It was shown violation each of five assumptions could produce spurious or artificial classes in the use of GMM.

Tofighi and Enders (2007) also supports Bauer's concern on the use of the mixture model under the possibility of producing spurious classes. In the context of GMM, Tofighi and Enders (2007) considered five factors that may affect the proportion of correctly identifying the number of classes: the number of repeated measures, sample sizes, separation of the latent classes, the mixing percentage, and within class distribution shape. Five fit measures were used to determine the number of classes: the Bayesian information criterion (BIC; Schwarz, 1978), a sample-size adjusted BIC (SABIC; Sclove, 1987), Akaike information criterion (AIC; Akaike, 1974), the consistent AIC (CAIC; Bozdogan, 1987), and a sample-size adjusted CAIC (SACAIC; Bozdogan, 1987). The results from simulation study showed that

the proportions of correctly identifying the number of classes varied dramatically with respect to each condition in the five factors and five fit measures mentioned above.

Similar work has been done in the IRT context. Li, Cohen, Kim, and Cho (2009) compared the five fit measures to investigate the proportions of correctly identifying the number of classes: AIC, BIC, deviance information coefficient (DIC; Spiegelhalter, Best, & Carlin, 1998), pseudo Bayes factor (PsBF; Geisser & Eddy, 1979; Gelfand & Dey, 1994), and posterior predictive model checks (PPMC; Gelman, Carlin, Stern, & Rubin, 1996). The results from Li et al. (2009) also showed that the proportion of correct identification for the class number depended on the selection of fit measure.

By introducing the mixture structure to a model, researchers would have more flexibility in a model to fit a given data. This flexibility comes from the fact that the mixture model assumes underlying heterogeneity in the population of interest, which is reflected in the fact that mixture models use a mixture density instead of a single density function. However, the classes identified by any mixture model could be spurious. In other words, factors other than actual heterogeneity could produce artificial classes. Therefore, it is important to investigate various factors that can affect the identification of classes in the mixture model.

On the other hand, the problem of missing data has been an issue in statistical modeling not particularly in the context of mixture modeling. Rubin (1976) classified missing data into three types: missing completely at random (MCAR), missing at random (MAR), and missing not at random (MNAR). The author investigated how the maximum likelihood estimation (MLE) and Bayesian estimation work in the presence of these three types of missing data and demonstrated that theoretically MLE is robust to MCAR and MAR. In the structural equation model (SEM) context, Enders and Bandalos (2001) showed that the full information maximum likelihood (FIML) method was best in terms of unbiasedness and efficiency compared to the other three methods: listwise deletion, pairwise deletion, and similar response pattern imputation. Sometimes, unavoidable missing data can be generated by design itself. The balance between cost effectiveness and design efficiency has been an

important issue in research design study (Graham, Taylor, Olchowski, & Cumsille, 2006). One of the ways to achieve cost effectiveness of design is to allow missing data. Different items are administrated to different respondents to reduce the amount of time for respondents to finish the survey (e.g., 3-forms design). In this case, it is important to investigate the extent to which the planned missing data affect the estimation of a statistical model based on the data set. Also, many studies have been done on the effect of missing data in the IRT context. Mislevy and Wu (1996) specified six mechanisms that can produce missing responses: alternative test forms, targeted testing, adaptive testing, not-reached items, omitted items, and examinee choice of tasks. Each of the six mechanisms was assumed to be one of the three missing mechanisms defined by Rubin (1976). They focused on whether the mechanism for missingness is ignorable or not based on the values of ability estimates. The impact of missing data on the differential item functioning (DIF) also has been studied (Finch, 2011; Robitzsch & Rupp, 2009). Missing data could be an issue in equating. In the equating, the non-equivalent groups with anchor test (NEAT) design involves missing data by design. The impact of missing data produced by NEAT equating design was also investigated (Sinharay & Holland, 2010).

1.2 Purpose of the Study

Given the fact that mixture IRT models are becoming more popular in many applications, it is worthwhile to study whether the ubiquitous sources of missing data mentioned above could be a potential threat to the validity of inference in mixture IRT models. More specifically, this study will consider the two dichotomous mixture IRT models, the mixture Rasch model and the two-parameter logistic model (2PL), and how various factors including the types of missing, the percentages of missing, sample size, and test length affect the estimation of these mixture IRT models. To see the impact of such missing conditions on the mixture model, the following quantities will be examined in the simulation study. First, item parameter estimates in each class will be examined to check whether the missing data causes bias in estimation.

The recovery evaluation will be done using root mean square error (RMSE). Second, the number of classes identified by the mixture model based on AIC and BIC will be monitored to see whether the missing data cause additional classes. A more detailed description of the simulation study will be presented in Chapter 3.

Chapter 2

THEORETICAL FRAMEWORK

As described in Chapter 1, the focus of this study is to investigate the impact of missing data on the dichotomous mixture IRT models. In this chapter, the theoretical background will be presented to discuss the stated problem. Since the mixture IRT models are the special case of mixture models, the definitions and the issues related to mixture models in general will be presented in section 2.1. This will include the definition of mixture model, incomplete data structure of mixture models, identifiability, and expectation-maximization (EM) algorithm for maximum likelihood estimation (MLE). Also, as a special case of mixture models, the mixture Rasch model will be discussed in section 2.2. In this study, class classification of the mixture IRT models is the main focus and it is done by using model fit indices such as AIC and BIC. Therefore, it is important to understand the idea behind those two model fit indices in order to discuss the class classification of mixture models made by them. The motivations and characteristics of those two model fit indices will be discussed in section 2.3. In section 2.4, the types of missing data and how the MLE is affected according to the types of missing data will be presented.

2.1 Mixture Models in General

The usefulness of a mixture model is its flexibility for modeling sub-populations within an overall population. Formally, a mixture model is a probabilistic model which uses the mixture distribution or density function represented by the sum of g component density functions (McLachlan, 2000). In the mixture model, an observation is considered to be drawn from a population G which consists of g groups in proportions $\pi_1, \cdots, \pi_g$. A component label

vector, which is zero-one indicator variable, could be introduced to define the component in the mixture model from which an observation was drawn. Since this component label vector is unknown, it could be considered as missing from the complete data vector. This conceptualization of the mixture model in terms of an observation and component label vector is useful in that it allows the maximum likelihood estimate (MLE) of the mixture distribution to be computed via the EM algorithm since the EM algorithm was initially developed to estimate parameters in the presence of missing data. Since the number of components in a mixture distribution is initially unknown, testing the number of components is an important issue in mixture modeling, which may produce spurious components. This testing is usually performed using the information criteria such as AIC and BIC. The following are more formal descriptions of the mixture model which was described above based on McLachlan (2000).

Definition of Mixture Model. Let's denote $\mathbf{Y}_1, \mathbf{Y}_2, ..., \mathbf{Y}_n$ a random sample of size n, where $\mathbf{Y}_j$ is a p-dimensional random vector with probability density function $f(\mathbf{y}_j)$ on $\mathbb{R}^p$. Suppose that the density function for $\mathbf{Y}_j$ has the following form:

$$f(\mathbf{y}_j; \mathbf{\Psi}) = \sum_{i=\mathbf{1}}^{g} \pi_i f_i(\mathbf{y}_j; \theta_i), \tag{2.1}$$

where the $f_i(\mathbf{y}_j; \theta_i)$ are any density functions and are called component densities of the mixture, the π_i $(i = 1, ..., g)$ are nonnegative numbers that sum to one and are called the mixing proportions or weights, and $\mathbf{\Psi}$ is the vector of all the parameters in the mixture model, that is, $\mathbf{\Psi} = (\pi_1, ..., \pi_{g-1}, \theta_1, ..., \theta_g)$. This mixture model represents the case where $\mathbf{Y}_j$ is drawn from a population G, which consists of g groups, $G_1, ..., G_g$, with proportions $\pi_1, ..., \pi_g$. In order to deal with membership information to a certain component, it is convenient to define g-dimensional component label vector $\mathbf{Z}_j$, where the ith element of $\mathbf{Z}_j$ is defined to be one or zero, according to whether $\mathbf{Y}_j$ is drawn from the component i or not. Thus, $\mathbf{Z}_j$ can be considered to be drawn from the following multinomial distribution:

$$\Pr[\mathbf{Z}_j = \mathbf{z}_j] = \pi_1^{z_{1j}} \pi_2^{z_{2j}} \cdots \pi_g^{z_{gj}}. \tag{2.2}$$

Here, π_i can be considered as the prior probability that the observation belongs to the ith component of the mixture, while the posterior probability $\tau_i(\mathbf{y}_j)$ that the observation belongs to the ith component after observing $\mathbf{y}_j$ is given by the following equation:

$$\begin{aligned} \tau_i(\mathbf{y}_j) &= \Pr[Z_{ij} = 1; \mathbf{y}_j] && (2.3) \\ &= \pi_i f_i(\mathbf{y}_j)/f(\mathbf{y}_j). && (2.4) \end{aligned}$$

Incomplete Data Structure of Mixture Models. In mixture models, we are estimating parameters of the mixture distribution or densities, which is given by Equation 2.1, given the data $\mathbf{y}_1, ..., \mathbf{y}_n$. The data $\mathbf{y}_1, ..., \mathbf{y}_n$ are assumed to be the realization of n independent and identically distributed ($i.i.d.$) random vectors $\mathbf{Y}_1, ..., \mathbf{Y}_n$ with common density $f(\mathbf{y}_j)$. One usual way to look at this problem is to consider the data $\mathbf{y}_1, ..., \mathbf{y}_n$ as incomplete since their corresponding component label vectors $\mathbf{z}_1, ..., \mathbf{z}_n$ are unknown. In this sense, the complete data vector, $\mathbf{y}_c$, for the estimation of the mixture density can be considered to have the following form:

$$\mathbf{y}_c = (\mathbf{y}, \mathbf{z}), \tag{2.5}$$

where $\mathbf{y}$ is observed data vector and $\mathbf{z}$ is the unobserved component label vector. In this context, the component label vector $\mathbf{z}$ can be considered as missing data from a complete data vector $\mathbf{y}_c$. This formulation of the problem of mixture models explains why the EM algorithm is so popular in estimating parameters in mixture models since the EM algorithm was initially developed to get maximum likelihood estimation in the presence of missing data (Dempster, Laird, & Rubin, 1977).

Identifiability of Mixture Models. The identifiability of distributions is the statement about uniqueness. Simply, it means that different values of parameters should correspond to different probability distributions. More formally, a parametric family of probability distributions $f(\mathbf{y}_j; \mathbf{\Psi})$ is identifiable if distinct values of the parameters $\mathbf{\Psi}$ correspond to distinct members of the family of probability distributions:

$$f(\mathbf{y}_j; \boldsymbol{\Psi}) = f(\mathbf{y}_j; \boldsymbol{\Psi}^*) \tag{2.6}$$

if and only if

$$\boldsymbol{\Psi} = \boldsymbol{\Psi}^*. \tag{2.7}$$

However, the identifiability of distributions in mixture models is slightly different from the one in general since the equation still holds even though the component labels are interchanged in $\boldsymbol{\Psi}$. If all the g components belong to the same family, then $f(\mathbf{y}_j; \boldsymbol{\Psi})$ is invariant under the $g!$ permutations of the component label switching in $\boldsymbol{\Psi}$. Thus, the definition of identifiability in mixture models should consider the issue of component label switching. In mixture models, the family of mixture distributions is defined to be identifiable if the following condition is met:

$$f(\mathbf{y}_j; \boldsymbol{\Psi}) = f(\mathbf{y}_j; \boldsymbol{\Psi}^*) \tag{2.8}$$

if and only if $g = g^*$ and there exists the permutation of component labels such that $\pi_i = \pi_i^*$ and $f_i(\mathbf{y}_i; \theta_i) = f_i(\mathbf{y}_i; \theta_i^*)$. This condition means that two mixture distributions are considered to be identifiable if we can find component labels for each mixture distribution that make the component densities and mixing proportions identical.

The term label switching was first used by Redner and Walker (1984) to describe the invariance of the likelihood under relabeling of the mixture components. If one is only interested in a point estimator of maximum likelihood estimation, this label switching problem is not of concern (McLachlan, 2000). However, in a Bayesian context, the invariance of likelihood under label switching can result in highly symmetrical and multimodal posterior distribution. This type of posterior distribution could be problematic since the usual practice of summarizing and estimating quantities of interest by the mean of marginalized posterior is often inappropriate (Stephens, 2000). Cho, Cohen, and Kim (2006) discussed two types of label switching. The first type of label switching occurs across iterations within a single chain of Bayesian estimation process. The second type of label switching illustrated the case

when the class labeled as class 1 when it was generated but labeled as class 2 when it was estimated.

In order to make unidentifiable $\mathbf{\Psi}$ due to the interchanging of components label be identifiable, a certain constraint is usually imposed on $\mathbf{\Psi}$. For example, Aitkin and Rubin (1985) imposed the following constraint that can be satisfied by only one permutation of the parameters:

$$\pi_1 \leq \pi_2 \leq ... \leq \pi_g. \tag{2.9}$$

However, many choices of constraint will not remove the symmetry and multimodality of the posterior distribution. In that case, the problem of label switching may remain after imposing an constraint (Stephens, 2000).

MLE for Mixture Models. Given the basic definition of a mixture density and other issues in mixture models, now we can discuss the maximum likelihood estimation (MLE) for mixture models. As with any other MLE procedure, a likelihood function $L(\mathbf{\Psi})$ can be constructed from the mixture density first and then the MLE can be obtained by solving the following equation:

$$\frac{\partial \log L(\mathbf{\Psi})}{\partial \mathbf{\Psi}} = 0. \tag{2.10}$$

The main difference of the MLE in mixture models from any other usual MLE is the inherent existence of missing data. As described in the previous section, the data for a mixture model are usually considered to be incomplete in the sense that the component label vector $\mathbf{Z}_j$ is unknown. The EM algorithm is an iterative method for finding the MLE with the joint likelihood function of $\mathbf{Y}_j$ and $\mathbf{Z}_j$ in the presence of incomplete data.

EM Algorithm for Mixture Models. The idea of the EM algorithm comes from the old method for handling missing data (Little & Rubin, 1987). One of the traditional methods for handling missing data is the following: 1) replace the missing data by initially estimated values, 2) estimate parameters of interest given the whole data including the ones estimated in step 1), and 3) repeat step 1 with the new parameter values obtained in step

2. This simple idea can be applied to get the MLE via the EM algorithm in the presence of an incomplete data. The EM algorithm consists of two steps: expectation step (E-step) and maximization step (M-step). The idea of E-step is the following: The likelihood is the function of parameters given data, that is, $L(\mathbf{\Psi}; \mathbf{y}_j, \mathbf{z}_j)$, where $\mathbf{y}_j$ and $\mathbf{z}_j$ represent observed data and missing data respectively.

The core idea of EM algorithm is to overcome the existence of missing data, $\mathbf{z}_j$, by iteratively working with the conditional expectation of the log likelihood for the complete data when only the observed data $\mathbf{y}_j$ are given. In other words, since we do not know the exact value of log likelihood for the complete data in the presence of missing data, we try to use expected or estimated values of the log likelihood for the complete data instead. Simply, we are guessing. This E-step corresponds to the Step 1 in the above example of handling missing data. In order to calculate the expectation of log likelihood for the complete data, the conditional distribution of $\mathbf{z}_j$ given $\mathbf{y}_j$, $f(\mathbf{z}_j; \mathbf{y}_j)$ is required. The parameters that are required to construct conditional distribution is given by the M-step. In the M-step, the parameters of interest are obtained by maximizing the expectation of log likelihood for the complete data obtained in the E-step. This M-step is the exactly same with the usual MLE except the fact that the likelihood is not the exact one but the expected one. The E and M steps are repeated until the difference in the log likelihoods for complete data between successive steps, $L(\mathbf{\Psi}^{(k+1)}) - L(\mathbf{\Psi}^{(k)})$, becomes less than some predetermined small amount, which is usually called convergence criterion. One of the beauties of using this EM algorithm for the mixture models is that it uses a complete data vector $\mathbf{y}_c$ instead of just using observed data vector $\mathbf{y}$ since each component density $f_i(\mathbf{y})$ could be estimated directly from the data known to come from it. Also, the MLE of mixture models, which could be considered as the MLE under incomplete data, is known as a classical example of a problem that is greatly simplified by the EM algorithm.

2.2 The Mixture Rasch Model

The role of measurement models is to map observed responses or manifest variables onto unobserved latent variables. The measurement models establish one to one correspondence between them. It is possible to categorize them based on whether the manifest variable and the latent variable are categorical or continuous. Table 2.1 below shows the classification of measurement models (Bartholomew, Steele, Moustaki, & Galbraith, 2002). As indicated in the table, the latent variable is assumed to be continuous in IRT, whereas the manifest variable of IRT is assumed to be categorical. Basically, the task of IRT is to map or locate respondents on the underlying latent continuum, which is usually called ability of respondents, based on the their responses to the items. In contrast, the latent variable in latent class analysis (LCA) is assumed to be categorical, whereas the manifest variable is still assumed to be categorical. Both IRT and LCA are the same in that both of them are used to model a categorical manifest variable. The mixture Rasch model (Rost, 1990) is an attempt to combine IRT and LCA.

Table 2.1: Classification of Measurement Models

Latent variable / Manifest variable	Continuous	Categorical
Continuous	Factor analysis	Latent trait analysis (IRT)
Categorical	Latent profile analysis	Latent class analysis

The Mixture Rasch Model as a Composite of IRT and LCA. The fact that a Rasch model assumes the continuous latent variable implies that the purpose of the Rasch model is to locate a person on the underlying latent continuum or to assign a number to each subject. On the contrary, the use of the categorical latent variable in LCA implies that its purpose is to classify subjects in the qualitative sense or to assign the class membership to subjects. In this sense, the purpose of the mixture Rasch model, which combines IRT and LCA, can be said to classify subjects and then to locate them within each class simultaneously. Based on the perspective of LCA, the feature of the mixture Rasch model that can locate subjects within a class could be considered as major progress since, in LCA, all

subjects within a class are considered to be the same in some sense. In LCA, subjects are classified into classes, which reflect qualitative differences among subjects, but no further distinctions are made among the subjects within a class. This has been pointed out as a limitation of LCA. However, the mixture Rasch model can classify subjects in a qualitative sense using classes and then further differentiate subjects within a class in the quantitative sense using the ability within a class. On the other hand, from the perspective of the Rasch model, the mixture Rasch model is the extension of the Rasch model in the sense that it can deal with the heterogeneity of a population. In the Rasch model, the parameters of items are assumed to be the same across the whole population, which reflects the implicit assumption that the population of interest is homogeneous. The mixture Rasch model, however, allows different item parameter values across latent classes, which gives us flexibility to model heterogeneity in a population. To be more specific, let's look into the equations of the Rasch and mixture Rasch model. First, the Rasch model is described by the following equation:

$$\Pr[Y_{ij}|\theta_i] = \frac{\exp(\theta_i - b_j)}{1 + \exp(\theta_i - b_j)}\,, \tag{2.11}$$

where Y_{ij} represents the response of individual i to j item, θ_i is the ability of the individual i , b_j is a difficulty parameter of the item, and $P[Y_{ij}|\theta_i]$ is the probability of the correct response conditional on the ability θ. One of the important implications of this equation is that the person and item are located on the same latent continuum, enabling us to make comparative statements about how a typical person might response to an item (De Ayala, 2009).

The mixture Rasch model could be considered as the composite of the Rasch model and LCA, which is described by the following equation:

$$\Pr[Y_{ij}|\theta_i] = \sum_{g=1}^{G} \pi_g \frac{\exp(\theta_i - b_{jg})}{1 + \exp(\theta_i - b_{jg})}\,, \tag{2.12}$$

where g is the index for a class. It should be noted that the difficulty parameter b_{jg} has an index g for the class membership, which implies that each class has its own item parameters.

In other words, an item could be easy for subjects in one class but difficult for subjects in another class. On the other hand, in a mixture Rasch model, a subject is assumed to be classified exclusively into only one class. Since this class membership of a subject is unknown, a person's ability parameters are estimated conditioning on the class membership in the estimation process. As discussed in the previous section, this is where the EM algorithm comes in for the estimation of parameters in mixture models. The membership parameters of subjects are treated as missing and estimated iteratively through the EM algorithm. Eventually, the probabilities of each subject belonging to classes are obtained by counting the number of frequencies of which each subject belonged to each class during the iteration process.

The Dimensionality in IRT. Like any other model in statistics, IRT will work best under the condition where all the underlying assumptions of the model are met. A model could produce unintended or undesirable results when the assumptions of a model are not met. Therefore, to get a valid result, it is necessary to check if the data at hand really meet the assumptions of a model or not. On the other hand, investigating the robustness of a model to the violation of assumptions could be useful in justifying the use of a model under the condition where some of the assumptions are not met. In this section, dimensionality will be discussed as an important assumption of IRT. This discussion on dimensionality in IRT could be useful when we discuss the multidimensionality nature of the mixture Rasch model.

Classical test theory (CTT) is often referred to as a weak model since the assumptions of CTT could be easily met by data. On the contrary, IRT is known as a strong model because of its more stringent assumptions which are less likely to be met by data. The assumptions of IRT are the following: unidimensionality, local independence, and functional form assumptions (De Ayala, 2009). The difference between unidimensionality and local independence could be confusing. Sometimes, unidimensionality seems to imply local independence. The rationale behind this idea might be the following: Suppose that unidimensionality holds,

then this means that the items at hand measure a unique trait. Then, by controlling the unique trait, the dependency among the item responses could be removed. In all, if unidimensionality holds, then local independence also holds. In this way, unidimensionality seems to imply local independence. However, in testlet situation, the dependency among items still remains even after controlling the latent trait, which could be a counter-example of the previous argument. According to Hattie (1985), local independence is more fundamental concept than unidimensionality. In fact, unidimensionality is defined by local independence. Hattie cited comment in Lord, Novick, and Birnbaum (1968/2008) on local independence:

> An individual's performance depends on a single underlying trait if, given his value on that trait, nothing further can be learned from him that can contribute to the explanation of his performance. The proposition is that the latent trait is the only important factor and, once a person's value on the trait is determined, the behavior is random, in the sense of statistical independence. (p. 538)

McDonald (1962) argued that local independence is actually the mathematical definition itself of latent trait. Once the definition of local independence is clarified, then it is possible to define unidimensionality. Unidimensionality is the assertion that there exists only one latent trait underlying the set of items. Hattie gave a more rigorous definition of it:

> As a working definition of unidimensionality, a set of items can be said to be unidimensional when it is possible to find a vector of values $\phi = \phi_i$ such that the probability of correctly answering an item g is $\pi_{ig} = f_g(\phi_i)$ and local independence holds for each value of ϕ.

As described above, unidimensionality is an important assumption of commonly used IRT models including the Rasch model. In fact, the mixture Rasch model could be considered as the generalization of the Rasch model by relaxing the unidimensionality assumption. Thus, it is worthwhile to review the methods for checking unidimensionality. Hattie reviewed over 30 methods for assessing unidimensionality and grouped them into 4 categories: indices based

on the answer pattern, reliability, principal components analysis (PCA) and factor analysis (FA), and latent trait model (or IRT). Indices from the answer pattern are based on the idea that unidimensionality can be determined by the deviation of an observed response pattern from the ideal response pattern which occurs when a total test score n is composed of correct answers to n easiest questions.

In Hattie's review, reliability methods for unidimensionality assessment focus on the coefficient α. It is a well-known measure of internal consistency or inter-correlations among test items. It is defined as the following equation:

$$\alpha = \frac{K}{K-1}\left[1 - \frac{\sum_{i=1}^{k}\sigma_{X_i}^2}{\sigma_X^2}\right], \tag{2.13}$$

where $X = \sum_{i=1}^{K} X_i$ and K is the number of components or items. The meaning of the coefficient α can be more clearly seen when the equation above is slightly modified into the following equation:

$$\alpha = \frac{K}{K-1}\left[\frac{\sigma_X^2 - \sum_{i=1}^{k}\sigma_{X_i}^2}{\sigma_X^2}\right] = \frac{K}{K-1}\left[\frac{\sum_i \mathrm{Cov}[X_i, Y_j]}{\sigma_X^2}\right], \tag{2.14}$$

The term $\sum_i \mathrm{Cov}[X_i, Y_j]$ represents the inter-correlation among items and explains why the coefficient α is a measure of degree to which items vary together. Based on the above definition of coefficient α, it seems that α could be used as a measure of unidimensionality since the inter-correlation among items would be high when the items measure a single construct. Hattie pointed out that unidimensionality implies a high level of internal consistency (i.e., large coefficient α) but the converse does not hold. A high level of internal consistency is a necessary but not sufficient condition for unidimensionality. That is, it is possible to get a large value of coefficient α when there exist multiple latent traits (Green, Lissitz, & Mulaik, 1977).

PCA and FA could also be used for assessing unidimensionality. Both of them are dimension reduction techniques. The difference between the two methods is that PCA reduces the number of variables using the linear combination of observed variables (i.e., component),

whereas FA do the same thing by assuming unobserved latent variable (i.e., factor). In the context of PCA and FA, unidimensionality could be defined by the existence of the single dominant component or factor that can explain the variability among the items.

Hattie reviewed around 10 unidimensionality assessment methods (e.g., Yen's Q and so on) based on IRT. Most of them are a kind of fit index based on the difference between observed values and expected values. Hattie cited a couple of comments as the rationale for these fit index approaches:

> If a given set of items fit the model, this is the evidence that they refer to as unidimensional ability (Wright & Panchapakesan, 1969).

However, Hattie pointed out that they appear to lack theoretical bases and little is known of their sampling distribution. Another problem he mentioned is that with large samples a chi-square test is almost certainly significant. The fact that all the assumptions of a model are satisfied might imply a good fit of a model to the data. However, the converse might not be true. A good fit does not necessarily guarantee that the unidimensional assumption is met. A good fit is necessary but not a sufficient condition for unidimensionality.

In addition to the brief review of unidimensionality assessment methods above, one of the research studies should be mentioned on the robustness of IRT to the violation of unidimentionality assumption. Reckase (1979) performed an interesting study regarding this issue. He suggested that the unidimensional IRT model could be used if the dominant factor in FA explains at least 20% of variability among items.

In summary, local independence is the statement about the existence of a latent trait that can explain the dependency among items. In other words, it could be considered as the definition of latent trait. Unidimensionality is the assertion that there is only one underlying latent trait and is a special case of local independence. Unidimensionality can be judged from various methods based on reliability, factor analysis, and IRT. The criteria for the judgment are high level of internal consistency, existence of dominant factor, and good fit, respectively. Hattie pointed out that dimensionality is the joint property of an item set and a particular

sample of examinees. The same set of items could be unidimensional for one group but not for another group. He also pointed out that a belief or point of view of the investigator is required when a researcher argues that items measure the same thing.

2.3 Model Selection Method

Likelihood Ratio Test. The notations and basis of discussion in this section come from Kuha's (2004) work. In statistics, the most common way for model selection is a likelihood ratio test (LRT). In the LRT, a statistical test is done to compare the fit of two nested models. The term nested model means that one model (reduced model) is a special case of the other (full model). Let's denote D as an observed sample data of size n. M_k represents a number of possible models for D with each model having a likelihood function $p(D|\theta_k; M_k)$. The likelihood function is fully specified by parameter vector θ_k with p_k number of parameters. Usually the log is taken to the likelihood function to produce the log-likelihood $l(\theta_k) = \log[p(D|\theta_k; M_k)]$. Given those notations, LRT can be summarized by the following equation:

$$G^2 = 2[l(\hat{\theta}_2) - l(\hat{\theta}_1)], \tag{2.15}$$

where M_1 is nested in M_2, G^2 is asymptotically distributed as χ^2 with $p_2 - p_1$ degree of freedom when M_1 is the true model. However, there are several limitations of the LRT. First, as with all other significant tests, a point null hypothesis test will be almost always significant when the sample size is large. Second, the LRT only works for two nested models. Third, the LRT always favors the complex model in the case of nested models.

Information Criteria. One alternative is to choose a model based on information-based statistics such as Akaike's (1974) information criterion (AIC) or Schwarz's (1978) Bayesian information criterion (BIC). The general form of information-based statistics is the following:

$$2[l(\hat{\theta}_2) - l(\hat{\theta}_1)] - a(p_2 - p_1), \tag{2.16}$$

where a is some known positive quantity. This information based statistics are also known as penalized model selection criteria since their form could be decomposed into two parts: The

first term reflects the difference between the fit of each of two models, which tends to favor the more complex model. The second term penalizes the complex model when we consider $p_2 - p_1$ as the increased complexity of M_2 over M_1. In other words, two terms in the penalized model selection criteria reflect the trading off between fit and the parsimonious model.There are a couple of advantages of the penalized criteria over the LRT. First, it can be used for a non-nested model. Second, it considers the penalty for the complex model, which makes a balance between fit and parsimoniousity. As a special case of this penalized criteria, AIC and BIC are defined by the following equation:

$$\text{AIC} = 2[l(\hat{\theta}_2) - l(\hat{\theta}_1)] - 2(p_2 - p_1) \tag{2.17}$$

$$\text{BIC} = 2[l(\hat{\theta}_2) - l(\hat{\theta}_1)] - \log(n)(p_2 - p_1) \tag{2.18}$$

AIC. The underlying idea behind AIC is to use some measure of distance between densities to compare models. Suppose that $f(y)$ is the true density function which generates the observed data D. A distance between true density $f(y)$ and the model density $p(D|\theta_k; M_k)$ can be defined in some way. Usually, the Kullback-Leibler (KL) distance is used for AIC, which is defined as

$$I[f(y), p(\theta_k)] = \int f(y) \log \left[\frac{f(y)}{p(y|\theta_k, M_k)} \right] dy, \tag{2.19}$$

where y is a random variable of the same size as D from the true density f. Given the distance between two densities, we can consider the following quantity T_A which reflects the distance between the densities of the two models:

$$T_A = 2E_x \left[I[f, p_1(\hat{\theta}_1^x)] - I[f, p_2(\hat{\theta}_2^x)] \right]. \tag{2.20}$$

Note that the definition of T_A involves two hypothetical data sets: x is used to estimate the parameters θ_k and y is used to judge the fit of the model. When T_A is positive, this means that the expected KL distance between M_2 and the true model f is smaller than the one between M_1 and f. The issue in calculating T_A is that it involves the true model f for D, which is unknown. The AIC_e, which is the more general version of AIC than what we usually

use, is a good approximation of T_A under certain conditions:

$$T_A \approx \text{AIC}_e = 2[l(\hat{\theta}_2) - l(\hat{\theta}_1)] - (1 + n/n_0)(p_2 - p_1), \tag{2.21}$$

where n_0 denotes the size of the hypothetical training sample x. AIC_e will be reduced to the usual AIC index when $n_0 = n$.

BIC. The basic idea of BIC is to compare the probabilities that each of the models is the true model. Let's consider the Bayesian theorem:

$$p(M_k|D) \propto p(D|M_k)p(M_k), \tag{2.22}$$

where $p(M_k|D)$, $p(D|M_k)$, and $p(M_k)$ represent posterior, likelihood, and prior, respectively. A comparison between the probabilities that each of models is the true model can be done based on the following ratio:

$$\frac{p(M_2|D)}{p(M_1|D)} = \frac{p(D|M_2)}{p(D|M_1)} \frac{p(M_2)}{p(M_1)}, \tag{2.23}$$

where $p(M_2|D)/p(M_1|D)$ is posterior odds, $p(M_2)/p(M_1)$ is prior odds, and $p(D|M_2)/p(D|M_1)$ is a Bayes factor (BF_{21}). Since the prior odds are updated to posterior odds by the ratio BF_{21}, the Bayes factor can be considered as the measure of the evidence based on the data in favor of M_2 over M_1. Usually, Bayes factor BF_{21} is transformed to $T_B = 2\log(\text{BF}_{21})$ in order to be on the same scale as the deviance statistics. If T_B is positive, this means that M_2 is more likely to be the true model given the data and the prior distribution $p(\theta_k|M_k)$. Usually, T_B should be at least 2 to be considered as a strong evidence. In principle, T_B can be used to compare any two models which are assumed to generate the data D. The definition of T_B does not require the two models to be nested. The real issue in calculating T_B is that there is an infinite number of prior distributions for any two models. Therefore, there could also be an infinite number of Bayes factors BF_{21} or T_B. In fact, the BIC index is the approximation of T_B. The equation below is a more general version of the BIC index than Equation 2.18:

$$T_B \approx \text{BIC}_e = 2[l(\hat{\theta}_2) - l(\hat{\theta}_1)] - \log(1 + n/n_0)(p_2 - p_1). \tag{2.24}$$

This BIC index is much simpler than T_B since it involves only the results from the standard maximum likelihood estimation and does not require any explicit specification of the prior distribution. This more general version of BIC will be reduced to the equation when $n_0 = 1$ and n is large. Thus, the BIC equation we usually use has a couple of underlying assumptions in order to be a good approximation of the original motivation of the BIC index, which is T_B. BIC_e itself is an approximation of T_B, which is derived under the the assumptions for the specific sets of prior distributions:

$$p(\theta_k|M_k) = N(\hat{\theta}_k, (n/n_0)(-[\partial^2 l(\hat{\theta}_k)/\partial\theta_k\partial\theta_k']^{-1}), \tag{2.25}$$

where $-[\partial^2 l(\hat{\theta}_k)/\partial\theta_k\partial\theta_k']^{-1}$ is the estimated variance matrix of the MLE $\hat{\theta}_k$, and n_0 is called the prior sample size which reflects how informative the prior is compared to the information provided by D. The usual BIC index equation, Equation 2.18, is the special case of Equation 2.24 when n_0 is set to 1, which is called the unit information prior. The use of unit information prior or setting n_0=1 can be considered from two aspects: First, from the point of view of the parameter estimation, unit information prior makes Bayesian estimates become very similar to the MLEs as n increases. The reason is that, as n increases, the variance of prior becomes large, which makes the prior uninformative. Thus, posterior density is dominated by the information from the data, which makes Bayesian estimates very similar to MLEs. Second, from the point of view of model selection, it turns out that setting n_0 to be fixed may penalize large models too heavily and make the Bayes factor prefer a smaller model (Spiegelhalter, Best, Carlin, & Linde, 2002)

Both AIC and BIC are penalized criteria based on the equation with a different form of a coefficient. However, the motivations of AIC and BIC are quite different. AIC is developed to identify the model which produces the closest, on average, density to the true density. On the other hand, BIC is designed to pick up the model that is most probable after conditioning on the data.

2.4 Missing Data and MLE

The Types of Missing Data. Rubin (1976) defined three types of missing data: missing completely at random (MCAR), missing at random (MAR), and missing not at random (MNAR). Shaffer and Graham (2002) summarized well on those types of missing data: Let $Y_{com} = [Y_{obs}, Y_{mis}] = (y_{ji})$ denote a complete data matrix which could be partitioned into an observed part Y_{obs} and a missing part Y_{mis}. Then the matrix with the same size as Y_{com} can be defined with element 1 or 0 to indicate whether corresponding elements are observed or missing respectively. Also, the indicator matrix $M = (m_{ij})$ can be defined for any data set, where $m_{ij} = 1$ if y_{ij} is missing and $m_{ij} = 0$ if y_{ij} is present. We refer to M as the missingness of the data. In Rubin's framework, missingness is treated using probabilistic models, in which M is a set of random variables having a joint distribution. Rubin describes the distribution of missingness (DOM) as the missing data mechanism. Schafer and Graham (2002) notes that it may not be necessary to specify a particular distribution of M since we are not likely to be interested in modeling the DOM. In fact, we may not have enough information to do so. Rubin's (1976) original purpose of introducing the DOM was to clarify the conditions under which it may be ignored. The missing data mechanism is defined by the conditional distribution of M given Y, which could be denoted by $f(M|Y, \phi)$, where ϕ represents unknown parameters. Using the terminologies above, the missing data are called MCAR if the following equation holds for all Y and ϕ:

$$f(M|Y, \phi) = f(M|\phi). \tag{2.26}$$

In other words, the probability of missing data does not depend on other measured variables or also the values of Y itself. In MCAR, our data can be considered as a relatively good representative of the complete data set. Also, the missing data is called MAR if the following equation holds for all Y_{mis} and ϕ:

$$f(M|Y, \phi) = f(M|Y_{obs}, \phi). \tag{2.27}$$

This implies that the probability of missingness may depend on the observed data Y_{obs} but not on missing data Y_{mis}. In other words, no relationship is assumed between the probability for missing data and the values of Y after controlling out other variables. In MAR, a systematic relationship exists between one or more measured variables and probability of missing. On the other hand, it should be noted that there is no way to test MAR because nothing can be verified about the presence or absence of the relationship without knowing the values of missing data. When the DOM depends on Y_{mis}, missing data are said to be MNAR. In MNAR, the probability of missing is related to the values of Y itself. Like MAR, nothing can be verified as MNAR without knowing the values of the missing variables (Enders, 2010).

MLE in the Presence of Missing Data. The discussion in this section is based on the book written by Little and Rubin (1987). Suppose that Y denotes the data and the data are assumed to be generated by a model described by a probability density function $f(Y|\theta)$. Here, θ represents a scalar or vector parameter, which is known to lie in a parameter space Ω_θ. The likelihood function $L(\theta|Y)$ is defined by $f(Y|\theta)$, that is, $L(\theta|Y) = f(Y|\theta)$. The log likelihood $l(\theta|Y)$ is defined by taking the natural logarithm to $L(\theta|Y)$. A maximum likelihood estimate (MLE) of θ is a value of $\theta \in \Omega_\theta$ that maximizes the likelihood $L(\theta|Y)$ or $l(\theta|Y)$. θ could be found by solving the following equation:

$$D_l(\theta) = \partial l(\theta|Y)/\partial\theta = 0, \tag{2.28}$$

where $D_l(\theta)$ is called the score function. MLE is known to have good asymptotic properties, which are consistency, normality, and efficiency.

Let $Y = (Y_{obs}, Y_{mis})$ denote the complete data set which can be decomposed into Y_{obs} and Y_{mis} and let $f(Y|\theta) = f(Y_{obs}, Y_{mis}|\theta)$ denote the probability density of joint distribution Y_{obs} and Y_{mis}. In order to get inference about θ in the presence of missing data, the new likelihood function can be defined as followings:

$$L_{ign}(\theta|Y_{obs}) = f(Y_{obs}|\theta) = \int f(Y_{obs}, Y_{mis}|\theta)dY_{mis}. \tag{2.29}$$

This likelihood represents the case when inference about θ is made by ignoring the missing data. On the other hand, in the full model, M is treated as a random variable and included in the joint distribution and M and Y. In order to represent this case, another likelihood function can be defined as the following equation:

$$L_{full}(\theta, \phi | Y_{obs}, M) = f(Y_{obs}, M | \theta, \phi) = \int f(Y_{obs}, Y_{mis}) | \theta) f(M | Y_{obs}, Y_{mis}, \phi) dY_{mis}, \quad (2.30)$$

where θ and ϕ represent the parameters of distribution of Y and missingness respectively. Here, what we want to find by defining these two likelihoods is the condition where the inferences about θ from L_{ign} and L_{full} are the same. Given the following definition of MAR,

$$f(M | Y_{obs}, Y_{mis}, \phi) = f(M | Y_{obs}, \phi) \text{ for all } Y_{mis} \text{ and } \phi, \quad (2.31)$$

we can derive the following identity,

$$\begin{aligned} L_{full}(\theta, \phi | Y_{obs}, M) &= f(Y_{obs}, M | \theta, \phi) \\ &= \int f(Y_{obs}, Y_{mis} | \theta) f(M | Y_{obs}, Y_{mis}, \phi) dY_{mis} \\ &= f(M | Y_{obs}, \phi) \times \int f(Y_{obs}, Y_{mis} | \theta) dY_{mis} \\ &= f(M | Y_{obs}, \phi) f(Y_{obs} | \theta) \\ &= f(M | Y_{obs}, \phi) L_{ign}(\theta | Y_{obs}). \end{aligned} \quad (2.32)$$

Since the two likelihoods L_{full} and L_{ign} are proportional to each other, the inferences about θ based on those two likelihoods should be the same. In other words, the MAR missing data mechanism is ignorable for the inference based on the maximum likelihood estimation. If the missing data mechanism is not MAR, then MLE needs a model for the missing data mechanism and maximization of the full likelihood function.

Chapter 3

SIMULATION AND EMPIRICAL STUDY

As described in the previous chapters, the primary purpose of using mixture models is to identify the heterogeneity of a population of interest. The usual practice for exploratory using mixture IRT models is to fit models with a different number of latent classes to data and to choose a model showing the best fit based on a certain fit index such as AIC or BIC. Although mixture models have been useful in many applications, there have been concerns on how accurately mixture models can identify the underlying heterogeneity of a population. Based on those concerns, the focus of this study is to investigate the impact of missing data on dichotomous mixture IRT models. In this chapter, the simulation study described below was conducted under 192 conditions to evaluate the performance of the dichotomous mixture IRT models in the presence of missing data. Also, an empirical data analysis was illustrated after the simulation study.

3.1 Design of Simulation Study

In this simulation study, the performance of the dichotomous mixture IRT models was evaluated based on two criteria:

1. the accuracy of item parameter recovery
2. the rate of correct identification of the number of classes.

The conditions for this simulation study include two dichotomous mixture IRT models for data generation (the mixture Rasch model and 2PL), two different numbers of classes in

data generation (1 class and 2 classes), two dichotomous mixture IRT models for data fitting (the mixture Rasch model and 2PL), two test lengths (10 and 30 items), two sample sizes (600 and 1200 examinees), three missing mechanisms (complete data, MAR, and MNAR) and two missing percentages (10% and 20%). One hundred replications were simulated for each of the 192 conditions (2 IRT models for data generation $\times$ 2 classes $\times$ 2 IRT models for fitting $\times$ 2 test lengths $\times$ 2 sample sizes $\times$ 3 missing conditions $\times$ 2 missing percentages = 192 conditions).

The Accuracy of Item Parameter Recovery. A recovery analysis on the item parameters was done to evaluate the performance of the dichotomous mixture IRT models in the presence of missing data. The bias and root mean square error (RMSE) were used to evaluate the quality of an estimator. By definition, an estimator $\hat{\theta}$ is a function of observations $x^{(1)}, \cdots, x^{(n)}$. Since an estimator depends on the observation or sample, it is a random variable. A particular realization of estimator is called estimate. The bias of estimator $\hat{\theta}$ for the true value of parameter θ is defined by the following equation:

$$\text{Bias}(\hat{\theta}) = E[\hat{\theta}] - \theta. \tag{3.1}$$

If $\text{Bias}(\hat{\theta}) = 0$ then the estimator $\hat{\theta}$ is said to be unbiased. $\text{Bias}(\hat{\theta}) = 0$ is the difference between the true value and the average of all possible estimates. In other words, if there are repeated sampling of n samples $x^{(1)}, \cdots, x^{(n)}$ then unbiased estimator $\hat{\theta}(x^{(1)}, \cdots, x^{(n)})$ will give the correct value of θ on the average. On the other hand, RMSE is defined by the following equation:

$$\text{RMSE}(\hat{\theta}) = \sqrt{E[(\hat{\theta} - \theta)^2]}. \tag{3.2}$$

RMSE captures the expected value of the squared error. An estimator is said to be consistent if the RMSE is reduced as we have more data. In comparing two estimators, one estimator with smaller RMSE is said to be more efficient than the other. In this study, the bias and root mean square error (RMSE) for the item difficulty parameter were calculated across items,

classes, and replications by the following equations:

$$\text{Bias}(\hat{\beta}) = \frac{\sum_{r=1}^{R}\sum_{i=1}^{I}\sum_{g=1}^{C}(\hat{\beta}_{igr} - \beta_{ig})}{RIC} \tag{3.3}$$

$$\text{RMSE}(\hat{\beta}) = \sqrt{\frac{\sum_{r=1}^{R}\sum_{i=1}^{I}\sum_{g=1}^{C}(\hat{\beta}_{igr} - \beta_{ig})^2}{RIC}}, \tag{3.4}$$

where $\hat{\beta}_{igr}$ is the estimated item difficulty parameter for item i of class g in replication r, $\beta_{ig}$is the true value of item parameter for item i in class g, R is the number of replications, I is the number of items, and C is the number of classes.

To make the item difficulty parameters across replications comparable before calculating the RMSE, it is necessary to put those parameters on a common metric. A simple mean equating method was used to put item difficulty parameters from each replication on the metric of true item difficulty parameters using the following equation (Kolen & Brennan, 2004):

$$b_j^* = b_j - (\bar{b}_j - \bar{b}_p), \tag{3.5}$$

where b_j^* denotes the equated or transformed item difficulty parameter on the metric of true item parameters, $\bar{b}_j$ denotes the mean of item difficulty parameters of each replication, and $\bar{b}_p$ represents the mean of true item difficulty parameters.

The Rate of Correct Identification for the Number of Classes. In this study, 100 replications per each condition were simulated. Given the predetermined true number of classes in the data generation stage, the rate r of correct identification was calculated using the following formula:

$$r = \frac{c}{T}, \tag{3.6}$$

where c denotes the number of replications which gave the correct number of classes based a certain fit index and T represents the total number of replications. Since only AIC and BIC were used to determine the number of classes, r was calculated for both AIC and BIC.

Test Length and Sample Size. Test length and sample size conditions were borrowed from Li et al. (2009). Their test lengths were 6, 15, and 30. In this study, the test length

conditions were 10 and 30 in order to use the 10 item difficulty parameters in Rost (1990). For sample size, 600 and 1200 examinees were used as in Li et al.

Item Difficulty Parameters. In this study, the Rasch model and 2PL were used to generate the simulation data. Each model was generated to have one class or two classes. The mechanism for generating two different classes was the same as used in Rost (1990). As in Rost, the item profile plots of two classes, which represent the values of item difficulty parameters across items, were assumed to be crossed to indicate that the examinees in one class did better in some portion of items but did worse in another portion of items than the other class. In this setting, the difference in item profile plots of two classes was assumed to be the underlying mechanism for generating different classes. For the Rasch model with 10 items, the item difficulty parameters of each class were the ones used in Rost used, as presented in Table 3.1. For 30 items, the same range of item difficulty parameters was divided into 30 equally spaced numbers. For the 2PL, the item difficulty parameters were assumed to have the same mechanism as the Rasch model. For the discrimination parameters, the value of 1 was assigned to difficult items and the value of 2 was assigned to easy items as in Li et al. The item parameters for 2PL are presented in Table 3.2.

Table 3.1: True Item Difficulty Parameters for the Mixture Rasch Model

Rasch Model			
	1 class	2 classes	
Item	b	b	b
1	-2.70	-2.70	2.60
2	-2.10	-2.20	2.10
3	-1.60	-1.50	1.50
4	-0.90	-0.90	0.90
5	-0.30	-0.30	0.30
6	0.30	0.30	0.30
7	1.00	1.00	-1.00
8	1.50	1.50	-1.40
9	2.10	2.10	-2.00
10	2.70	2.70	-2.80

Table 3.2: True Item Parameters for the 2PL

2PL						
	1 class		2 classes			
Item	a	b	a	b	a	b
1	2	-2.70	2	-2.70	1	2.60
2	2	-2.10	2	-2.20	1	2.10
3	2	-1.60	2	-1.50	1	1.50
4	2	-0.90	2	-0.90	1	0.90
5	2	-0.30	2	-0.30	1	0.30
6	1	0.30	1	0.30	2	0.30
7	1	1.00	1	1.00	2	-1.00
8	1	1.50	1	1.50	2	-1.40
9	1	2.10	1	2.10	2	-2.00
10	1	2.70	1	2.70	2	-2.80

Spurious Classes from Fitting the Wrong Model. In this study, the simulation data were generated based on the Rasch model and 2PL as described in the previous section. A previous study (Alexeev, Templin, & Cohen, 2011) found that applying the wrong model could produce spurious classes in mixture models. In other words, if the Mixture Rasch model is used to fit the data which was produced by higher IRT models such as 2PL, then it is very likely to produce additional spurious classes. Since this study could also be considered as the one to see the effect of missing data on the spurious classes in the mixture model, the possibility of producing spurious classes due to mismatching models for data generation and fitting was considered as a simulation condition. The data generated by the Rasch model were fitted using the mixture Rasch model and 2PL. Also the data generated by the 2PL were fitted using the mixture Rasch model and 2PL.

Missing Data Condition. Given the complete data set for a specific simulation condition, the MAR and MNAR missing observations were generated following Finch (2008). In Finch's work, it was assumed that the type of missing data mechanism is MAR if the probabilities of missing responses in the target items were inversely proportional to the sum of the number of correct items, other than the target items. The missing observations generated in this way could be considered as MAR since the probability of missing value is related to another observed variable, which is the sum of the number of correct items other than the target items. The following are the more specific procedures for MAR missing data generation. First, the simulees were grouped into four fractiles based on total scores for the items except target items. By target items, they mean some portion of items which can have missing responses. For example, for the complete data set with 10 items, the three most difficult items were chosen as target items. The total score was calculated based on the remaining seven items. Based on the total score other than target items, the simulees were divided into four fractiles (0–1, 2–3, 4–5, 6–7). The four different values of probabilities of missing responses on the target items, with lower scores having a higher probability of a missing value, were assigned to the four fractiles. The average of these probabilities across

fractiles was designed to be equal to the total percentages of missing responses, which are 10% and 20%. In this study, the probabilities of missing responses across fractiles were set to .3, .2, .1, and 0 for 20% total missing response condition and .15, .1, .05, and 0 for 10% missing response condition. The total number of missing responses were exactly controlled to be equal to the desired number by counting missing responses. By generating missing responses in this way, simulees with high abilities based on the total scores for items other than target items were less likely to have a missing response than those with lower abilities and vice versa.

It was also assumed that the type of missing data mechanism is MNAR if the probability of having a missing response on the target items was directly related to whether the simulees gave a correct answer or not on the target items. Given the complete data set, a lower probability of missing responses was assigned to the simulees who got the item correct and a higher probability was assigned to the ones who got the item incorrect. The average of these higher and lower probabilities was set to be equal to the total percentages of missing responses for the simulation study, which are 10% and 20%. For the 20% of total missing responses, 10% of the probability of missing was assigned to the simulees who got the item correct and 30% of the probability was assigned to the ones who got the item incorrect. For the 10% of total missing, 5% and 15% were used. The total percentages of missing responses were also exactly controlled by counting missing responses. The R code for generating MAR and MNAR missing responses are presented in Appendix B.

In this study, the total percentages of missing responses were set to 10% and 20% to be in a similar range with the previous research. De Ayala et al. (2001) used 5.1%, 10.3%, 15.4%, and 20.5% as missing percentages for the simulation study. Enders (2004) imposed 11% of missing data and Finch (2008) used 5%, 15%, and 30% as missing percentage conditions.

Software for Data Simulation. The M*plus* Version 6.11 (Muthén & Muthén, 1998-2011) was used to fit models in this simulation study. M*plus* was set to use a maximum likelihood estimation with robust standard errors by specifying ALGORITHM=INTEGRATION.

The practical issue of this simulation study was how to deal with large numbers of input and output files. 192 conditions with 100 replications required 19200 input and output files respectively. In order to find the best model based on the fit index, models having three different numbers of classes had to be fitted, which made the total number of input and output files 57600. It is almost impossible to handle this amount of input and output files by hand. Therefore, the M*plus* automation package in the R software was used to create and run input files automatically. The R codes for the mixture Rasch model and 2PL were presented in Appendix C. Linux shell script using AWK utility was used in order to extract item difficulty parameters, AIC, and BIC from output files.

3.2 Result

The results of this simulation study are presented in three parts. First, the RMSE and Bias for item difficulty parameters for 192 conditions are presented and analyzed using the general linear model. Second, the number of classes identified by the mixture IRT models for the simulation conditions are also presented and analyzed using the generalized linear model. The notation for the simulation conditions was designed to represent the combination of factors for a given condition. For example, R211060010 represents the following conditions: the Rasch model was used to fit the data (R), 2PL was used to generate data (2), simulated data contain one class (1), the number of items was ten (10), the number of examinees was six hundred (600), and the total 10% of the missing percentage was imposed.

3.2.1 The Impact of Missing Data on Bias and RMSE

The Table 3.3 below summarizes the results for the Bias and RMSE across various conditions. 100 replications were simulated for each condition. The Bias and RMSE was calculated for the item difficulty parameters across items, classes, and replications for each condition using Equation 3.3 and Equation 3.4, respectively.

Table 3.3: Bias and RMSE for Simulation Conditions

		Complete		MAR		MNAR	
	Condition	Bias	RMSE	Bias	RMSE	Bias	RMSE
1	R111060010	0.001	0.098	0.002	0.102	0.180	0.257
2	R111060020	0.012	0.096	0.016	0.193	0.444	0.601
3	R113060010	-0.000	0.053	0.000	0.054	0.054	0.131
4	R113060020	-0.001	0.054	-0.001	0.058	0.132	0.311
5	R121060010	0.003	0.246	0.025	0.415	0.140	0.316
6	R121060020	-0.002	0.163	-0.004	0.335	0.314	0.553
7	R123060010	0.008	0.174	0.005	0.139	0.053	0.176
8	R123060020	-0.000	0.083	-0.003	0.095	0.110	0.285
9	R211060010	-0.249	0.401	-0.257	0.403	-0.069	0.538
10	R211060020	-0.259	0.419	-0.279	0.424	-0.069	0.538
11	R213060010	-0.075	0.241	-0.074	0.246	-0.019	0.330
12	R213060020	-0.075	0.242	-0.074	0.254	0.061	0.544
13	R221060010	-0.567	1.156	-0.542	1.162	-0.437	1.253
14	R221060020	-0.583	1.285	-0.566	1.234	-0.289	1.488
15	R223060010	0.254	0.758	0.253	0.759	0.307	0.847
16	R223060020	0.252	0.749	0.246	0.749	0.380	0.994
17	R1110120010	0.005	0.069	0.004	0.072	0.178	0.237
18	R1110120020	0.001	0.067	0.002	0.075	0.437	0.60
19	R1130120010	0.000	0.037	0.000	0.039	0.056	0.131
20	R1130120020	0.000	0.126	0.000	0.041	0.136	0.320
21	R1210120010	0.015	0.221	0.005	0.227	0.152	0.352
22	R1210120020	0.004	0.220	0.018	0.450	0.308	0.558
23	R1230120010	0.004	0.126	0.004	0.124	0.052	0.164
24	R1230120020	0.002	0.123	-0.002	0.066	0.110	0.276
25	R2110120010	-0.239	0.422	-0.248	0.393	-0.062	0.519
26	R2110120020	-0.233	0.383	-0.246	0.386	0.211	0.930

27	R2130120010	-0.073	0.236	-0.072	0.241	-0.016	0.322
28	R2130120020	-0.071	0.235	-0.069	0.246	0.066	0.540
29	R2210120010	-0.475	0.921	-0.475	0.916	-0.375	1.005
30	R2210120020	-0.442	0.939	-0.441	1.018	-0.211	1.106
31	R2230120010	0.258	0.760	0.258	0.765	0.312	0.851
32	R2230120020	0.263	0.769	0.256	0.759	0.389	1.006
33	T111060010	-0.001	0.121	0.000	0.127	0.175	0.267
34	T111060020	0.012	0.118	0.021	0.145	0.449	0.614
35	T113060010	0.000	0.060	0.000	0.064	0.054	0.137
36	T113060020	0.000	0.061	-0.001	0.070	0.133	0.313
37	T121060010	-0.022	0.24	-0.021	0.457	0.143	0.509
38	T121060020	-0.012	0.480	-0.043	0.897	0.311	0.822
39	T123060010	0.000	0.099	0.000	0.186	0.050	0.156
40	T123060020	0.000	0.148	-0.003	0.120	0.116	0.324
41	T211060010	-0.499	0.766	-0.500	0.785	-0.321	0.852
42	T211060020	-0.496	0.772	-0.515	0.796	-0.321	0.852
43	T213060010	-0.152	0.404	-0.154	0.407	-0.096	0.445
44	T213060020	-0.152	0.409	-0.158	0.417	-0.018	0.609
45	T221060010	-1.745	10.56	-1.773	6.663	-1.419	5.707
46	T221060020	-1.617	5.864	-2.178	10.291	-1.811	8.187
47	T223060010	0.178	0.749	0.178	0.756	0.233	0.838
48	T223060020	0.177	0.746	0.166	0.738	0.310	1.000
49	T1110120010	0.005	0.181	0.007	0.089	0.186	0.239
50	T1110120020	0.004	0.087	0.005	0.100	0.435	0.596
51	T1130120010	0.000	0.041	0.001	0.045	0.057	0.135
52	T1130120020	0.000	0.04	0.000	0.049	0.136	0.318
53	T1210120010	0.002	0.147	-0.001	0.497	0.132	0.500
54	T1210120020	-0.005	0.394	-0.014	0.770	0.333	0.598
55	T1230120010	0.002	0.128	0.000	0.131	0.052	0.172
56	T1230120020	-0.001	0.070	-0.003	0.084	0.111	0.268

57	T2110120010	-0.481	0.745	-0.485	0.746	-0.303	0.814
58	T2110120020	-0.477	0.747	-0.484	0.756	-0.040	1.128
59	T2130120010	-0.150	0.402	-0.151	0.403	-0.094	0.438
60	T2130120020	-0.149	0.402	-0.152	0.406	-0.014	0.588
61	T2210120010	-1.086	2.421	-1.175	2.754	-0.957	2.415
62	T2210120020	-1.028	2.245	-1.371	5.660	-0.715	2.350
63	T2230120010	0.183	0.746	0.188	0.764	0.240	0.837
64	T2230120020	0.189	0.746	0.182	0.734	0.318	0.980

Minimum Sample Size for the Mixture IRT Models. Before discussing the main focus of this simulation study, which is missing data, it is necessary to briefly mention the minimum sample size for the mixture IRT models. In Table 3.3, the values of Bias and RMSE for conditions 45-46 and 61-62 were unusually large compared to other values. These unusual large values indicate that the estimations of mixture IRT models failed in those simulation conditions. In fact, those conditions can be said to be the most severe ones in the sense that the most complex models were trying to be estimated given the least amount of information. In both ranges of conditions, the mixture 2PL was fitted to the data which was generated by 2PL with two classes, whereas there were only 10 items. Thus, those values of RMSE for conditions 45-46 and 61-62 were excluded from the analysis of this simulation study.

General Linear Model Results for RMSE. The general linear model procedure in the SPSS software, version 17 (SPSS, Chicago, IL), was used to summarize the results in Table 3.3. In the general linear model, RMSE in Table 3.3 was set to be a dependent variable and seven factors (i.e., model, data, samples, items, percentage, type, and class) were set to be independent variables. The model indicates the type of the mixture IRT model used to fit the data, the data represents the type of the mixture IRT model used to generate data, class represents the number of classes in the generated data, percentage represents the total percentage of missing responses in data, and type represents the type of missing data. The results of the general linear model for Table 3.3 are presented in Table 3.4. Overall, the

results in Table 3.4 coincide with the expected ones. The more items the data had, the more accurate the estimation was. The MNAR missing data increased RMSE compared

Statistical Significance and Effect Size. It is well known that the p value for judging statistical significance is confounded with sample size so that it is not a useful index for the study effect. Thus, the 6th edition of the Publication Manual of the American Psychological Association recommended reporting effect size along with the p value of the statistical significant test. In SPSS, partial η^2 is the default effect size measure for several ANOVA procedures. In the presence of multiple predictors, partial η^2 measures the proportion of the total variance explained by a given predictor after excluding variance explained by other predictors. However, the values of partial η^2 are not additive to one. They are not additive to one since the partial η^2 value is calculated by partialing out other predictors, which makes the interpretation of partial eta square value difficult. Therefore, to make a comparative statement for effect sizes, η^2 was calculated by using the following equation based on the sum of squared values from SPSS:

$$\eta^2 = \frac{SS_{\text{between}}}{SS_{\text{total}}} \tag{3.7}$$

According to Cohen (1988), .001, .059, and .138 of η^2 values correspond to small, medium, and large effect respectively. In this analysis, the effect size measures were considered in interpreting the results in Table 3.4.

The Effect of Complexity in Modeling on RMSE. In Table 3.4, the model, data, and class factors can be considered as the ones that represent the complexity in modeling. The model factor ($\beta = 0.070$, $p < .01$, small effect size based on $\eta^2 = .010$) represents the type of mixture IRT models which were used to fit the data, the data factor ($\beta = 0.496$, $p < .01$, large effect size based on $\eta^2 = .494$) represents the type of mixture IRT models which were used to generated the data, and class factor ($\beta = 0.279$, $p < .01$, large effect size based on $\eta^2 = .156$) represents the number of classes in the data. It can be said that the model factor reflects the complexity in the models and the data and class factors reflect the complexity in the data themselves. All three factors were statistically significant. Also, the

Table 3.4: General Linear Model Results for RMSE

Parameter	β	Std.Error	*df*	Sig	η_p^2	η^2	Power	*M*	*SD*
(Intercept)	0.066	0.0310	1	.03	.822		1.00		
[Model=2PL]	0.070	0.0212	1	$< .01$	.057	.010	.34	0.442	0.032
[Model=1PL]	0							0.440	0.036
[Data=2PL]	0.496	0.0212	1	$< .01$	.753	.494	1.00	0.683	0.031
[Data=1PL]	0							0.228	0.019
[Samples=1200]	-0.027	0.0210	1	.19	.009	.001	.15	0.427	0.033
[Samples=600]	0							0.454	0.036
[Items=30]	-0.228	0.212	1	$< .01$	.391	.104	1.00	0.363	0.029
[Items=10]	0							0.530	0.038
[Percentage=20]	0.081	0.0210	1	$< .01$	.076	.013	.76	0.481	0.036
[Percentage=10]	0							0.400	0.032
[Type=mnar]	0.192	0.0257	1	$< .01$	.237	.056	1.00	0.557	0.041
[Type=mar]	0.034	0.0257	1	.18	.009		.15	0.400	0.042
[Type=comp]	0							0.365	0.041
[Class=2]	0.279	0.0212	1	$< .01$	.490	.156	1.00	0.569	0.040
[Class=1]	0							0.337	0.026

Note. M = mean, SD = standard deviation.

coefficients of three factors were all positive, which means that the complexity in a model introduces more RMSEs into the parameter estimation of mixture IRT models. Based on the effect size measure by η^2, the complexity in the data introduced more RMSEs than the one of the fitting model. Overall, complexity in modeling introduced more RMSEs into the parameter estimation of mixture IRT models.

The Effect of the Amount of Information in Data on RMSE. The sample and item factors in Table 3.4 can be considered as ones that represent the amount of information in the data. Only the coefficient of items was statistically significant ($\beta = -0.228$, $p < .01$, medium effect size based on $\eta^2 = .104$). The negative sign of the coefficient indicates that the values of RMSE decreased as the number of items increased, which coincides with the common knowledge in statistics that more information ensures more accurate estimation. However, increasing sample size from 600 to 1200 did not affect the values of RMSE in the sense that the coefficient of samples was not statistically significant.

The Effect of Missing Data on RMSE. The percentage and type factor in Table 3.4 represent the total percentages and types of missing responses respectively. The percentage factor ($\beta = 0.081$, $p < .01$, small effect size based on $\eta^2 = .013$) was statistically significant with a positive coefficient, which indicates that increasing missing percentage introduces more RMSE. This seems to be natural in that increasing the percentage of missing responses corresponds to decreasing the amount of information in the data. The type factor has three levels, comp representing complete data, mar representing MAR missing data, and mnar representing MNAR missing data. The comp level was set as a reference level. The RMSE values of MNAR missing data were statistically significantly higher than those of the complete data ($\beta = 0.192$, $p < .01$, medium effect size based on $\eta^2 = .056$). However, there was no statistically significant difference in the values of RMSE between MAR and complete data. This was also an expected result since the inferences of MLE based on the likelihood function with data without any missing and data with MAR missing should be the same as indicated in Equation 2.32 (Little & Rubin, 1987). The fact that the coefficient for MNAR missing was

statistically significant indicates that the inference of MLE based on the likelihood function with MNAR missing data introduces bias in the estimation.

The Effect of Missing Data on Bias. Based on Table 3.3, the effect of seven factors (i.e., model, data, samples, items, percentage, type, and class) on the bias of difficulty parameters were also investigated using the general linear model. The main effect of type of missing yielded an F ratio of $F(2, 111) = 13.822$, $p < .01$. Combined with the post-hoc analysis using Tukey's HSD, this indicate that the bias was significantly greater for MNAR ($M = 0.127$, $SD = 0.027$) than for complete data ($M = -0.041$, $SD = 0.024$). In other words, the mixture IRT models are more likely to overestimate item difficulty parameters in the presence of MNAR missing data. There was no statistically significant difference in the values between complete and MAR data. This was qualified by the interaction between the type of missing and the number of items, $F(2, 111) = 5.314$, $p < .01$, $\eta_p^2 = .087$, and between the type of missing and the percentage of missing, $F(2, 111) = 3.455$, $p = .03$, $\eta_p^2 = .059$. Therefore, the effect of the type of missing data on bias could be changed according to the number of items and the percentage of missing.

Dependency of Simulation Data. In this study, three different types of missing data, complete, MAR, and MNAR, were used as a simulation condition. The data for MAR and MNAR were created from the data for complete condition by removing some responses. Thus, it was suspected that the results from those data could be highly correlated. Therefore, a mixed between-within subject ANOVA was used to take care of this dependency.

For the values of RMSE, the significant main effects were exactly the same with the ones found in the general linear model above. The model, data, class, item, and percentage factors were significant in both cases. A mixed between-within subjects analysis of variance was conducted to assess the impact of six factors (model, data, class, sample, item, and percentage) on RMSE with type of missing as a within-subject factor. The main effects of model yielded the F ratio, p value, and, η_p^2 of $F(1, 53) = 4.397$, $p = .04$, and $\eta_p^2 = .07$. The main effects of data yielded $F(1, 53) = 221.370$, $p < .01$, and $\eta_p^2 = .80$. The main

effects of class yielded $F(1, 53) = 69.871$, $p < .01$, and $\eta_p^2 = .56$. The main effects of item yielded $F(1, 53) = 46.619$, $p < .01$, and $\eta_p^2 = .46$. The main effects of percentage yielded $F(1, 53) = 5.99$, $p = .02$, and $\eta_p^2 = .10$. The within-subject factor, the type of missing data, was also significant, $F(2, 53) = 127.529$, $p < .01$, and $\eta_p^2 = .10$. There were statistically significant interactions between the type of missing and the number of items, $F(2, 53) = 7.5$, $p < .01$, $\eta_p^2 = .124$, and between the type of missing and the percentage of missing, $F(2, 53) = 34.393$, $p < .01$, $\eta_p^2 = .394$. Therefore, the effect of the type of missing data could be changed according to the number of items and the percentage of missing.

The same model was used for the values for bias. The within-subject factor, the type of missing data, was significant, $F(2, 53) = 582.094$, $p < .01$, and $\eta_p^2 = .917$. Also, there were statistically significant interactions between the type of missing and the number of items, $F(2, 53) = 123.726$, $p < .01$, $\eta_p^2 = .700$, and between the type of missing and the percentage of missing, $F(2, 53) = 84.239$, $p < .01$, $\eta_p^2 = .614$.

3.2.2 The Impact of Missing Data on the Class Classification

In the previous section, the quantities of interest were RMSEs for the simulation conditions. The RMSE for each condition was calculated across items, classes, and replications in a condition. In this section, the rate of correct identification for the number of classes was presented and analyzed based on the similar framework used in the analysis of RMSEs. The following is the description of how the rate was calculated. For each simulation condition, mixture IRT models with one, two, and three classes were fitted to the data and then the number of classes for the given data was chosen as the number of classes of the model showing minimum AIC or BIC. The frequencies of the number of classes for the simulation conditions are presented from Table 1 to Table 128 in Appendix A. In the tables, the rate of correct identification can be seen from the column corresponding to the true number of classes in the simulation notation. For example, in the table for the simulation condition R1210120010, the second number after R, which is 2, represents the true number of classes. Therefore, the

frequencies in the second column of the table for the simulation condition R1210120010 indicates the rate of correct identification. Since the optimal number of classes for each condition can be determined by both AIC and BIC, the rate of correct identification can be obtained based on both AIC and BIC. In order to analyze the rate of correct identification and find some meanings from it, the general linear model procedure in the SPSS software, version 17 (SPSS, Chicago, IL), was used as in the previous section. The general linear model results for the rate of correct identification based on AIC and BIC are presented in Table 3.5 and Table 3.6 respectively.

Comments on the Results in the Tables. In most of the simulation conditions, the BIC index was almost perfect in identifying the true number of classes in the data except one condition in which the mixture Rasch model was fitted to the data generated using the 2PL. In fact, in that exception, the BIC index almost always failed to identify the correct number of classes in the data. Since the difference in the rate of correct classification was so large between those two groups, it was decided to exclude the data from the conditions in which the mixture Rasch model was fitted to the data that was generated using 2PL from the analysis using the general linear model. Therefore, only data from the remaining conditions were used to run the general linear model for the rate based on the BIC. The discussion on those excluded data will be presented in section 3.2.3. On the other hand, overall, the performance of the AIC in identifying the true number of classes in data was not as good as the one of the BIC. However, the results from the BIC were perfect. Therefore, only the general linear model result for the rate from the AIC based on Table 3.5 will be discussed in this section. In running the general linear model for the AIC, the conditions showing a significant drop in the rate were also excluded like the case of the BIC. The general linear model result for the rate from the BIC is presented in Table 3.6 without further discussion as almost all the coefficients were not statistically significant, .

The Effect of Complexity in Modeling on the Rate Based on AIC. As with the previous analysis on RMSEs in the previous section, the model, data, and class factors can

be considered the ones representing the complexity in the modeling. All the three factors, model ($\beta = 18.875$, $p < .01$, large effect size based on $\eta^2 = .305$), data ($\beta = 11.938$, $p < .01$, large effect size based on $\eta^2 = .101$), and class ($\beta = 4.444$, $p < .01$, small effect size based on $\eta^2 = .02$) were statistically significant. The positive sign of the coefficients indicate that the rate of correct identification for the true number of classes in data increases as the complexity in the modeling increases. At first, this result seemed to be quite counter-intuitive. It seemed that the rate of correct identification should be low when the complex model was used since it is less likely to get correct estimation for the complex model in general. Note that, in the previous section, it was found that AIC and BIC almost always failed for the conditions in which the mixture Rasch model was fitted to data generated using the 2PL. In my understanding, the finding of this section that the more complex model, which is the 2PL, shows a higher rate of correct identification is just another way of saying the finding of the previous section. One of the possible conclusions from those two findings could be the following: The mixture models might use their allowed degree of freedom in the number of classes to explain the complexity in data even when the complexity in data does not come from the heterogeneity of a population. This potential explanation for the findings will be discussed more in the discussion section of this chapter.

The Effect of the Amount of Information in Data on the Rate based on AIC. Both samples ($\beta = -3.864$, $p = .01$, small effect size based on $\eta^2 = .01$) and items ($\beta = -11.556$, $p < .01$, large effect size based on $\eta^2 = .15$) factors were statistically significant with negative values of coefficients indicating that the rate of correct identification drops as the sample size or the number of items increases. This result also seemed to be counter-intuitive at first. However, after reviewing previous research, it turned out that this inverse relationship between sample size or the number of items and the rate of correct identification should not be easily generalized since the performance of information criteria seem to be confounded with other factors including sample size and separation between classes. This will be discussed more in the discussion section.

The Effect of Missing Data on the Rate Based on AIC. The percentage factor was not statistically significant. However, the rate of correct identification for data with MNAR missing ($\beta = -7.705$, $p < .01$) and MAR missing ($\beta = -4.091$, $p = .03$) was lower than that of the complete data set. In this simulation study, when the mixture model failed to identify the correct number of classes, this always resulted in an over-extraction, not an under-extraction. Therefore, when the rate of correct identification drops, it means that spurious classes are generated.

Dependency of Simulation Data. In order to take care of the dependency of simulation data across complete, MAR, and MNAR condition, a mixed between-within subject ANOVA was used to see the effect of factors of interest on the AIC. The main effects of model yielded the F ratio, p value, and, η_p^2 of $F(1, 37) = 54.417$, $p < .01$, and $\eta_p^2 = .595$. The main effects of data yielded $F(1, 37) = 17.925$, $p < .01$, and $\eta_p^2 = .326$. The main effects of item yielded $F(1, 37) = 26.770$, $p < .01$, and $\eta_p^2 = .420$. The within-subject factor, the type of missing data, was also significant, $F(1, 37) = 6.304$, $p = 017$, and $\eta_p^2 = .146$. There was statistically significant interaction between the type of missing and the number of items, $F(2, 37) = 12.765$, $p < .01$, $\eta_p^2 = .257$.

3.2.3 Applying the Mixture Rasch Model to the Data from 2PL

As described in the previous section, the AIC and BIC showed poor performance in identifying the correct number of classes in the data in which the mixture Rasch models were fitted to data that was generated using the 2PL. The AIC almost perfectly failed in those simulation conditions. Interestingly, in those simulation conditions, the BIC worked perfectly well in some conditions whereas failed perfectly in some other conditions. In this section, some observations on the pattern of the AIC and BIC in those simulation conditions will be presented based on Figure 3.1 below. Recall the notation for the simulation condition. For example, R2130 represents that the mixture Rasch model was used (R), the data was generated using the 2PL (2), the true number of classes in data was one (1), and the number

Table 3.5: General Linear Model Results for AIC

Parameter	β	Std.Error	*df*	Sig	η_p^2	η^2	Power	*M*	*SD*
(Intercept)	71.076	2.2204	1	$< .01$	.905		1.00		
[Model=2PL]	18.875	1.7774	1	$< .01$	.461	.305	1.00	84.792	1.077
[Model=1PL]	0							64.945	2.104
[Data=2PL]	11.938	1.9586	1	$< .01$	.753	.101	1.00	90.08	0.835
[Data=1PL]	0							78.38	1.622
[Samples=1200]	-3.864	1.5158	1	.01	.047	.01	.68	74.553	2.120
[Samples=600]	0							78.412	1.823
[Items=30]	-11.556	1.5514	1	$< .01$	.296	.15	1.00	72.723	2.290
[Items=10]	0							83.983	1.186
[Percentage=20]	0.561	1.5158	1	.71	.001	.00	.06	76.762	2.074
[Percentage=10]	0							76.203	1.904
[Type=mnar]	-7.705	1.8564	1	$< .01$	.115	.05	.98	72.703	2.799
[Type=mar]	-4.091	1.8564	1	.03	.056		.46	76.324	2.328
[Type=comp]	0							80.412	2.010
[Class=2]	4.444	1.5514	1	$< .01$	.059	.02	.78	80.231	2.002
[Class=1]	0							76.238	1.955

Note. M = mean, SD = standard deviation.

Table 3.6: General Linear Model Results for BIC

Parameter	β	Std.Error	*df*	Sig
(Intercept)	100.124	0.9527	1	$< .01$
[Model=2PL]	-0.958	0.7626	1	.21
[Model=1PL]	0			
[Data=2PL]	0.652	0.8404	1	.44
[Data=1PL]	0			
[Samples=1200]	0.409	0.6504	1	.52
[Samples=600]	0			
[Items=30]	0.996	0.6657	1	.13
[Items=10]	0			
[Percentage=20]	-0.500	0.6504	1	.44
[Percentage=10]	0			
[Type=mnar]	0.159	0.7965	1	.84
[Type=mar]	-1.000	0.7965	1	.21
[Type=comp]	0			
[Class=2]	-2.218	0.6657	1	$< .01$
[Class=1]	0			

of items was 30 (30). Therefore, the simulation conditions of interest in this section are 48 conditions starting with R2 among the total 192 simulation conditions. The pattern of the AIC and BIC in those 48 conditions was presented in Figure 3.1 containing four sub-figures. In each sub-figure, the rate of correct identification was plotted using the box-plot according to four subsets of conditions among those 48 conditions: R2110, R2130, R2210, and R2230. Each set of conditions contains 12 simulation conditions. The whole results of class classification for those 48 conditions were presented in the tables in Appendix A: from Table A.33 to A64. The following are some observations based on Figure 3.1.

The complexity in data due to 2PL can produce spurious classes. This observation can be made based on Figure 3.1 (a) and (b). Those two sub-figures show that the AIC failed perfectly in those 48 conditions regardless of the sample size. Since, in this simulation study, mis-classification always means over-extraction, it can be said that the complexity in the data due to 2PL produced spurious classes. In fact, this was what Alexeev et al. (2011) found in their study. Basically, what they found was the general feature of mixture models that applying an overly restricted model to data may produce spurious classes. This will be discussed more in the discussion section of this chapter.

The spurious classes due to 2PL may or may not be realized depending on the sample size. Another interesting observation can be found by comparing the rate of the BIC for the condition R2230 in sub-figures (b) and (d) of Figure 3.1. The rate of the BIC for the condition R2230 in sub-figure (d) was almost zero, which indicates that spurious classes were produced by the complexity in data due to 2PL. However, the rate of the BIC for the same conditions R2230 in sub-figure (b) was almost perfect even in the presence of the same source of complexity in the data. The only difference between those two cases was the sample size. Given the same source of complexity in the data, the BIC did not produce any spurious classes when the sample size was 600, which is shown in sub-figure (b), but the BIC did perfectly produce spurious classes when the sample size was 1200, which is shown in sub-figure (d). This observation suggests that the complexity in data due to 2PL does not

always work to produce spurious classes. The spurious classes due to 2PL may or may not be realized depending on the sample size.

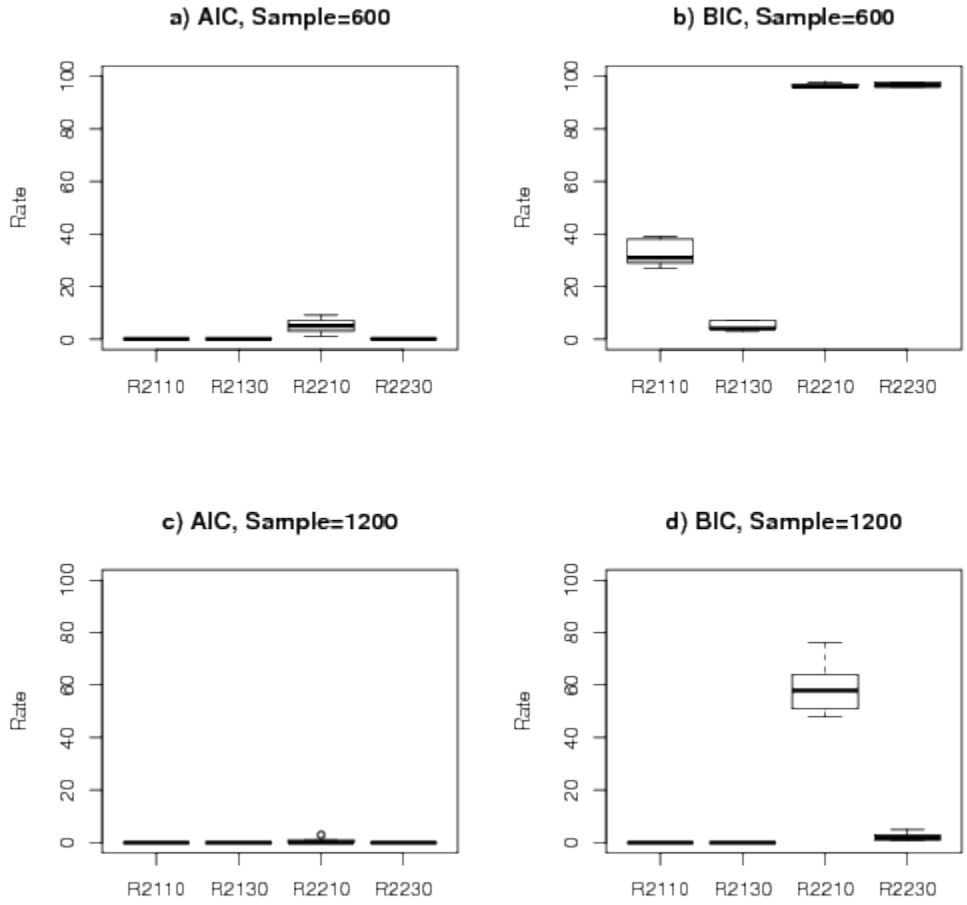
The spurious classes due to 2PL may or may not be realized depending on the number of items. This can be observed by comparing R2110 and R2130 in sub-figure (b) or R2210 and R2230 in sub-figure (d). With 10 items, the complexity in data due to 2PL seemed to be partially realized as spurious classes. However, with 30 items, it seems that the complexity was almost perfectly realized as spurious classes.

The spurious classes due to 2PL may or may not be realized depending on the other complexity in the data. This observation can be obtained by comparing R2110 and R2210 or R2130 and R2230 in sub-figure (b). For example, the data for R2130 was generated using the 2PL with just one class. It can be said that the complexity in the data comes from the 2PL. On the other hand, the data for R2230 was generated using the 2PL with two classes. It can be said that the complexity in the data comes from both the 2PL and two classes. However, the spurious classes were generated almost perfectly in the case of R2130 but were not generated almost perfectly in the case of R2230. One of the possible explanations for this fact is that, in R2230, the complexity from the two classes overwhelmed the one from the 2PL so that the bigger complexity from two classes suppressed the power of the complexity from the 2PL to produce spurious classes.

3.3 Summary and Discussion of the Results

The design and the descriptions of the results of this simulation study are presented in the previous sections 3.1 and 3.2. In this section, the discussion of the results of this simulation study will be presented in two parts. In the first part, the impact of missing data on the mixture IRT models will be presented. In the second part, the issue of spurious classes in mixture models will be discussed from the point of view that overly restricted models can produce spurious classes.

Figure 3.1: The Rate of Correct Identification When the Mixture Rasch Models Were Fitted to the Data Generated by 2PL



3.3.1 The Impact of Missing Data on the Mixture IRT Models

As was described in the first section of this chapter, the goal of this simulation study was to investigate the impact of missing data on the dichotomous mixture IRT models. The evaluation criteria for the impact were the following: the accuracy of item parameter recovery measured by RMSE and the rate of correct identification for the number of classes. Seven factors, which are model for fitting, model for data generation, the number of classes, sample

size, the number of items, percentage of missing, and the types of missing were manipulated as the simulation conditions.

RMSE. The impact of these seven factors on RMSE was summarized in Table 3.4. The results in Table 3.4 coincide with our knowledge on the accuracy of estimation in general. Complexity in data, which was represented by data, and class factors, resulted in less accurate parameter estimations. The more items the data had, the more accurate the estimation was. Also, the impact of missing data coincides with that which was expected by theory. According to Little and Rubin (1987), the estimation of MLE based on the likelihood function with MAR missing data and without any missing data should be the same. In this simulation study, the MNAR missing data increased RMSE compared to the complete data but the MAR missing did not. Also the RMSE was increased as the percentage of missing in data was increased.

On the other hand, the above statements about the impact of various factors on RMSE were based on the p-values of statistical significant tests. It is well known that a p-value is an index confounded with sample size. In other words, statistical significance does not necessarily imply practical significance. In this case, the effect size measure could be useful to consider practical or relative importance of an effect. In this study, η^2 values for effects of various factors on RMSE were also presented in Table 3.4. Data factor (large effect size based on $\eta^2 = .494$), which represents the models for generating data, and class factor (large effect size based on $\eta^2 = .156$), which represents the number of classes in data, showed high η^2 values. However, these two factors are actually the characteristics of data itself and cannot be manipulated in practical situations. Therefore, once those factors are considered to be given, then the number of items (medium effect size based on $\eta^2 = .104$) and the types of missing (medium effect size based on $\eta^2 = .056$) were the factors having the most high values of η^2. This kind of comparison might give some sense on the magnitude of effect of missing data. In these special conditions of this simulation study, the effect size of the types of missing data was around 54% of that of the items, which might suggest that dealing with

missing data appropriately is as important as increasing the number of items in terms of RMSE. It might be pointed out that increased RMSE might be natural in the presence of missing data since the sample size was reduced in the missing data conditions. However, the RMSE of MAR missing data was in the similar range with the RMSE of the complete data ($p = .18$), which indicates that reduced sample size by the missing data does not necessarily introduce more RMSE.

It might be possible that there exists dependency across complete, MAR, and MNAR data since the data for MAR and MNAR were created from the data for complete condition by removing some responses. In order to take care of this dependency, a mixed between-within subject ANOVA was used with the type of missing as a within-subject factor. The significant main effects of a mixed between-within subject ANOVA were exactly same as the one of the general linear model in Table 3.4: the model, data, item, percentage, class, and type factors were statistically significant.

Bias. It was found that the item difficulty parameters for the MNAR data were overestimated compared to the ones for the complete data. Also, it was found that the effect of the type of missing data on bias could be changed according to the number of items and the percentage of missing based on the statistically significant interaction terms.

Correct Classification Rate. Another quantity of interest in this simulation study was the rate of correct identification for the number of classes in data. As can be seen from the tables in Appendix A, the BIC was almost perfect in identifying the true number of classes except some conditions in which the mixture Rasch models were fitted to the data that was generated using the 2PL. When the performance of any index in mixture models is being discussed, it is very important to realize that the performance of an index is actually conditioned on many other factors. For example, in the context of latent profile analysis, the BIC completely failed in identifying the correct number of class in small class separation but completely succeeded in large class separation, given the same sample size $N = 300$, (Lubke & Neale, 2006). In that study, the separation between classes was defined as the Mahalanobis

distance, $(\mu_1 - \mu_2)\mathbf{\Sigma}^{-1}(\mu_1 - \mu_2)$, since the observed variable in the latent profile analysis is continuous. Tofighi and Enders (2007) also used the class separation as an important factor in the context of the growth mixture model. However, an appropriate definition of class separation could not be found in the context of the IRT, which requires further studies on this topic. One of the possible ways for defining class separation in the IRT context could be using the Euclidean distance between item parameter vectors of classes. This might be natural in the IRT context since usually the difference in classes are characterized by the difference in the item parameters.

Keeping in mind that the performance of the AIC and BIC is conditioned on many other factors such as class separation, the performance of the AIC summarized by the general linear model was presented in Table 3.5. Since the performance of the BIC was almost perfect, there was nothing to be summarized using the general linear model. The fact that the rate of correct identification was more high ($\beta = 18.875$, $p < .01$, large effect size based on $\eta^2 = .305$) when the 2PL was used to fit data seems to represent that fitting overly restricted model, which is the mixture Rasch model here, is more likely to produce spurious classes. Both MAR missing ($\beta = -4.091$, $p = .03$) and MNAR missing ($\beta = -7.705$, $p < .01$) resulted in less accurate estimation in terms of the rate of correct identification but the percentage of missing in this simulation study didn't affect the rate ($p = .71$) indicating that the complexity in data introduced by missing data could be the source of spurious classes.

Interestingly, the rate of correct identification decreased as the sample size ($\beta = -3.864$, $p = .01$, small effect size based on $\eta^2 = .01$) and the number of items ($\beta = -11.556$, $p < .01$, large effect size based on $\eta^2 = .15$) increased indicating that the mixture IRT model is more likely to produce spurious classes as the sample size and the number of items increase. This result is consistent with the one from Li et al. (2009) in which the AIC performed worse when the sample size increased in the mixture IRT context. Janssen and De Boeck (1999) also found that the AIC preferred more complex model in the multidimensional IRT context. However, there were other studies showing that the performance of the AIC becomes

better as the sample size increases. Tofighi and Enders (2007) tested four sample sizes, which were 400, 700, 1000, and 2000, as a factor that can affect the performance of the AIC and other indices on the class classification in the growth model context. The rate of correct identification by the AIC increased up to the sample size 1000 but started to decrease from 1000 to 2000. Lubke and Neale (2006) tested seven sample sizes, which were 25, 50, 75, 150, 200, 300, and 1000, in the context of latent profile analysis and found that the performance of the AIC was becoming better as the sample size increased. Based on those results, it seems that the relationship between the performance of the AIC and the sample size depends on the condition in which the simulation study was conducted. In fact, McDonald and Marsh (1990) already warned the use of AIC for the model selection in the structural equation models. They argued that the AIC would depend on the sample size, like the conventional chi-square test and would tend to prefer saturated models in large samples and models with few estimated parameters in small samples. In other words, the complexity of the model selected by the AIC increases with the sample size. Based on their argument, the fact that the AIC performed worse as the sample size increased in some range of sample sizes might be explained by the tendency of the AIC that prefers more complex model in large sample size.

3.3.2 The Issue of Spurious Classes in Mixture Models

This simulation study is for investigating the impact of missing data on the mixture IRT models, In other words, this study is about how the complexity in data introduced by missing data affect the estimation of the mixture IRT models in terms of RMSE and spurious classes. The possibility of mixture models to misidentify spurious classes has been concerned by many researchers. Basically, this concern comes from the flexibility of mixture models in which multiple classes are allowed to model the heterogeneity of a population. This flexibility of mixture models allows the possibility of misidentification since the mixture models might take advantage of their flexibility to explain some complexities in data due to factors other

than population heterogeneity. When overly restricted model is fitted to data, it's more likely to produce spurious classes. For example, it is apparent that applying the mixture Rasch model to the data that was generated by the 2PL is more likely to produce spurious classes based on this simulation study and earlier research (Alexeev et al, 2011). In this case, the complexity in data comes from the complexity of model for data generation. In this framework, the missing data could be considered as another source of complexity in data. Therefore, at this point, it would be interesting to compare the effect of missing data and the 2PL in producing spurious classes. In this simulation study, it seems that the effect of missing data on the spurious class is much smaller than that of the 2PL. Even though missing data lowered (MAR: $\beta = -4.091$, $p = .03$), MNAR: $\beta = -7.705$, $p < .01$) the rate of correct identification, it was still more than 60% (68.17% for complete data, 64.79% for MAR, and 61.75% for MNAR). However, the rate for the case in which the mixture Rasch model was used to 2PL data was almost zero, which indicates that it almost always produced spurious classes.

3.3.3 Summary of the Simulation Study

In this simulation study, the impact of missing data on the mixture IRT models were tested in terms of RMSE and the rate of correct identification. The MNAR missing data increased RMSE compared to the complete data but the MAR missing didn't. Also the RMSE was increased as the percentage of missing in data was increased. Based on the rate of correct identification, it seems that both MNAR and MAR missing data produced spurious classes. However, the effect of missing data in producing spurious classes was much smaller than that of mis-specification of a model, that is, applying the mixture Rasch model to 2PL data.

3.4 Empirical Example

In this section, an example was illustrated to check the tendency of model selection indices and the impact of missing data which found in the previous section.

Description of Data. The data were collected to evaluate the efficacy of Enhanced Anchored Instruction (EAI) (Bottge, 1999). EAI extends AI by affording students additional opportunities to practice their skills as they solve new but analogous problems in applied and challenging contexts. Students first solve a problem in a multimedia format and then apply what they learn in related hands-on problems (e.g., building skateboard ramps, designing and manufacturing hovercrafts). One important advantage of EAI is its ability to directly immerse students in problem contexts, thus helping to eliminate the comprehension difficulties students with low skills in math and reading often experience with complex text-based problems. A pretest-posttest cluster randomized school based trial will be conducted in year 2 and year 3 to test the efficacy of EAI. In each study, 23 schools, 2-3 teachers per school, and 5-10 students per teacher will participate. The data used in this example is the pretest data. The number of items was 73 and the number of examinees was 351.

Estimation Method. In an exploratory analysis, the number of classes were determined based on the same procedure in the simulation study. Models with different classes were fitted to the data and the number of classes of the model showing best fit was determined as the number of classes. The M*plus* Version 6.11 (Muthén & Muthén, 1998-2011) was used to fit models in this simulation study. M*plus* was set to use a maximum likelihood estimation with robust standard errors by specifying ALGORITHM=INTEGRATION.

Results. The tendency of AIC and BIC was similar with the one found in the simulation study and earlier researches. In Table 3.7, the model fit results of the mixture Rasch model and the 2PL for the complete data were presented. First, the tendency of the AIC that prefers complex models was also checked in this example. The number of classes identified by the AIC was always larger than by the BIC. Based on the BIC, the optimal number of classes were two for the Rasch model and one for the 2PL. However, based on the AIC, the optimal number of classes was two for the 2PL. For the Rasch model, the AIC didn't show the minimum value up to four classes. Also, the fitting the Rasch model to the data produced more classes than fitting the 2PL. As discussed in the previous section, the Rasch

model is more restricted model than the 2PL so that a certain complexity in the data might be expressed as the multiple classes in the Rasch model. In Table 3.8 and 3.9, the model fit results for the mixture Rasch model for the data with MAR and MNAR missing data were presented respectively. As also discussed in the previous section, both MNAR and MAR missing data have a tendency to produce spurious classes. However, in this example, the spurious classes by those missing data was not observed. Figure 3.2 illustrate the item difficulty parameters across the whole 73 items when the mixture Rasch model with 2 classes was fitted to the actual complete data. Figure 3.3 and Figure 3.4 showed the cross pattern of item difficulty parameter profiles that characterize the existence of latent classes.

Table 3.7: Model Comparison Results for Example Data

	Rasch (complete)				2PL		
	1 class	2 classes	3 classes	4 classes	1 class	2 classes	3 classes
AIC	27875.578	27554.896	27481.442	27426.484	27464.001	**27289.299**	27358.047
BIC	28161.276	**28126.293**	28334.676	28565.416	**28026.676**	28420.509	28493.118

Table 3.8: Model Comparison Results for Example Data

	Rasch (MAR)			
	1 class	2 classes	3 classes	4 classes
AIC	22023.043	21840.617	21749.298	21707.831
BIC	22504.880	**22408.153**	22602.532	22846.763

Discussion. The indices that are used in model selection in mixture IRT models does not always give the same results, which suggests some more decision making procedures to decide the number of classes in a population. In a certain condition, mixture models can produce spurious classes. In fact, this kind of limitation of mixture models as an exploratory tool was the motivation of this study because knowledge on the behavior of those fit indices

Table 3.9: Model Comparison Results for Example Data

	Rasch (MNAR)			
	1 class	2 classes	3 classes	4 classes
AIC	22350.177	21872.284	21773.972	21760.230
BIC	22632.014	**22439.820**	22627.206	22899.162

in various conditions could be helpful in deciding the number of classes. The limitation of mixture models will be discussed more in the next chapter.

Figure 3.2: The Item Difficulty Parameters for the Whole 73 Items

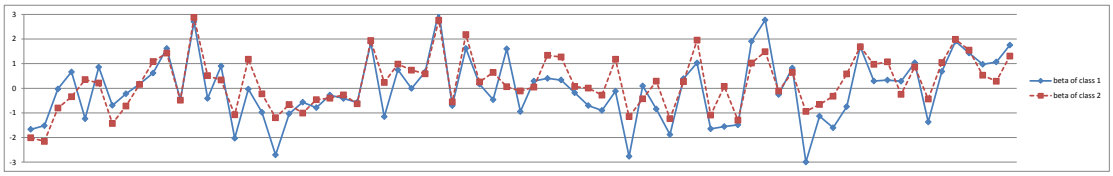


Figure 3.3: The Item Difficulty Parameters for the Items 1-9

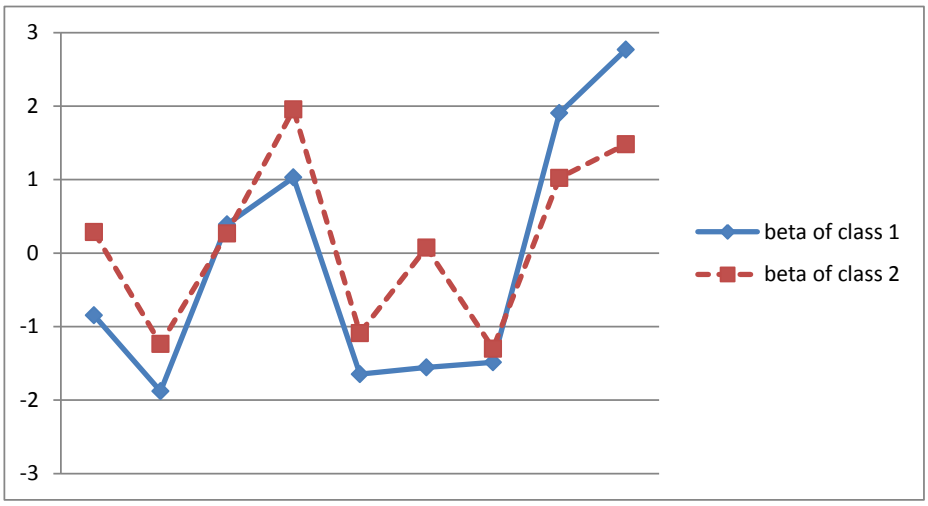
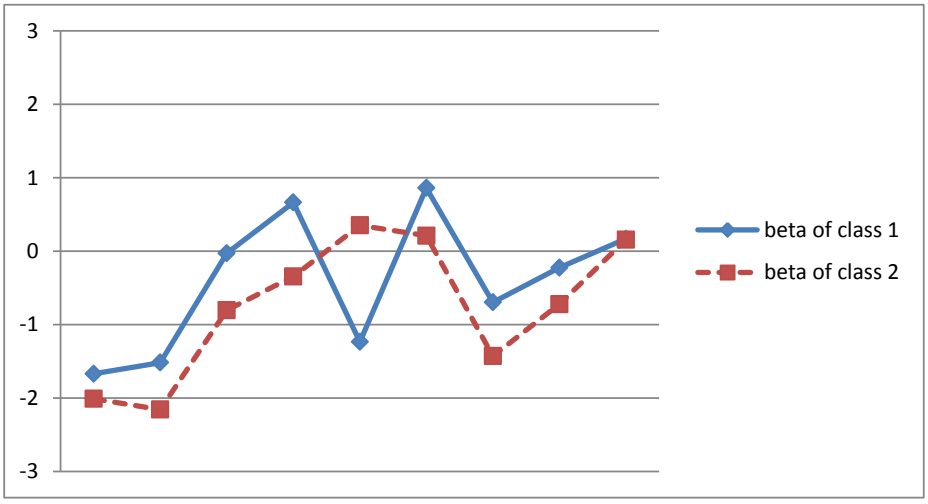


Figure 3.4: The Item Difficulty Parameters for the Items 49-57



Chapter 4

CONCLUSION AND DISCUSSION

In this study, the impact of missing data on the dichotomous mixture IRT models was investigated in various conditions. The motivation of this study was the concern about spurious classes in the mixture IRT models, which is actually rooted from the relaxation of unidimensional assumption of the Rasch model. In the conditions of this simulation study, spurious classes were generated by the data having MNAR and MAR missing but not as much as by the data that was generated by the 2PL and was fitted by the mixture Rasch model. Even though the impact of missing data was not that much in the conditions of this study, two things should be noted here: First, the impact of any factor on spurious classes is conditioned on other factors. For example, as described in section 3.2.3, the spurious classes due to the 2PL may or may not be realized depending on the sample size. Given the data that were generated by the 2PL and were fitted by the mixture Rasch model, the BIC didn't produce any spurious classes when the sample size was 600 but did perfectly produced spurious classes when the sample size was 1200. This example implies that the impact of missing data could be serious in some other conditions, which requires further study. Second, the motivation of this study still seems to be quite important and worth while to discuss little bit more in this discussion chapter.

The Unidimensionality Assumption in IRT. The most commonly used models in IRT would be the Rasch model, the 2PL, and the 3PL. The unidimensionality of a latent trait is one of the most important underlying assumptions in those IRT models. By assuming the unidimensionality of a latent trait, the dependencies between the responses of a participant

are accounted for by the unidimensional latent trait. In other words, once the unidimensional latent trait is controlled, then the dependencies are expected to be removed from responses. However, as with any other model in statistics, the violation of this unidimensionality assumption could be a potential threat to the valid estimation of those IRT models. Given this situation, it would be very natural to consider the generalization of those models to deal with the violation of the unidimensionality assumption.

Rijmen and De Boeck (2005) compared two generalizations of the Rasch model that relax the unidimensionality assumption: the beween-item multidimensional model (Adams, Wilson, & Wang, 1997), and the mixture Rasch model (Rost, 1990). In order to describe those two generalizations, they decomposed the unidimensionality assumption further. According to them, the unidimensionality means that all items are located on the same scale, and that this scale is the same for all persons. In the between-item multidimensional model, the assumption that all items are located on the same scale is relaxed, and instead the assumption that a test consists of K subgroups of items that each can be modeled by a Rasch model is used. The mixture Rasch model is another generalization of the Rasch model by relaxing the assumption that the scale a test is measuring is the same for all persons. In the mixture Rasch model, the total group of persons are assumed to consist of T qualitatively different subgroups, within which the unidimensional Rasch model is assumed to hold. In this way, the mixture Rasch model could be considered as the generalization of the Rasch model to deal with the violation of the unidimensionality assumption.

The Limitation of the Mixture Models as an Exploratory Analysis Tool. As described above, the flexibility of the mixture Rasch model was motivated to deal with the violation of the unidimensional assumption. To be more specific, the mixture Rasch model is more flexible than the Rasch model in that the scale a test is measuring is allowed to differ across subgroups of a population. The problem is that the flexibility of the mixture model could be used to fit the complexity in data caused by another sources other than hetetogeneity of a population. The classes in mixture model that is produced by another source other than

heterogeneity of a population are called spurious classes. In the IRT context, Alexeev et al. (2011) found that data that were generated by 2PL caused spurious classes when the data were fitted using the mixture Rasch model. In this case, the mixture Rasch model could be considered as an overly restricted model for the given data. It could be said that the complexity in data due to the 2PL was fitted using the flexibility of the classes in the mixture Rasch model. Bauer (2007) also presented a similar problem in the growth mixture model context (GMM). In the GMM, spurious classes were produced by violations of five assumptions of GMM: within-class normality, correct specification of covariance structure, correct specification of linear or non-linear effect of covariates, MAR assumptions for missing data, and independence of sample individual. Bauer pointed out that it is the non-normality of distribution that is used to identify the existence of classes in growth mixture model. Growth mixture model, however, seems to be unable to disentangle the sources of non-normality: whether it comes from the real heterogeneity in population or other mechanisms. The existence of classes representing subpopulation may not be the only plausible explanation for the non-normality of distribution. Bauer's comment could be applied to the mixture model in general. The limitation of mixture model as an exploratory tool seems to be its incapability of discerning the confounding sources of generating mixture density.

The motivation of this study was to investigate whether missing data could be a potential source of spurious classes or not. Given the conditions of this study, missing data produced spurious classes but not severely. The reason why 3PL was excluded from the simulation conditions was that fitting 3PL seems to be impossible in M*plus*. Also, based on similar research in other context such as GMM, the class separation seems to be important factor that can affect the realization of the spurious classes. However, appropriate definition of class separation in the IRT context could not be found, which suggests future study.

References

Adams, R. J., Wilson, M., & Wang, W. C. (1997). The multidimensional random coefficients multinomial logit model. *Applied Psychological Measurement, 21*, 1-23.

Aitkin, M. & Rubin, D. B. (1985). Estimation and hypothesis testing in finite mixture models. *Journal of the Royal Statistics Society: Series B, 47*, 67-75.

Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transaction on Automatic Control, 19*, 716-723.

Alexeev, N., Templin, J., & Cohen, A. (2011). Spurious latent classes in the mixture Rasch model. *Journal of Educational Measurement, 48,* 313-332.

Bartholomew, D. J., Steele, F., Moustaki, I., & Galbraith, J. I. (2002). *The analysis and interpretation of multivariate data for social scientists.* Boca Raton: Chapman & Hall/CRC Press.

Bauer, D. J. (2007). Observations on the use of growth mixture models in psychological research. *Multivariate Behavioral Research, 42*, 757-786.

Bauer, D. J., & Curran, P. J. (2003). Overextraction of latent trajectory classes: Much ado about nothing? Reply to Rindskopf (2003), Muthén (2003), and Cudeck and Henly (2003). *Psychological Methods, 8*, 384-393.

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

Bottge, B. A., & Yehle, A. (1999). Making standards-based instruction meaningful for all. *Journal for Vocational Special Needs Education, 22*, 23-32.

Bozdogan, H. (1987). Model selection and Akaike's information criterion (AIC): The general theory and its analytic extensions. *Psychometrika, 52*, 345-370.

Cho, S.-J., Cohen, A. S., & Kim, S.-H. (2006, June). *An investigation of priors on the probabilities of mix- tures in the mixture Rasch model.* Paper presented at the International Meeting of

the Psychometric Society: The 71st annual meeting of the Psychometric Society, Montreal, Canada.

Cohen, A. S., & Bolt, D. M. (2005). A mixture model analysis of differential item functioning. *Journal of Educational Measurement, 42*, 133-148.

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

Cohen, J. (1988). *Statistical power analysis for the behavioral science*. Hillsdale, NJ: Lawrence Earlbaum Associates.

Day, N. E. (1969). Estimating the components of a mixture of two normal distributions. *Biometrika, 56*, 463-474.

De Ayala, R. J., Plake, B. S., & Impara, J. C. (2001). The impact of omitted responses on the accuracy of ability estimation in item response theory. *Journal of Educational Measurement,* 213-234.

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

Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm (with discussion). *Journal of the Royal Statistical Society: Series B, 39*, 213-234.

Enders, C. K. & Bandalos, D. L. (2001). The relative performance of full information maximum likelihood estimation for missing data in structural equation models. *Structural Equation Modeling, 8*, 430-457.

Enders, C. K. (2010). *Applied missing data analysis*. New York: Guilford Press.

Enders, C. K. (2004). The impact of missing data on sample reliability estimates: Implications for reliability reporting practices. *Educational and Psychological Measurement, 64,* 419-436.

Finch, W. H. (2008). Estimation of item response theory parameters in the presence of missing data. *Journal of Educational Measurement, 45*, 225-245.

Finch, W. H. (2011). The impact of missing data on the detection of nonuniform differential item functioning. *Educational and Psychological Measurement, 71*, 663-683.

Flora, D., Curran, P., Hussong, A., & Edwards, M. (2008). Incorporating measurement nonequivalence in a cross-study latent growth curve analysis. *Structural Equation Modeling, 15*, 676-704.

Geisser, S., & Eddy, W. F. (1979). A predictive approach to model selection. *Journal of the American Statistical Association, 74*, 153-160.

Gelfand, A. E., & Dey, D. K. (1994). Bayesian model choice: Asymptotics and exact calculations. *Journal of the Royal Statistical Society: Series B, 56*, 501-514.

Gelman, A., Carlin, J. B., Stern, H. S., & Rubin, D. B. (1996). *Bayesian data analysis*. London: Chapman & Hall.

Graham, J. W., Taylor, B. J., Olchowski, A. E., & Cumsille, P. E. (2006). Planned missing data designs in psychological research. *Psychological Methods, 11,* 323-343.

Green, S., Lissits, R., & Mulaik, S. (1977). Limitations of coefficient alpha as an index of test unidimensionality. *Educational and Psychometrical Measurement, 37*, 827-835.

Hattie, J. (1985). Methodology review: Assessing unidimensionality of tests and items. *Applied Psychometric Measurement, 9*, 139-164.

Janssen, R., & De Boeck, P. (1999). Confirmatory analyses of componential test structure using multidimensional item response theory. *Multivariate Behavioral Research, 34*, 245-268.

Kolen, M. J., & Brennan, R. L. (2004). *Test equating, scaling, and linking: Methods and practices*. New York: Springer.

Kuha, J. (2004). AIC and BIC: Comparisons of assumptions and performance. *Sociological Methods & Research, 33,* 188-229.

Li, F., Cohen, A. S., Kim, S.-H., & Cho, S.-J. (2009). Model selection methods for dichotomous mixture IRT models. *Applied Psychological Measurement, 33,* 353-373.

Little, R. J. A., & Rubin, D. B. (1987). *Statistical analysis with missing data*. New York: Wiley.

Lord, F. M., Novick, M. R., & Birnbaum, A. (1968/2008). *Statistical theories of mental test scores.* Reading, MA: Addison-Wesley.

Lubke, G., & Neale, M. C. (2006). Distinguishing between latent classes and continuous factors: Resolution by maximum likelihood? *Multivariate Behavioral Research, 41*, 499-532.

McDonald, R. P. (1962). A note on the derivation of the general latent class model. *Psychometrika, 27*, 203-206.

McDonald, R. P., & Marsh, H. W. (1990). Choosing a multivariate model: Noncentrality and goodness of fit. *Psychological Bulletin, 107*, 247-255.

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

Mislevy, R. J., & Wu, P.-K. (1996). *Missing responses and IRT ability estimation: omits, choice, time limits, and adaptive testing* (Research Report RR-96-30- ONR). Princeton, NJ: Educational Testing Service.

Muthén, B., & Shedden, K. (1999). Finite mixture modeling with mixture outcomes using the EM algorithm. *Biometrics, 55,* 463-469.

Muthén, B. O. (2001). Beyond SEM: General latent variable modeling. *Behaviormetrika, 29*, 81-117.

Muthén, L. K., & Muthén, B. O. (1998-2011). *Mplus user's guide* (6th ed.). Los Angeles, CA: Muthén & Muthén.

Ntzoufras, I. (2009). *Bayesian modeling using WinBUGS.* Hoboken, NJ: Wiley & Sons, Inc.

Parrila, R., Aunola, K., Leskinen, E., Nurmi, J.-E., & Kirby, J. R. (2005). Development of individual differences in reading: results from longitudinal studies in english and finnish. *Journal of Educational Psychology, 97*, 299-319.

Rasch, G. (1960/80). *Probabilistic models for some intelligence and attainment tests.* (Copenhagen, Danish Institute for Educational Research), expanded edition (1980) with fore-word and afterword by B. D. Wright. Chicago: The University of Chicago Press.

Reckase, M. D. (1979). Unifactor latent trait models applied to multifactor tests: Results and implications. *Journal of Educational Statistics, 4*, 207-230.

Redner, R. A., & Walker, H. F. (1984). Mixture densities, maximum likelihood and the EM algorithm. *Society of Industrial and Applied Mathematics Review, 26*, 195-239.

Rijmen, F., & De Boeck, P. (2005). A relationship between a between item multidimensional IRT model and the mixture Rasch model. *Psychometrika, 70*, 481-496.

Robitzsch, A., & Rupp, A. A. (2009). Impact of missing data on the detection of differential item functioning. *Educational and Psychological Measurement, 69*, 18-34.

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

Rubin, D. B. (1976). Inference and missing data. *Biometrika, 63,* 581-592.

Rumberger, R. W. (1987). High school dropouts: A review of issues and evidence. *Review of Educational Research, 57*, 101-121.

Schafer, J. L., & Graham, J. W. (2002). Missing data: Our view of the state of the art. *Psychological Methods, 7*, 147-177.

Schwarz, G. (1978). Estimating the dimension of a model. *Annals of Statistics, 6*, 461-464.

Sclove, L. S. (1987). Application of model-selection criteria to some problems in multivariate analysis. *Psychometrika, 52*, 333-343.

Segawa, E., Ngwe, J. E., Li, Y., & Flay, B. R. (2005). Evaluation of the effects of the Aban Aya youth project in reducing violence among African American adolescent males using latent class growth mixture modeling techniques. *Evaluation Review, 29*, 128-148.

Sinharay, S., & Holland, P. (2010). The missing data assumptions of the NEAT design and their implications for test equating. *Psychometrika, 75*, 309-327.

Spiegelhalter, D. J., Best, N. G., & Carlin, B. P. (1998). *Bayesian deviance, the effective number of para- meters, and the comparison of arbitrarily complex models.* Technical report, MRC Biostatistics Unit, Cambridge.

Spiegelhalter, D. J., Thomas, A., Best, N. G., & Lunn, D. (2003). WinBUGS 1.4 [Computer program]. Cambridge, UK: MRC Biostatistics Unit, Cambridge University.

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

Stephens, M. (2000). Dealing with label switching in mixture model. *Journal of the Royal Statistics Society: Series B, 62*, 795-809.

Stoolmiller, M., Kim, H. K., & Capaldi, D. M. (2005). The course of depressive symptoms in men from early adolescence to young adulthood: Identifying latent trajectories and early predictors. *Journal of Abnormal Psychology, 114*, 331-345.

Tofighi, D., & Enders, C. K. (2007). Identifying the correct number of classes in a growth mixture model. In G. R. Hancock (Ed.). *Mixture models in latent variable research* (pp. 317–341). Greenwich, CT: Information Age.

von Davier, M. (2001). WINMIRA [Computer program]. St. Paul, MN: Assessment Systems Corporation.

Wehlage, G. G., & Rutter, R. A. (1986). Dropping out: How much do schools contribute to the problem? *Teachers College Record, 87*, 374-392.

Wright, B. D., & Panchapakesan N. (1969). A procedure for sample-free item analysis. *Educational and Psychological Measurement, 29*, 23-48.

Appendix A

The Rate of Correct Identification for Simulation Conditions

Table A.1: R1110120010 (AIC)

	1 class	2 classes	3 classes
comp	81	12	7
mar	74	23	3
mnar	75	18	7

Table A.2: R1110120010 (BIC)

	1 class	2 classes	3 classes
comp	100	0	0
mar	100	0	0
mnar	100	0	0

Table A.3: R1110120020 (AIC)

	1 class	2 classes	3 classes
comp	88	9	3
mar	77	16	7
mnar	74	22	4

Table A.4: R1110120020 (BIC)

	1 class	2 classes	3 classes
comp	100	0	0
mar	100	0	0
mnar	100	0	0

Table A.5: R111060010 (AIC)

	1 class	2 classes	3 classes
comp	75	18	7
mar	80	14	6
mnar	80	16	4

Table A.6: R111060010 (BIC)

	1 class	2 classes	3 classes
comp	100	0	0
mar	100	0	0
mnar	100	0	0

Table A.7: R111060020 (AIC)

	1 class	2 classes	3 classes
comp	80	17	3
mar	81	13	6
mnar	81	10	9

Table A.8: R111060020 (BIC)

	1 class	2 classes	3 classes
comp	100	0	0
mar	98	0	2
mnar	100	0	0

Table A.9: R1130120010 (AIC)

	1 class	2 classes	3 classes
comp	49	30	21
mar	49	25	26
mnar	30	40	30

Table A.10: R1130120010 (BIC)

	1 class	2 classes	3 classes
comp	100	0	0
mar	100	0	0
mnar	100	0	0

Table A.11: R1130120020 (AIC)

	1 class	2 classes	3 classes
comp	63	23	14
mar	40	39	21
mnar	20	39	41

Table A.12: R1130120020 (BIC)

	1 class	2 classes	3 classes
comp	97	0	3
mar	100	0	0
mnar	100	0	0

Table A.13: R113060010 (AIC)

	1 class	2 classes	3 classes
comp	59	27	14
mar	50	24	26
mnar	47	32	21

Table A.14: R113060010 (BIC)

	1 class	2 classes	3 classes
comp	100	0	0
mar	100	0	0
mnar	100	0	0

Table A.15: R113060020 (AIC)

	1 class	2 classes	3 classes
comp	65	17	18
mar	50	27	23
mnar	39	33	28

Table A.16: R113060020 (BIC)

	1 class	2 classes	3 classes
comp	100	0	0
mar	100	0	0
mnar	100	0	0

Table A.17: R1210120010 (AIC)

	1 class	2 classes	3 classes
comp	0	69	31
mar	0	70	30
mnar	0	69	31

Table A.18: R1210120010 (BIC)

	1 class	2 classes	3 classes
comp	0	98	2
mar	0	100	0
mnar	0	97	3

Table A.19: R1210120020 (AIC)

	1 class	2 classes	3 classes
comp	0	62	38
mar	0	62	38
mnar	0	68	32

Table A.20: R1210120020 (BIC)

	1 class	2 classes	3 classes
comp	0	98	2
mar	0	96	4
mnar	0	96	4

Table A.21: R121060010 (AIC)

	1 class	2 classes	3 classes
comp	0	64	36
mar	0	71	29
mnar	0	72	28

Table A.22: R121060010 (BIC)

	1 class	2 classes	3 classes
comp	0	100	0
mar	0	97	3
mnar	0	100	0

Table A.23: R121060020 (AIC)

	1 class	2 classes	3 classes
comp	0	69	31
mar	0	74	26
mnar	0	64	36

Table A.24: R121060020 (BIC)

	1 class	2 classes	3 classes
comp	0	100	0
mar	0	100	0
mnar	0	100	0

Table A.25: R1230120010 (AIC)

	1 class	2 classes	3 classes
comp	0	50	50
mar	0	52	48
mnar	0	52	48

Table A.26: R1230120010 (BIC)

	1 class	2 classes	3 classes
comp	0	100	0
mar	0	97	3
mnar	0	98	2

Table A.27: R1230120020 (AIC)

	1 class	2 classes	3 classes
comp	0	60	40
mar	0	47	53
mnar	0	43	57

Table A.28: R1230120020 (BIC)

	1 class	2 classes	3 classes
comp	0	98	2
mar	0	100	0
mnar	0	98	2

Table A.29: R123060010 (AIC)

	1 class	2 classes	3 classes
comp	0	61	39
mar	0	61	39
mnar	0	49	51

Table A.30: R123060010 (BIC)

	1 class	2 classes	3 classes
comp	0	97	3
mar	0	98	2
mnar	0	100	0

Table A.31: R123060020 (AIC)

	1 class	2 classes	3 classes
comp	0	63	37
mar	0	56	44
mnar	0	58	42

Table A.32: R123060020 (BIC)

	1 class	2 classes	3 classes
comp	0	100	0
mar	0	100	0
mnar	0	98	2

Table A.33: R2110120010 (AIC)

	1 class	2 classes	3 classes
comp	0	69	31
mar	0	60	40
mnar	0	55	45

Table A.34: R2110120010 (BIC)

	1 class	2 classes	3 classes
comp	0	100	0
mar	0	98	2
mnar	0	100	0

Table A.35: R2110120020 (AIC)

	1 class	2 classes	3 classes
comp	0	77	23
mar	0	64	36
mnar	0	59	41

Table A.36: R2110120020 (BIC)

	1 class	2 classes	3 classes
comp	1	97	2
mar	1	97	2
mnar	2	97	1

Table A.37: R211060010 (AIC)

	1 class	2 classes	3 classes
comp	0	84	16
mar	0	79	21
mnar	0	70	30

Table A.38: R211060010 (BIC)

	1 class	2 classes	3 classes
comp	31	68	1
mar	31	68	1
mnar	27	72	1

Table A.39: R211060020 (AIC)

	1 class	2 classes	3 classes
comp	0	81	19
mar	0	83	17
mnar	0	81	19

Table A.40: R211060020 (BIC)

	1 class	2 classes	3 classes
comp	39	60	1
mar	38	61	1
mnar	29	70	1

Table A.41: R2130120010 (AIC)

	1 class	2 classes	3 classes
comp	0	21	79
mar	0	17	83
mnar	0	14	86

Table A.42: R2130120010 (BIC)

	1 class	2 classes	3 classes
comp	0	95	5
mar	0	96	4
mnar	0	100	0

Table A.43: R2130120020 (AIC)

	1 class	2 classes	3 classes
comp	0	26	74
mar	0	26	74
mnar	0	19	81

Table A.44: R2130120020 (BIC)

	1 class	2 classes	3 classes
comp	0	98	2
mar	0	97	3
mnar	0	100	0

Table A.45: R213060010 (AIC)

	1 class	2 classes	3 classes
comp	0	50	50
mar	0	43	57
mnar	0	42	58

Table A.46: R213060010 (BIC)

	1 class	2 classes	3 classes
comp	4	95	1
mar	3	96	1
mnar	4	95	1

Table A.47: R213060020 (AIC)

	1 class	2 classes	3 classes
comp	0	59	41
mar	0	53	47
mnar	0	43	57

Table A.48: R213060020 (BIC)

	1 class	2 classes	3 classes
comp	4	95	1
mar	7	92	1
mnar	7	92	1

Table A.49: R2210120010 (AIC)

	1 class	2 classes	3 classes
comp	0	0	100
mar	0	0	100
mnar	0	3	97

Table A.50: R2210120010 (BIC)

	1 class	2 classes	3 classes
comp	0	51	49
mar	0	61	39
mnar	0	55	45

Table A.51: R2210120020 (AIC)

	1 class	2 classes	3 classes
comp	0	0	100
mar	0	1	99
mnar	0	0	100

Table A.52: R2210120020 (BIC)

	1 class	2 classes	3 classes
comp	0	48	52
mar	0	76	24
mnar	0	64	36

Table A.53: R221060010 (AIC)

	1 class	2 classes	3 classes
comp	0	6	94
mar	0	4	96
mnar	0	9	91

Table A.54: R221060010 (BIC)

	1 class	2 classes	3 classes
comp	0	96	4
mar	0	96	4
mnar	0	96	4

Table A.55: R221060020 (AIC)

	1 class	2 classes	3 classes
comp	0	1	99
mar	0	7	93
mnar	0	3	97

Table A.56: R221060020 (BIC)

	1 class	2 classes	3 classes
comp	0	97	3
mar	0	98	2
mnar	0	96	4

Table A.57: R2230120010 (AIC)

	1 class	2 classes	3 classes
comp	0	0	100
mar	0	0	100
mnar	0	0	100

Table A.58: R2230120010 (BIC)

	1 class	2 classes	3 classes
comp	0	1	99
mar	0	1	99
mnar	0	1	99

Table A.59: R2230120020 (AIC)

	1 class	2 classes	3 classes
comp	0	0	100
mar	0	0	100
mnar	0	0	100

Table A.60: R2230120020 (BIC)

	1 class	2 classes	3 classes
comp	0	3	97
mar	0	5	95
mnar	0	3	97

Table A.61: R223060010 (AIC)

	1 class	2 classes	3 classes
comp	0	0	100
mar	0	0	100
mnar	0	0	100

Table A.62: R223060010 (BIC)

	1 class	2 classes	3 classes
comp	0	96	4
mar	0	99	1
mnar	0	96	4

Table A.63: R223060020 (AIC)

	1 class	2 classes	3 classes
comp	0	0	100
mar	0	0	100
mnar	0	0	100

Table A.64: R223060020 (BIC)

	1 class	2 classes	3 classes
comp	0	97	3
mar	0	98	2
mnar	0	98	2

Table A.65: T1110120010 (AIC)

	1 class	2 classes	3 classes
comp	84	13	3
mar	83	15	2
mnar	76	18	6

Table A.66: T1110120010 (BIC)

	1 class	2 classes	3 classes
comp	98	0	2
mar	100	0	0
mnar	100	0	0

Table A.67: T1110120020 (AIC)

	1 class	2 classes	3 classes
comp	91	8	1
mar	87	11	2
mnar	82	17	1

Table A.68: T1110120020 (BIC)

	1 class	2 classes	3 classes
comp	100	0	0
mar	100	0	0
mnar	100	0	0

Table A.69: T111060010 (AIC)

	1 class	2 classes	3 classes
comp	92	7	1
mar	77	21	2
mnar	86	12	2

Table A.70: T111060010 (BIC)

	1 class	2 classes	3 classes
comp	100	0	0
mar	100	0	0
mnar	100	0	0

Table A.71: T111060020 (AIC)

	1 class	2 classes	3 classes
comp	89	10	1
mar	84	12	4
mnar	81	15	4

Table A.72: T111060020 (BIC)

	1 class	2 classes	3 classes
comp	100	0	0
mar	100	0	0
mnar	100	0	0

Table A.73: T1130120010 (AIC)

	1 class	2 classes	3 classes
comp	75	14	11
mar	60	27	13
mnar	53	27	20

Table A.74: T1130120010 (BIC)

	1 class	2 classes	3 classes
comp	100	0	0
mar	100	0	0
mnar	100	0	0

Table A.75: T1130120020 (AIC)

	1 class	2 classes	3 classes
comp	90	5	5
mar	78	12	10
mnar	48	31	21

Table A.76: T1130120020 (BIC)

	1 class	2 classes	3 classes
comp	100	0	0
mar	100	0	0
mnar	100	0	0

Table A.77: T113060010 (AIC)

	1 class	2 classes	3 classes
comp	89	10	1
mar	68	27	5
mnar	60	30	10

Table A.78: T113060010 (BIC)

	1 class	2 classes	3 classes
comp	100	0	0
mar	100	0	0
mnar	100	0	0

Table A.79: T113060020 (AIC)

	1 class	2 classes	3 classes
comp	87	8	5
mar	73	19	8
mnar	64	25	11

Table A.80: T113060020 (BIC)

	1 class	2 classes	3 classes
comp	100	0	0
mar	100	0	0
mnar	100	0	0

Table A.81: T1210120010 (AIC)

	1 class	2 classes	3 classes
comp	0	87	13
mar	0	83	17
mnar	0	86	14

Table A.82: T1210120010 (BIC)

	1 class	2 classes	3 classes
comp	0	100	0
mar	0	100	0
mnar	0	98	2

Table A.83: T1210120020 (AIC)

	1 class	2 classes	3 classes
comp	0	88	12
mar	0	78	22
mnar	0	87	13

Table A.84: T1210120020 (BIC)

	1 class	2 classes	3 classes
comp	0	98	2
mar	0	100	0
mnar	0	97	3

Table A.85: T121060010 (AIC)

	1 class	2 classes	3 classes
comp	0	91	9
mar	0	84	16
mnar	0	84	16

Table A.86: T121060010 (BIC)

	1 class	2 classes	3 classes
comp	0	98	2
mar	2	97	1
mnar	1	97	2

Table A.87: T121060020 (AIC)

	1 class	2 classes	3 classes
comp	0	82	18
mar	0	61	39
mnar	0	88	12

Table A.88: T121060020 (BIC)

	1 class	2 classes	3 classes
comp	1	97	2
mar	43	56	1
mnar	3	94	3

Table A.89: T1230120010 (AIC)

	1 class	2 classes	3 classes
comp	0	79	21
mar	0	86	14
mnar	0	76	24

Table A.90: T1230120010 (BIC)

	1 class	2 classes	3 classes
comp	0	98	2
mar	0	98	2
mnar	0	98	2

Table A.91: T1230120020 (AIC)

	1 class	2 classes	3 classes
comp	0	78	22
mar	0	93	7
mnar	0	72	28

Table A.92: T1230120020 (BIC)

	1 class	2 classes	3 classes
comp	0	100	0
mar	0	100	0
mnar	0	100	0

Table A.93: T123060010 (AIC)

	1 class	2 classes	3 classes
comp	0	92	8
mar	0	91	9
mnar	0	91	9

Table A.94: T123060010 (BIC)

	1 class	2 classes	3 classes
comp	0	98	2
mar	0	97	3
mnar	0	100	0

Table A.95: T123060020 (AIC)

	1 class	2 classes	3 classes
comp	0	93	7
mar	0	0	100
mnar	0	85	15

Table A.96: T123060020 (BIC)

	1 class	2 classes	3 classes
comp	0	98	2
mar	0	0	100
mnar	0	98	2

Table A.97: T2110120010 (AIC)

	1 class	2 classes	3 classes
comp	93	3	4
mar	88	11	1
mnar	84	14	2

Table A.98: T2110120010 (BIC)

	1 class	2 classes	3 classes
comp	100	0	0
mar	100	0	0
mnar	100	0	0

Table A.99: T2110120020 (AIC)

	1 class	2 classes	3 classes
comp	90	7	3
mar	91	8	1
mnar	83	16	1

Table A.100: T2110120020 (BIC)

	1 class	2 classes	3 classes
comp	100	0	0
mar	100	0	0
mnar	100	0	0

Table A.101: T211060010 (AIC)

	1 class	2 classes	3 classes
comp	91	8	1
mar	95	3	2
mnar	89	10	1

Table A.102: T211060010 (BIC)

	1 class	2 classes	3 classes
comp	100	0	0
mar	98	0	2
mnar	100	0	0

Table A.103: T211060020 (AIC)

	1 class	2 classes	3 classes
comp	91	8	1
mar	100	0	0
mnar	93	6	1

Table A.104: T211060020 (BIC)

	1 class	2 classes	3 classes
comp	100	0	0
mar	100	0	0
mnar	100	0	0

Table A.105: T2130120010 (AIC)

	1 class	2 classes	3 classes
comp	86	11	3
mar	85	10	5
mnar	83	15	2

Table A.106: T2130120010 (BIC)

	1 class	2 classes	3 classes
comp	100	0	0
mar	100	0	0
mnar	100	0	0

Table A.107: T2130120020 (AIC)

	1 class	2 classes	3 classes
comp	95	4	1
mar	92	7	1
mnar	81	14	5

Table A.108: T2130120020 (BIC)

	1 class	2 classes	3 classes
comp	100	0	0
mar	100	0	0
mnar	100	0	0

Table A.109: T213060010 (AIC)

	1 class	2 classes	3 classes
comp	80	14	6
mar	86	10	4
mnar	85	11	4

Table A.110: T213060010 (BIC)

	1 class	2 classes	3 classes
comp	100	0	0
mar	100	0	0
mnar	100	0	0

Table A.111: T213060020 (AIC)

	1 class	2 classes	3 classes
comp	89	7	4
mar	93	5	2
mnar	84	15	1

Table A.112: T213060020 (BIC)

	1 class	2 classes	3 classes
comp	100	0	0
mar	100	0	0
mnar	100	0	0

Table A.113: T2210120010 (AIC)

	1 class	2 classes	3 classes
comp	0	84	16
mar	0	88	12
mnar	0	79	21

Table A.114: T2210120010 (BIC)

	1 class	2 classes	3 classes
comp	0	96	4
mar	0	97	3
mnar	0	92	8

Table A.115: T2210120020 (AIC)

	1 class	2 classes	3 classes
comp	0	88	12
mar	0	69	31
mnar	0	93	7

Table A.116: T2210120020 (BIC)

	1 class	2 classes	3 classes
comp	0	97	3
mar	0	90	10
mnar	0	91	9

Table A.117: T221060010 (AIC)

	1 class	2 classes	3 classes
comp	0	93	7
mar	0	86	14
mnar	0	89	11

Table A.118: T221060010 (BIC)

	1 class	2 classes	3 classes
comp	0	91	9
mar	0	88	12
mnar	0	93	7

Table A.119: T221060020 (AIC)

	1 class	2 classes	3 classes
comp	0	76	24
mar	0	61	39
mnar	0	85	15

Table A.120: T221060020 (BIC)

	1 class	2 classes	3 classes
comp	0	91	9
mar	9	79	12
mnar	2	83	15

Table A.121: T2230120010 (AIC)

	1 class	2 classes	3 classes
comp	0	93	7
mar	0	87	13
mnar	0	96	4

Table A.122: T2230120010 (BIC)

	1 class	2 classes	3 classes
comp	0	100	0
mar	0	93	7
mnar	0	96	4

Table A.123: T2230120020 (AIC)

	1 class	2 classes	3 classes
comp	0	94	6
mar	0	86	14
mnar	0	88	12

Table A.124: T2230120020 (BIC)

	1 class	2 classes	3 classes
comp	0	97	3
mar	0	97	3
mnar	0	100	0

Table A.125: T223060010 (AIC)

	1 class	2 classes	3 classes
comp	0	96	4
mar	0	92	8
mnar	0	97	3

Table A.126: T223060010 (BIC)

	1 class	2 classes	3 classes
comp	0	100	0
mar	0	98	2
mnar	0	100	0

Table A.127: T223060020 (AIC)

	1 class	2 classes	3 classes
comp	0	95	5
mar	0	95	5
mnar	0	96	4

Table A.128: T223060020 (BIC)

	1 class	2 classes	3 classes
comp	0	100	0
mar	0	100	0
mnar	0	100	0

Appendix B

R Code for Generating Complete and Missing data

```
nItems<-10                  # number of items
nSubj<-1000                # number of subjects
permissing<-0.10
nClass<-2                     # number of classes
pClass<-c(0.5,0.5)
meanTheta1<-0          # mean of normal for class 1
meanTheta2<-0          # mean of normal for class 2
sdTheta1<-1
sdTheta2<-1
beta1<-c(-2.7, -2.2, -1.5, -0.9, -0.3, 0.3, 1.0, 1.5, 2.1, 2.7)
# difficulty parameters for class 1
beta2<-c(2.6, 2.1, 1.5, 0.9, 0.3, 0.3, -1.0, -1.4, -2.0, -2.8)
# difficulty parameters for class 2
for (iter in 1:100) {
 ### generating complete data set ###
 response<-mat.or.vec(nSubj,nItems+1)
 for (row in 1:nSubj) {
  r<-runif(1,0,1)
  if (r<=0.5) {
   class<-1
   beta<-beta1
   theta<-rnorm(1,mean=meanTheta1,sd=sdTheta1)
  }
  else {
   class<-2
   beta<-beta2
   theta<-rnorm(1,mean=meanTheta2,sd=sdTheta2)
  }
  for (column in 1:nItems) {
   p=exp(theta-beta[column])/(1+exp(theta-beta[column]))
   r<-runif(1,0,1)
   if (r<=p) {
    response[row,column]<-1
   }
   else {
    response[row,column]<-0
   }
```

```
 }
 response[row,nItems+1]<-class
}
### end of generating complete data set ###
### generating MAR data set ###
temp<-response[,1:nItems]
sum<-mat.or.vec(nSubj,1)
group<-mat.or.vec(nSubj,1)
missing<-mat.or.vec(nSubj,1)
mar<-cbind(response,sum,group,missing)
mnar<-cbind(response,sum,group,missing)
for (row in 1:nSubj) {
 c<-mar[row,nItems+1]
 if (c==1) {
  s<-sum(mar[row,1:6])
  if (s<=1) {
   mar[row,nItems+3]<-1
  }
  else if (s>=2 & s<=4) {
   mar[row,nItems+3]<-2
  }
  else if (s>=5 & s<=7) {
   mar[row,nItems+3]<-3
  }
  else if (s>=8) {
   mar[row,nItems+3]<-4
  }
 }
 else if (c==2) {
  s<-sum(mar[row,5:10])
  if (s<=1) {
   mar[row,nItems+3]<-1
  }
  else if (s>=2 & s<=4) {
   mar[row,nItems+3]<-2
  }
  else if (s>=5 & s<=7) {
   mar[row,nItems+3]<-3
  }
  else if (s>=8) {
   mar[row,nItems+3]<-4
  }
 }
}
count<-1
while (count<=nSubj*nItems*permissing) {
 for (row in 1:nSubj) {
  if (count<=nSubj*nItems*permissing) {
```

```
86      g<-mar[row,nItems+3]
87      c<-mar[row,nItems+1]
88      if (c==1) {
89       if (g==1) {
90        for (i in 1:4) {
91         r<-runif(1,0,1)
92         if (r<0.6) {
93          if (mar[row,nItems-3+i]!=9) {
94           mar[row,nItems-3+i]<-9
95           mar[row,nItems+4]<-1
96           # 1 = missing, 0 = not missing
97           count<-count+1
98          }
99         }
100       }
101      }
102      else if (g==2) {
103       for (i in 1:4) {
104        r<-runif(1,0,1)
105        if (r<0.4) {
106         if (mar[row,nItems-3+i]!=9) {
107          mar[row,nItems-3+i]<-9
108          mar[row,nItems+4]<-1
109          # 1 = missing, 0 = not missing
110          count<-count+1
111         }
112        }
113       }
114      }
115      else if (g==3) {
116       for (i in 1:4) {
117        r<-runif(1,0,1)
118        if (r<0.2) {
119         if (mar[row,nItems-3+i]!=9) {
120          mar[row,nItems-3+i]<-9
121          mar[row,nItems+4]<-1
122          # 1 = missing, 0 = not missing
123          count<-count+1
124         }
125        }
126       }
127      }
128     }
129     else if (c==2) {
130      if (g==1) {
131       for (i in 1:4) {
132        r<-runif(1,0,1)
133        if (r<0.6) {
```

```
134           if (mar[row,i]!=9) {
135            mar[row,i]<-9
136            mar[row,nItems+4]<-1
137            # 1 = missing, 0 = not missing
138            count<-count+1
139           }
140          }
141         }
142        }
143        else if (g==2) {
144         for (i in 1:4) {
145          r<-runif(1,0,1)
146          if (r<0.4) {
147           if (mar[row,i]!=9) {
148            mar[row,i]<-9
149            mar[row,nItems+4]<-1
150            # 1 = missing, 0 = not missing
151            count<-count+1
152           }
153          }
154         }
155        }
156        else if (g==3) {
157         for (i in 1:4) {
158          r<-runif(1,0,1)
159          if (r<0.2) {
160           if (mar[row,i]!=9) {
161            mar[row,i]<-9
162            mar[row,nItems+4]<-1
163            # 1 = missing, 0 = not missing
164            count<-count+1
165           }
166          }
167         }
168        }
169       }
170     }
171     else break
172    }
173   }
174   ### generating MNAR data set ###
175   count<-1
176   while (count<=nSubj*nItems*permissing) {
177    for (row in 1:nSubj) {
178     if (count<=nSubj*nItems*permissing) {
179      c<-mnar[row,nItems+1]
180      if (c==1) {
181       for (i in 1:4) {
```

```
      if (mnar[row,nItems-3+i]!=9) {
       if (mnar[row,nItems-3+i]==1) {
        r<-runif(1,0,1)
        if (r<0.3) {
         mnar[row,nItems-3+i]<-9
         mnar[row,nItems+4]<-1
         # 1 = missing, 0 = not missing
         count<-count+1
        }
       }
       else if (mnar[row,nItems-3+i]==0) {
        if (r<0.1) {
         mnar[row,nItems-3+i]<-9
         mnar[row,nItems+4]<-1
         # 1 = missing, 0 = not missing
         count<-count+1
        }
       }
      }
     }
     else if (c==2) {
      for (i in 1:4) {
       if (mnar[row,i]!=9) {
        if (mnar[row,i]==1) {
         r<-runif(1,0,1)
         if (r<0.3) {
          mnar[row,i]<-9
          mnar[row,nItems+4]<-1
          # 1 = missing, 0 = not missing
          count<-count+1
         }
        }
        else if (mnar[row,i]==0) {
         if (r<0.1) {
          mnar[row,i]<-9
          mnar[row,nItems+4]<-1
          # 1 = missing, 0 = not missing
          count<-count+1
         }
        }
       }
      }
     }
    }
    else break
   }
  }
```

```
230  # generating output file for complete data
231  # response : item + 1 = class
232  comp<-response[,1:nItems]
233  filename1<-paste("comp1pl2c",iter,".txt",sep="")
234  # file for item calibration
235  filename2<-paste("comp1pl2c_o",iter,".txt",sep="")
236  # original file containing other information
237  write.table(comp,file=filename1,sep='\t',row.names=FALSE,
238  +col.names=FALSE)
239  write.table(response,file=filename2,sep='\t',row.names=FALSE,
240  +col.names=FALSE)
241  # generating output file for mar data
242  # mar : item + 1 = class, item + 2 = sum, item + 3 = group,
243  +item + 4 = missing
244  # 1 = missing, 0 = not missing
245  resmar<-mar[,1:nItems]
246  filename3<-paste("mar1pl2c",iter,".txt",sep="")
247  filename4<-paste("mar1pl2c_o",iter,".txt",sep="")
248  write.table(resmar,file=filename3,sep='\t',row.names=FALSE,
249  +col.names=FALSE)
250  write.table(mar,file=filename4,sep='\t',row.names=FALSE,
251  +col.names=FALSE)
252  # generating output file for mnar data
253  # mnar : item + 1 = class, item + 4 = missing
254  # 1 = missing, 0 = not missing
255  resmnar<-mnar[,1:nItems]
256  filename5<-paste("mnar1pl2c",iter,".txt",sep="")
257  filename6<-paste("mnar1pl2c_o",iter,".txt",sep="")
258  write.table(resmnar,file=filename5,sep='\t',row.names=FALSE,
259  +col.names=FALSE)
260  write.table(mnar,file=filename6,sep='\t',row.names=FALSE,
261  +col.names=FALSE)
262 }
```

Appendix C

R Code for Mplus Automation Package

C.1 The Rasch Model

```
[[init]]
iterators = replications;
replications= 1:100;
filename = "replications[[replications]].inp";
outputDirectory = "C:/mplus/R111060010/comp/C2";
[[/init]]

TITLE:  Code for the Rasch model
(IRT) analysis with binary latent class and factor indicators
DATA: FILE = "C:\R111060010\comp\comp1pl1c[[replications]].txt";
VARIABLE: NAMES = u1-u10;
          CATEGORICAL = u1-u10;
          CLASSES = c (2);
          MISSING ARE ALL (9);
ANALYSIS: TYPE = MIXTURE;
          ALGORITHM = INTEGRATION;
MODEL:    %OVERALL%
          f BY u1-u10;
          [f@0];
          %c#1%
          f BY u1-u10@1;
          f@1;
          [u1$1-u10$1];
          %c#2%
          f BY u1-u10@1;
          f@1;
          [u1$1-u10$1];
Savedata: file is save[[replications]].txt ;
          save is cprob;
          format is free;
OUTPUT:   TECH1 TECH8 TECH11;
```

C.2 2PL

```
[[init]]
iterators = replications;
replications= 1:100;
filename = "replications[[replications]].inp";
outputDirectory = "C:/mplus/T111060010/comp/C2";
[[/init]]

TITLE:  Code for the 2PL
(IRT) analysis with binary latent class and factor indicators
DATA: FILE = "C:\T111060010\comp\comp1pl1c[[replications]].txt";
VARIABLE: NAMES = u1-u10;
          CATEGORICAL = u1-u10;
          CLASSES = c (2);
      MISSING ARE ALL (9);
ANALYSIS: TYPE = MIXTURE;
          ALGORITHM = INTEGRATION;
MODEL:    %OVERALL%
          f BY u1-u10;
          [f@0];
          %c#1%
          f BY u1-u10;
          f@1;
          [u1$1-u10$1];
          %c#2%
          f BY u1-u10;
          f@1;
          [u1$1-u10$1];
Savedata: file is save[[replications]].txt ;
          save is cprob;
          format is free;
OUTPUT:   TECH1 TECH8 TECH11;
```