

2013

Modeling Mixed Unfolding/Monotone Dichotomous Item Exams

Na Yang

University of South Carolina - Columbia

Follow this and additional works at: <https://scholarcommons.sc.edu/etd>



Part of the [Statistics and Probability Commons](#)

Recommended Citation

Yang, N.(2013). *Modeling Mixed Unfolding/Monotone Dichotomous Item Exams*. (Doctoral dissertation). Retrieved from <https://scholarcommons.sc.edu/etd/2682>

This Open Access Dissertation is brought to you by Scholar Commons. It has been accepted for inclusion in Theses and Dissertations by an authorized administrator of Scholar Commons. For more information, please contact dillarda@mailbox.sc.edu.

MODELING MIXED UNFOLDING/MONOTONE DICHOTOMOUS ITEM EXAMS

By

Na Yang

Bachelor of Economics
Shanghai University of Finance and Economics, 2002

Master of Science
University of South Carolina, 2009

Submitted in Partial Fulfillment of the Requirements
for the Degree of Doctor of Philosophy in
Statistics
College of Arts and Sciences
University of South Carolina
2012

Accepted by:

Brian Habing, Chair

John Grego, Committee Member

Xiaoyan Lin, Committee Member

Lee Van Horn, Outside Committee Member

Lacy Ford, Vice Provost and Dean of Graduate Studies

© Copyright by Na Yang, 2012
All Rights Reserved.

DEDICATION

I dedicate my dissertation work to my family. You are my love and reason for being. This is dedicated to my loving parents, Xiulian Wang and Yongming Yang for millions of support and encouragement, to my husband, Jianfeng Li for not letting me quit and always being with me, to my precious children, Angus Li and Lily Yang, for enjoying the wonder of life with me, to my grandparents for showing me the meaning of life, to my parents-in-law, Aiying Li and Dongming Li for their understanding and kindness, to my loved uncles, aunts and my little brother.

ACKNOWLEDGMENTS

This dissertation marks the end of my own 7 years journey to the doctorate. I could not get to this point on my own without help along the way.

Special thanks go to my advisor and dissertation chair Dr. Brian Habing for his many years of support, guidance and most of all patience throughout the whole process. Thank you Dr. Don Edwards, Dr. John Grego, Dr. Lee Van Horn and Dr. Xiaoyan Lin for your generosity with your expertise, precious time, tremendous help and support.

I would like to acknowledge and thank all the professors in the department, who showed me the wonderland of Statistics, particularly, to Dr. Edsel Pena, Dr. David Hitchcock, and Dr. Joshua Tebbs for their wonderful classes. Thanks to all of the staff in the department for their continuous support. Thanks also to all my friends and colleagues.

Finally to my dear husband, Jianfeng Li. Your love, support and encouragement lead me to my dream. I love you.

This material is based upon work supported by the National Science Foundation under Grant No. *SES* – 0720072. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

ABSTRACT

Item response theory (IRT) is widely applied to analyze educational and psychological assessments. Readily available IRT implementations allow for two common types of models: monotone models used for dominance scales (Guttman 1950; Rasch 1960/1980; Birnbaum 1968; Mokken 1971) and unfolding models used for proximity scales (Coombs, 1964; Andrich, 1996; Roberts, Donoghue and Laughlin, 2000).

When an exam contains items following both types of models, there is currently no method to distinguish the item types, estimate their characteristics, or estimate the examinee characteristics. Thus, there is no existing methodology to simultaneously analyze items like “At a minimum, I am in favor of the economically disadvantaged receiving publicly funded private school vouchers” and “I am in favor of publicly funded private school vouchers only for the economically disadvantaged” on the same survey. The former is a monotone increasing item (all those favoring some sort of voucher are likely to agree) whereas the latter is an unfolding item (one could disagree because they were against all vouchers, or because they were in favor of everyone receiving them). This situation forces analysts either to choose subscales of only one item type or risk incorrect and misleading results. The goal for this study is to find a reasonable means to solve this problem in the dichotomous (or binary) case when unidimensionality holds.

The first portion of this study (Chapter 2) attempts to identify these two item types in the mixed unfolding/monotone items exam (we call this a mixed exam). Two methods are discussed: the manifest monotonicity test (Sijtsma and Molenaar, 2002) and the p-value/biserial method based on classical item statistics. Simulation shows

that the manifest monotonicity test does a good job of separating these two types when the items are located in the middle of the ability range. However, this method does not work for the extreme items (including higher or lower location values for both item types). The p-value/biserial method is limited to lower location values.

In the second portion of this study (Chapters 3, 4, and 5), a mixed model is proposed which combines both model types. The marginalized Bayes modal estimation algorithm is implemented to estimate the model parameters. The estimates for each parameter in the new model are also discussed. The new model and marginalized Bayes modal estimation can successfully identify the unfolding items and gives very good estimates for the item parameters

CONTENTS

DEDICATION	iii
ACKNOWLEDGMENTS	iv
ABSTRACT	v
LIST OF TABLES	ix
LIST OF FIGURES	xii
CHAPTER 1. TEST THEORY	1
1.1. Classical Test Theory	2
1.2. Item Response Theory	5
1.3. Mokken Scale Analysis	10
1.4. Mean/Mean Linking	11
1.5. Overview Of the Subsequent Chapters	13
CHAPTER 2. IDENTIFYING ITEM TYPES BY THE MANIFEST MONOTONICITY TEST AND P-VALUE/BISERIAL METHOD	14
2.1. The Manifest Monotonicity Test	14
2.2. The Simulation Study for the Manifest Monotonicity Test	17
2.3. The P-value/biserial Method	32
2.4. The Simulation Study for the P-value/Biserial Method	33
2.5. Conclusion	36
CHAPTER 3. THE MIXTURE MODEL WITH THE MARGINALIZED BAYES MODAL ESTIMATION PROCEDURE	39
3.1. The Mixture Model	39
3.2. The Marginalized Bayes Modal Estimation	40

3.3. The Log-likelihood Component	42
CHAPTER 4. THE INITIAL ESTIMATES OF PARAMETERS IN MIXTURE MODEL AND SIMULATION STUDY	48
4.1. The Formulas for Initial Estimates	48
4.2. The Simulation Study for the Initial Value Of π	51
4.3. Classification Trees for the P-value/Biserial Method	56
4.4. The Study about Initial Estimated π_i	61
CHAPTER 5. SIMULATION STUDY FOR MIXTURE MODEL	65
5.1. The Mixture Model Estimation Simulation Study	65
5.2. Sign Reversal Problem	69
5.3. Question of Estimated π for the Monotone Items	69
5.4. Simulation Study for All Three Methods	77
CHAPTER 6. SUMMARY AND FUTURE WORK	90
BIBLIOGRAPHY	93
APPENDIX A: GAUSS-HERMITE INTEGRATION	96
APPENDIX B: EQUATION FOR DERIVATIVE	97
APPENDIX C: ITEM PARAMETERS FOR CHAPTER 2	100
APPENDIX D: R CODE FOR CHAPTER 2	101
APPENDIX E: R CODE FOR MARGINALIZED BAYES MODAL ESTIMATION . .	112
APPENDIX F: R CODE FOR MEAN/MEAN LINKING	131
APPENDIX G: ERROR RATE REPORT AND CLASSIFICATION TREE FOR SECTION 4.3	136

LIST OF TABLES

Table 2.1	The Example for the Manifest Monotonicity	16
Table 2.2	The Percentage of Time Each Item is Identified as Non-monotone. (1 – 10 monotone, 11 – 20 unfolding)	19
Table 2.3	Item Parameters for Items 8, 9, 15 and 16	19
Table 2.4	The Studies Summary	21
Table 2.5	Manifest Monotonicity Test Study Report for the Monotone Items. Reported values are the proportion identified as unfolding . . .	22
Table 2.6	The Manifest Monotonicity Test Study Report for the Monotone Items (Cont.)	23
Table 2.7	The Manifest Monotonicity Test Study Report for the Unfolding Items.	24
Table 2.8	The Manifest Monotonicity Test Study Report for the Unfolding Items (Cont.)	25
Table 2.9	The Manifest Monotonicity Test Study Report for the Monotone Item with $minisize = 100$	28
Table 2.10	The Manifest Monotonicity Test Study Report for the Unfolding Item with $minisize = 100$	29
Table 2.11	The Manifest Monotonicity Test Study Report for the Monotone Item with $minvi = 0.06$	30
Table 2.12	The Manifest Monotonicity Test Study Report for the Unfolding Item with $minvi = 0.06$	31

Table 2.13	The Formula to Calculate the <i>estimatedb</i> by Using the <i>P – value</i> and Biserial Correlation.	33
Table 4.1	Parameter Estimation for the Monotone Items when the Initial $\pi = 0$	63
Table 4.2	Parameter Estimation for the Monotone Items when the Initial $\pi = 1$	63
Table 4.3	Parameter Estimation for the Unfolding Items when the Initial $\pi = 0$	64
Table 4.4	Parameter Estimation for the Unfolding Items when the Initial $\pi = 1$	64
Table 5.1	Parameter Estimation for the Monotone Items	67
Table 5.2	Parameter Estimation for the Unfolding Items	68
Table 5.3	Parameter Estimation for Setting 1	71
Table 5.4	Parameter Estimation for Setting 2	72
Table 5.5	Parameter Estimation for Setting 3	73
Table 5.6	Parameter Estimation for Setting 4	75
Table 5.7	Parameter Estimation for Setting 5	76
Table 5.8	Item parameters and Estimates from Mixture model for Simulation Run 1	78
Table 5.9	Item parameters and Estimates from Mixture model for Simulation Run 2	79
Table 5.10	Item parameters and Estimates from Mixture model for Simulation Run 3	80
Table 5.11	Item parameters and Estimates from Mixture model for Simulation Run 4	81

Table 5.12	Item parameters and Estimates from Mixture model for Simulation Run 5	82
Table 5.13	Item parameters and Estimates from Mixture model for Simulation Run 6	83
Table 5.14	Item parameters and Estimates from Mixture model for Simulation Run 7	84
Table 5.15	Item parameters and Estimates from Mixture model for Simulation Run 8	85
Table 5.16	Item parameters and Estimates from Mixture model for Simulation Run 9	86
Table 5.17	Item parameters and Estimates from Mixture model for Simulation Run 10	87
Table 5.18	Mean and Standard Deviation of Estimators for Simulation Run 1 to Simulation Run 10	88
Table 5.19	Proportion of “M”, “U”, “C” for Each Indicators Across Simulation Run 1 to Simulation Run 10	89
Table 6.1	Error Rate with N=500 for the Original Data	136
Table 6.2	Error Rate with N=1000 for the Original Data	136
Table 6.3	Error Rate with N=2000 for the Original Data	136
Table 6.4	Error Rate with N=4000 for the Original Data	136
Table 6.5	Error Rate with N=500 for the Transformed Data	138
Table 6.6	Error Rate with N=1000 for the Transformed Data	138
Table 6.7	Error Rate with N=2000 for the Transformed Data	138
Table 6.8	Error Rate with N=4000 for the Transformed Data	142

LIST OF FIGURES

Figure 1.1	IRF for Monotone Model and Unfolding Model	6
Figure 2.1	IRF for 10 GGUM items: “5” indicates item 15 and “6” indicates item 16	19
Figure 2.2	The P-value/Biserial Method for 62 Items under Different Scenarios 35	
Figure 2.3	The P-value/Biserial Method for 20 Items	36
Figure 2.4	The P-value/Biserial Method for 9 Simulations	37
Figure 2.5	The P-value/Biserial Method for the Whole Transformed Data Set	38
Figure 4.1	Theory Result by Using the P-value/Biserial Method	52
Figure 4.2	P-value vs Estimated b Value for the Original Data Set	53
Figure 4.3	P-value vs Estimated b Value for the Transformed Data Set . .	54
Figure 4.4	Classification Tree for $N = 500$ with the Original and the Transformed Data	59
Figure 4.5	Classification Tree for $N = 1000$ with the Original Data and the Transformed Data	60
Figure 4.6	Classification Tree for $N = 2000$ with the Original Data and the Transformed Data	60
Figure 4.7	Classification Tree for $N = 4000$ with the Original Data and the Transformed Data	61
Figure 5.1	IRFs for a Few Items under Setting 1	71

Figure 5.2	IRFs for a Few Items under Setting 2	72
Figure 5.3	IRFs for a Few Items under Setting 3	73
Figure 5.4	IRFs for a Few Items under Setting 4	75
Figure 5.5	IRFs for a Few Items under Setting 5	76
Figure 6.1	Classification Tree for $N = 500$ with the Transformed Data . .	137
Figure 6.2	Classification Tree for $N = 1000$ with the Transformed Data . .	139
Figure 6.3	Classification Tree for $N = 2000$ with the Transformed Data . .	140
Figure 6.4	Classification Tree for $N = 4000$ with the Transformed Data . .	141

CHAPTER 1

TEST THEORY

Multiple choice educational tests are designed to have one correct response and several incorrect but distracting responses. If the examinees get an item correct, a 1 is recorded and a 0 otherwise. These types of binary questions are defined as dichotomous items and they are the types studied throughout this research. Compared to dichotomous items, Polytomous item types include those with responses ranging from 1 =strongly disagree, 2 =disagree, 3 =neutral, 4 =agree, to 5 =strong agree, or partial credit educational items with grades on a scale 0 to 5, where 0 means totally wrong and 5 means perfect. Discoveries in the dichotomous case should provide insight into applying our findings to the polytomous case. Once the scores are collected, a test can be analyzed based on the individuals' responses and the item statistics. Classical Test Theory and Item Response Theory are the two different measurement theory frameworks which the test analysis is based upon.

The main purpose of this research is to model unfolding and monotone IRT items simultaneously. Monotone increasing items are those that have stronger agreement directly related to the respondent having a higher value on the underlying latent trait. Items where a higher score is related to a lower value on the latent trait are monotone decreasing items. Unfolding items are the ones where the relationship between the strength of the response and the value of the underlying latent trait is non-monotonic, depending instead on the distance between the location of the item and the respondent on the same latent continuum. Before discussing the methods and simulation study, classical test theory (CTT) and item response theory (IRT) will be introduced in this chapter.

1.1. CLASSICAL TEST THEORY

Spearman's (1904) ideas of classical test theory (CTT) served as the major approach in educational measurement for most of the 20th century. CTT is built under the weak assumption that the observed test score consists of the examinees true score and an independent measurement error. This assumption can be easily met by most tests and has been applied to a wide variety of test development and test score analysis (Hambleton, 1985).

CTT introduces three basic measurement concepts (1) observed test score, (2) true score when no errors exist in measurement, and (3) error score. Classical test analysis believes that the difference between true score and the observed score is the measurement error, and that:

$$X = T + E \tag{1.1}$$

where

X is the observed exam score;

T is true score which can be defined as examinee's expected test score over repeated administrations of the test;

E is the random error with mean 0 that is uncorrelated with T or other error scores.

While CTT mainly focuses on the test level information, item statistics such as item difficulty, point biserial correlation and biserial correlation supplement the classical test theory framework by allowing for analysis at the item level. Their use is sometime referred to as classical item analysis. If a test is dichotomously scored, the item difficulty p_i (p-value) is the proportion of examinees who answered the i th item correctly:

$$p_i = \frac{\sum_{j=1}^N X_{ij}}{N} \tag{1.2}$$

where

X_{ij} is the score for the i th item and j th examinee. X_{ij} is in (0 or 1) under the dichotomous setting;

N is the total number of examinees.

The p-value is an inverse indicator of item difficulty (Wood 1960); a higher p-value means a lower difficulty of the item.

Point biserial correlation and biserial correlation are two statistics used to measure the strength of the relationship between the single items and the total score on a test, also called the item discrimination. An item having a higher discrimination means this item discriminates between examinees of differing true scores better than the item with lower discrimination.

Point biserial correlation (ρ_{pbis}) is the correlation between a dichotomous variable and a continuous variable (here, the observed score X treated as a continuous variables). For dichotomous items, it can be written:

$$\rho_{pbis} = \frac{\mu_+ - \mu_X}{\sigma_X} \sqrt{(p_i/q_i)} \quad (1.3)$$

where

μ_+ is the mean of X for those examinees who answer the item correctly;

μ_X is the mean of entire score X and σ_X is the standard deviation of X ;

p_i is the item difficulty;

q_i is $1 - p_i$.

The point biserial correlation is positive if large values of ability are associated with $X_i = 1$ and small values of ability are associated with $X_i = 0$.

If we assume that a normally distributed latent trait underlies the dichotomous response, a biserial correlation can be derived. Biserial correlation (ρ_{bis}) is the correlation between a dichotomous variable and a continuous variable (latent trait) where it is calculated by assuming the dichotomous variable is created by dichotomizing

a normal random variable. This statistic, first derived by Pearson (1909) can be computed by the formula (Crocker & Algina 1986, Lord & Novick 1968):

$$\rho_{bis} = \frac{\mu_+ - \mu_X}{\sigma_X} (p_i / \gamma_i) \quad (1.4)$$

where

μ_+ is the mean of X for those examinees who answer the item correctly;

μ_X is the mean of X and σ_X is the standard deviation of X ;

p_i is the item difficulty;

γ_i is the X ordinate of the standard normal curve at the z-score associated with the p_i value for the i th item.

Compared with the point biserial correlation, the biserial correlation is less sensitive to the item difficulty (Crocker & Algina 1986). Mathematically, the relationship between biserial and point biserial correlations is

$$\rho_{bis} = \frac{\sqrt{p_i q_i}}{\gamma_i} \rho_{pbis} \quad (1.5)$$

Test reliability is one of the fundamental concepts in test theory. It measures the consistency or reproducibility of an examinee's performance on the test. If one examinee took a test with higher reliability on two occasions, you would be very likely to reach the same conclusion about the examinee's performance on both times. There are four basic forms of assessing test score reliability: test/retest, parallel form, split-half, and internal consistency (Crocker & Algina 1986). In test/retest analysis, the same test is given to the same group of examinee over a designed period. Then the score from different time periods are correlated. With parallel forms, each student takes two different forms of the exam that result in the the same true score and the same observed score variance for both forms for every individual. The two scores are again correlated. The split-half reliability is calculated by dividing the test into two parts and correlating the scores from the parts. The correlation is then adjusted for

the reduced exam length. Then there is coefficient α :

$$\alpha = \frac{N}{N-1} (1 - (\sum \sigma_i^2 / \sigma_X^2)) \quad (1.6)$$

where

N is the total number of items;

σ_i^2 is the variance of scores on item i ;

σ_X^2 is the variance of total test score X .

This is the average of all split half reliabilities (Crocker & Algina 1986).

One of the advantages of CTT is that it has a weak assumption that is relatively easy to meet in the real data. However, Hambleton (1995) discusses several shortcomings of CTT. First is that both item difficulty and item correlation statistics are dependent upon the particular examinees sample. Moreover, test reliability is defined in terms of parallel forms which are difficult to achieve in practice. Further, CTT presumes that the variance of errors of measurement is the same for all examinees. These shortcomings limit CTT's application in many cases.

1.2. ITEM RESPONSE THEORY

Item Response Theory (IRT) is the modern test theory currently used for modeling large scale testing data. One benefit of IRT over CTT is that it overcomes the major weakness of CTT's sample dependency. In contrast, it is based on strong assumptions. IRT focuses on measuring some latent trait or traits, which are possessed by the examinees, usually ability. When the model fits, it produces item statistics which can be interpreted independently from the sample test-takers or other items. IRT uses a mathematical model to express the probability of an examinee answering an item correctly as a function of the examinee's ability (θ).

Assuming that the latent trait θ is unidimensional and item responses are locally independent (conditionally independent given θ), readily available IRT implementations are primarily for monotone increasing items. A higher score on a monotone

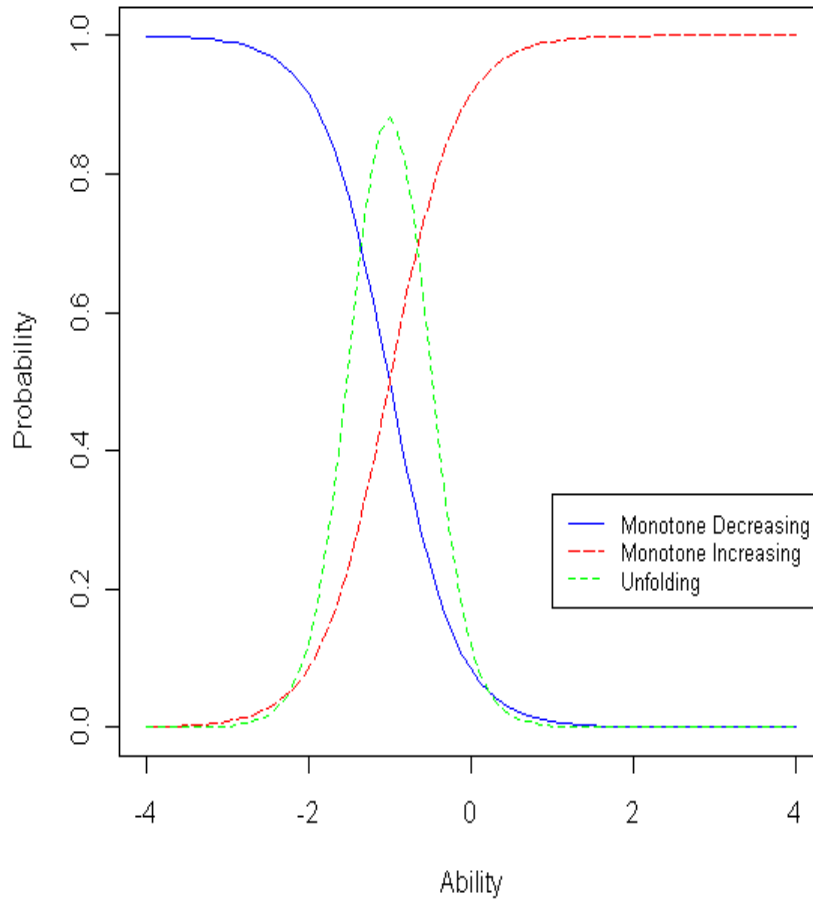


FIGURE 1.1. IRF for Monotone Model and Unfolding Model

increasing items corresponds to a higher value on the underlying latent trait. The item responses and the underlying latent trait (θ) are linked by the item response functions (IRFs). The item response function for these items is monotone increasing (see Figure 1.1). It is also possible to have monotone decreasing items where higher scores are related to lower values of latent trait (see Figure 1.1). Dichotomous monotone item exams are commonly analyzed by using the Rasch model (Rasch, 1964), two parameter logistic model (2PL) or three parameter logistic model (3PL; Birnbaum, 1968). In this study, the 2PL model is applied to analyze all the monotone items.

The particular logistic model is often specified as a special case of Birnbaum's (1968) three-parameter logistic (3PL) model:

$$P(X_{ij} = k|\theta, (a_i, b_i, c_i)) = c_i + (1 - c_i) \frac{\exp(Da_i(\theta_j - b_i))}{1 + \exp(Da_i(\theta_j - b_i))} \quad (1.7)$$

where

X_{ij} is the response of subject $j \in \{1, 2, \dots, N\}$ to item $i \in \{1, 2, \dots, I\}$;

a_i : the slope or discrimination parameter of item i ;

b_i : the location or difficulty parameter of item i ;

c_i : lower asymptote or guessing parameter of item i ;

θ_j : is j th examinee's ability.

A scaling parameter, $D = 1.7$, is often included to make the scale of the logistic form similar to that of the normal ogive model. Setting c_i to zero produces the two-parameter logistic model (2PL). Setting a_i to be negative produces a monotone decreasing model. Setting both c_i to zero and fixing a_i to be constant across items produces the one-parameter logistic (1PL) or Rasch model.

Instruments which are designed to measure attitudes or other latent constructs may, however, be composed of unfolding (or proximity) items. For these items, the relationship between the strength of the response and the value of the underlying latent trait is non-monotonic, depending instead on the distance between the location of the item and the respondent on the same latent continuum. The level of response is highest for items that most closely match the subject's level of the latent trait. The item response function for this type of items is single peaked (see Figure 1.1).

The unfolding model is defined by setting the probability of giving a positive response to an item to be a decreasing function of the distance (i.e., the absolute difference τ_{ik}) between a subject's location (e.g. ability θ_j) and the item location (e.g. difficulty δ_i). Several unfolding IRT models have been proposed (e.g. Andrich, 1988; Hoijsink, 1990). The generalized graded unfolding model (GGUM; Roberts et al. 2000) is constructed by assuming that an unobserved generalized partial credit model

(GPCM, Muraki 1992) models the subject' level of agreement within the question. The GPCM is defined as

$$P(X_{ij} = k|\theta_j, \varphi_i) = \frac{\exp(\alpha_i[k(\theta_j - \delta_i)] - \sum_{v=0}^k \tau_{iv})}{\sum_{w=0}^K \exp(\alpha_i[w(\theta_j - \delta_i)] - \sum_{v=0}^w \tau_{iv})} \quad (1.8)$$

where

$$\varphi_i = \{\alpha_i, \delta_i, \tau_{ik}\};$$

X_{ij} is the response of subject $j \in \{1, 2, \dots, N\}$ to item $i \in \{1, 2, \dots, I\}$;

θ_j : is j th examinees ability;

$k \in \{0, 1, 2, \dots, K\}$, where 0 corresponds to the strongest possible negative response and K corresponds to the strongest positive response;

K is the number of subjective response categories minus 1;

α_i is the discrimination of item i ;

δ_i : is the location parameter of item i ;

τ_{iv} is the location of the v th subjective response categories threshold on the attitude continuum relative to the location of the i th item, subjected to the constraint that $\sum_{w=0}^K \tau_{iw} = 0$ (Roberts, 2000). Note that the value of τ_{i0} is defined to be 0 in Equation 1.8.

If there are four observable response categories: strongly disagree, disagree, agree and strongly agree, there are eight subjective response categories (SRC) behind the associated probability functions in the GPCM: strongly disagree from below, strongly disagree from above, disagree from below, disagree from above, agree from below, agree from above, strongly agree from below and strongly agree from above. There are two unique SRCs corresponding to each of the four observed response categories (ORCs) based on the signed distance between a persons attitude position and the location of the item ($\theta_j - \delta_i$). The agreement and disagreement categories from above and from below (e.g. “disagree from below” and “disagree from above”) are combined into simply “disagree” (SRCs combine to ORCs). This results in an IRF for

the GGUM of the form:

$$P(X_{ij} = k | \theta_j, \varphi_i) = \frac{\exp(\alpha_i(k(\theta_j - \delta_i) - \sum_{v=0}^k \tau_{iv})) + \exp(\alpha_i((M-k)(\theta_j - \delta_i) - \sum_{v=0}^k \tau_{iv}))}{\sum_{w=0}^K \exp(\alpha_i(w(\theta_j - \delta_i) - \sum_{v=0}^w \tau_{iv})) + \exp(\alpha_i((M-w)(\theta_j - \delta_i) - \sum_{v=0}^w \tau_{iv}))} \quad (1.9)$$

where

$$\varphi_i = \{\alpha_i, \delta_i, \tau_{ik}\};$$

X_{ij} is the response of subject $j \in \{1, 2, \dots, N\}$ to item $i \in \{1, 2, \dots, I\}$;

$k \in \{0, 1, 2, \dots, K\}$, where 0 corresponds to the strongest possible negative response and K corresponds to the strongest positive response;

α_i : is the discrimination parameter of item i ;

δ_i : is the location parameter of item i ;

θ_j : is j th examinees ability;

τ_{iv} : is the location of the v th ability threshold where a response of k becomes more likely than a response of $v - 1$ in GPCM model but without such an interpretation in the GGUM. A symmetry restriction set $\tau_{ik} = \tau_{i(2K+1-k)}$ for $k \neq 0$ and $\tau_{i0} = 0$.

As an initial exploration into combining monotone and unfolding items, this research only focuses on dichotomous responses where X_{ij} is in $(0, 1)$. In the GGUM model, there are only three τ_i s in the IRF for each item, where $\tau_{i1} = -\tau_{i2}$ and $\tau_{i0} = 0$.

Item parameters for IRT models can be estimated using the marginal maximum likelihood (Bock and Aitkin, 1981; Bock and Lieberman, 1970), based on the expectation-maximization (EM) algorithm. The corresponding Bayesian method is marginalized Bayes modal estimation.

1.3. MOKKEN SCALE ANALYSIS

Mokken scale analysis is a nonparametric method for analyzing monotone item exams. It is used for constructing scales of items. It is based on the scalability coefficients that are defined for pairs of items (H_{ij}), for each item relative to the remaining items (H_i) and for the total scale (H):

$$H_{ij} = \frac{\text{cov}(X_i, X_j)}{\text{cov}_{max}} = \frac{p_{ij} - p_i p_j}{p_i(1 - p_j)} \text{ for } p_i < p_j \quad (1.10)$$

$$H_i = \frac{\text{cov}(X_i, R_{(i)})}{\text{cov}_{max}(X_i, R_{(i)})} = \frac{\sum_{j \neq i} (p_{ij} - p_i p_j)}{\sum_{j > i} (p_i - p_i p_j) + \sum_{j < i} (p_j - p_i p_j)}$$

where

$R_{(i)}$ is the rest score of item i (the total score excluding that item);

p_i is the probability of getting the i th item correct;

p_{ij} is the probability of getting both i th and j th items correct.

The calculation of these H -coefficients depend on comparing the probability of an error in the ranking of item p -values to the probability of such a ranking occurring if the items are unrelated. The assumptions of unidimensionality, local independence and monotonicity imply that:

$$\begin{aligned} 0 < H_{ij} &\leq 1 \text{ for all } i \neq j \\ 0 < H_i &\leq 1 \text{ for all } i \\ 0 < H &\leq 1 \end{aligned} \quad (1.11)$$

Mokken (1971) proposed a rule of thumb, for which H_i and H values of exceed some positive constant ρ is considered as weak association. A lower bound of .30 is advocated as a suitable value of ρ (Sijtsma and Molenaar, 2002). This result holds true for monotone item exams when all items have a value $\in (-\infty, 0)$ or all items have $a \in (0, \infty)$.

In our research, the problem becomes more complicated because we allow a combination of monotone increasing items and monotone decreasing items where $a \in (-\infty, \infty)$. For the mixed exam which only contains monotone items with both monotone increasing and monotone decreasing items, the sign of the H_i coefficient for each item is uncertain. However, all monotone increasing items will have the same sign which is opposite that of the monotone decreasing items.

Mokken scale analysis is not applicable for unfolding items. The sign of the H -coefficient can be either positive or negative. Therefore, when adding the unfolding items to the mixed exam, the sign of the H -coefficient cannot easily tell the items apart. However, through the simulation studies in Chapter 2 and Chapter 3, two ad-hoc rules can still be derived based on Mokken scale analysis. Moreover, Mokken scale analysis has been utilized to separate the monotone increasing items and monotone decreasing items for a mixed exam (see in Chapter 5). For an item with negative H value, we defined the transformed score which is calculated by 1 minus the current score. The transformed score is utilized to distinguish these two types of monotone items and make sure all the monotone items are on the same scale by calculating the H coefficient. The item score will be changed to $1-X_{ij}$ for the items with negative H coefficient. Since the H coefficient for unfolding items could also be negative, the test score for some of the unfolding items are also transformed.

1.4. MEAN/MEAN LINKING

IRT is sample independent in the sense that the shape of the underlying IRFs are hypothesized to be the same for any population for whom the model fits. The scales (mean and standard deviation) of the ability distribution are arbitrary though. As such it is necessary to link sets of item parameters on different scales. “When there are only two tests, this involves finding a set of linking constants to transform the parameters from one test (the from scale) to the scale of the other (the to scale).” (Weeks, 2010). Following Kolen and Brennan (2004), when x and y are two ability

scales, the equation to transform from scale x to y is:

$$\theta_x = A\theta_y + B \tag{1.12}$$

“The mean/mean (Loyd and Hoover 1980) method, known as moment method, is the simplest approach to estimating A and B because they only require the computation of means and standard deviations for various item parameters.” (Weeks, 2010).

That is,

$$\begin{aligned} a_x &= a_y/A \\ b_x &= Ab_y + B \end{aligned} \tag{1.13}$$

where

a_x is the a parameter (discrimination parameter) in $2PL$ model on x scale;

a_y is the a parameter in $2PL$ model on y scale;

b_x is the b parameter (difficulty parameter) in $2PL$ model on x scale;

b_y is the b parameter (difficulty parameter) in $2PL$ model on y scale.

A and B are found using

$$\begin{aligned} A &= \text{mean}(a_y)/\text{mean}(a_x) \\ B &= \text{mean}(b_x) - A \text{mean}(b_y) \end{aligned} \tag{1.14}$$

In the case of items with a negative a , the absolute value of a is used to find the linking constraints. This is equivalent to the $1 - X$ transform (Section 1.3).

The linking function works on the parameters in the GGUM model also. Once we have the A and B ,

$$\begin{aligned} \alpha_x &= \alpha_y/A \\ \delta_x &= A\delta_y + B \\ \tau_x &= A\tau_y \end{aligned} \tag{1.15}$$

The R package *plink* (Weeks, 2010) has been applied to do the mean/mean linking. This linking will be utilized throughout to compare the true values and estimated values in the various simulations in Chapter 5.

1.5. OVERVIEW OF THE SUBSEQUENT CHAPTERS

Chapter 2 reviews the nonparametric manifest monotonicity test under non-parametric theory and develops the p-value/biserial method under the CTT framework. Using these two methods, we attempt to identify the items types in the “mixed exam”. Several simulation studies are conducted to access the pros and cons of these two methods. In Chapter 3, a new mixture model and associated marginalized Bayes modal estimation are proposed. The initial estimates for all the parameters in this mixture model are discussed in Chapter 4. Chapter 5 applies this mixture model and marginalized Bayes modal estimation for several simulation studies. Chapter 6 summarizes the research and provides the several directions for future study.

CHAPTER 2

IDENTIFYING ITEM TYPES BY THE MANIFEST MONOTONICITY TEST AND P-VALUE/BISERIAL METHOD

The first step of this study is try to identify the item types on a mixed exam. All the methods discussed in this chapter were originally designed for the monotone items. We investigated these methods and their use for distinguishing between monotone and unfolding items. We started by exploring an existing nonparametric method, the manifest monotonicity test (Sijtsma and Molenaar, 2002). We also proposed a new method based on classical test statistics, named the p-value/biserial method. Several simulation studies were conducted to show the advantages and disadvantages of these two methods for identifying the item type.

2.1. THE MANIFEST MONOTONICITY TEST

The assumption of increasing monotonicity means that the IRF, or the probability of answering the item correctly, is a non-decreasing function of the latent value (θ). This is very crucial for distinguishing monotone models and unfolding models. Latent monotonicity is defined as follows

$$P(X_i \geq x|\theta_a) \geq P(X_i \geq x|\theta_b) \text{ for all } \theta_a > \theta_b. \quad (2.1)$$

where $i = 1, 2, \dots, I$ and $x = 1, \dots, N$;

“Manifest monotonicity is an observable property of the test data” (Van Der Ark, 2007), and is defined as

$$P(X_i \geq x | R_{-i} = s) \geq P(X_i \geq x | R_{-i} = r) \text{ for all } i, x, s > r \quad (2.2)$$

where R_{-j} is the rest score, the total score excluding item i . This is the basis of the manifest monotonicity test which can be applied to detect violations of monotonicity in the dichotomous case. Failure to reject manifest monotonicity does not guarantee monotonicity (Junker and Sijtsma 2000).

Without the sampling error, if manifest monotonicity holds, the probability of correctly answering item i with rest score s will be greater than the probability of correctly answering the same item with a lower rest score (i.e. $r < s$) for any pair of r and s . If the reverse order is observed (i.e. the difficulty parameter p_i with rest score s is less than p_i with rest score r), then manifest monotonicity is violated (Molenaar and Sijtsma, 2000). Two possible concerns are the cases of the small sample sizes and counting of too many small violations. These problems occur purely by sampling fluctuation even when monotonicity holds in the populations and are either caused by the number of observations in r or the proportion of violations (vi) being too small.

To solve the first problem, the observations in the rest score groups will be combined if they contain less than a certain *Minisize* number of observations. For the second problem, let vi_i be the percentage of violations related to the total number of paired groups for item i for different pairs of r and s . The minimum violation (*Minvi*) is used to control the small violation problem with *Minvi* = 0.03 in default. Then the violation will be counted if the difference of probability for rest group r and group s for any ($r < s$) is greater than *Minvi* (see details in Molenaar and Sijtsma, 2000 and the example below).

Consider the example in Table 2.1. For item i , there are 3 rest score groups (h) after applying the *minisize* so that $N_h \geq \text{minisize}$. By default *minsize* = $\frac{N}{10}$ if $N \leq 500$; *minsize* = $\frac{N}{5}$ if $250 \leq N < 500$; and *minsize* = $\max(\frac{N}{3}, 50)$, if $N < 250$

(Molenaar and Sijtsma, 2000). Note that $\sum_{h=1}^3 N_h = N$ where N is the total number of examinees and N_h is the size of the rest score group.

TABLE 2.1. The Example for the Manifest Monotonicity

Group	Lo	Hi	N	$F0$	$F1$	$P(X = 1)$
1	9	22	$N_1 = (a + b)$	a	b	b/N_1
2	23	26	$N_2 = (c + d)$	c	d	d/N_2
3	27	50	$N_3 = (e + f)$	e	f	f/N_3

where

Lo is the lowest score of that group;

Hi is highest score of that group;

$F0$ and $F1$ is the frequency distribution of the item scores for item i for that group j ;

$P(X = 1)$ is the proportion of respondents obtaining score 1 in that group.

For this example, manifest monotonicity means $d/N_2 > b/N_1$, $f/N_3 > b/N_1$ and $f/N_3 > d/N_2$. If the difference between two probabilities is too small, it might be caused by the sampling error. So, the minimum violation is applied. The probability difference is calculated and if the difference is larger than $Minvi$, a violation will be recorded.

In any two comparison groups for item i , a z-statistics is computed (Molenaar and Sijtsma, 2000). For group 1 and 2 in our example, z is calculated by

$$z = \frac{2(\sqrt{(a+1)(d+1)} - \sqrt{bc})}{\sqrt{(t-1)}} \quad (2.3)$$

where

a is the frequency of $X_i=0$ in group 1;

b is the frequency of $X_i=1$ in group 1;

c is the frequency of $X_i=0$ in group 2;

d is the frequency of $X_i=1$ in group 2;

$$t = a + b + c + d;$$

The z -statistics will be compared with critical value with default $\alpha = 0.05$ (two tailed normal test) for each item. Any item with at least one significant z or with at least one violation is counted as an item which violates the manifest monotonicity in our research. The entire manifest monotonicity test can be implemented in the R package Mokken where the test statistics is given in the result of *check.monotonicity* function (Van Der Ark, 2007).

2.2. THE SIMULATION STUDY FOR THE MANIFEST MONOTONICITY TEST

The effectiveness of the manifest monotonicity test was investigated by using a mixed exam containing both monotone items and unfolding items. An item bank for the mixed exam consists of three types of items' parameters. One is monotone increasing items which are generated from the *2PL* model with a parameters between 0.9 and 1.5 by 0.1, b parameters between -1.5 to 1.5 by 0.1, and c parameter 0. Another type is monotone decreasing items with a parameters between -1.5 and -0.9 by 0.1, b parameters between -1.5 to 1.5 by 0.1, and c parameter 0. Unfolding items were calculated based on each monotone item in the bank as indicated below.

For the corresponding unfolding items' parameters, we made each unfolding item have the same location parameter as a paired monotone item ($\delta_i = b_i$) and set these paired items to have the same average information when θ follow the standard normal distribution. So $\int_{\theta} I_m(\theta, \varphi_{mi})d\theta = \int_{\theta} I_u(\theta, \varphi_{ui})d\theta$ where I is the information function. Also, for simplicity, in this study, τ_{i1} is set to -0.2 for all the unfolding items. Then $\tau_{i2} = 0.2$ by definition. There are 868 items in the item bank. All the simulation items (in Chapters 2 – 5) in this research were randomly picked from this item bank.

Twenty mixed items were randomly picked from the item bank containing 10 monotone items and their paired unfolding monotone items. Items 1 to 10 are monotone items including both monotone increasing items and monotone decreasing items. Items 11 – 20 are unfolding items. A data set was generated from these 20 items with sample size $N=1000$. For the items with negative H coefficients, the transformed score was applied (see Section 1.3). The R function: *check.monotonicity* showed which items were detected as non-monotone item (Section 2.1).

We repeated this simulation 1000 times with the same item parameters. The frequency of non-monotonicity for i th item is given in Table 2.2. In this table, the first 10 items are monotone items and the second 10 items are unfolding items. The numbers on the 2nd row and 4th row are the observed probability that the given item has been identified as non-monotone item in 1000 simulations. For the first 10 items, all the proportions are < 0.05 except item 8 and 9 with frequency 0.06 and 0.08 which are slightly higher than 0.05. From 11 to 20, the proportions for items 15 and 16 are relatively small. This informs us that these items were more difficult to distinguish by using the manifest monotonicity test (the parameters for these two items are shown in Table 2.3). However, the other items from 11 to 20 are identified as non-monotone items with higher probability.

Figure 2.1 contains the IRFs for all the unfolding items and shows that item 15 and 16 are the “extreme unfolding items” with very large or small location values. As seen above, this method is not very effective for these items (all the parameters for this simulation are recorded in Appendix C).

TABLE 2.2. The Percentage of Time Each Item is Identified as Non-monotone. (1 – 10 monotone, 11 – 20 unfolding)

Item	1	2	3	4	5
Freq	0.03	0.03	0.03	0.04	0.01
Item	6	7	8	9	10
Freq	0.05	0.02	0.06	0.08	0.04
Item	11	12	13	14	15
Freq	1	1	1	0.97	0.13
Item	16	17	18	19	20
Freq	0.22	1	0.9	0.879	1

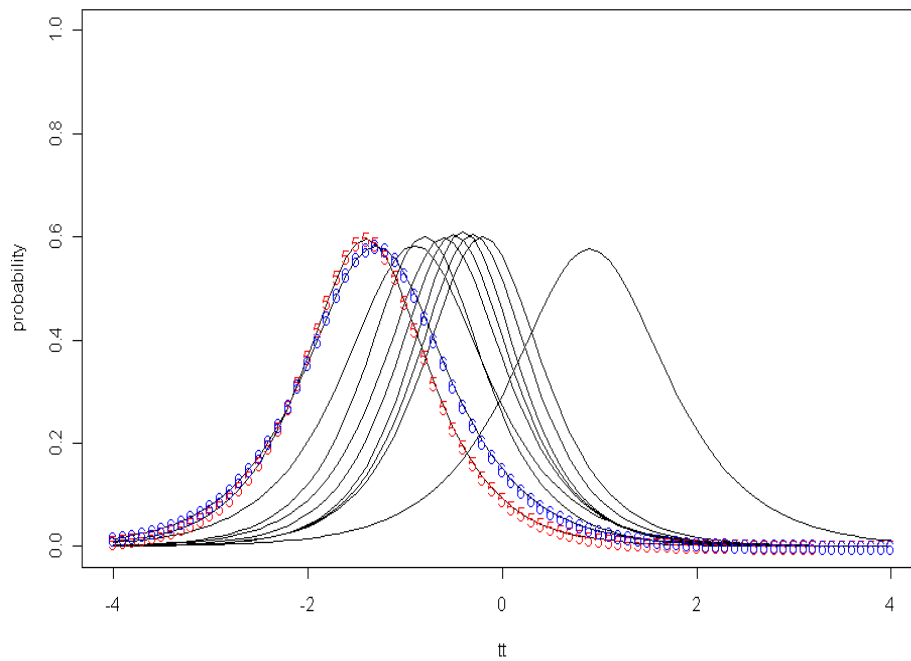


FIGURE 2.1. IRF for 10 GGUM items: “5” indicates item 15 and “6” indicates item 16

TABLE 2.3. Item Parameters for Items 8, 9, 15 and 16

Item	8	9		15	16
a	1.1	1	α	2	1.7
b	-0.9	0.9	δ	-1.4	-1.3

In the simulation study conducted above, the manifest monotonicity test can tell the items apart for the limited location values and failed when the unfolding item located in extremes and θ follow the standard normal distribution. When θ follows the standard normal distribution, 95 percent of the examinees are located within 3 standard deviations, so that 5 percent of examinees who are located in the extremes maybe too few to identify the items type. It seems plausible that when θ has a different distribution, like the uniform distribution, the manifest monotonicity test should work better on the extremes. Moreover, the manifest monotonicity test should also perform better with large examinees.

Thus we propose two hypothesises as follows.

- Hypothesis 1: the manifest monotonicity test performs better for large numbers of examinees since there will be more examinees located in the extremes.
- Hypothesis 2: the manifest monotonicity test performs better for detecting unfolding items when θ follows the uniform $[-3, 3]$ distribution.

We systematically analyzed the performance of the manifest monotonicity test for different θ distributions and various numbers of examinees. θ was set to follow either a normal distribution or a uniform $[-3, 3]$ distribution. The number of examinees was either 500, 1000, 2000 or 4000. The a parameters in monotone items were in $(-1.5, -0.9, 0.9, 1.5)$ and b s were in $(-1.5, -1, 0, 1, 1.5)$. The α parameters for unfolding item were in $(0.5, 1, 1.5, 2)$, δ s were in $(-1.5, -1, 0, 1, 1.5)$ and τ s were in $(-0.2, -0.4, -0.6)$. Then 20 monotone items applied in the simulation data set were a combination of parameter a and b . The 20 unfolding items were a combination of α and δ parameters. The effect of τ was studies in this simulation as well. In each study, 3 sub-studies were conducted for different τ values in $(-0.2, -0.4, -0.6)$ (Table 2.4).

TABLE 2.4. The Studies Summary

	Study1	Study2	Study3	Study4	Study5	Study6	Study7	Study8
θ	normal	normal	normal	normal	uniform	uniform	uniform	uniform
N	500	1000	2000	4000	500	1000	2000	4000

TABLE 2.5. Manifest Monotonicity Test Study Report for the Monotone Items. Reported values are the proportion identified as unfolding

item	Parameter	Study1 (Normal, $N = 500$)		Study2 (Normal, $N = 1000$)		Study3 (Normal, $N = 2000$)		Study4 (Normal, $N = 4000$)	
		$\tau = -0.2$	$\tau = -0.4$	$\tau = -0.2$	$\tau = -0.4$	$\tau = -0.2$	$\tau = -0.4$	$\tau = -0.2$	$\tau = -0.4$
1	a -1.5	0.002	0.002	0.016	0.022	0	0	0	0
2	-1.5	0	0	0.024	0.028	0.016	0.002	0	0
3	-1.5	0	0	0.018	0.014	0.010	0	0.002	0
4	-1.5	0	0	0.016	0.018	0.030	0	0	0
5	-1.5	0	0	0.022	0.014	0.012	0.002	0	0
6	-0.9	0.010	0.018	0.138	0.118	0.134	0.028	0.012	0.002
7	-0.9	0.014	0.006	0.116	0.132	0.114	0.026	0.010	0
8	-0.9	0	0	0.080	0.054	0.076	0.008	0.004	0
9	-0.9	0.002	0	0.100	0.108	0.086	0.012	0.008	0.002
10	-0.9	0.002	0	0.138	0.108	0.112	0.010	0.012	0
11	0.9	0.008	0.004	0.130	0.120	0.108	0.016	0.010	0.002
12	0.9	0.006	0	0.110	0.110	0.092	0.014	0.010	0.002
13	0.9	0	0	0.076	0.046	0.058	0.004	0.004	0
14	0.9	0	0	0.074	0.092	0.094	0.012	0.012	0
15	0.9	0.006	0	0.108	0.108	0.098	0.016	0.008	0.004
16	1.5	0	0.004	0.022	0.026	0.024	0	0	0
17	1.5	0.004	0	0.034	0.022	0.044	0.004	0	0
18	1.5	0	0	0.008	0.014	0.006	0	0	0
19	1.5	0	0	0.024	0.026	0.018	0	0	0
20	1.5	0	0	0.014	0.016	0.012	0	0	0

TABLE 2.6. The Manifest Monotonicity Test Study Report for the Monotone Items (Cont.)

item	Parameter	Study5 (Uniform, $N = 500$)		Study6 (Uniform, $N = 1000$)		Study7 (Uniform, $N = 2000$)		Study8 (Uniform, $N = 4000$)	
		$\tau = -0.2$	$\tau = -0.4$	$\tau = -0.2$	$\tau = -0.4$	$\tau = -0.2$	$\tau = -0.4$	$\tau = -0.2$	$\tau = -0.4$
1	a -1.5	0	0	0.020	0.024	0	0.010	0	0
	b -1.5	0	0	0	0.032	0	0.006	0	0.002
2	-1.5	0	0	0.004	0.008	0.002	0	0	0
3	-1.5	0	0	0.018	0.020	0	0	0	0
4	-1.5	0	0	0.044	0.076	0	0.028	0	0.010
5	-1.5	0	0.002	0.156	0.222	0.072	0.142	0.020	0.058
6	-0.9	-1.5	0	0.072	0.080	0.006	0.010	0.004	0.012
7	-0.9	-1	0	0.056	0.072	0.004	0.006	0.014	0
8	-0.9	0	0.002	0.092	0.118	0.020	0.040	0.020	0.004
9	-0.9	1	0	0.094	0.196	0.034	0.102	0.054	0.050
10	-0.9	1.5	0	0.214	0.262	0.096	0.190	0.162	0.092
11	0.9	-1.5	0	0.072	0.060	0.010	0.014	0.032	0
12	0.9	-1	0	0.056	0.058	0.012	0.004	0.012	0.002
13	0.9	0	0.004	0.090	0.136	0.026	0.024	0	0.008
14	0.9	1	0	0.122	0.200	0.044	0.132	0.066	0.038
15	0.9	1.5	0.002	0.180	0.246	0.088	0.182	0.066	0.052
16	1.5	-1.5	0	0.028	0.018	0.002	0.002	0	0
17	1.5	-1	0	0.012	0.002	0	0	0	0
18	1.5	0	0.002	0.014	0.024	0.002	0.002	0	0
19	1.5	1	0	0.032	0.080	0.002	0.016	0.008	0
20	1.5	1.5	0	0.142	0.228	0.068	0.166	0.074	0.054

TABLE 2.7. The Manifest Monotonicity Test Study Report for the Unfolding Items.

item	Parameter α	δ	Study1 (Normal, $N = 500$)		Study2 (Normal, $N = 1000$)		Study3 (Normal, $N = 2000$)		Study4 (Normal, $N = 4000$)					
			$\tau = -0.2$	$\tau = -0.4$	$\tau = -0.2$	$\tau = -0.4$	$\tau = -0.2$	$\tau = -0.4$	$\tau = -0.2$	$\tau = -0.4$	$\tau = -0.2$	$\tau = -0.4$		
21	0.5	-1.5	0.164	0.18	0.172	0.724	0.732	0.686	0.414	0.406	0.434	0.160	0.110	0.132
22	0.5	-1	0.394	0.404	0.418	0.868	0.844	0.880	0.704	0.704	0.708	0.508	0.500	0.594
23	0.5	0	0.886	0.888	0.906	0.996	1	0.996	1	1	1	1	1	1
24	0.5	1	0.446	0.418	0.454	0.878	0.858	0.878	0.734	0.672	0.722	0.532	0.470	0.510
25	0.5	1.5	0.204	0.190	0.206	0.708	0.708	0.720	0.480	0.426	0.398	0.158	0.150	0.152
26	1	-1.5	0.006	0.004	0.006	0.308	0.284	0.306	0.124	0.064	0.044	0.016	0.010	0.010
27	1	-1	0.092	0.066	0.056	0.746	0.760	0.780	0.702	0.712	0.754	0.630	0.712	0.808
28	1	0	0.998	0.998	0.998	1	1	1	1	1	1	1	1	1
29	1	1	0.244	0.242	0.244	0.788	0.732	0.782	0.658	0.646	0.636	0.642	0.586	0.626
30	1	1.5	0.018	0.014	0.018	0.340	0.318	0.334	0.082	0.078	0.086	0.006	0.006	0.014
31	1.5	-1.5	0.002	0.004	0.002	0.126	0.106	0.110	0.024	0.018	0.014	0.006	0.002	0.002
32	1.5	-1	0.018	0.014	0.012	0.734	0.748	0.776	0.740	0.792	0.844	0.774	0.778	0.912
33	1.5	0	1	1	1	1	1	1	1	1	1	1	1	1
34	1.5	1	0.134	0.122	0.172	0.766	0.752	0.772	0.764	0.746	0.814	0.844	0.784	0.856
35	1.5	1.5	0.004	0.004	0	0.136	0.136	0.132	0.022	0.012	0.014	0	0	0
36	2	-1.5	0.010	0.004	0	0.052	0.064	0.052	0.008	0.008	0.008	0	0.002	0
37	2	-1	0.014	0.006	0.002	0.736	0.814	0.856	0.778	0.822	0.880	0.808	0.808	0.882
38	2	0	1	1	1	1	1	1	1	1	1	1	1	1
39	2	1	0.1	0.084	0.098	0.798	0.786	0.848	0.806	0.804	0.908	0.926	0.866	0.956
40	2	1.5	0	0.002	0.002	0.098	0.062	0.076	0.008	0.006	0.002	0	0	0

TABLE 2.8. The Manifest Monotonicity Test Study Report for the Unfolding Items (Cont.)

item	Parameter	Study5 (Uniform, $N = 500$)		Study6 (Uniform, $N = 1000$)		Study7 (Uniform, $N = 2000$)		Study8 (Uniform, $N = 4000$)			
		δ	τ	δ	τ	δ	τ	δ	τ		
21	0.5 -1.5	0.160	0.134	0.798	0.840	0.652	0.672	0.720	0.606	0.572	0.570
22	0.5 -1	0.470	0.468	0.966	0.982	0.970	0.976	0.974	0.978	0.984	0.988
23	0.5 0	0.988	0.992	1	1	1	1	1	1	1	1
24	0.5 1	0.634	0.652	0.956	0.958	0.952	0.946	0.940	0.932	0.898	0.908
25	0.5 1.5	0.350	0.332	0.758	0.756	0.77	0.566	0.632	0.444	0.354	0.382
26	1 -1.5	0.020	0.030	0.832	0.810	0.864	0.826	0.830	0.834	0.796	0.856
27	1 -1	0.606	0.708	0.566	0.998	0.998	1	1	1	1	1
28	1 0	1	1	1	1	1	1	1	1	1	1
29	1 1	0.842	0.860	0.998	0.994	0.994	0.998	1	0.998	1	1
30	1 1.5	0.238	0.284	0.688	0.644	0.612	0.586	0.534	0.464	0.410	0.338
31	1.5 -1.5	0.012	0.004	0.010	0.906	0.916	0.926	0.940	0.970	0.974	0.974
32	1.5 -1	0.61	0.752	0.614	1	1	1	1	1	1	1
33	1.5 0	1	1	1	1	1	1	1	1	1	1
34	1.5 1	0.96	0.958	0.946	1	0.998	1	1	1	1	1
35	1.5 1.5	0.256	0.260	0.166	0.612	0.654	0.636	0.542	0.592	0.422	0.338
36	2 -1.5	0.004	0.002	0.002	0.948	0.948	0.978	0.990	0.996	0.998	1
37	2 -1	0.598	0.778	0.668	1	1	1	1	1	1	1
38	2 0	1	1	1	1	1	1	1	1	1	1
39	2 1	0.978	0.99	0.974	1	1	1	1	1	1	1
40	2 1.5	0.218	0.232	0.198	0.706	0.630	0.654	0.528	0.630	0.426	0.464

Tables 2.5 and 2.6 report the results for monotone items, indicating the proportion identified as unfolding. Table 2.5 is the study for θ following the standard normal distribution and Table 2.6 is for θ following a uniform $[-3, 3]$. In these two tables, when the number of examinees N changes from 500 to 1000, the frequency of the item to be unfolding increases slightly. Then it goes down when N increases from 1000 to 2000 and to 4000. Therefore, hypothesis 1 has support for monotone items when $N \geq 1000$. The frequencies are universally larger when $b = -1.5$ or $b = 1.5$ in both tables which indicated that the manifest monotonicity method has limitation on the extremes. The frequencies with θ following a uniform distribution are higher compared to θ following a standard normal distribution. However, all the frequencies of monotone items are fairly small (Min=0, Max=0.28, Mean=0.026).

Tables 2.7 and 2.8 report the result for unfolding items. Table 2.7 is when θ follows the standard normal distribution and table 2.8 is for θ following a uniform $[-3, 3]$. The frequencies in Table 2.8 are higher than the frequencies in Table 2.7, which indicates that this method has an easier time identifying the unfolding items under a uniform distribution, as conjectured in hypothesis 2. Hypothesis 1, that the frequencies are higher when N increases, is not well supported. In Table 2.7 when θ has standard normal distribution, the frequency is decreasing with N increases for $\alpha = 0.5$ and $\alpha = 1$. When α increases, the frequency is increasing with N when δ is in the middle with values $-1, 0, 1$. However, when δ is in the extremes -1.5 and 1.5 , the frequency is decreasing with N . In Table 2.8, there is a similar pattern that for the larger δ value $(-1.5, 1.5)$, the frequency is decreasing when N increases. For both tables, the frequency numbers are higher in the middle ($\delta = -1, 0, 1$) and lower in the extremes $\delta = -1.5, 1.5$.

Therefore, when θ follows uniform distribution, the manifest monotonicity test works better to detect the unfolding items. But it is easier to give more misleading results for monotone items than when θ follows standard normal distribution. It gives

more accurate results when N increases for the middle items and fails more often with larger δ values. The trend for different τ is not obvious in these four tables.

We also studied the effect of *minsize* and *minvi* (Tables 2.9 to 2.12). The default of *minsize* in R is $minsize = \frac{N}{10}$ if $N \leq 500$; $minsize = \frac{N}{5}$ if $250 \leq N < 500$; and $minsize = \max(\frac{N}{3}, 50)$, if $N < 250$ (Molenaar and Sijtsma, 2000). Tables 2.9 and 2.10 list the reports when $N = 2000$ with *minsize* = 100 which is applied to compare with Tables 2.4 to 2.7 with $N = 2000$ and default *minsize* = 200. The frequencies of both items increase if the *minsize* are larger. Tables 2.11 and 2.12 list the reports for $N = 2000$ and *minvi* = 0.06. The default value for *minvi* is 0.03. When increasing *minvi*, the frequencies get smaller for both monotone items and unfolding items. After comparison, the default value from R gives more reasonable result.

TABLE 2.9. The Manifest Monotonicity Test Study Report for the Monotone Item with $minimize = 100$

item	Parameter		Study9 (Normal, $N = 2000$)			Study10 (Unifrom, $N = 2000$)		
	a	b	$\tau = -0.2$	$\tau = -0.4$	$\tau = -0.6$	$\tau = -0.2$	$\tau = -0.4$	$\tau = -0.6$
1	-1.5	-1.5	0.068	0.050	0.058	0.376	0.342	0.426
2	-1.5	-1	0.158	0.098	0.062	0.228	0.186	0.234
3	-1.5	0	0.266	0.194	0.116	0.092	0.100	0.108
4	-1.5	1	0.208	0.164	0.114	0.328	0.444	0.356
5	-1.5	1.5	0.120	0.096	0.092	0.574	0.754	0.636
6	-0.9	-1.5	0.396	0.310	0.326	0.450	0.528	0.626
7	-0.9	-1	0.430	0.372	0.334	0.476	0.422	0.504
8	-0.9	0	0.424	0.402	0.372	0.546	0.528	0.570
9	-0.9	1	0.516	0.406	0.394	0.584	0.680	0.638
10	-0.9	1.5	0.404	0.374	0.374	0.696	0.790	0.738
11	0.9	-1.5	0.402	0.368	0.336	0.462	0.498	0.554
12	0.9	-1	0.420	0.400	0.308	0.492	0.466	0.544
13	0.9	0	0.450	0.378	0.342	0.454	0.560	0.536
14	0.9	1	0.460	0.434	0.400	0.550	0.718	0.578
15	0.9	1.5	0.428	0.396	0.330	0.674	0.764	0.714
16	1.5	-1.5	0.074	0.062	0.046	0.332	0.348	0.432
17	1.5	-1	0.194	0.090	0.078	0.212	0.198	0.220
18	1.5	0	0.246	0.208	0.162	0.074	0.112	0.122
19	1.5	1	0.224	0.150	0.088	0.348	0.450	0.374
20	1.5	1.5	0.136	0.084	0.076	0.590	0.752	0.656

TABLE 2.10. The Manifest Monotonicity Test Study Report for the Unfolding Item with $minsize = 100$

item	Parameter		Study9 (Normal, $N = 2000$)			Study10 (Unifrom, $N = 2000$)		
	α	δ	$\tau = -0.2$	$\tau = -0.4$	$\tau = -0.6$	$\tau = -0.2$	$\tau = -0.4$	$\tau = -0.6$
21	0.5	-1.5	0.976	0.968	0.960	0.994	0.996	0.996
22	0.5	-1	0.992	0.992	0.982	1	1	1
23	0.5	0	1	1	1	1	1	1
24	0.5	1	0.988	0.978	0.990	1	1	1
25	0.5	1.5	0.952	0.962	0.958	0.994	0.992	0.984
26	1	-1.5	0.838	0.802	0.768	0.996	0.996	0.998
27	1	-1	0.998	0.996	0.996	1	1	1
28	1	0	1	1	1	1	1	1
29	1	1	0.980	0.966	0.964	1	1	1
30	1	1.5	0.784	0.780	0.710	0.968	0.966	0.966
31	1.5	-1.5	0.672	0.650	0.596	0.998	0.994	0.998
32	1.5	-1	0.996	0.998	1	1	1	1
33	1.5	0	1	1	1	1	1	1
34	1.5	1	0.976	0.974	0.960	1	1	1
35	1.5	1.5	0.508	0.522	0.432	0.968	0.936	0.926
36	2	-1.5	0.474	0.466	0.482	1	0.998	1
37	2	-1	1	1	1	1	1	1
38	2	0	1	1	1	1	1	1
39	2	1	0.986	0.982	0.992	1	1	1
40	2	1.5	0.376	0.350	0.334	0.976	0.962	0.940

TABLE 2.1.1. The Manifest Monotonicity Test Study Report for the Monotone Item with $minvi = 0.06$

Item	Parameter		Study11 (Normal, $N = 2000$)			Study12 (Unifrom, $N = 2000$)		
	a	b	$\tau = -0.2$	$\tau = -0.4$	$\tau = -0.6$	$\tau = -0.2$	$\tau = -0.4$	$\tau = -0.6$
1	-1.5	-1.5	0	0	0	0.002	0	0
2	-1.5	-1	0	0	0	0	0	0
3	-1.5	0	0	0	0	0	0	0
4	-1.5	1	0	0	0	0	0.002	0
5	-1.5	1.5	0	0	0	0.014	0.034	0.016
6	-0.9	-1.5	0	0	0	0	0.006	0.002
7	-0.9	-1	0.004	0	0	0.002	0.002	0
8	-0.9	0	0.002	0	0.002	0	0	0
9	-0.9	1	0	0	0	0.004	0.024	0.008
10	-0.9	1.5	0	0	0	0.032	0.056	0.040
11	0.9	-1.5	0	0.002	0	0.002	0.002	0
12	0.9	-1	0	0	0	0	0	0
13	0.9	0	0.004	0	0.002	0.002	0.008	0
14	0.9	1	0.004	0	0	0.004	0.022	0.004
15	0.9	1.5	0.002	0	0	0.042	0.058	0.036
16	1.5	-1.5	0	0	0	0	0	0
17	1.5	-1	0	0	0	0	0	0
18	1.5	0	0	0	0	0	0	0
19	1.5	1	0	0	0	0	0	0
20	1.5	1.5	0	0	0	0.012	0.042	0.016

TABLE 2.12. The Manifest Monotonicity Test Study Report for the Unfolding Item with $minvi = 0.06$

item	Parameter		Study11 (Normal, $N = 2000$)			Study12 (Unifrom, $N = 2000$)		
	α	δ	$\tau = -0.2$	$\tau = -0.4$	$\tau = -0.6$	$\tau = -0.2$	$\tau = -0.4$	$\tau = -0.6$
21	0.5	-1.5	0.128	0.098	0.114	0.304	0.338	0.316
22	0.5	-1	0.364	0.346	0.380	0.782	0.81	0.818
23	0.5	0	0.982	0.984	0.988	1	1	1
24	0.5	1	0.358	0.306	0.288	0.688	0.670	0.672
25	0.5	1.5	0.114	0.126	0.120	0.254	0.222	0.270
26	1	-1.5	0.008	0.008	0.016	0.576	0.464	0.552
27	1	-1	0.392	0.452	0.506	1	1	1
28	1	0	1	1	1	1	1	1
29	1	1	0.382	0.328	0.342	0.988	0.968	0.976
30	1	1.5	0.016	0.012	0.010	0.280	0.198	0.224
31	1.5	-1.5	0	0.002	0	0.764	0.780	0.794
32	1.5	-1	0.588	0.592	0.708	1	1	1
33	1.5	0	1	1	1	1	1	1
34	1.5	1	0.472	0.456	0.498	1	1	1
35	1.5	1.5	0	0.004	0	0.330	0.236	0.244
36	2	-1.5	0	0	0	0.864	0.912	0.932
37	2	-1	0.678	0.682	0.786	1	1	1
38	2	0	1	1	1	1	1	1
39	2	1	0.598	0.604	0.670	1	1	1
40	2	1.5	0	0	0.002	0.338	0.266	0.246

2.3. THE P-VALUE/BISERIAL METHOD

Classical item analysis is another easily implemented theoretical framework where we might find the solution for this problem. The relationships between the normal ogive model (two parameter model using the normal cdf instead of logistic curve) and classical item statistics are very easy to calculate. Lord and Novick (1968) showed that, when θ follows the standard normal distribution, the parameters for the normal ogive model are:

$$a_i = \frac{\rho_{bis}}{\sqrt{1 - \rho_{bis}^2}} \quad (2.4)$$

and

$$b_i = \frac{-\phi^{-1}(p_i)}{\rho_{bis}} \quad (2.5)$$

where

b_i is the location (difficulty) parameter;

p_i is the p-value for i th item;

ρ_{bis} is the biserial correlation for i th item;

ϕ is the pdf of standard normal distribution.

Since this relationship is for the normal ogive monotone IRT model, which is very close to the $2PL$, we expect unfolding items mixed with monotone items to have different behavior. Therefore the p-value and biserial correlation are calculated and compared in one graph for all items. Table 2.13 is the summary of all the formulas we used for this method. In the Table 2.13, the formulas on the left hand side are when the true IRFs and θ are known (the theory results). The right hand side contains the formulas when the true values are unknown and all the parameters are estimated from the simulation data set (the simulation results).

TABLE 2.13. The Formula to Calculate the *estimatedb* by Using the *P* – *value* and Biserial Correlation.

Theory results	Simulation results
$p = E(IRF)$	$\hat{p} = \frac{\sum_{j=1}^J k_j}{J}$
$p_{pbis} = corr(X, \theta)$	$\hat{\rho}_{pbis} = corr((X_i), R_{-i})$
$\rho_{bis} = \rho_{pbis} \frac{\sqrt{p(1-p)}}{\varphi(\phi^{-1}(p))}$	$\hat{\rho}_{bis} = \hat{\rho}_{pbis} \frac{\sqrt{\hat{p}(1-\hat{p})}}{\varphi(\phi^{-1}(\hat{p}))}$
$b = \frac{-\varphi^{-1}(p)}{\rho_{bis}}$	$\hat{b} = \frac{-\varphi^{-1}(\hat{p})}{\hat{\rho}_{bis}}$

As noted above, for a mixed exams containing only monotone increasing items and monotone decreasing items, the items can be changed to the same direction (changing from monotone decreasing item to monotone increasing item) by transforming the items with negative H values. To make the problem easier, in this preliminary analysis, an assumption is made that the separation of monotone items and unfolding items is known. The transformed scores in this chapter are only used for monotone items with negative H values and the items' scores X_i are transformed to $1 - X_i$. We extend this transformation to all the items with negative H values in Chapter 3.

Once we have the p-values and estimated b values for the items with transformed scores, the plot of the p-value vs estimated b was drawn. Then an ad-hoc rule can be derived by the simulation study. This method is named the p-value/biserial method.

2.4. THE SIMULATION STUDY FOR THE P-VALUE/BISERIAL METHOD

Before applying the method to the data set in the last simulation, we generated a much simpler data set to check the pattern of data based on the p-value/biserial methodology. Set the a parameter for all the monotone items to be equal to 1 and the b parameters to range from -1.5 to 1.5 by 0.1 . Then 31 monotone items were generated. Following the method described in the last simulation, we generated the corresponding 31 unfolding items. This gave us a mixed exam with 62 items. Then a data set with sample size $N = 1000$ was generated based on these item parameters.

Using the formulas from Table 2.13, we calculated the p-value and estimated b value for each item under both the theory and the simulation scenarios. The graphs of p-value vs. estimated b were constructed under these two scenarios (see Figure 2.2). In Figure 2.2, the plot on the left is the theoretical result using the true b value and p-value. We used “u” to stand for the unfolding items and “m” to stand for the monotone items. The plot on the right is the simulation result for estimated b vs. p-value. Both plots show that the unfolding items and monotone items are clearly separated except for the area in the lower-right corner. The items with the p-value above 0.4 and the items with the estimated b value between -1 and 1 are monotone items. The items with p-value below 0.4 and the estimated b value below -1 are unfolding items.

The next question is to check if this method works for a more complex data set. Using one of the data sets from the first simulation study, we calculated the p-value and estimated b value for each item under both theory and simulation scenarios by Table 2.4. Then the plot for the p-value vs. estimated b was drawn (Figure 2.3). This figure shows the same pattern that all the p-values are above 0.4 or the estimated b value is between -1 to 1 are monotone items. The items with estimated b value below -1 and p-value below 0.4 are unfolding items. Figure 2.4 shows more simulation results. They have the same pattern as Figure 2.2 and Figure 2.3. Notice that the items in the lower-right corner usually have relatively large location values (higher b values in monotone items and higher δ values in unfolding items).

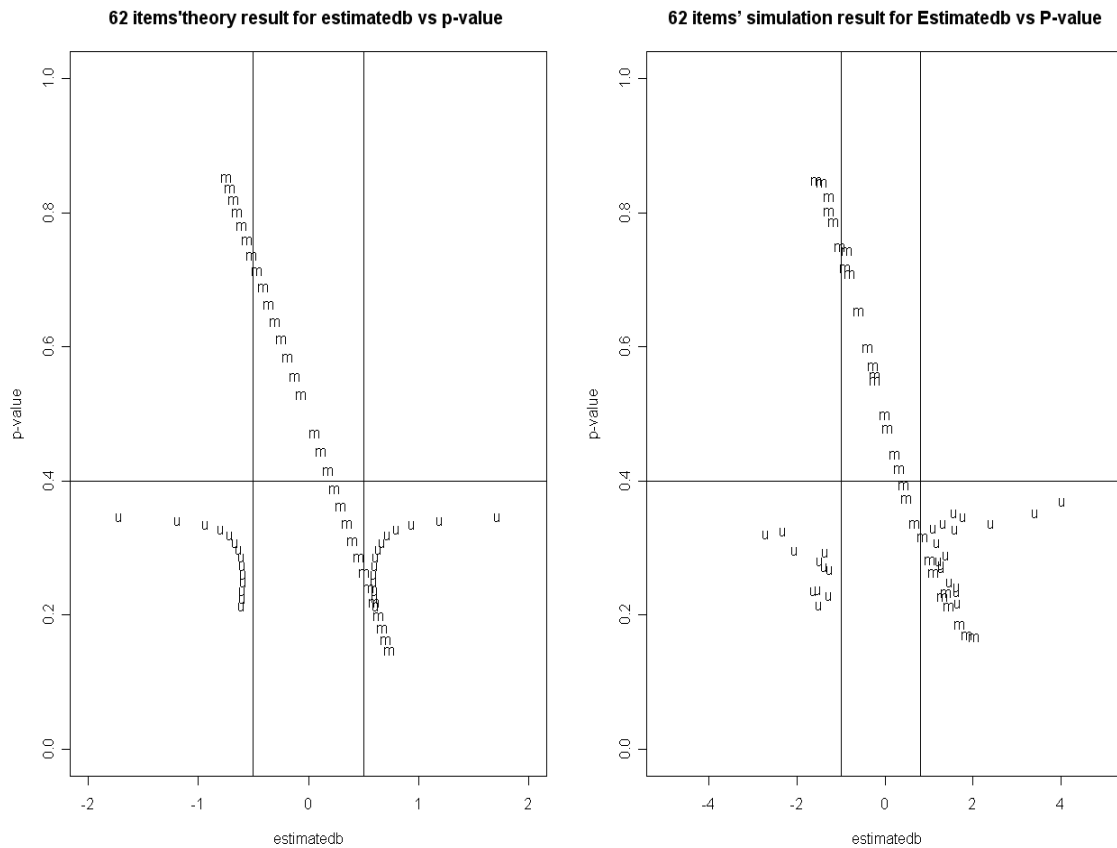


FIGURE 2.2. The P-value/Biserial Method for 62 Items under Different Scenarios

The the ad-hoc rules of identifying the item's type given below were observed from simulation study if the separation between the monotone items and the unfolding items were known:

- The items with p-value above 0.4 are monotone items.
- The items with estimated b value between -1 to 1 are monotone items.
- The items with p-value below 0.4 and estimated b value below -1 are unfolding items.

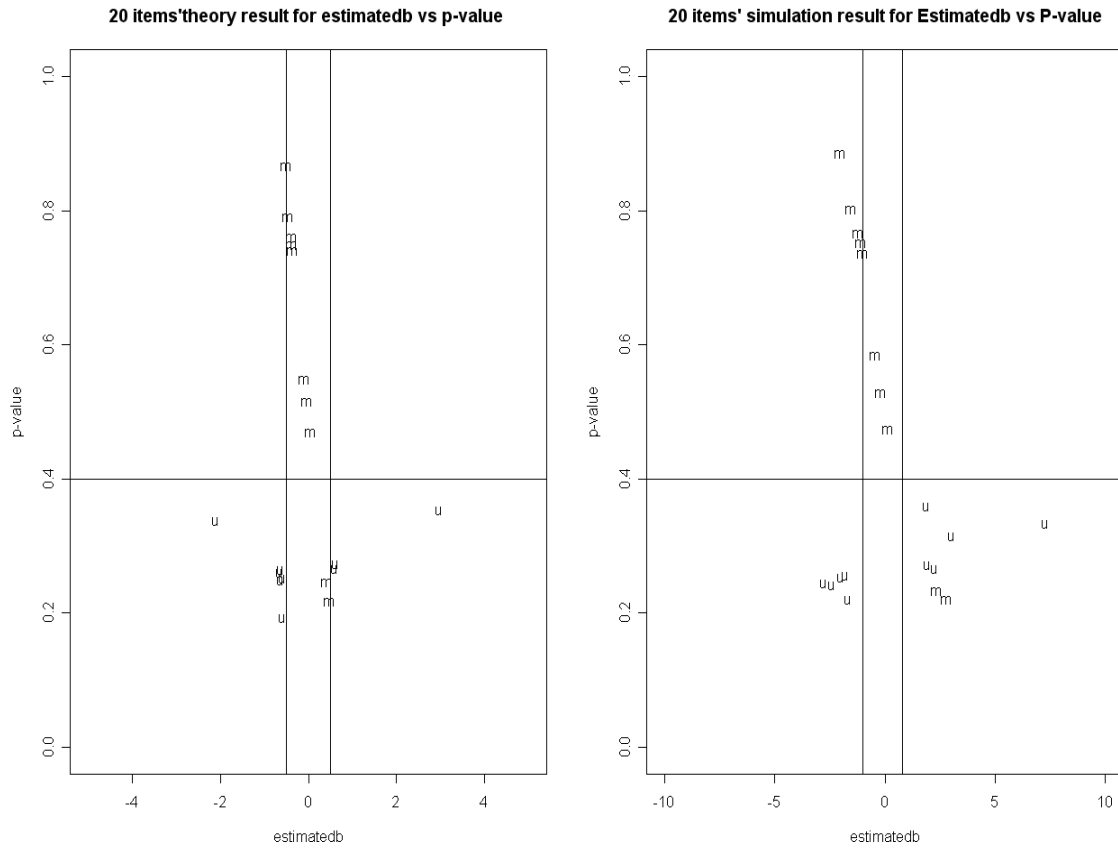


FIGURE 2.3. The P-value/Biserial Method for 20 Items

2.5. CONCLUSION

The manifest monotonicity test gives us some useful information about the item's monotonicity and non-monotonicity but this method only works for limited location values. A similar conclusion can be also derived for the p-value/biserial method. The limitation here is that the simulations for p-value/biserial method were conducted with the assumption that transformation is only applied for the monotone items with negative H value in this chapter. The plot (Figure 2.5) is different if we transformed all the item with the negative H value. Chapter 4 will discuss this in detail and another ad-hoc rule will be proposed and applied to attain the initial estimated π for subsequent the mixture model.

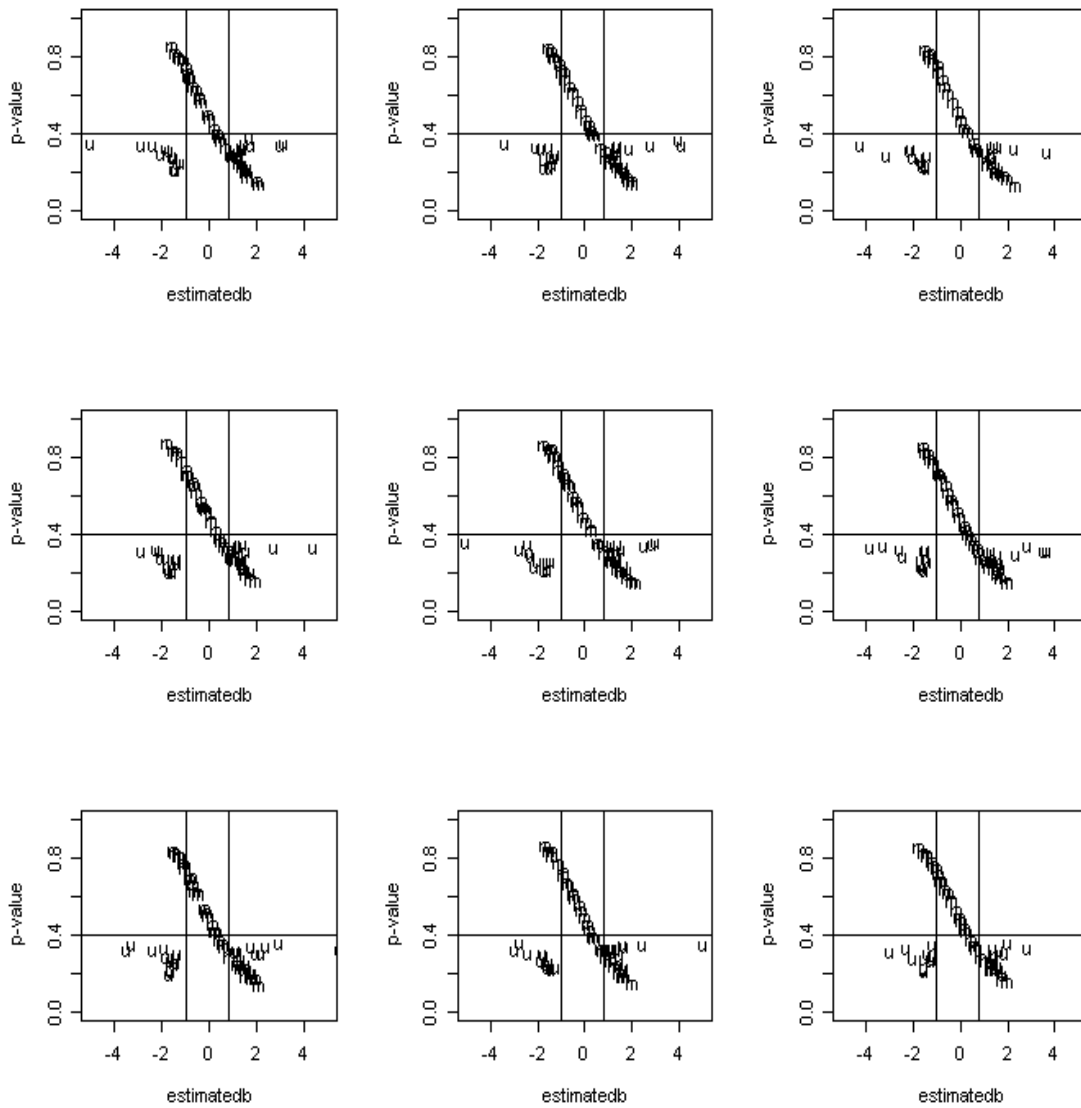


FIGURE 2.4. The P-value/Biserial Method for 9 Simulations

items' simulation result for Estimateddb vs P-value

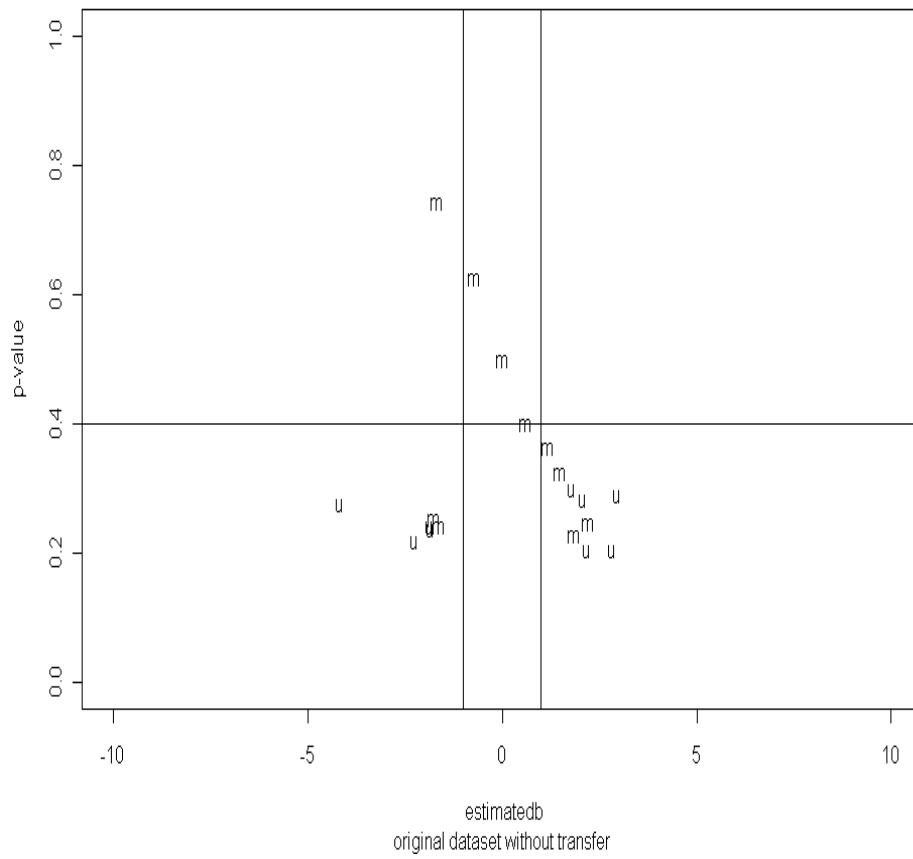


FIGURE 2.5. The P-value/Biserial Method for the Whole Transformed Data Set

CHAPTER 3

THE MIXTURE MODEL WITH THE MARGINALIZED BAYES MODAL ESTIMATION PROCEDURE

While the manifest monotonicity test and p-value/biserial method may provide some valuable information about identifying the types of each item, they do not attempt to estimate the item parameters for each item on a mixed exam. In order to solve this problem, a new model has been proposed which combines the monotone model and unfolding model together. This model can be estimated through marginalized Bayes modal estimation (e.g. Baker, 1992). The initial estimates for this algorithm (Chapter 4) and simulation studies of its effectiveness (Chapter 5) are discussed in the following chapters.

3.1. THE MIXTURE MODEL

We consider an exam which contains items that are appropriately modeled by either the 2PL (possibly with negative discrimination values) or dichotomous item GGUM. This results in an item response function that allows for both item types.

$$P(X_{ij} = k|\theta_j, \pi_i, \varphi_i) = P_u(X_{ij} = k|\theta_j, \varphi_{iu})^{\pi_i} P_m(X_{ij} = k|\theta_j, \varphi_{im})^{1-\pi_i} \quad (3.1)$$

where

$$\pi_i = \begin{cases} 0 & \text{if the } i \text{ th item is a monotone item} \\ 1 & \text{if the } i \text{ th item is a unfolding item} \end{cases}$$

$$i = 1 \dots I;$$

$$\varphi_{iu} = \{a_i, b_i\} \text{ (parameters for monotone items) and } \varphi_{im} = \{\alpha_i, \delta_i, \tau_i\} \text{ (parameters$$

for unfolding items) where u stands for unfolding and m stand for monotone; P_m is the IRF of the monotone model ($2PL$ with $a_i \in (-\infty, \infty)$, Equation (1.7)); P_u is the IRF of the unfolding model (GGUM, Equation (1.9)). k can take value 0 or 1 under dichotomous assumption.

From a Bayesian stand point, π_i is probability that item i is unfolding. If $\pi_i = 0$, then item i is monotone item, and $\pi_i = 1$ means that this item is unfolding item. Below we consider the case where the item type is not known a priori. The estimation simplifies greatly if π_i is fixed to 0 or 1 in advance.

3.2. THE MARGINALIZED BAYES MODAL ESTIMATION

Recall that a Bayesian approach produces a posterior distribution that depends on the contribution of both prior information about the parameters and information obtained from the sample of item response data. In IRT, an EM algorithm is typically applied to get the posterior estimates.

The posterior distribution given X is

$$g(\theta, \xi, \varphi, \eta|X) \propto L(X|\theta, \varphi)g(\theta|\xi)g(\xi)g(\varphi|\eta)g(\eta) \quad (3.2)$$

The $g(\theta|\xi)$ term is the conditional probability of an examinee's ability (θ) conditioned on the population parameters ξ where

$$\xi = \{\mu_\theta, \sigma_\theta^2\}$$

$g(\varphi_i|\eta)$ is the probability distribution of item parameters for i th item, which is contained in the φ_i vector where

$$\varphi_i = \{a_i, b_i, \gamma_i, \delta_i, t_i\}$$

conditional on the population parameters in vector η where

$$\eta = \{\mu_{al}, \sigma_{al}^2, \mu_b, \sigma_b^2, \mu_\gamma, \sigma_\gamma^2, \mu_\delta, \sigma_\delta^2, \mu_t, \sigma_t^2\}$$

In the 2PL model, the a_i parameters can be either positive (monotone increasing items) or negative (monotone decreasing items). We assign the absolute value of a_i a log-normal distribution which is different from Baker (1992). The transformation $al_i = \log|a_i|$ results in each al_i having a normal prior distribution. The sign ζ_i of a_i can be decided by the sign of the H coefficient. The normal prior distribution of each al_i is defined by its hyper-parameters μ_{al} and σ_{al}^2 . The difficulty parameter b_i follows the normal distribution with hyper-parameters μ_b and σ_b^2 (Baker 1992). Under the unfolding model setting, α_i and $-\tau_i$ are always positive and we assign them a log-normal prior. The similar transformation method is used where $\gamma_i = \log(\alpha_i)$ with hyper-parameters μ_γ and σ_γ^2 and $t_i = \log(-\tau_i)$ with hyper-parameters μ_t and σ_t^2 . The location parameter δ_i follows the normal distribution with hyper-parameters μ_δ and σ_δ^2 (Roberts et al, 2000). Typically, the prior distribution of θ is a normal distribution with common mean (μ_θ) and variance (σ_θ^2). In our simulation studies for this research, we assume θ follows the standard normal distribution. Then the IRF for P_u will become

$$P(X_{ij} = k | \theta_j, \varphi_i) = \frac{\exp(\exp(\gamma)(k(\theta_j - \delta_i) + \sum_{v=0}^k \exp(t_{iv}))) + \exp(\exp(\gamma)((M-k)(\theta_j - \delta_i) + \sum_{v=0}^k \exp(t_{iv}))}{\sum_{w=0}^K \exp(\exp(\gamma)(w(\theta_j - \delta_i) + \sum_{v=0}^w \exp(t_{iv})) + \exp(\exp(\gamma)((M-w)(\theta_j - \delta_i) + \sum_{v=0}^w \exp(t_{iv})))} \quad (3.3)$$

We use ζ_i to control the sign of a_i . Then P_m become

$$P(X_{ij} = k | \theta, (al_i, b_i)) = \frac{\exp(Dexp(al_i)\zeta_i(\theta_j - b_i))}{1 + \exp(Dexp(al_i)\zeta_i(\theta_j - b_i))} \quad (3.4)$$

Integrating Equation 3.2 over the probability distribution of ability $g(\theta|\xi)$ with respect to θ and item population parameters $g(\eta)$ with respect to η leads to a marginalized posterior distribution

$$g(\varphi, \xi) \propto \int \int L(X|\theta, \varphi)g(\theta|\xi)g(\xi)g(\varphi|\eta)g(\eta)d\theta d\eta \propto L(X|\varphi, \xi)g(\varphi)g(\xi) \quad (3.5)$$

$L(X|\varphi, \xi)$ is the marginal likelihood resulting from $L(X|\varphi, \theta)$ having been integrated with respect to ability. Then the parameters can be estimated by maximizing the log of Equation 3.5. That is, taking the partial derivatives of the log of Equation 3.5 with respect to each item's parameter and setting them equal to zero. The system of Bayesian estimation equations is given by

$$\frac{\partial \log L(X|\varphi, \eta)}{\partial \varphi_i} + \frac{\partial \log g(\varphi)}{\partial \varphi_i} + \frac{\partial \log g(\xi)}{\partial \varphi_i} = 0 \quad (3.6)$$

Since ξ is independent of φ_i , the last part is 0. Then Equation 3.6 becomes:

$$\frac{\partial \log L(X|\varphi, \eta)}{\partial \varphi_i} + \frac{\partial \log g(\varphi)}{\partial \varphi_i} = 0 \quad (3.7)$$

This equation is composed of two components. One is log-likelihood component and the other one is the prior distribution component.

3.3. THE LOG-LIKELIHOOD COMPONENT

Following Baker (1992) and Roberts et al. (2000), we let X_s be one of the S distinct response vectors for a given data set with $s = 1, 2, \dots, S$ and let x_{is} be the response to the i th item for response string X_s . r_s is the number of examinees whose response vector has the X_s pattern. Then under the assumption of the local independence, the conditional probability of observing a particular response vector X_s , given θ , is equal to

$$P(X_s|\theta) = \prod_{i=1}^I P(X_{is} = x_{is}|\theta) \quad (3.8)$$

Let $g(\theta)$ be the distribution function of θ . Then

$$P(X_s) = \int P(X_s|\theta)g(\theta)d\theta \quad (3.9)$$

Let N be the number of examinees and the likelihood function is as follows,

$$L(X|\varphi, \eta) = \frac{N!}{\prod_{s=1}^S r_s!} \prod_{s=1}^S P(X_s)^{r_s} \quad (3.10)$$

Then the log likelihood function is

$$\log L = \log(N!) - \sum_{s=1}^S \log r_s! + \sum_{s=1}^S r_s \log P(X_s) \quad (3.11)$$

Then under the two different models, we calculate the first-order derivatives of Equation 3.11 with respect to each parameter and then set these derivatives equal to 0. For example, for each item we obtain the al_i, b_i by assuming the item is monotone and maximizing that log-likelihood. Similarly, we obtain the estimates of γ_i, δ_i, t_i by assuming the item is unfolding. al_i, γ_i and t_i are the transformation of a_i, α_i and τ_i separately.

Mimicking Roberts et al. (2000), the general form of the first-order partial derivative of the log likelihood function which respect to a particular item parameter φ_i under dichotomous data setting, is given by

$$\begin{aligned} \frac{\partial \log L}{\partial \varphi_i} &= \sum_{s=1}^S \frac{1}{P(X_s)} \frac{\partial P(X_s)}{\partial \varphi_i} \\ &= \sum_{s=1}^S \frac{r_s}{P(X_s)} \int \frac{P(X_{is} = x_{is}|\theta)g(\theta)(x_{is} - P(X_{is} = 1|\theta))}{P(X_{is} = 1|\theta)(1 - P(X = 1|\theta))} \frac{\partial P(X_{is} = 1|\theta)}{\partial \varphi_i} d\theta \\ &\approx \sum_{f=1}^F \frac{\bar{r}_{if} - \bar{n}_{if}P(X_i = 1|x_f)}{P(X_i = 1|x_f)(1 - P(X_i = 1|x_f))} \frac{\partial P(X_i = 1|\theta = x_f)}{\partial \varphi_i} \end{aligned} \quad (3.12)$$

Where

$$\bar{n}_{if} = \sum_{s=1}^S \frac{r_s P(X_i = x_{is} | x_f) A(x_f)}{\bar{P}(X_s)} \quad (3.13)$$

$$\bar{r}_{if} = \sum_{s=1}^S \frac{r_s P(X_i = x_{is} | x_f) A(x_f) x_{is}}{\bar{P}(X_s)} \quad (3.14)$$

$$\bar{P}(X_s) = \sum_{f=1}^F P(X_s = x_s | \theta = x_f) A(x_f) \quad (3.15)$$

Gauss-Hermite integration has been applied to do the integration (See Appendix A). x_f is the a quadrature point and $A(x_f)$ is the rescaled density of $g(\theta)$ at x_f . The scale of $A(x_f)$ is such that

$$\sum_{f=1}^F A(x_f) = 1.$$

$\frac{\partial P(X_i=1|\theta=x_f)}{\partial \varphi_i}$ must be evaluated separately for each parameter to compute the associated first-order partial derivative. The derivation of this component is given in Appendix B for each item parameter under one of the two types of models. Let c be the number of parameter needed to estimated. The information function in the EM algorithm is given by

$$I(\varphi_{1i}, \dots, \varphi_{ci}) = \begin{pmatrix} I_{\varphi_{1i}\varphi_{1i}} & I_{\varphi_{1i}\varphi_{2i}} & \dots & I_{\varphi_{1i}\varphi_{ci}} \\ I_{\varphi_{2i}\varphi_{1i}} & I_{\varphi_{2i}\varphi_{2i}} & \dots & I_{\varphi_{2i}\varphi_{ci}} \\ \dots & \dots & \dots & \dots \\ I_{\varphi_{ci}\varphi_{1i}} & I_{\varphi_{ci}\varphi_{2i}} & \dots & I_{\varphi_{ci}\varphi_{ci}} \end{pmatrix} \quad (3.16)$$

Let P be the item response function for both model types. The elements of the information function matrix are equal to (Roberts et al. 2000):

$$I_{\varphi_i \varphi'_i} = \sum_{f=1}^F \frac{\bar{n}_{if}}{(P(X_i = 1|x_f)(1 - P(X_i = 1|x_f)))} \frac{\partial P(X_i = 1|v_f)}{\partial \varphi_i} \frac{\partial P(X_i = 1|v_f)}{\partial \varphi'_i} \quad (3.17)$$

Note that all of the priors are normally distributed or straight-forward transformations of a normal distribution. Thus the forms of their first-order derivatives are very similar with respect to each φ_i .

$$g(\varphi) = \frac{1}{\sqrt{2\pi\sigma_\varphi^2}} \exp\left[-\frac{1}{2}\left(\frac{\varphi - \mu_\varphi}{\sigma_\varphi}\right)^2\right] \quad (3.18)$$

Then,

$$\frac{\partial \log g(\varphi_i)}{\partial \varphi_i} = -\frac{(\varphi_i - \mu_\varphi)}{\sigma_\varphi^2} \quad (3.19)$$

Equation 3.7 becomes

$$\sum_{f=1}^F \frac{\bar{r}_{if} - \bar{n}_{if}P(X_i = 1|x_f)}{P(X_i = 1|x_f)(1 - P(X_i = 1|x_f))} \frac{\partial P(X_i = 1|\theta = x_f)}{\partial \varphi_i} - \frac{(\varphi_i - \mu_\varphi)}{\sigma_\varphi^2} = 0 \quad (3.20)$$

The information matrix will change also. The diagonal elements become

$$I_{\varphi_i\varphi_i} = \sum_{f=1}^F \frac{\bar{n}_{if}}{(P(X_i = 1|x_f)(1 - P(X_i = 1|x_f)))} \frac{\partial P(X_i = 1|v_f)}{\partial \varphi_i} \frac{\partial P(X_i = 1|v_f)}{\partial \varphi_i} + \frac{1}{\sigma_\varphi^2} \quad (3.21)$$

but the off-diagonal elements do not change since the distribution of different parameters are independent.

The EM algorithm, similar to that described by Muraki (1992) and Roberts et al. (2000), is used to solve the likelihood equations for a_i, b_i under the monotone model and γ_i, δ_i, t_i under the GGUM model. In the E stage of the algorithm, \bar{r}_{if} is computed for different item response function. $P(X_{is} = x_{is}|\theta)$ can be obtained from the observed response and the current item parameter estimates

$$P(X_{is} = x_{is}|\theta) = \pi_i P_u(X_{is} = X_{is}|\theta_j, \varphi_{iu}) + (1 - \pi_i) P_m(X_{is} = X_{is}|\theta_j, \varphi_{im}) \quad (3.22)$$

Where all the parameters are obtained from the previous iteration's estimates.

In the M step, we use Fisher's method of scoring to compute the most likely item parameter estimates for all items. In Fisher's method of scoring, the update function used to calculate the parameters on the q th iteration is given by

$$\begin{pmatrix} \hat{\varphi}_{1i} \\ \hat{\varphi}_{2i} \\ \dots \\ \hat{\varphi}_{ci} \end{pmatrix}_q = \begin{pmatrix} \hat{\varphi}_{1i} \\ \hat{\varphi}_{2i} \\ \dots \\ \hat{\varphi}_{ci} \end{pmatrix}_{q-1} + (I)^{-1} \begin{pmatrix} \frac{\partial \log L}{\partial \varphi_{1i}} \\ \frac{\partial \log L}{\partial \varphi_{2i}} \\ \dots \\ \frac{\partial \log L}{\partial \varphi_{ci}} \end{pmatrix} \quad (3.23)$$

When we get the parameter estimates for each item for each type in the q th iteration, we compute the estimated π_i for each item where π_i is the probability of i th item to be unfolding. Let L be the likelihood,

$$\pi_i = \frac{L_u}{L_u + L_m} \quad (3.24)$$

In the q th iteration, these two likelihood functions are as follows

$$\begin{aligned} L_{qu} = & \frac{N!}{\prod_{s=1}^S r_s!} \int \prod_{i \neq h, h_1^I} (\pi_{(q-1, h)} P_u(X_{hj} = x_{hs} | \theta) \\ & + (1 - \pi_{(q-1, h)}) P_m(X_{hj} = x_{hs})) P_u(X_{ij} | \theta) d\theta \end{aligned} \quad (3.25)$$

$$\begin{aligned} L_{qm} = & \frac{N!}{\prod_{s=1}^S r_s!} \int \prod_{i \neq h, h_1^I} (\pi_{(q-1, h)} P_u(X_{hj} = x_{hs} | \theta) \\ & + (1 - \pi_{(q-1, h)}) P_m(X_{hj} = x_{hs})) P_m(X_{ij} | \theta) d\theta \end{aligned} \quad (3.26)$$

Then for the $(q + 1)$ th iteration,

$$\pi_{q+1, i} = \frac{L_{qu}}{L_{qu} + L_{qm}} \quad (3.27)$$

The completion of a single E stage followed by a single M stage constitutes one cycle of the EM algorithm. Additional cycles are conducted until the largest change in any item parameter estimate from one cycle to the next is arbitrarily small (< 0.0005 , Roberts et al. 2000). Usually, for a data set with 40 items and 1000 examinees, it takes

7 days to run the EM algorithm (CPU Intel(R) Core(TM) i5 M520 @ 2.40GHz) when it is coded in R. It is expected to improve drastically if programmed in FORTRAN or a similar language.

The EM algorithm requires carefully selecting the initial estimates for every parameter to avoid a local maxima. The initial estimates for parameters are discussed in Chapter 4.

CHAPTER 4

THE INITIAL ESTIMATES OF PARAMETERS IN MIXTURE MODEL AND SIMULATION STUDY

The EM algorithm requires carefully selecting the initial estimates for every parameter to avoid local maxima (Roberts et al., 2000). In this chapter, we investigate the appropriate choice of the initial parameter values.

4.1. THE FORMULAS FOR INITIAL ESTIMATES

For monotone items, the a_i and the b_i parameters were estimated using the p-value and biserial correlations (Table 2.13). All the initial estimates of the a_i s and the b_i s were calculated from the original data set. The formulas for calculating these initial values can be found in Equation 4.1, 4.2, and 2.5. Since there are two types of monotone items (increasing and decreasing), our initial estimation method is different from what was traditionally used for the 2PL model. In this chapter, we calculated the transformed score for all the items with negative H_i values. This transformed score was applied for the initial estimates for π_i . The symbol, al_i , is the log of $|a_i|$. Under the assumption that θ follows the standard normal distribution, the initial estimates of the a_i for the 2PL model and the b were calculated based on the original data set.

$$\hat{a}_i = \frac{\hat{\rho}_{bis}}{\sqrt{1 - \hat{\rho}_{bis}^2}} \quad (4.1)$$

Then

$$al_i = \log(|a_i|) \quad (4.2)$$

and

$$|a_i| = \exp(al_i) \quad (4.3)$$

We use ζ_i to control the sign of a_i . Then

$$a_i = \exp(al_i)\zeta_i \quad (4.4)$$

The difficulty parameter b can be estimated based on the original data set by Equation 2.5.

The initial parameter estimates for the GGUM came from the constrained version of GGUM (Roberts, 1996). For the dichotomous case of this constrained model, there is only one τ_i parameter that needs to be estimated. “The initial estimate of τ_i is obtained by first setting all $\theta_j = 0$ and then setting the δ_i value of the most frequently endorsed item to 0. Consequently, the estimate of τ must be large enough to account for the proportion of endorsements observed for the most frequently endorsed item” (Roberts, 1996). Presuming that item h is the most frequently endorsed item which is located at $\delta_h=0$ and all $\theta_j = 0$, the expected proportion of endorsement for item h is

$$P(X_h = 1|\theta_j) = \frac{\exp(-\tau_i) + \exp(-\tau_i)}{\exp(0) + \exp(-\tau_i) + \exp(-\tau_i) + \exp(0)} = \frac{\exp(-\tau_i)}{1 + \exp(-\tau_i)} \quad (4.5)$$

Now, let the x_h be the item score of item h . Then set the observed proportion of this item equal to the expected proportion.

Then solve for τ_i as

$$\frac{\exp(-\tau_i)}{1 + \exp(-\tau_i)} = \frac{x_h}{N} \quad (4.6)$$

and

$$\tau_i = -\ln \frac{x_h}{N - x_h} \quad (4.7)$$

This estimate of τ_i can be used to derive the initial estimate of δ_i . If it is assumed that all $\theta_j = 0$ and that the location of the item with the largest proportion of

endorsements is $\delta_h = 0$. Then the absolute value of this item's location (δ_i) can be obtained by solving the following equation with respect to δ_i (for $i \neq h$):

$$P(X_i = 1|\theta_j) = \frac{x_i}{N} = 0 = \frac{\exp(-\delta_i - \tau_i) + \exp(-2\delta_i - \tau_i)}{\exp(0) + \exp(-\delta_i - \tau_i) + \exp(-2\delta_i - \tau_i) + \exp(-3\delta_i)} - \frac{x_i}{N} \quad (4.8)$$

where x_i refers to the item score for item i (Roberts, 1996). A bisection algorithm has been applied to numerically solve this equation since there is no closed form (Roberts, 1996). This solution only gives the absolute value of the δ_i . “The sign of each estimate is obtained from the signs of the pattern loadings from the first principal component of the inter item correlation matrix. Davison (1977) showed that these signs correspond to the direction of the item (the sign of discrimination parameter) when responses exhibit an unfolding structure” (Roberts, 1996).

4.2. THE SIMULATION STUDY FOR THE INITIAL VALUE OF π

The p-value/biserial method can be applied to obtain the initial estimates of probability π_i . However in Chapter 2, we assumed that there is existing information about the items' separation between unfolding and monotone model types. This section will study if there exists a rule without this information. Several plots were utilized to compare the relationship of the p-value and biserial correlation between the original data set and transformed data set similarly as in Chapter 2. We use these graphs to construct a second ad-hoc rule about identifying item types. Based on this rule, the items can be assigned as either monotone items, indeterminant items or unfolding items. The initial estimates of π_i for these three classes were set to 0, 0.5, and 1 respectively. Later, a further discussion about the robustness of the initial π_i estimate is conducted.

We randomly picked 40 paired items from the item bank (Section 2.2) with 1 – 20 being monotone and 21 – 40 being unfolding. The “mixed exam” data set was generated with $N = 1000$. In Chapter 2, assuming the separation between monotone items and unfolding items were known and only the score for monotone items with the negative H coefficients were transformed, the p-value/biserial method provided an ad-hoc rule to get an idea of the item type for a subset of the items. The pattern in the graph was not clear without the information about the separation into monotone items and unfolding items (Figure 2.5). However, some conclusions can still be achieved after comparison of several graphs below. In this simulation study, the transformed score was calculated for any item with negative H value, including the unfolding items.

Based on the formula on the right hand side of Table 2.13, we used the true parameters and simulated data set to compute the p-value and estimated b value, which leads to Figure 4.1 to Figure 4.3. In each case, “m” stands for monotones items and “u” stands for unfolding items. Figure 4.1 is the comparison between the p-values and biserial correlations using the true parameter values. The unfolding

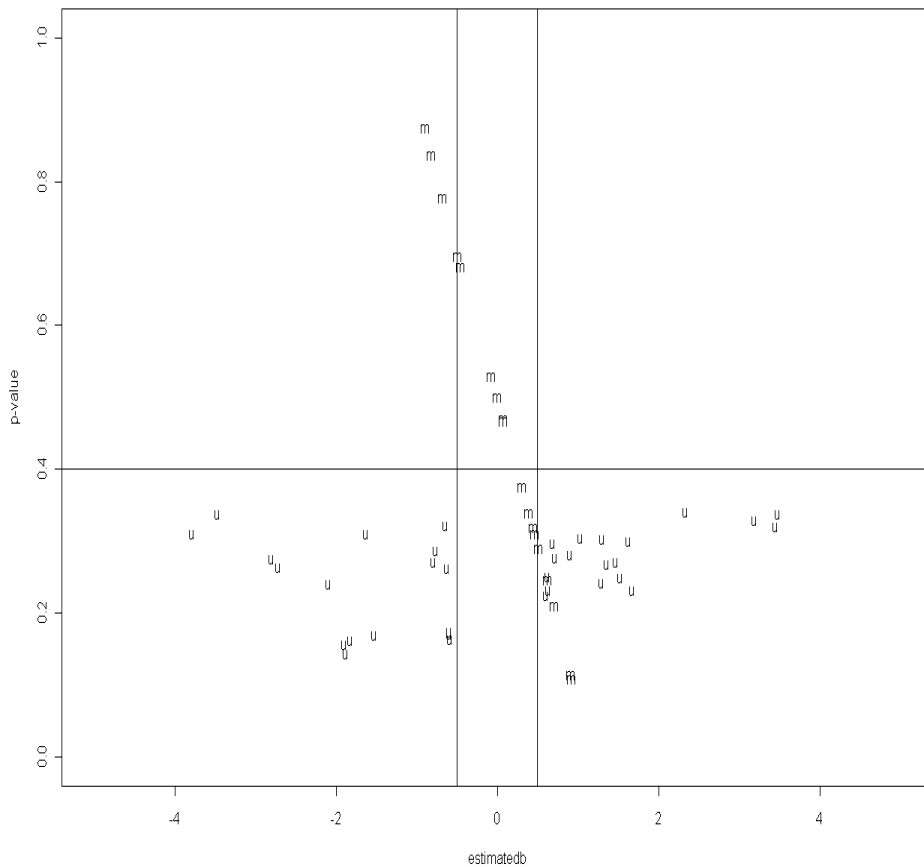


FIGURE 4.1. Theory Result by Using the P-value/Biserial Method

items and monotone items in this plot are clearly separated except for the area in the lower-right corner. Figure 4.2 uses the untransformed simulation data set to calculate the p-value and estimated b value. In the left hand side of Figure 4.2, though all the “m”s and “u”s are mixed up when the p-value is below the 0.4 line, there are only “m” above the 0.4 line which means that all the items above the 0.4 line are monotone items. The graph on the right hand side contains the item numbers. All items with p-value above 0.4 and estimated b between -1 and 1 are monotone items, including item 2, 4, 5, 7, 8, 9, 11, 12, 16, 17, 20.

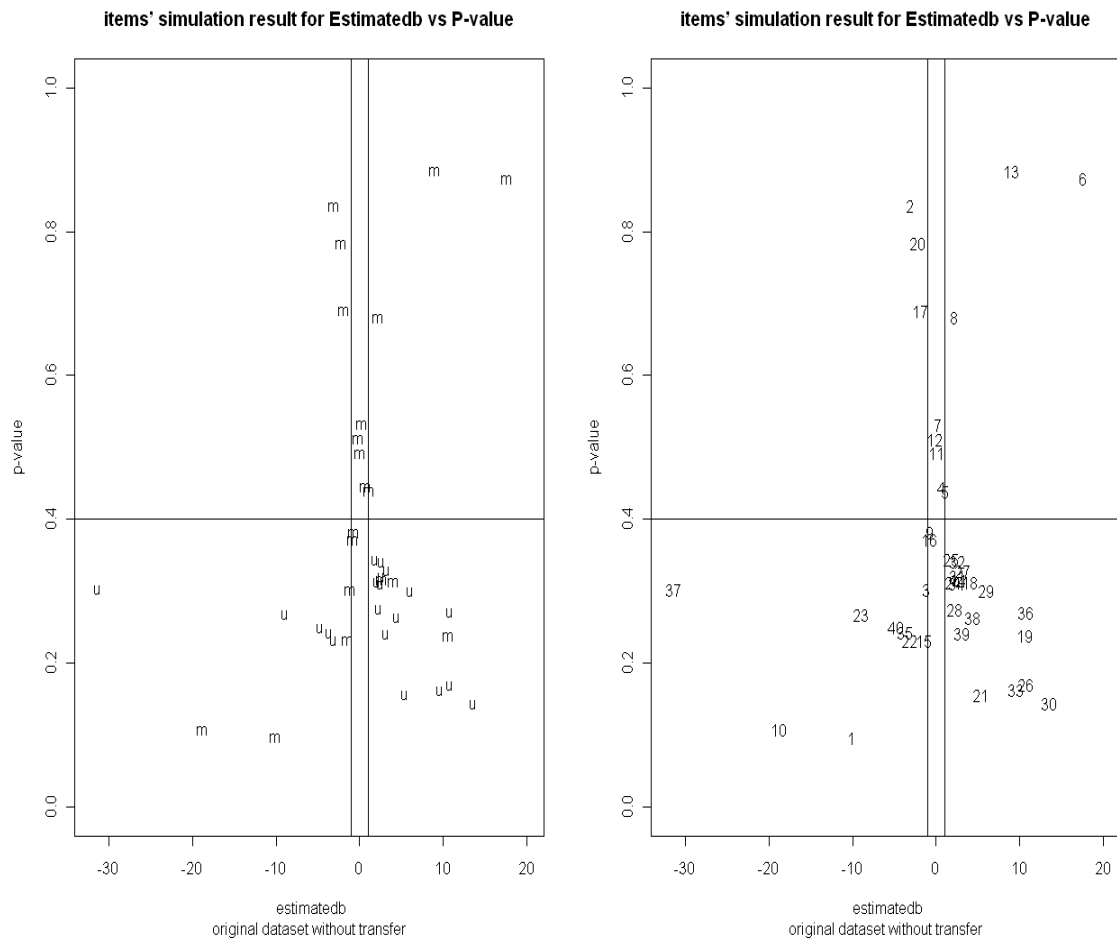


FIGURE 4.2. P-value vs Estimated b Value for the Original Data Set

Figure 4.3 was constructed using the transformed score which recoded all the items with negative H values. In this graph, the items on the lower left corner and the items that are the furthest away from the other items are unfolding items. All the items with estimated b value between -1 to 1 are monotone items. After recoding all the item scores for negative H value, the p-value and estimated b value were calculated. In this graph for the comparison of p-value and estimated b value for transformed score, items on the lower left corner and further away from the other items are unfolding items, 24, 27, 31, 32, 35, 36, 37, 40. All the items in the middle are monotone items, 3, 4, 5, 7, 8, 9, 11, 12, 14, 15, 16, 17, 18, 19.

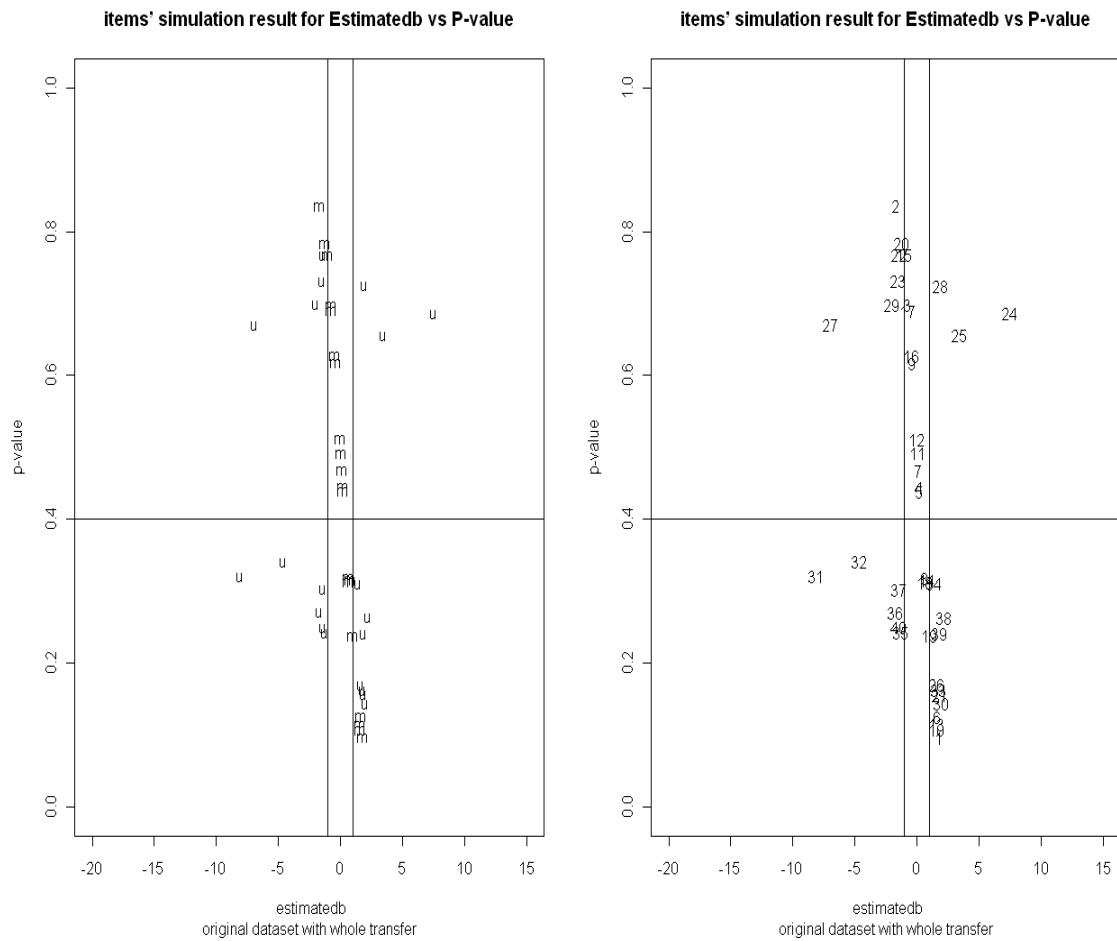


FIGURE 4.3. P-value vs Estimated b Value for the Transformed Data Set

Therefore, another ad-hoc rule based on the Figure 4.1 to Figure 4.3 is

- In the p-value vs estimated b plot with the original data set, the items with p-value above 0.4 are monotone items.
- In the p-value vs estimated b plot with the original data set, the items with estimated b value between -1 to 1 are monotone items
- In the p-value vs estimated b plot with the transformed data set, the items with p-value below 0.4 and estimated b value below -1 are unfolding items.

Applying this rule, the 40 items can be grouped as

(1) Monotone item: 2, 3, 4, 5, 7, 8, 9, 11, 12, 14, 15, 16, 17, 18, 19, 20

(2) Unfolding item: 24, 27, 31, 32, 35, 36, 37, 40

(3) indeterminant item: 1, 6, 10, 13, 21, 22, 23, 25, 26, 28, 29, 30, 33, 34, 38, 39

For the monotone items, we assigned 0 as the initial estimates for π_i , 1 for unfolding items. For the other uncertain items, probability of 0.5 was assigned.

4.3. CLASSIFICATION TREES FOR THE P-VALUE/BISERIAL METHOD

In Chapter 2 and Chapter 4, we proposed the P-value/biserial method based on the comparison of the graphs from a limited number of simulation studies. In this section, we use classification tree methodology (e.g. Breiman, Friedman, Olshen, Stone, 1984) to examine the prediction rules for item type using the p – *value* and *estimatedb* value for larger amounts of simulated data.

A classification tree is designed to predict the class membership of various observations based on the values of their corresponding independent variables. The classification rule is represented by a tree where each of the ends of the tree (the leaves) give the estimated membership for any observations which belong to that leaf. Given a training data set, at each stage the methodology can be thought of as asking the observations at that stage a question about their features. The feature that best separates the data into its true category (say one of the independent variables is greater than some value) is chosen and a branching occurs based on that question. This continues until all of the observations have been classified as well as possible given the data, or until some stopping rule is met. As such, a tree can be grown to be quite large, perhaps perfectly fitting the data and having only one observation on each leaf. However, this may simply be the result of overfitting (Moisen, 2008). An appropriate tree size can be determined in a number of ways. In this work, we use the cross-validated error rate to trim the tree down to an appropriate size.

In our study, the categorical variable being predicted is the item type: “m” for the monotone items and “u” for unfolding items. The independent variables are the p value and *estimatedb* value. The Rpart package (Therneau, Atkinson, and Ripley, 2006) in R was utilized to construct the trees.

The item parameters used for this section are from Section 2.2 where the a s are in $(-1.5, -0.9, 0.9, 1.5)$ and b s are in $(-1.5, -1, 0, 1, 1.5)$. The α parameters for the unfolding item are in $(0.5, 1, 1.5, 2)$, δ s are in $(-1.5, -1, 0, 1, 1.5)$ and τ s are in $(-0.2, -0.4, -0.6)$. The 20 monotone items are constructed by the combination of a , b

parameters and another 20 items are built by the combination of α and δ parameters. So, there are 40 items in each mixed exam. Further, each of these item sets was crossed with the three possible values of τ (one value for the entire exam). Moreover, the θ has two distributions: standard normal distribution and uniform distribution $[-0.3, 0.3]$ with N from 500, 1000, 2000 to 4000. For each item set, 100 data sets are generated for both distributions of θ and for different N . Then the p value, estimated b value, p value under transformed data (pvt) and estimated b value under transformed data ($estimatedbt$) are calculated for each item. These values are combined for each N . The classification trees are built to predict the item type for different N s (Figures 4.4 to 4.7). The classification trees for the original data are relatively simple. The trees for the transformed data are much more complex, with extra nodes only leading to gradual improvement. In this case, a relative error rate of 0.5 was used to trim the tree. Items located on trimmed branches are marked as being of uncertain type, and are indicated by a “C” in later analysis. The graphs on the right hand side of Figures 4.4 to 4.7 are trimmed for the transformed data. The whole tree structure and the error rate for the transformed data are reported in Appendix *G*.

Take Figure 4.5 for example. The graph on the left hand side is the classification tree for the original data set for $N = 1000$ and the one on the right hand side is the one for the transformed data set. The decision rules for the original data set are:

The monotone items are located at

- $estimatedb < -5.506$ and $p \geq 0.3775$;
- $estimatedb$ is between $[-5.506, -0.2874]$;
- $estimatedb \geq 0.2008$ and $p \geq 0.4925$.

The unfolding items are located at

- $estimatedb < -5.506$ and $p < 0.3775$;
- $estimatedb$ is between $(-0.2874, 0.2008)$ and $p \geq 0.4925$;
- $estimatedb > -0.2874$ and $p < 0.4925$.

In order to limit the complexity and maintain ease of comparison with the other rules, for the larger trees generated from the transformed data the number of splits that gave a relative error less than 0.5 was used (in Appendix G). So the decision rule for the transformed data is:

The monotone items are located at:

- *estimatedb* is between $[-1.68, -0.8278)$ and $p \geq 0.7645$;
- *estimatedb* is between $[-0.8278, 0.7103)$ and $p \geq 0.6225$.

The unfolding items are located at:

- *estimatedb* ≥ 0.7103 and $p \geq 0.2885$;
- *estimatedb* is between $[-1.68, -0.8278)$ and $p \geq 0.7645$;
- *estimatedb* ≤ -0.8278 and $p < 0.7645$.

For the items located outside the decision rules, they named as uncertain items and in later analysis, "c" is assigned.

Let us compare these two decision rules with the ad-hoc rule in Section 4.2. Review the decision rule of Section 4.2.

- In the p-value vs estimated *b* plot with the original data set, the items with p-value above 0.4 are monotone items.
- In the p-value vs estimated *b* plot with the original data set, the items with estimated *b* value between -1 to 1 are monotone items.
- In the p-value vs estimated *b* plot with the transformed data set, the items with p-value below 0.4 and estimated *b* value below -1 are unfolding items.

Some of the decisions for the monotone items from these two rules overlap: the items with $p > 0.4925$ and the items with *estimatedb* between $(-1, -0, 2874)$ are the monotone items under both rules. The third rule in the ad-hoc rule overlaps with the second rule for the transformed data set: the items with *p - value* below 0.4 and *estimatedb* value below -1 are the unfolding items. As the ad-hoc rules are of a form that could have been selected by the classification tree, it is expected that the

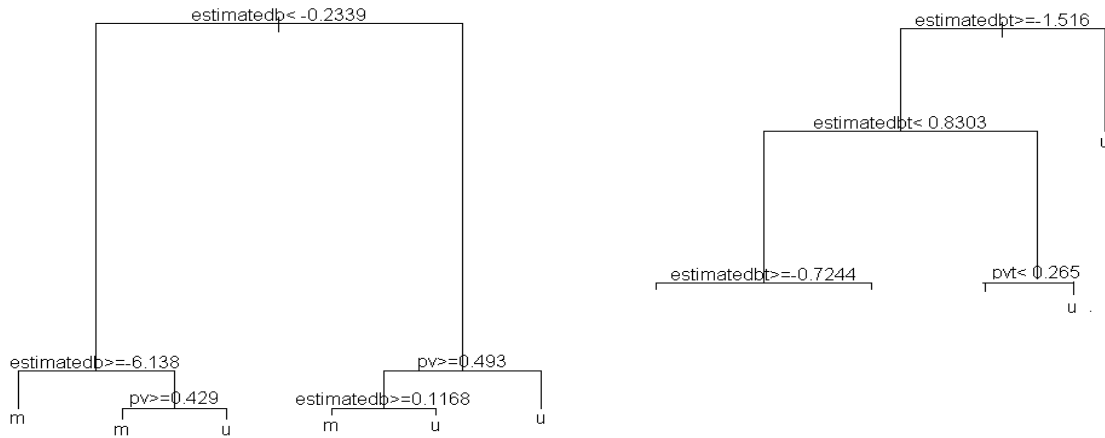


FIGURE 4.4. Classification Tree for $N = 500$ with the Original and the Transformed Data

classification tree result should be more accurate. Chapter 5 includes a simulation study to compare these decision rules to each other as well as the others that have been discussed.

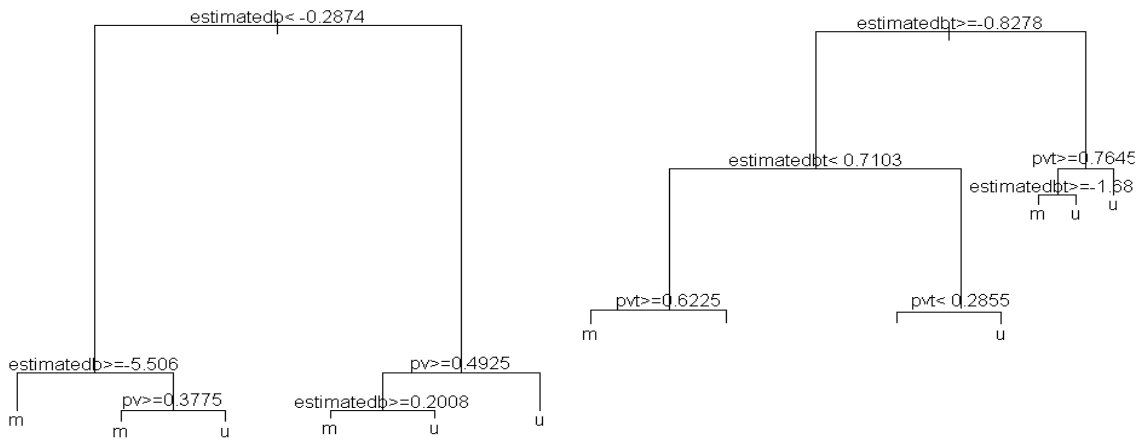


FIGURE 4.5. Classification Tree for $N = 1000$ with the Original Data and the Transformed Data

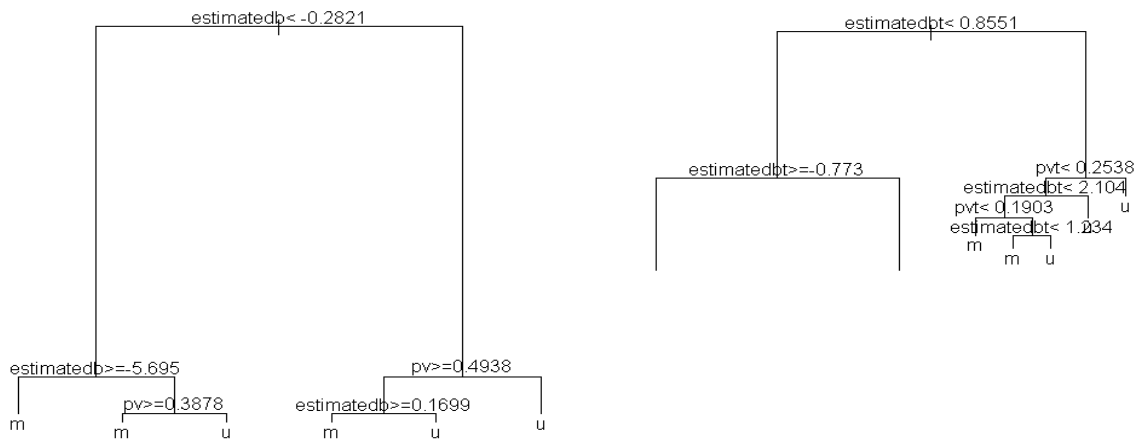


FIGURE 4.6. Classification Tree for $N = 2000$ with the Original Data and the Transformed Data

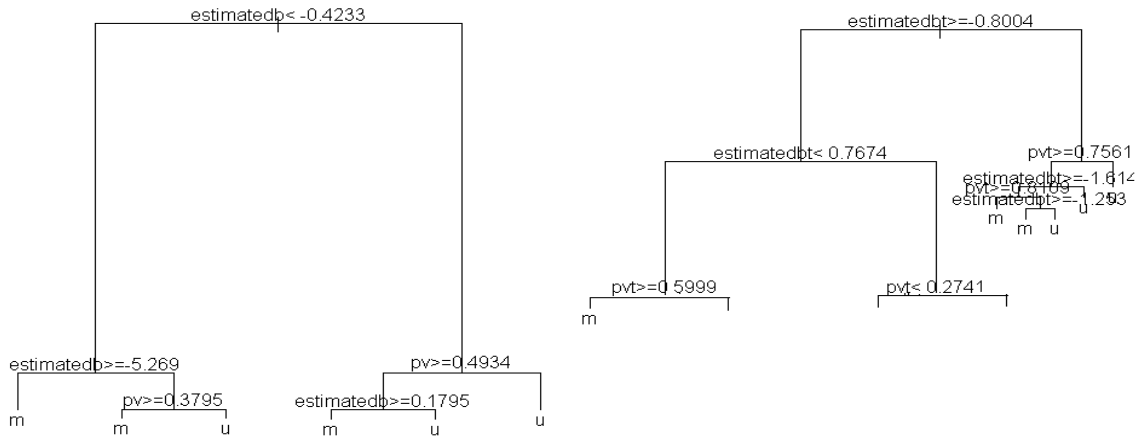


FIGURE 4.7. Classification Tree for $N = 4000$ with the Original Data and the Transformed Data

4.4. THE STUDY ABOUT INITIAL ESTIMATED π_i

In last section, we used the ad-hoc rule to get the initial estimate of π_i . Since the ad-hoc rule was concluded from a simulation study, there is no strong theoretical support arriving in these cut-off points for the estimated π -value. A resulting question is: Does the marginalized Bayes modal estimation still work if the initial π were badly misspecified?

We examined this by a simulation study which applied different initial π s to the data set. 20 items were randomly picked, 10 monotone items and 10 unfolding items from the item bank in Chapter 2. Assuming that θ follows the standard normal distribution, a data set was generated for these 20 items with $N = 1000$. In the first simulation, we set the initial $\pi_i = 0$ for all items. The estimated results are in Tables 4.1 and 4.3. The second simulation was conducted with the same data but with the initial $\pi_i = 1$ for all items. Those results are in Tables 4.2 and 4.4.

The estimates for both sets of initial values are similar and close to the true estimates. The π estimates are close in both simulations. The only difference was the running time as seen by comparing the times acquired here to those in Chapter 5. The simulation with the initial π close to the true π will take less time to converge, saving up to 1/3 of the computing time. However it does not affect the final estimation quality.

TABLE 4.1. Parameter Estimation for the Monotone Items when the Initial $\pi = 0$

Item	a	b	Initial π	Estimated a^*	Estimated b^*	Estimated π	true π
1	1.2	0.7	0	1.3718	0.5859	0.44	0
2	1.5	-0.4	0	1.8043	-0.4067	0.54	0
3	1.3	-0.5	0	1.4702	-0.4702	0.47	0
4	-1.2	-0.5	0	-1.2557	-0.6375	0.54	0
5	-1.1	-0.8	0	-1.2559	-0.7738	0.67	0
6	1.4	0	0	1.3413	0.0087	0.59	0
7	1	-0.9	0	1.0330	-0.7459	0.53	0
8	-1.3	-0.4	0	-1.5177	-0.2932	0.51	0
9	0.9	-0.8	0	1.0877	-0.7896	0.58	0
10	1.4	1	0	1.3643	1.0277	0.49	0

* The results above are reported after conducting the mean/mean linking.

TABLE 4.2. Parameter Estimation for the Monotone Items when the Initial $\pi = 1$

Item	a	b	Initial π	Estimated a^*	Estimated b^*	Estimated π	true π
1	1.2	0.7	1	1.3572	0.5754	0.48	0
2	1.5	-0.4	1	1.8424	-0.4203	0.55	0
3	1.3	-0.5	1	1.4606	-0.4878	0.47	0
4	-1.2	-0.5	1	-1.2297	-0.6608	0.53	0
5	-1.1	-0.8	1	-1.2921	-0.7757	0.47	0
6	1.4	0	1	1.3180	-0.0060	0.60	0
7	1	-0.9	1	1.0133	-0.7701	0.53	0
8	-1.3	-0.4	1	-1.5215	-0.3089	0.54	0
9	0.9	-0.8	1	1.1115	-0.7927	0.52	0
10	1.4	1	1	1.3555	1.0178	0.49	0

* The results above are reported after conducting the mean/mean linking.

TABLE 4.3. Parameter Estimation for the Unfolding Items when the Initial $\pi = 0$

Item	α	δ	Initial π	Estimated α^*	Estimated δ^*	Estimated π	True π
11	1.84	0.7	0	1.8576	0.7370	1.00	1.00
12	2.25	-0.4	0	2.2696	-0.3348	1.00	1.00
13	2.00	-0.5	0	1.7931	-0.5347	1.00	1.00
14	1.87	-0.5	0	1.5946	-0.5500	1.00	1.00
15	1.70	-0.8	0	1.7186	-0.7957	1.00	1.00
16	2.15	0	0	2.2802	0.0119	1.00	1.00
17	1.55	-0.9	0	1.4995	-0.8564	1.00	1.00
18	2.01	-0.4	0	1.6552	-0.4641	1.00	1.00
19	1.44	-0.8	0	1.2273	-0.9064	1.00	1.00
20	2.01	1	0	1.7223	0.9881	1.00	1.00

* The results above are reported by conducting the mean/mean linking.

TABLE 4.4. Parameter Estimation for the Unfolding Items when the Initial $\pi = 1$

Item	α	δ	Initial π	Estimated α^*	Estimated δ^*	Estimated π	True π
11	1.84	0.7	1	1.8576	0.7504	1.00	1.00
12	2.25	-0.4	1	2.2696	-0.3215	1.00	1.00
13	2.00	-0.5	1	1.7930	-0.5212	1.00	1.00
14	1.87	-0.5	1	1.5945	-0.5365	1.00	1.00
15	1.70	-0.8	1	1.7185	-0.7822	1.00	1.00
16	2.15	0	1	2.2801	0.0253	1.00	1.00
17	1.55	-0.9	1	1.4994	-0.8430	1.00	1.00
18	2.01	-0.4	1	1.6552	-0.4507	1.00	1.00
19	1.44	-0.8	1	1.2272	-0.8931	1.00	1.00
20	2.01	1	1	1.7222	1.0015	1.00	1.00

* The results above are reported by conducting the mean/mean linking.

CHAPTER 5

SIMULATION STUDY FOR MIXTURE MODEL

In this chapter, we discuss the effectiveness of the mixture model with the marginalized Bayes modal estimation algorithm. One simulation for 20 monotone items and 20 unfolding items is conducted and the result is analyzed in section 5.1. Then in section 5.2, we discuss the reverse sign problem which was found during the simulation study. Section 5.3 utilizes five settings to explain the reason why the estimated π_i s are always close to 0.5 for monotone items. Finally, Section 5.4 examines all the methods discussed so far in through simulation: the manifest monotonicity test, the p-value/biserial method, and the mixture model.

5.1. THE MIXTURE MODEL ESTIMATION SIMULATION STUDY

As a first investigation into the estimation of the mixture model with the marginalized Bayes modal estimation algorithm, one mixed exam containing 20 monotone items and 20 unfolding items was simulated with the parameters being randomly taken from the item bank (Section 2.2, first simulation). 1000 examinees were randomly generated from the standard normal distribution. The initial estimates for all item parameters in the mixture model were calculated by the formula in Chapter 4. Then the marginalized Bayes modal estimation algorithm was applied to find the estimates for all the parameters in our model.

Table 5.1 and Table 5.2 list all the simulation estimates vs. the true parameters. In the Table 5.1, the first 4 columns list all the true parameters for the 20 monotone items and the next three columns are the estimates from marginalized Bayes modal

estimation. The last column is the true π value we want to achieve. The true parameters and estimates parameters for the unfolding items are listed in Table 5.2. All of the estimates have been placed on the simulated parameter scale using mean/mean linking (Section 1.4). The linked values are indicated by an asterisk.

The signs for all the estimates for parameters were originally reversed from the true parameters because there are more monotone items with negative a parameters which caused the scale switch. They were corrected to the simulation scale in the linking. The estimates for all the item parameters (estimated a , b , α , δ) are fairly accurate. The differences between the value of estimated a^* and the value of the true a are close to or less than 0.1 except for item 6 which has a large a value of 1.5. The estimated b values are also fairly close to their true values (given their scale), with the largest difference between being 0.39. The estimation for α are fairly good (the differences are around 0.1) except for item 35 (0.35) and item 37 (0.24). The Estimated τ_i s are all very close to the true values across all the items. The estimated probability of being an unfolding item were all items are all 1 or very close to 1 for items that were actually unfolding. However, the estimates of π for the monotone items were very close to 0.5 instead of the desired value 0. This is discussed more in Section 5.3. .

TABLE 5.1. Parameter Estimation for the Monotone Items

heightItem	a	b	Est a^*	Est b^*	Est π	true π
1	-1.4	-1.5	-1.4864	-1.3194	0.5207	0
2	-1	1.4	-0.9910	1.0361	0.4986	0
3	1.3	0.7	1.2743	0.4602	0.5017	0
4	-1.4	-0.1	-1.3970	-0.1987	0.4693	0
5	-1.1	-0.1	-1.0678	-0.2297	0.5475	0
6	1.5	-1.4	1.2918	-1.2236	0.4999	0
7	1.2	0	1.2896	-0.1443	0.4997	0
8	1.3	-0.6	1.3331	-0.5287	0.4991	0
9	1.3	0.4	1.3219	0.2354	0.5049	0
10	-1.5	-1.5	-1.6760	-1.2157	0.4923	0
11	-1.1	0	-1.2187	-0.0843	0.4372	0
12	-1.1	0.1	-1.0344	-0.0306	0.5037	0
13	1.5	-1.4	1.5696	-1.2019	0.4999	0
14	-0.9	-0.7	-0.8320	-0.6541	0.6160	0
15	1.4	1	1.5592	0.6295	0.4223	0
16	1.5	0.5	1.4100	0.2547	0.4921	0
17	-1.1	0.7	-1.0598	0.4758	0.5011	0
18	-1.5	-0.6	-1.4307	-0.5334	0.5066	0
19	-1.2	-0.9	-1.1098	-0.8254	0.5008	0
20	-1.2	1	-1.3066	0.7223	0.4999	0

* The values for the 2PL items were estimated without including the $D = 1.7$ term for the exam in Table 5.1 and 5.2. The results above are reported by taking the originally estimated a divided by 1.7 and conducting the mean/mean linking.

TABLE 5.2. Parameter Estimation for the Unfolding Items

Item	α	δ	τ_1	Est α	Est δ	Est τ_{i1}	Est π	true π
21	1.7905	-1.5	-0.2	1.6914	-1.4328	-0.0678	1	1
22	1.4073	1.4	-0.2	1.4208	1.3225	-0.1501	0.9994	1
23	1.9632	0.7	-0.2	2.0327	0.7508	-0.1811	1	1
24	2.1510	-0.1	-0.2	2.0123	-0.0901	-0.1520	1	1
25	1.7749	-0.1	-0.2	1.6377	-0.1379	-0.1582	1	1
26	1.9503	-1.4	-0.2	1.9221	-1.4407	-0.2388	1	1
27	1.9041	0	-0.2	2.1185	0.0510	-0.2173	1	1
28	1.9811	-0.6	-0.2	1.7677	-0.6279	-0.1083	1	1
29	2.0082	0.4	-0.2	1.9840	0.3223	-0.1442	1	1
30	1.8920	-1.5	-0.2	1.8066	-1.4265	-0.0555	1	1
31	1.7761	0	-0.2	1.7897	0.0701	-0.1272	1	1
32	1.7749	0.1	-0.2	1.7506	0.0959	-0.1767	1	1
33	1.9503	-1.4	-0.2	1.7101	-1.3726	-0.0688	1	1
34	1.4567	-0.7	-0.2	1.6060	-0.7129	-0.2174	1	1
35	2.0068	1	-0.2	2.3558	0.9374	-0.2492	1	1
36	2.2369	0.5	-0.2	2.1562	0.5252	-0.1377	1	1
37	1.7154	0.7	-0.2	1.4738	0.6670	-0.0932	1	1
38	2.2206	-0.6	-0.2	2.0092	-0.5603	-0.1108	1	1
39	1.7977	-0.9	-0.2	1.9304	-0.7913	-0.1176	1	1
40	1.7716	1	-0.2	1.5326	1.1146	-0.1413	1	1

5.2. SIGN REVERSAL PROBLEM

In the simulation study in Section 5.1, the pre-linked signs of the a , b and δ parameters are reversed when compared with the true parameters. However, this is not always the case since it depends upon the sign of true parameters. If the negative monotone items with negative a dominate the whole item setting such as in Section 5.1 (where 12 out of 20 are negative monotone items), the estimated ability scale is reversed. If we took out all the monotone items with $\pi = 0.5$ and checked their biserials, the signs of biserials are the same as the sign of a parameters. For simulation, we can simply reverse all the signs as needed. In a real data set, the true sign is unknown or arbitrary. We suggest studying the content of the item questions and comparing them with the distribution of the score. Then it is easier to decide the direction of scale. For example, for an item like “Abortion is unacceptable under any circumstances”, negative estimates for a value means the scale is from disagreement with legal abortion to agreement with legal abortion. Positive a estimates means the scale is from agreement with legal abortion to disagreement with legal abortion. It is arbitrary from a statistical point of view.

5.3. QUESTION OF ESTIMATED π FOR THE MONOTONE ITEMS

In Section 5.1, we notice that the estimated π values for monotone items are always close to 0.5. This means that the estimated likelihoods of this item being monotone or unfolding are approximately the same. In this section, we created five different settings to investigate this problem.

- Setting 1: An exam consisting of 20 monotone items.
- Setting 2: An exam consisting of 20 unfolding items.
- Setting 3: An exam consisting of 18 monotone items and 2 unfolding items.

In these 2 unfolding item, one is located in the middle and other is located in the extreme place (one item with δ value in the middle and other item with δ in the extreme).

- Setting 4: An exam consisting of 18 unfolding items and 2 monotone items with one in the middle and one in the extreme.
- Setting 5: An exam consisting of 10 monotone items and 10 unfolding items.

For each setting, we obtained the estimates for each parameter. In addition to comparing the estimates to each other in tabular form, a comparison for each item can be made based on its IRF for both monotone estimates and unfolding estimates. Table 5.3 contains the estimates for first setting for 20 monotone items. The estimates for monotone items are close to the true estimates and the estimated π are around 0.5. Figure 5.1 shows the IRFs for 4 items (The one in blue is the IRF with the monotone estimates and the red one is the IRF with the unfolding estimates for this item). The IRFs for items 1, 11 and 17 overlap in the middle and are only separated at the extremes. The IRFs for Items 19 overlap for most of the range of ability. Table 5.4 contains the results for Setting 2 with 20 unfolding items in the data set. Estimated π s are 1 or close to 1. The IRFs for some of the items are in Figure 5.2.

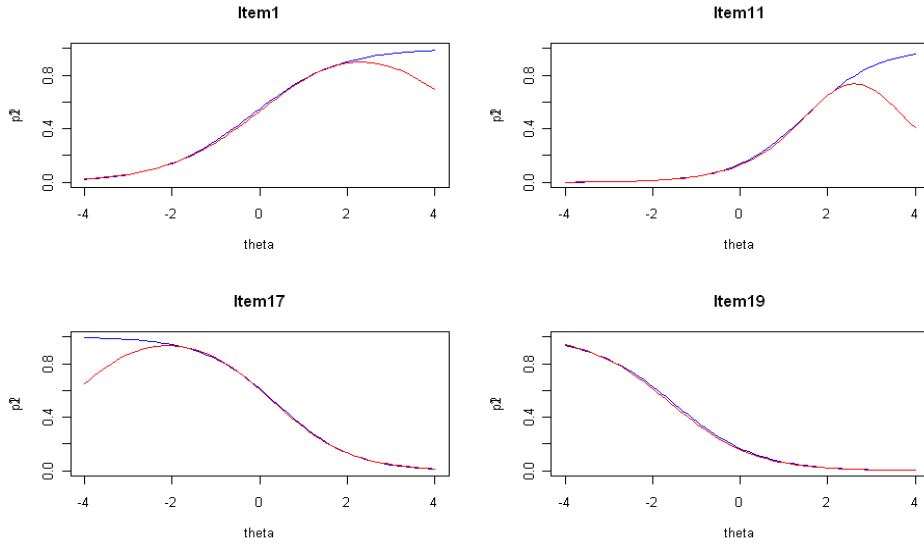


FIGURE 5.1. IRFs for a Few Items under Setting 1

TABLE 5.3. Parameter Estimation for Setting 1

Item	a	b	Est a^*	Est b^*	Est α	Est δ	Est τ	Est π
1	1.2	-0.2	1.0275	-0.1492	0.9238	2.3265	-2.3451	0.44
2	-0.9	-0.9	-0.8760	-1.1794	0.7803	-2.8745	-1.5308	0.51
3	1	0.1	0.9086	0.1578	0.8158	2.7766	-2.4783	0.49
4	-1.5	-0.9	-1.3887	-1.0600	1.2773	-2.4612	-1.3727	0.50
5	1.4	1.3	1.4190	1.4335	1.3113	2.3970	-0.9516	0.56
6	-1	-1	-1.0852	-1.0575	0.9849	-2.4395	-1.2808	0.54
7	1.4	0.9	1.3893	1.1029	1.2791	2.3713	-1.2404	0.52
8	0.9	1.3	0.8540	1.5739	0.7502	2.7847	-1.0022	0.42
9	1.1	1.2	1.0784	1.2636	0.9865	2.2736	-0.9118	0.56
10	-1.5	1	-1.6498	0.9184	1.5769	-1.7626	-2.5909	0.56
11	1.1	1.5	1.2598	1.6716	1.1445	2.6121	-0.8981	0.50
12	-1.3	-1.5	-1.2503	-1.6715	1.1623	-2.4199	-0.7227	0.64
13	0.9	0.3	0.9441	0.2987	0.8456	2.5920	-2.1494	0.48
14	-0.9	-0.4	-0.9666	-0.4321	0.8712	-2.7643	-2.2090	0.51
15	1	0.6	1.0489	0.6223	0.9445	2.3414	-1.5950	0.47
16	1.1	0.7	1.1525	0.7656	1.0434	2.6121	-1.7716	0.45
17	-1.2	0.4	-1.2024	0.4165	1.1086	-2.0812	-2.3906	0.58
18	1.3	0.1	1.3515	0.1560	1.2465	2.3586	-2.1495	0.51
19	-1.1	-1.5	-1.0930	-1.6971	1.0518	-5.1380	-3.5422	0.51
20	1	-0.5	0.8543	-0.6340	0.8088	4.7023	-5.2752	0.49

* The results above are reported after conducting the mean/mean linking.

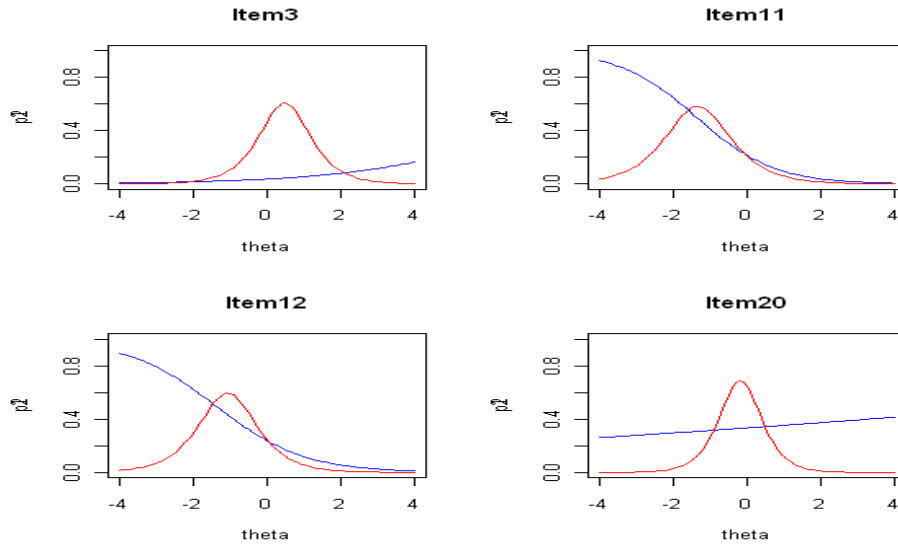


FIGURE 5.2. IRFs for a Few Items under Setting 2

TABLE 5.4. Parameter Estimation for Setting 2

Item	Est a	Est b	α	δ	τ	Est α^*	Est δ^*	Est τ^*	Est π
1	0.1630	3.4420	1.5	-0.2	-0.2	1.5881	-0.1483	-0.2623	1
2	0.0502	7.7188	1	0.3	-0.2	0.9580	0.3427	-0.1207	1
3	0.4565	1.9092	2	-0.5	-0.2	1.9333	-0.4738	-0.2232	1
4	0.7389	1.7591	2.4	-0.7	-0.2	2.4320	-0.7265	-0.2219	1
5	0.1006	9.0523	2.5	0.2	-0.2	1.9619	0.2391	-0.1385	1
6	0.8632	1.6293	2.1	-0.9	-0.2	2.4182	-0.8330	-0.2440	1
7	-1.3804	-1.8334	2.2	1.4	-0.2	2.4463	1.1639	-0.0622	1
8	0.8692	1.8038	1.9	-1	-0.2	1.6303	-1.0991	-0.0944	1
9	0.0744	7.7914	1.8	0.1	-0.2	1.7854	0.0967	-0.2387	1
10	-1.3937	-1.5144	1.6	1.5	-0.2	1.5250	1.6555	-0.3211	0.97
11	-1.0549	-1.3664	1.5	1.1	-0.2	1.4606	1.1985	-0.2254	1
12	-0.9230	-1.3911	1.7	0.9	-0.2	1.6908	0.9524	-0.2292	1
13	-0.7757	-1.6064	1.1	1.4	-0.2	0.9606	1.3636	-0.0536	0.99
14	1.5776	1.7617	2.2	-1.5	-0.2	2.1344	-1.6260	-0.2422	1
15	0.3175	3.1373	2.4	-0.3	-0.2	2.2572	-0.2729	-0.1710	1
16	0.5260	1.5470	1.8	-0.5	-0.2	1.9093	-0.5211	-0.2718	1
17	0.5931	1.4630	1.6	-0.6	-0.2	1.6549	-0.6661	-0.2444	1
18	-0.8276	-0.9942	1.4	0.7	-0.2	1.7419	0.7527	-0.3883	1
19	0.4872	1.8870	2	-0.5	-0.2	1.7442	-0.5338	-0.1746	1
20	0.0963	7.9392	2.14	0.1	-0.2	2.6078	0.1355	-0.3059	1

* The results above are reported after conducting the mean/mean linking.

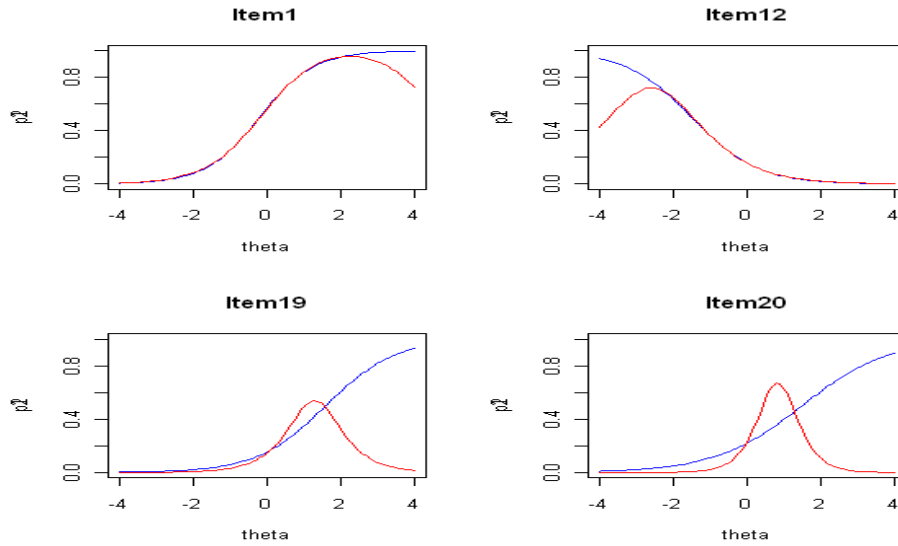


FIGURE 5.3. IRFs for a Few Items under Setting 3

TABLE 5.5. Parameter Estimation for Setting 3

Item	a	b	Est a^*	Est b^*	α	δ	τ	Est α^*	Est δ^*	Est τ^*	Est π
1	1.2	-0.2	1.3694	-0.2247				1.2975	2.2615	-2.4103	0.4
2	-1	-0.9	-0.9300	-0.9945				0.8576	-2.3150	-1.1911	0.52
3	1	0.2	1.1890	0.2249				1.1123	2.4384	-2.1360	0.5
4	1.5	-0.8	1.4478	-0.8426				1.3898	1.9377	-2.6680	0.54
5	1.4	1.3	1.6129	1.3287				1.5311	2.5674	-1.2757	0.52
6	-1	-1.1	-0.8455	-1.3646				0.7757	-2.5187	-1.0012	0.54
7	1.2	0.9	1.0339	1.1563				0.9737	2.2134	0.9700	0.61
8	0.9	1	0.8597	1.0545				0.7787	2.6976	-1.4725	0.44
9	1.2	1.2	1.2255	1.3142				1.1436	2.6375	-1.2988	0.58
10	-1.5	0.9	-1.4883	0.9040				1.4357	-1.7210	-2.5187	0.54
11	1.1	1.5	1.2043	1.5874				1.1206	2.6161	-1.0043	0.52
12	-1.2	-1.5	-1.1205	-1.6552				1.0385	-2.5967	-0.9115	0.57
13	0.9	0.3	0.7729	0.4153				0.7058	2.5452	-1.9030	0.51
14	-0.9	-0.4	-0.908	-0.5701				0.8318	-2.5235	-1.8136	0.49
15	1	-0.6	1.0495	-0.7076				0.9780	2.1609	-2.7112	0.5
16	1.1	0.7	0.9506	0.7993				0.9020	2.0550	-1.1273	0.64
17	-1.2	-0.5	-1.2237	-0.5155				1.1511	-2.2162	-1.6470	0.57
18	1.3	0.1	1.3684	0.1905				1.2912	2.2629	-2.0143	0.39
19			1.1839	1.5979	1.5	1.5	-0.2	1.4471	1.4138	-0.1105	1
20			0.9462	1.4912	2	0.9	-0.2	2.2593	0.9106	-0.3203	1

* The results above are reported after conducting the mean/mean linking.

Table 5.5 lists all the estimates for the 3rd setting which contains 18 monotone items and 2 unfolding items with $\delta_{19} = 1.5$ as an extreme location item and $\delta_{20} = 0.9$ as a middle location item. The estimated π for monotone items are between 0.39 to 0.61 and 1 for the two unfolding items. The estimates for all the parameters are close to the true parameters. Figure 5.3 contains the IRFs for 2 monotone items and 2 unfolding items. Item 1 and item 12 are monotone items with monotone IRF and unfolding IRF overlap in the middle and separate at the ends.

Table 5.6 and 5.7 contain the results for Setting 4 and Setting 5 where setting 4 has 2 monotone items and 18 unfolding items and setting 5 has 20 unfolding items. Figure 5.4 is IRFs for 4 items in setting 4. Item 1 and item 2 are monotone items which have similar behavior as in the previous studies. However, item 13 is an unfolding item. Its IRFs are off since the estimates are off from the true unfolding parameters, which causes the π to be around 0.53. After 1700 iterations and 2 weeks, the estimation for this item does not converge, with differences between the current iteration and the previous iteration being fairly large (> 0.1).

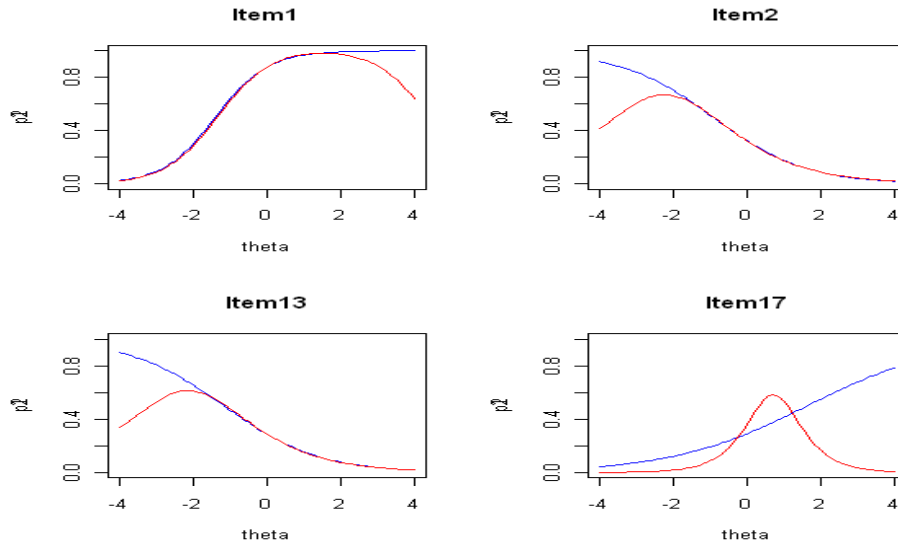


FIGURE 5.4. IRFs for a Few Items under Setting 4

TABLE 5.6. Parameter Estimation for Setting 4

Item	a	b	Est a^*	Est b^*	α	δ	τ	Est α^*	Est δ^*	Est τ^*	Est π
1	-1.4	1.5	-1.5791	1.4155				1.3946	1.5306	-4.6210	0.25
2	0.9	0.9	0.8983	0.9461				0.7259	-2.2339	-0.1071	0.36
3			0.2423	2.9956	1.5	-0.2	-0.2	1.4903	-0.3299	-0.1702	1
4			0.4377	1.6764	1	-0.7	-0.2	1.14	-0.7057	-0.1326	1
5			-0.3287	-2.6800	2	0.2	-0.2	2.2801	0.1422	-0.2564	1
6			1.0682	1.5750	2.1	-0.9	-0.2	1.9755	-1.1424	-0.1969	1
7			-1.4801	-1.4055	2.2	1.4	-0.2	2.6683	1.0845	-0.2866	1
8			1.2530	1.4286	1.9	-1	-0.2	2.1177	-1.2213	-0.2670	1
9			0.1147	5.7690	1.8	0.1	-0.2	1.6271	0.1346	-0.2271	1
10			-1.3367	-1.3558	1.6	1.5	-0.2	1.7269	1.2873	-0.2748	1
11			-1.0115	-1.2201	1.5	1.1	-0.2	1.7789	0.8914	-0.2900	1
12			-0.8662	-1.3969	1.7	0.9	-0.2	1.7193	0.794	-0.2099	1
13			-0.8593	-1.1699	1.1	1.4	-0.2	0.7400	2.0551	-0.0414	0.53
14			1.8775	1.6943	2.2	-1.5	-0.2	2.2647	-1.8049	-0.2898	1
15			0.3480	3.1078	2.4	-0.3	-0.2	2.2067	-0.4175	-0.1493	1
16			0.5628	1.7279	1.8	-0.5	-0.2	1.6254	-0.7073	-0.1831	1
17			0.6064	1.6328	1.6	-0.6	-0.2	1.6395	-0.7490	-0.1996	1
18			-0.6249	-1.2062	1.4	0.7	-0.2	1.4902	0.5378	-0.2682	1
19			0.6103	1.8924	2	-0.5	-0.2	1.6978	-0.7439	-0.1178	1
20			0.1210	7.2195	2.14	0.1	-0.2	1.7516	0.0359	-0.1540	1

* The results above are reported after conducting the mean/mean linking.

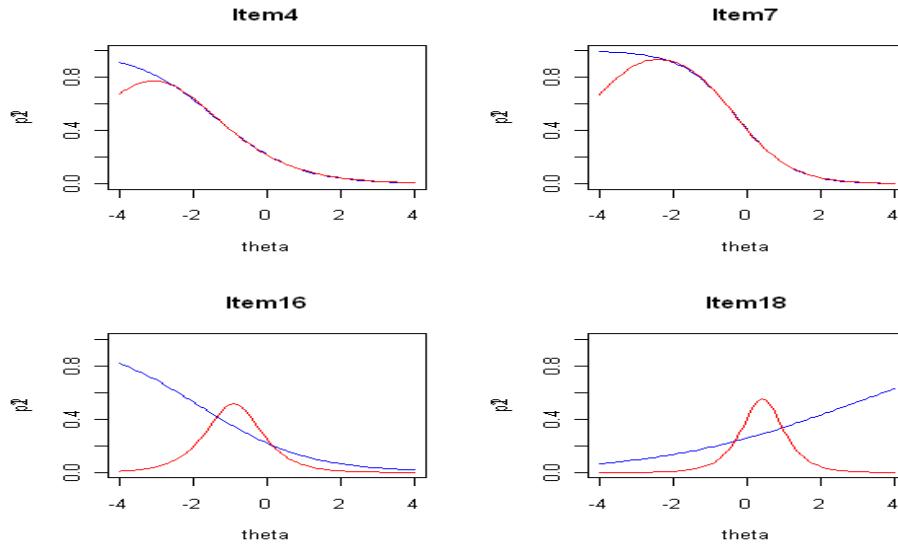


FIGURE 5.5. IRFs for a Few Items under Setting 5

TABLE 5.7. Parameter Estimation for Setting 5

Item	a	b	Est a^*	Est b^*	α	δ	τ	Est α^*	Est δ^*	Est τ^*	Est π
1	1.5	-0.6	1.5394	-0.6600				1.3345	2.2896	-3.1338	0.43
2	-1.4	0.2	-1.4996	0.1595				1.2887	-2.3638	-1.8623	0.4
3	-1.5	-0.3	-1.6008	-0.2903				1.3825	-2.5517	-0.6082	0.47
4	-1	-1.4	-1.0124	-1.4271				0.8366	-3.0886	-0.3211	0.45
5	-1.1	1.4	-1.0402	1.6148				0.8881	-1.8313	-4.2822	0.63
6	1.1	-1.1	1.2849	-1.1106				1.1047	2.3286	-4.1725	0.47
7	-1.5	-0.2	-1.5036	-0.2997				1.2863	-2.3875	-0.7032	0.49
8	1.4	0.4	1.4650	0.4473				1.2816	2.0713	-0.4589	0.72
9	1.3	0.2	1.5678	0.1392				1.3511	2.2175	-1.1221	0.56
10	-1.5	0.8	-1.6976	0.8074				1.4861	-2.0367	-3.7992	0.51
11			-0.7178	-1.3971	2.2206	-0.6	-0.2	2.0712	-0.6675	-0.8695	1
12			0.3051	2.7461	2.1469	0.2	-0.2	2.1828	0.2404	-0.9256	1
13			-0.3008	-3.0457	2.2601	-0.3	-0.2	2.1110	-0.2471	-0.6699	1
14			-1.0518	-1.7314	1.4073	-1.4	-0.2	1.2712	-1.5067	-0.3328	1
15			1.1577	1.6560	1.52	1.4	-0.2	1.3905	1.5025	-0.3502	1
16			-0.7650	-1.8324	1.6228	-1.1	-0.2	1.5338	-0.9019	-0.3154	1
17			-0.2402	-3.4407	2.2673	-0.2	-0.2	2.5836	-0.1595	-0.8492	1
18			0.4395	2.6942	2.1302	0.4	-0.2	2.1089	0.4238	-0.4891	1
19			0.1930	4.5234	2.0242	0.2	-0.2	1.7992	0.1672	-0.3720	1
20			0.8012	1.6570	2.1782	0.8	-0.2	1.7165	0.9625	-0.3855	1

* The results above are reported after conducting the mean/mean linking.

In this section, all the simulations studies under different settings show a common pattern through the estimated π and IRFs. The estimates of π s for monotone items are around 0.5 since the two IRFs overlap in most places except the extreme locations and split in the end of the scale. The reason for the estimates of $\pi = 0.5$ is because the data set does not have enough data (examinees) in the extreme location (θ in the end of scale) when θ follows the standard normal distribution. Without enough information, the model cannot capture the activities in that location. Therefore, two models have the similar likelihood which caused π be close to 0.5.

5.4. SIMULATION STUDY FOR ALL THREE METHODS

In this section, we conducted an analysis for all three methods discussed previously: the manifest monotonicity test, the p-value/biserial method, and the mixture model on 10 data sets with 1000 examinees. These 10 data sets were generated for 20 items (assuming θ follows a standard normal distributions). The first 10 items are monotone items and the last 10 items are unfolding items. Each method is utilized to identify the item type for each data set. The estimates are reported along with the standard errors in Tables 5.8 to 5.17 (see Roberts, 2000). In these ten tables, Ind_{MM} is the indicator from applying the manifest monotonicity test, Ind_{GTO} is the indicator using the p-value/biserial method with the separation criteria from the classification tree for the original data, and Ind_{GTT} is the indicator for the p-value/biserial method with the separation criteria through the classification tree for the transformed data. In each case, “M” stands for monotone item, “U” stands for unfolding item and “C” is for uncertain item.

TABLE 5.8. Item parameters and Estimates from Mixture model for Simulation Run 1

item	a	b	ea^*	eb^*	α	δ	τ	$e\alpha^*$	$e\delta^*$	$e\tau^*$	$e\pi$	$stda$	$stdb$	$stdb$	$std\delta$	$std\alpha$	$std\tau$	Ind_{MM}	Ind_{GTO}	Ind_{GTT}
1	-1.5	-1.5	-1.698	-1.208				2.860	-2.565	-1.347	0.504	0.081	0.054	0.055	0.084	0.033	M	M	C	
2	-1.5	-1	-1.293	-0.899				2.154	-2.350	-1.428	0.573	0.073	0.054	0.057	0.077	0.032	M	M	C	
3	-0.9	0	-0.978	0.132				1.628	-2.496	-2.613	0.497	0.067	0.047	0.047	0.072	0.018	M	M	C	
4	-0.9	1.5	-0.887	1.734				1.488	-2.386	-4.122	0.471	0.087	0.106	0.108	0.094	0.017	M	M	U	
5	0.9	-1.5	0.893	-1.412				1.498	2.640	-4.053	0.462	0.086	0.100	0.101	0.092	0.016	M	M	C	
6	0.9	-1	0.994	-0.804				1.666	2.513	-3.313	0.501	0.074	0.062	0.062	0.078	0.016	M	M	C	
7	-1.5	0	-1.640	0.163				2.776	-2.158	-2.319	0.492	0.061	0.033	0.033	0.063	0.014	M	M	C	
8	1.5	1	1.463	1.166				2.451	2.809	-1.630	0.524	0.072	0.051	0.052	0.075	0.027	M	U	M	
9	1.5	1.5	1.264	1.851				2.620	2.062	-0.351	0.524	0.092	0.091	0.078	0.108	0.145	M	U	U	
10	0.9	0	0.783	0.228				1.284	2.913	-2.649	0.528	0.073	0.056	0.058	0.080	0.022	M	M	C	
11			-0.321	-0.650	0.5	-1.5	-0.2	0.698	-1.161	-0.435	0.973	0.131	0.154	0.176	0.128	0.217	M	M	U	
12			-0.184	-0.243	0.5	-1	-0.4	0.523	-0.832	-0.481	0.944	0.212	0.222	0.226	0.168	0.251	U	M	C	
13			-0.054	-0.103	0.5	0	-0.6	0.543	0.364	-0.475	0.997	0.691	0.715	0.172	0.160	0.243	U	M	C	
14			-0.385	1.188	1	1	-0.2	0.992	1.179	-0.252	1	0.116	0.153	0.115	0.098	0.275	M	U	U	
15			0.524	1.308	1	1.5	-0.4	1.061	1.577	-0.315	0.991	0.097	0.124	0.128	0.099	0.223	M	U	U	
16			0.038	-0.006	1	0	-0.6	1.119	0.048	-0.572	1	0.952	0.970	0.065	0.093	0.103	U	M	C	
17			-0.808	-1.231	1.5	-1.5	-0.2	1.645	-1.509	-0.383	0.992	0.084	0.095	0.082	0.082	0.143	M	U	C	
18			0.531	1.161	1.5	1	-0.4	1.317	1.244	-0.337	1	0.094	0.113	0.088	0.089	0.168	M	U	U	
19			0.894	1.748	2	1.5	-0.2	1.734	1.925	-0.265	0.927	0.088	0.107	0.091	0.087	0.223	M	U	U	
20			0.594	1.202	2	1	-0.4	1.975	1.112	-0.390	1	0.089	0.105	0.050	0.081	0.104	U	U	U	

* The results above are reported after conducting the mean/mean linking. Ind_{MM} is the indicator for manifest monotonicity test. Ind_{GTO} is the indicator for p-value/biserial under classification tree for original data. Ind_{GTT} is the indicator for p-value/biserial method for transformed data. "M" for monotone item, "U" for unfolding item and "C" is uncertain item

TABLE 5.9. Item parameters and Estimates from Mixture model for Simulation Run 2

item	a	b	ea^*	eb^*	α	δ	τ	$e\alpha^*$	$e\delta^*$	$e\tau^*$	$e\pi$	$stda$	$stdb$	$stdb$	$std\delta$	$std\alpha$	$std\tau$	Ind_{MM}	Ind_{GTO}	Ind_{GTT}
1	-1.5	-1.5	-1.467	-1.543				2.463	-2.992	-1.434	0.504	0.081	0.054	0.055	0.084	0.033	M	M	C	
2	-1.5	-1	-1.679	-0.973				1.323	-1.224	-0.012	0.573	0.073	0.054	0.057	0.077	0.032	M	M	C	
3	-0.9	0	-0.955	-0.003				1.593	-2.800	-2.785	0.497	0.067	0.047	0.047	0.072	0.018	M	M	C	
4	-0.9	1.5	-0.971	1.428				1.632	-2.795	-4.228	0.471	0.087	0.106	0.108	0.094	0.017	M	M	U	
5	0.9	-1.5	0.911	-1.519				1.527	2.477	-3.999	0.462	0.086	0.100	0.101	0.092	0.016	M	M	C	
6	0.9	-1	1.015	-0.92				1.699	2.329	-3.246	0.501	0.074	0.062	0.062	0.078	0.016	M	M	C	
7	-1.5	0	-1.315	-0.036				2.223	-2.809	-2.772	0.492	0.061	0.033	0.033	0.063	0.014	M	M	C	
8	1.5	1	1.481	1.049				2.494	2.799	-1.741	0.524	0.072	0.051	0.052	0.075	0.027	M	U	M	
9	1.5	1.5	1.738	1.401				2.930	2.836	-1.427	0.528	0.092	0.091	0.078	0.108	0.145	M	U	U	
10			0.971	-0.017				1.603	2.330	-2.322	0.524	0.073	0.056	0.058	0.080	0.022	M	M	C	
11			-0.215	-1.084	0.5	-1.5	-0.2	0.379	-2.213	-0.515	0.973	0.131	0.154	0.176	0.128	0.217	M	M	U	
12			-0.182	-0.839	0.5	-1	-0.4	0.601	-0.843	-0.249	0.944	0.212	0.222	0.226	0.168	0.251	U	M	C	
13			-0.007	-0.604	0.5	0	-0.6	0.504	-0.25	-0.448	0.997	0.691	0.715	0.172	0.160	0.243	U	M	C	
14			-0.315	1.325	1	1	-0.2	0.983	0.871	-0.104	1	0.116	0.153	0.115	0.098	0.275	M	U	U	
15			0.459	1.059	1	1.5	-0.4	1.076	1.195	-0.315	0.991	0.097	0.124	0.128	0.099	0.223	M	U	U	
16			0.054	0.423	1	0	-0.6	0.983	-0.135	-0.616	1	0.952	0.970	0.065	0.093	0.103	U	M	C	
17			-0.697	-1.470	1.5	-1.5	-0.2	1.301	-1.737	-0.273	0.992	0.084	0.095	0.082	0.082	0.143	M	U	C	
18			0.467	1.056	1.5	1	-0.4	1.502	0.919	-0.344	1	0.094	0.113	0.088	0.089	0.168	M	U	U	
19			0.844	1.528	2	1.5	-0.2	1.772	1.633	-0.262	0.927	0.088	0.107	0.091	0.087	0.223	M	U	U	
20			0.542	1.058	2	1	-0.4	1.898	0.897	-0.375	1	0.089	0.105	0.050	0.081	0.104	U	U	U	

* The results above are reported after conducting the mean/mean linking. Ind_{MM} is the indicator for manifest monotonicity test. Ind_{GTO} is the indicator for p-value/biserial under classification tree for original data. Ind_{GTT} is the indicator for p-value/biserial method for transformed data. "M" for monotone item, "U" for unfolding item and "C" is uncertain item

TABLE 5.10. Item parameters and Estimates from Mixture model for Simulation Run 3

item	a	b	ea^*	eb^*	α	δ	τ	$e\alpha^*$	$e\delta^*$	$e\tau^*$	$e\pi$	$stda$	$stdb$	$std\delta$	$std\alpha$	$std\tau$	Ind_{MM}	Ind_{GTO}	Ind_{GTT}
1	-1.5	-1.5	-1.687	1.471	0.5	-1.5	-0.2	0.133	-3.342	0.559	0.569	0.088	0.060	3.841	0.215	0.931	M	M	C
2	-1.5	-1	-1.563	0.981	0.5	-1	-0.4	1.325	1.305	0.021	0.524	0.071	0.046	0.228	0.119	1.175	M	M	C
3	-0.9	0	-0.861	-0.024	0.5	0	-0.6	1.447	1.785	1.756	0.401	0.071	0.052	0.052	0.075	0.030	M	M	C
4	-0.9	1.5	-0.831	-1.566	1	1	-0.2	1.377	2.286	3.855	0.502	0.089	0.110	0.111	0.098	0.018	M	M	U
5	0.9	-1.5	0.881	1.526	1	1.5	-0.4	1.561	-1.169	2.633	0.572	0.085	0.100	0.076	0.079	0.024	M	M	C
6	0.9	-1	0.975	0.972	1	0	-0.6	1.621	-2.139	3.106	0.571	0.074	0.065	0.064	0.079	0.017	M	M	C
7	-1.5	0	-1.521	0.022	1.5	-1.5	-0.2	2.568	2.116	2.090	0.506	0.062	0.035	0.035	0.064	0.016	M	M	C
8	1.5	1	1.641	-0.891	1.5	1	-0.4	0.065	17.714	0.042	0.424	0.072	0.044	3.829	0.228	1.263	M	U	M
9	1.5	1.5	1.544	-1.341	2	1.5	-0.2	0.123	17.661	1.014	0.579	0.081	0.060	3.843	0.216	0.656	M	U	U
10	0.9	0	0.788	-0.041	2	1	-0.4	1.271	-2.041	1.931	0.471	0.073	0.056	0.058	0.079	0.030	M	M	C
11	-1.5	-1.5	-0.250	1.149	0.5	-1.5	-0.2	0.389	1.453	1.113	0.961	0.168	0.248	0.537	0.223	1.251	U	M	U
12	-1.5	-1	-0.176	0.464	0.5	-1	-0.4	0.354	0.966	0.821	0.945	0.231	0.253	0.515	0.273	0.764	U	M	C
13	-0.9	0	-0.030	2.367	0.5	0	-0.6	0.706	0.050	0.669	1.000	0.989	2.482	0.117	0.128	0.136	U	M	U
14	-0.9	1.5	-0.432	-0.933	1	1	-0.2	0.935	-1.266	0.375	0.998	0.108	0.130	0.131	0.106	0.195	M	U	U
15	0.9	-1.5	0.607	-0.929	1	1.5	-0.4	0.959	-1.471	1.218	0.985	0.089	0.099	0.157	0.109	0.208	M	U	U
16	0.9	-1	0.038	-0.280	1	0	-0.6	1.031	0.128	0.586	1.000	0.948	0.998	0.073	0.097	0.109	U	M	U
17	-1.5	0	-0.621	1.594	1.5	-1.5	-0.2	1.573	1.314	0.111	1.000	0.095	0.137	0.099	0.088	0.483	M	M	C
18	1.5	1	0.464	-0.968	1.5	1	-0.4	1.406	-0.910	0.337	1.000	0.103	0.124	0.066	0.080	0.151	U	U	U
19	1.5	1.5	0.972	-1.442	2	1.5	-0.2	1.988	-1.707	0.384	0.996	0.084	0.091	0.070	0.081	0.132	M	U	U
20	0.9	0	0.620	-0.953	2	1	-0.4	1.868	-1.022	0.429	1.000	0.088	0.096	0.052	0.082	0.097	M	U	U

* The results above are reported after conducting the mean/mean linking. Ind_{MM} is the indicator for manifest monotonicity test. Ind_{GTO} is the indicator for p-value/biserial under classification tree for original data. Ind_{GTT} is the indicator for p-value/biserial method for transformed data. "M" for monotone item, "U" for unfolding item and "C" is uncertain item

TABLE 5.1.1. Item parameters and Estimates from Mixture model for Simulation Run 4

item	a	b	ea^*	eb^*	α	δ	τ	$e\alpha^*$	$e\delta^*$	$e\tau^*$	$e\pi$	$stda$	$stdb$	$std\delta$	$std\alpha$	$std\tau$	Ind_{MM}	Ind_{GTO}	Ind_{GTT}
1	-1.5	-1.5	-1.354	-1.561				2.256	-2.863	-1.277	0.499	0.089	0.077	0.081	0.092	0.043	M	M	C
2	-1.5	-1	-1.364	-1.064				2.277	-2.594	-1.511	0.457	0.073	0.053	0.055	0.076	0.030	M	M	C
3	-0.9	0	-0.907	-0.051				1.494	-2.519	-2.440	0.407	0.068	0.051	0.052	0.074	0.021	M	M	C
4	-0.9	1.5	-0.868	1.486				1.432	-2.098	-3.587	0.513	0.085	0.100	0.100	0.094	0.019	M	M	U
5	0.9	-1.5	1.024	-1.463				1.711	2	-3.465	0.530	0.084	0.086	0.087	0.090	0.017	M	M	C
6	0.9	-1	0.876	-1.062				1.447	2.226	-3.283	0.476	0.077	0.075	0.074	0.085	0.018	M	M	C
7	-1.5	0	-1.583	-0.083				2.673	-2.081	-1.993	0.483	0.061	0.035	0.035	0.064	0.017	M	M	C
8	1.5	1	1.422	1.020				2.382	2.640	-1.606	0.482	0.072	0.051	0.052	0.075	0.027	M	U	M
9	1.5	1.5	1.413	1.406				2.353	2.711	-1.281	0.498	0.083	0.066	0.069	0.087	0.040	M	U	U
10	0.9	0	0.926	0.026				1.530	2.558	-2.508	0.460	0.068	0.050	0.051	0.073	0.020	M	M	C
11			-0.278	-1.120	0.5	-1.5	-0.2	0.591	-1.377	-0.159	0.915	0.150	0.215	0.272	0.154	0.622	M	M	U
12			-0.213	-0.335	0.5	-1	-0.4	0.564	-0.996	-0.538	0.967	0.185	0.194	0.211	0.159	0.210	U	M	C
13			-0.028	-0.457	0.5	0	-0.6	0.593	-0.078	-0.401	0.999	1.264	1.410	0.150	0.143	0.263	U	M	U
14			-0.413	0.786	1	1	-0.2	1.076	0.949	-0.383	1	0.108	0.125	0.096	0.094	0.167	U	U	M
15			0.504	1.082	1	1.5	-0.4	1.073	1.300	-0.303	0.999	0.097	0.122	0.124	0.099	0.223	M	U	U
16			0.037	-1.609	1	0	-0.6	0.785	0.070	-0.657	1	0.916	1.749	0.101	0.118	0.125	U	M	U
17			-0.818	-1.422	1.5	-1.5	-0.2	1.468	-1.813	-0.367	0.892	0.085	0.100	0.102	0.088	0.169	M	M	C
18			0.504	0.923	1.5	1	-0.4	1.497	0.964	-0.401	1	0.096	0.112	0.066	0.083	0.123	M	U	U
19			0.897	1.504	2	1.5	-0.2	1.835	1.643	-0.261	0.998	0.085	0.099	0.077	0.080	0.205	M	U	U
20			0.662	0.845	2	1	-0.4	2.279	0.941	-0.495	1	0.082	0.084	0.040	0.080	0.072	U	U	U

* The results above are reported after conducting the mean/mean linking. Ind_{MM} is the indicator for manifest monotonicity test. Ind_{GTO} is the indicator for p-value/biserial under classification tree for original data. Ind_{GTT} is the indicator for p-value/biserial method for transformed data. "M" for monotone item, "U" for unfolding item and "C" is uncertain item

TABLE 5.12. Item parameters and Estimates from Mixture model for Simulation Run 5

item	a	b	ea^*	eb^*	α	δ	τ	$e\alpha^*$	$e\delta^*$	$e\tau^*$	$e\pi$	$stda$	$stdb$	$std\delta$	$std\alpha$	$std\tau$	Ind_{MM}	Ind_{GTO}	Ind_{GTT}
1	-1.5	-1.5	-1.320	-1.571				2.198	-2.881	-1.284	0.544	0.085	0.072	0.076	0.089	0.043	M	M	C
2	-1.5	-1	-1.418	-1.150				2.376	-2.766	-1.602	0.489	0.073	0.051	0.053	0.076	0.028	M	M	C
3	-0.9	0	-0.779	-0.081				1.267	-2.448	-2.315	0.637	0.072	0.056	0.057	0.079	0.025	M	M	C
4	-0.9	1.5	-0.851	1.365				1.407	-2.300	-3.666	0.531	0.083	0.094	0.094	0.091	0.018	M	M	U
5	0.9	-1.5	0.826	-1.697				1.367	2.202	-3.903	0.508	0.089	0.110	0.111	0.098	0.018	M	M	C
6	0.9	-1	0.769	-1.016				1.308	1.359	-2.314	0.847	0.078	0.076	0.062	0.077	0.027	M	M	C
7	-1.5	0	-1.571	-0.050				2.660	-2.318	-2.265	0.474	0.061	0.034	0.034	0.063	0.015	M	M	C
8	1.5	1	1.549	0.911				2.607	2.559	-1.638	0.513	0.071	0.046	0.047	0.073	0.025	M	U	M
9	1.5	1.5	1.304	1.429				2.191	2.450	-0.992	0.640	0.083	0.071	0.075	0.088	0.055	M	U	U
10	0.9	0	0.904	0.012				1.483	2.379	-2.336	0.514	0.068	0.050	0.051	0.074	0.022	M	M	C
11			-0.213	-1.429	0.5	-1.5	-0.2	0.492	-1.534	-0.073	0.873	0.186	0.302	0.348	0.180	1.041	U	M	U
12			-0.265	-0.588	0.5	-1	-0.4	0.843	-0.872	-0.482	1	0.150	0.163	0.112	0.107	0.162	U	M	C
13			-0.024	-1.168	0.5	0	-0.6	0.637	-0.050	-0.553	1	1.34	1.941	0.131	0.136	0.180	U	M	U
14			-0.357	0.991	1	1	-0.2	0.994	0.930	-0.271	1	0.121	0.159	0.107	0.095	0.254	U	U	U
15			0.493	0.704	1	1.5	-0.4	0.947	1.441	-0.664	0.980	0.095	0.102	0.119	0.101	0.112	M	U	M
16			0.025	-0.320	1	0	-0.6	1.046	-0.045	-0.598	1	1.402	1.462	0.069	0.095	0.105	U	M	C
17			-0.711	-1.728	1.5	-1.5	-0.2	1.391	-1.796	-0.152	0.965	0.093	0.126	0.111	0.089	0.414	M	M	C
18			0.533	0.854	1.5	1	-0.4	1.532	0.945	-0.436	1	0.092	0.103	0.062	0.081	0.112	M	U	U
19			0.871	1.466	2	1.5	-0.2	2.148	1.377	-0.216	1	0.085	0.099	0.077	0.081	0.219	M	U	U
20			0.553	0.992	2	1	-0.4	2.180	0.822	-0.419	1	0.091	0.107	0.038	0.072	0.088	M	U	U

* The results above are reported after conducting the mean/mean linking. Ind_{MM} is the indicator for manifest monotonicity test. Ind_{GTO} is the indicator for p-value/biserial under classification tree for original data. Ind_{GTT} is the indicator for p-value/biserial method for transformed data. "M" for monotone item, "U" for unfolding item and "C" is uncertain item

TABLE 5.13. Item parameters and Estimates from Mixture model for Simulation Run 6

item	a	b	ea^*	eb^*	α	δ	τ	$e\alpha^*$	$e\delta^*$	$e\tau^*$	$e\pi$	$stda$	$stdb$	$std\delta$	$std\alpha$	$std\tau$	Ind MM	Ind GTO	Ind GTT
1	-1.5	-1.5	-1.769	-1.483			1.492	-1.457	-0.007	0.468	0.090	0.056	0.229	0.121	1.265	M	M	C	
2	-1.5	-1	-1.772	-0.896			1.194	-1.283	-0.018	0.515	0.069	0.040	0.236	0.120	1.194	M	M	C	
3	-0.9	0	-0.928	-0.011			1.533	-2.613	-2.582	0.532	0.067	0.049	0.050	0.073	0.019	M	M	C	
4	-0.9	1.5	-0.848	1.556			1.399	-2.225	-3.789	0.577	0.089	0.108	0.110	0.099	0.019	M	M	U	
5	0.9	-1.5	0.923	-1.343			1.532	2.049	-3.390	0.487	0.079	0.082	0.081	0.087	0.018	M	M	C	
6	0.9	-1	0.776	-1.025			1.272	2.347	-3.358	0.506	0.078	0.078	0.076	0.087	0.019	M	M	C	
7	-1.5	0	-1.466	0.022			2.613	-1.491	-1.507	0.391	0.062	0.036	0.034	0.064	0.023	M	M	C	
8	1.5	1	1.264	0.927			2.107	2.598	-1.650	0.492	0.072	0.052	0.055	0.075	0.027	M	U	M	
9	1.5	1.5	1.554	1.544			2.602	2.859	-1.299	0.489	0.093	0.070	0.073	0.097	0.040	M	U	U	
10	0.9	0	0.878	0.001			1.449	2.567	-2.538	0.371	0.069	0.052	0.052	0.074	0.020	M	M	C	
11			-0.277	-0.994	0.5	-1.5	0.452	-1.467	-0.412	0.914	0.148	0.195	0.300	0.153	0.159	M	M	U	
12			-0.111	-0.715	0.5	-1	0.345	-1.091	-0.242	0.980	0.342	0.415	0.507	0.293	0.675	U	M	U	
13			-0.05	-1.136	0.5	0	0.458	-0.247	-0.184	0.966	0.732	1.096	0.220	0.174	0.654	U	M	U	
14			-0.445	0.957	1	1	1.210	0.938	-0.292	1	0.104	0.127	0.085	0.088	0.201	U	U	U	
15			0.581	0.876	1	1.5	-0.4	1.132	1.439	-0.542	0.998	0.089	0.096	0.106	0.094	0.120	M	U	U
16			0.021	-1.332	1	0	-0.6	1.032	-0.646	1	1.581	2.948	0.071	0.098	0.097	U	M	U	
17			-0.706	-1.517	1.5	-1.5	-0.2	1.399	-1.508	-0.082	0.999	0.088	0.114	0.113	0.089	0.728	M	M	C
18			0.582	0.826	1.5	1	-0.4	1.528	0.982	-0.414	1	0.088	0.093	0.065	0.085	0.118	M	U	U
19			0.893	1.557	2	1.5	-0.2	2.016	1.489	-0.151	1	0.089	0.104	0.077	0.080	0.329	M	U	U
20			0.609	1.015	2	1	-0.4	1.751	1.012	-0.354	1	0.088	0.099	0.061	0.085	0.127	M	U	U

* The results above are reported after conducting the mean/mean linking. Ind MM is the indicator for manifest monotonicity test. Ind GTO is the indicator for p-value/biserial under classification tree for original data. Ind GTT is the indicator for p-value/biserial method for transformed data. "M" for monotone item, "U" for unfolding item and "C" is uncertain item

TABLE 5.14. Item parameters and Estimates from Mixture model for Simulation Run 7

item	a	b	ea^*	eb^*	α	δ	τ	$e\alpha^*$	$e\delta^*$	$e\tau^*$	$e\pi$	$stda$	$stdb$	$std\delta$	$std\sigma$	$std\tau$	Ind _{MM}	Ind _{GTO}	Ind _{GTT}
1	-1.5	-1.5	-1.561	-1.456	0.141	2.714	-0.852	0.388	0.388	0.388	0.388	0.088	0.067	3.827	0.220	0.710	M	M	C
2	-1.5	-1	-1.378	-1.090	2.298	-2.638	-1.530	0.530	0.530	0.530	0.530	0.074	0.056	0.058	0.078	0.029	M	M	C
3	-0.9	0	-0.840	0.030	1.392	-2.851	-2.860	0.485	0.485	0.485	0.485	0.072	0.054	0.054	0.077	0.018	U	M	C
4	-0.9	1.5	-0.950	1.562	1.594	-2.529	-4.096	0.478	0.478	0.478	0.478	0.087	0.103	0.105	0.093	0.016	M	M	U
5	0.9	-1.5	0.848	-1.496	1.399	2.131	-3.630	0.567	0.567	0.567	0.567	0.088	0.107	0.107	0.098	0.019	M	M	C
6	0.9	-1	0.832	-1.175	1.383	2.480	-3.650	0.468	0.468	0.468	0.468	0.082	0.087	0.086	0.090	0.017	M	M	C
7	-1.5	0	-1.504	-0.043	2.539	-2.254	-2.208	0.500	0.500	0.500	0.500	0.062	0.036	0.036	0.064	0.016	M	M	C
8	1.5	1	1.896	0.887	3.201	2.391	-1.498	0.490	0.490	0.490	0.490	0.071	0.041	0.041	0.073	0.024	M	U	M
9	1.5	1.5	1.522	1.449	2.547	2.806	-1.341	0.487	0.487	0.487	0.487	0.086	0.068	0.070	0.089	0.038	M	U	U
10	0.9	0	0.798	-0.062	1.289	2.280	-2.292	0.534	0.534	0.534	0.534	0.073	0.057	0.057	0.080	0.025	M	M	C
11			-0.306	-1.069	0.5	-1.5	-0.2	0.504	-1.365	-0.607	0.986	0.143	0.197	0.304	0.154	0.216	U	M	U
12			-0.133	-0.378	0.5	-1	-0.4	0.613	-0.907	-0.371	0.998	0.297	0.316	0.169	0.146	0.275	U	M	C
13			-0.041	-1.986	0.5	0	-0.6	0.500	0.096	0.666	0.990	0.848	1.914	0.197	0.187	0.189	U	M	U
14			-0.355	1.019	1	1	-0.2	0.985	0.924	-0.219	1	0.126	0.167	0.113	0.097	0.311	U	U	U
15			0.466	1.076	1	1.5	-0.4	0.929	1.419	-0.324	0.984	0.105	0.135	0.151	0.109	0.235	M	U	U
16			0.024	0.248	1	0	-0.6	1.005	0.053	-0.523	1	1.510	1.559	0.077	0.099	0.124	U	M	C
17			-0.716	-1.401	1.5	-1.5	-0.2	1.075	-1.73	-0.009	0.83	0.09	0.114	0.173	0.106	1.265	M	M	C
18			0.551	0.877	1.5	1	-0.4	1.509	1.005	-0.415	1	0.093	0.104	0.069	0.085	0.119	M	U	U
19			1.088	1.334	2	1.5	-0.2	2.162	1.724	-0.472	0.992	0.080	0.079	0.065	0.082	0.102	M	U	M
20			0.615	1.066	2	1	-0.4	2.088	0.950	-0.367	1	0.090	0.105	0.048	0.080	0.105	M	U	U

* The results above are reported after conducting the mean/mean linking. Ind_{MM} is the indicator for manifest monotonicity test. Ind_{GTO} is the indicator for p-value/biserial under classification tree for original data. Ind_{GTT} is the indicator for p-value/biserial method for transformed data. "M" for monotone item, "U" for unfolding item and "C" is uncertain item

TABLE 5.15. Item parameters and Estimates from Mixture model for Simulation Run 8

item	a	b	ea^*	eb^*	α	δ	τ	$e\alpha^*$	$e\delta^*$	$e\tau^*$	$e\pi$	$stda$	$stdb$	$std\delta$	$std\alpha$	$std\tau$	Ind_{MM}	Ind_{GTO}	Ind_{GTT}
1	-1.5	-1.5	-1.531	-1.441	0.112	2.679	-0.007	0.498	0.088	0.069	0.823	0.219	1.265	M	M	C			
2	-1.5	-1	-1.522	-0.956	2.552	-2.534	-1.566	0.508	0.072	0.049	0.051	0.074	0.026	M	M	C			
3	-0.9	0	-0.934	0.085	1.551	-2.651	-2.721	0.500	0.069	0.051	0.051	0.074	0.018	M	M	C			
4	-0.9	1.5	-0.866	1.690	1.432	-2.174	-3.874	0.520	0.092	0.121	0.124	0.101	0.019	M	M	U			
5	0.9	-1.5	1.037	-1.479	1.731	1.690	-3.168	0.503	0.086	0.094	0.092	0.092	0.019	M	M	C			
6	0.9	-1	0.908	-0.953	1.491	1.868	-2.808	0.459	0.076	0.073	0.069	0.083	0.020	M	M	C			
7	-1.5	0	-1.401	0.012	2.361	-2.296	-2.303	0.492	0.063	0.038	0.038	0.065	0.016	M	M	C			
8	1.5	1	1.532	0.948	2.573	2.565	-1.606	0.473	0.071	0.048	0.049	0.074	0.025	M	U	M			
9	1.5	1.5	1.919	1.355	3.236	2.702	-1.341	0.494	0.087	0.053	0.054	0.090	0.032	M	U	U			
10	0.9	0	0.994	0.076	1.652	1.947	-1.836	0.771	0.067	0.049	0.049	0.073	0.025	M	M	C			
11			-0.279	-0.983	0.5	-1.5	-0.2	0.474	-1.774	-0.491	0.905	0.153	0.212	0.329	0.163	0.233	M	U	U
12			-0.144	-0.292	0.5	-1	-0.4	0.601	-0.594	-0.421	0.995	0.276	0.292	0.183	0.153	0.249	U	M	C
13			-0.034	-1.120	0.5	0	-0.6	0.494	0.305	-0.514	0.935	1.045	1.621	0.484	0.380	0.398	U	M	U
14			-0.405	1.070	1	1	-0.2	1.086	0.973	-0.212	1	0.115	0.153	0.105	0.092	0.298	M	U	U
15			0.480	1.205	1	1.5	-0.4	0.919	1.497	-0.240	0.932	0.104	0.142	0.165	0.111	0.321	M	U	U
16			0.066	-0.503	1	0	-0.6	0.886	0.180	-0.589	1	0.587	0.678	0.092	0.109	0.124	U	M	C
17			-0.767	-1.279	1.5	-1.5	-0.2	1.318	-1.634	-0.010	0.988	0.086	0.103	0.164	0.103	1.265	M	M	C
18			0.558	0.940	1.5	1	-0.4	1.772	0.953	-0.406	1	0.093	0.106	0.055	0.080	0.106	U	U	U
19			0.848	1.526	2	1.5	-0.2	1.865	1.540	-0.213	1	0.088	0.109	0.080	0.079	0.243	M	U	U
20			0.650	0.968	2	1	-0.4	1.841	1.075	-0.420	1	0.086	0.095	0.062	0.085	0.102	M	U	U

* The results above are reported after conducting the mean/mean linking. Ind_{MM} is the indicator for manifest monotonicity test. Ind_{GTO} is the indicator for p-value/biserial under classification tree for original data. Ind_{GTT} is the indicator for p-value/biserial method for transformed data. "M" for monotone item, "U" for unfolding item and "C" is uncertain item

TABLE 5.16. Item parameters and Estimates from Mixture model for Simulation Run 9

item	a	b	ea^*	eb^*	α	δ	τ	$e\alpha^*$	$e\delta^*$	$e\tau^*$	$e\pi$	$stda$	$stdb$	$std\delta$	$std\alpha$	$std\tau$	Ind_{MM}	Ind_{GTO}	Ind_{GTT}
1	-1.5	-1.5	-1.480	-1.475				2.366	2.656	-0.171	0.391	0.088	0.071	0.272	0.947	0.947	M	M	C
2	-1.5	-1	-1.591	-0.836				2.093	2.654	-0.171	0.487	0.071	0.044	0.268	0.931	0.930	M	M	C
3	-0.9	0	-0.973	0.227				1.632	-2.812	-3.031	0.491	0.067	0.049	0.048	0.071	0.016	M	M	C
4	-0.9	1.5	-0.934	1.588				1.564	-2.483	-4.077	0.527	0.084	0.095	0.097	0.090	0.016	M	M	U
5	0.9	-1.5	0.882	-1.361				1.478	2.673	-4.035	0.471	0.084	0.094	0.095	0.090	0.016	M	M	C
6	0.9	-1	0.857	-0.845				1.421	2.491	-3.329	0.539	0.076	0.070	0.068	0.083	0.017	M	M	C
7	-1.5	0	-1.525	0.048				2.583	-2.416	-2.462	0.488	0.062	0.035	0.035	0.063	0.014	M	M	C
8	1.5	1	1.307	1.164				2.179	2.774	-1.589	0.527	0.073	0.056	0.059	0.076	0.030	M	U	M
9	1.5	1.5	1.564	1.478				2.612	2.802	-1.308	0.530	0.084	0.061	0.063	0.087	0.037	M	U	U
10	0.9	0	0.926	0.088				1.520	2.398	-2.280	0.521	0.068	0.050	0.050	0.074	0.022	M	M	C
11			-0.222	-1.198	0.5	-1.5	-0.2	0.363	-1.594	-0.018	0.999	0.181	0.284	0.593	0.233	1.255	U	M	U
12			-0.229	-0.162	0.5	-1	-0.4	0.687	-0.792	-0.604	0.999	0.172	0.174	0.146	0.128	0.154	U	M	C
13			-0.019	-1.026	0.5	0	-0.6	0.460	0.055	-0.474	0.978	1.606	2.296	0.216	0.193	0.284	U	M	U
14			-0.423	1.073	1	1	-0.2	1.292	0.998	-0.341	1	0.108	0.136	0.076	0.084	0.163	U	U	U
15			0.483	1.194	1	1.5	-0.4	0.919	1.604	-0.354	0.966	0.100	0.129	0.151	0.107	0.221	U	U	U
16			0.051	-0.792	1	0	-0.6	1.076	0.167	-0.662	1	0.719	0.963	0.069	0.095	0.093	U	M	U
17			-0.655	-1.394	1.5	-1.5	-0.2	1.222	-1.502	-0.095	0.993	0.092	0.121	0.131	0.095	0.688	M	M	C
18			0.605	0.923	1.5	1	-0.4	1.615	1.140	-0.484	1	0.086	0.091	0.066	0.085	0.098	U	U	U
19			0.868	1.740	2	1.5	-0.2	2.018	1.624	-0.157	1	0.089	0.112	0.071	0.076	0.322	M	U	U
20			0.517	1.396	2	1	-0.4	1.808	1.019	-0.257	1	0.099	0.136	0.055	0.077	0.170	M	U	U

* The results above are reported after conducting the mean/mean linking. Ind_{MM} is the indicator for manifest monotonicity test. Ind_{GTO} is the indicator for p-value/biserial under classification tree for original data. Ind_{GTT} is the indicator for p-value/biserial method for transformed data. "M" for monotone item, "U" for unfolding item and "C" is uncertain item

TABLE 5.17. Item parameters and Estimates from Mixture model for Simulation Run 10

item	a	b	ea^*	eb^*	α	δ	τ	$e\alpha^*$	$e\delta^*$	$e\tau^*$	$e\pi$	$stda$	$stdb$	$std\delta$	$std\alpha$	$std\tau$	Ind_{MM}	Ind_{GTO}	Ind_{GTT}
1	-1.5	-1.5	-1.410	-1.530				2.477	-1.446	-0.174	0.657	0.089	0.073	0.275	0.960	0.951	M	U	C
2	-1.5	-1	-1.513	-0.866				1.247	-0.952	-0.016	0.531	0.071	0.045	0.200	0.114	1.265	M	M	C
3	-0.9	0	-1.048	0.067				1.757	-2.730	-2.789	0.508	0.066	0.045	0.045	0.069	0.016	M	M	C
4	-0.9	1.5	-0.973	1.489				1.641	-2.768	-4.260	0.484	0.082	0.086	0.087	0.085	0.014	M	M	U
5	0.9	-1.5	0.883	-1.364				1.485	2.900	-4.265	0.470	0.084	0.090	0.091	0.089	0.015	M	M	C
6	0.9	-1	0.993	-0.879				1.670	2.795	-3.673	0.517	0.073	0.061	0.061	0.077	0.014	U	M	C
7	-1.5	0	-1.495	0.060				2.537	-2.845	-2.905	0.502	0.062	0.035	0.035	0.062	0.012	M	M	C
8	1.5	1	1.606	0.998				2.706	2.705	-1.700	0.480	0.070	0.045	0.045	0.072	0.024	M	U	M
9	1.5	1.5	1.534	1.509				2.571	2.835	-1.310	0.523	0.086	0.063	0.065	0.089	0.038	M	U	U
10	0.9	0	0.901	-0.022				1.481	2.521	-2.518	0.548	0.069	0.050	0.050	0.075	0.020	M	M	C
11			-0.226	-1.070	0.5	-1.5	-0.2	0.403	-1.446	-0.369	0.994	0.177	0.252	0.414	0.194	0.416	U	M	U
12			-0.168	-0.571	0.5	-1	-0.4	0.401	-0.952	-0.370	0.967	0.230	0.264	0.379	0.222	0.409	M	M	U
13			-0.040	2.341	0.5	0	-0.6	0.649	-0.002	-0.726	1	0.816	1.028	0.128	0.138	0.136	U	M	U
14			-0.422	1.049	1	1	-0.2	1.220	0.972	-0.312	1	0.107	0.134	0.080	0.085	0.187	M	U	U
15			0.505	1.192	1	1.5	-0.4	0.848	1.982	-0.584	0.773	0.097	0.124	0.162	0.108	0.148	M	U	U
16			0.079	0.089	1	0	-0.6	1.285	-0.073	-0.622	1	0.470	0.473	0.056	0.087	0.084	U	M	C
17			-0.729	-1.452	1.5	-1.5	-0.2	1.097	-1.446	-0.011	0.869	0.090	0.111	0.163	0.104	1.265	M	M	C
18			0.591	0.959	1.5	1	-0.4	1.811	1.031	-0.457	1	0.087	0.095	0.052	0.080	0.094	M	U	U
19			0.864	1.570	2	1.5	-0.2	1.785	1.705	-0.274	0.999	0.085	0.100	0.078	0.079	0.201	M	U	U
20			0.627	1.079	2	1	-0.4	1.645	1.192	-0.409	1	0.086	0.096	0.071	0.087	0.118	M	U	U

* The results above are reported after conducting the mean/mean linking. Ind_{MM} is the indicator for manifest monotonicity test. Ind_{GTO} is the indicator for p-value/biserial under classification tree for original data. Ind_{GTT} is the indicator for p-value/biserial method for transformed data. "M" for monotone item, "U" for unfolding item and "C" is uncertain item

TABLE 5.18. Mean and Standard Deviation of Estimators for Simulation Run 1 to Simulation Run 10

item	a	b	$Mean_{ea}^*$	$Mean_{eb}^*$	α	δ	τ	$Mean_{\alpha}^*$	$Mean_{\delta}^*$	$Mean_{\pi}^*$	std_{ea}^*	std_{eb}	std_{δ}	std_{α}	std_{τ}	
1	-1.5	-1.5	-1.528	-1.474				1.646	2.076	0.711	0.502	0.152	0.104	2.548	1.101	0.595
2	-1.5	-1	-1.509	-0.971				1.884	-1.499	0.788	0.519	0.149	0.103	1.619	0.541	0.782
3	-0.9	0	-0.920	0.042				1.529	-2.571	2.589	0.495	0.077	0.091	0.310	0.138	0.359
4	-0.9	1.5	-0.898	1.547				1.497	-2.405	3.955	0.507	0.054	0.111	0.238	0.102	0.234
5	0.9	-1.5	0.911	-1.466				1.529	2.193	3.654	0.503	0.069	0.105	0.513	0.117	0.499
6	0.9	-1	0.900	-0.965				1.498	2.255	3.208	0.539	0.092	0.110	0.401	0.157	0.400
7	-1.5	0	-1.502	0.007				2.553	-2.278	2.282	0.482	0.093	0.071	0.384	0.159	0.393
8	1.5	1	1.516	0.996				2.276	0.612	1.470	0.493	0.179	0.104	6.441	0.834	0.506
9	1.5	1.5	1.536	1.476				2.379	0.640	1.166	0.570	0.192	0.147	6.435	0.843	0.320
10	0.9	0	0.887	0.037				1.456	2.393	2.321	0.524	0.075	0.081	0.276	0.134	0.262
11			-0.259	-1.075	0.5	-1.5	-0.2	0.474	-1.538	0.419	0.949	0.039	0.195	0.286	0.105	0.313
12			-0.180	-0.459	0.5	-1	-0.4	0.553	-0.845	0.458	0.974	0.046	0.217	0.180	0.155	0.172
13			-0.033	-0.763	0.5	0	-0.6	0.554	0.014	0.511	0.986	0.014	1.281	0.203	0.087	0.157
14			-0.395	1.039	1	1	-0.2	1.077	1.000	0.276	1.000	0.041	0.146	0.124	0.123	0.085
15			0.510	1.063	1	1.5	-0.4	0.986	1.492	0.486	0.960	0.048	0.181	0.210	0.092	0.294
16			0.043	-0.352	1	0	-0.6	1.025	0.011	0.607	1.000	0.019	0.702	0.111	0.133	0.043
17			-0.723	-1.449	1.5	-1.5	-0.2	1.349	-1.599	0.149	0.952	0.062	0.144	0.167	0.187	0.143
18			0.539	0.949	1.5	1	-0.4	1.549	1.009	0.403	1.000	0.049	0.099	0.106	0.151	0.050
19			0.904	1.542	2	1.5	-0.2	1.932	1.637	0.265	0.984	0.074	0.126	0.149	0.155	0.098
20			0.599	1.057	2	1	-0.4	1.933	1.004	0.391	1.000	0.047	0.151	0.107	0.198	0.062

TABLE 5.19. Proportion of “M”, “U”, “C” for Each Indicators Across Simulation Run 1 to Simulation Run 10

Item	Parameter			Ind _{MM}			Ind _{GTO}			Ind _{GTT}			Mean _π			
	a	b		α	δ	τ	M	U	C	M	U	C		M	U	C
1	-1.5	-1.5					1	0	0	0.9	0.1	0	0	0	1	0.502
2	-1.5	-1					1	0	0	1	0	0	0	0	1	0.519
3	-0.9	0					0.9	0.1	0	1	0	0	0	0	1	0.495
4	-0.9	1.5					1	0	0	1	0	0	0	1	0	0.507
5	0.9	-1.5					1	0	0	1	0	0	0	0	1	0.503
6	0.9	-1					0.9	0.1	0	1	0	0	0	0	1	0.539
7	-1.5	0					1	0	0	1	0	0	0	0	1	0.482
8	1.5	1					1	0	0	0.1	0.9	0	0.9	0	0.1	0.493
9	1.5	1.5					1	0	0	0	1	0	0	1	0	0.570
10	0.9	0					1	0	0	1	0	0	0	0	1	0.524
11			0.5	-1.5	-0.2	0.5	0.5	0	1	0	0	0	1	0	0	0.949
12			0.5	-1	-0.4	0.1	0.9	0	1	0	0	0	0.2	0.8	0	0.974
13			0.5	0	-0.6	0	1	0	1	0	0	0	0.7	0.3	0	0.986
14			1	1	-0.2	0.4	0.6	0	0.1	0.9	0	0	0.1	0.9	0	1
15			1	1.5	-0.4	0.9	0.1	0	0	1	0	0.1	0.9	0	0	0.961
16			1	0	-0.6	0	1	0	1	0	0	0	0.4	0.6	0	1
17			1.5	-1.5	-0.2	1	0	0	0.8	0.2	0	0	0	1	0	0.952
18			1.5	1	-0.4	0.7	0.3	0	0	1	0	0	1	0	0	1
19			2	1.5	-0.2	1	0	0	0	1	0	0.1	0.9	0	0	0.984
20			2	1	-0.4	0.6	0.4	0	0	1	0	0	1	0	0	1

Table 5.18 is the summary of these 10 tables by taking the means and standard deviations of the parameter estimates. The means of these estimates in all ten tables are close to the true parameters and the standard deviations are small. Table 5.19 is the summary of the 10 tables by taking the proportion of "M", "U", "C" for each indicator for all 20 items. As observed from this table, Ind_{MM} almost always correctly identified monotone items, Ind_{GTO} had difficulty with two items, and as expected from its cross-validation error rates, Ind_{GTT} performed poorly. For the unfolding items, Ind_{GTO} performed well with roughly half the items (14 – 15, 18 – 20), but poorly on the rest. Ind_{MM} , on the other hand, performed well on items 12 – 13 and 16. Ind_{GTT} again performed poorly. The estimates from the mixture model on the other hand consistently gave the unfolding items posterior probabilities of being unfolding near one, while giving the monotone items posterior probabilities much closer to 0.5

CHAPTER 6

SUMMARY AND FUTURE WORK

A great deal of educational and psychological research involves the use of questionnaires and other instruments to measure latent traits such as attitudes. Historically, instrument designers have been restricted to only using one item type, either monotone items or unfolding items. This is due to the lack of statistical methodology for modeling both item types at the same time.

This research is a first step in opening up the use of both item types simultaneously, allowing researchers much greater flexibility in the types of instruments that they can use. In this initial work, the monotone items in our simulation study were all $2PL$ items and unfolding item with $\tau = -0.2$. We explored three methods to identify the item types: the manifest monotonicity test, the p-value/biserial method, and the mixture model with marginalized Bayes modal estimation.

When the manifest monotonicity test and the p-value/biserial method are utilized to try to find the right item types, they both can provide valuable information. However, both methods have limitations. The manifest method is more useful for the items with locations near the center of the ability distribution. For the items located in either extreme, the method does not work. An ad-hoc rule can be established by using p-value/biserial method which has been applied to attain the initial value of π for the mixture model $(0, 0.5, 1)$. This rule can separate the items into monotone, unfolding or uncertain item type through the comparison of the p-value and biserial correlation calculated from both the original data and transformed data. This method still has limitations on the extremes.

To better distinguish the item types and to estimate the parameters of the model, a mixture model was proposed. Marginalized Bayes modal estimation has been utilized for estimating this model. The method of initial parameter estimation for each parameter has been discussed in Chapter 4. It was found that the initial estimate of π had little effect on the final estimates, but a better estimate of it could lead to using fewer iterations and much shorter running time. Through the simulation study, the estimates for all the parameters were close to the true parameters. The estimated probability of each item to be unfolding from the model were either close to 1 for the unfolding items or close to 0.5 for the monotone items. In Chapter 5, we used five simulation studies to study the reason for estimated $\pi = 0.5$. We found that the IRFs for the monotone items with monotone estimates and unfolding estimates overlapped in the middle and then separated in the end. Since we assumed θ followed the standard normal distribution, the values of θ are too few in the extremes to make a large difference between two the likelihoods of L_u and L_m . Therefore, the π for these items are close to 0.5.

There are a few concerns for this method. The first is the computing time. The algorithm typically took 7 days to run on a computer (CPU Intel(R) Core(TM) i5 M520 @ 2.40GHz) for the data set with 40 items and 1000 examinees with more than 1000 iteration. Another problem is the mixture model was able to clearly identify the unfolding items, but the monotone items were found to be equally likely to be monotone or unfolding for the simulation settings used.

In future studies, we might use Fortran to take the place of R which will provide much quicker computation. Additional future research should also consider other methods of determining which model is appropriate (AIC, BIC or adjusted BIC). Since the two models (unfolding and monotone) have the approximately same likelihood, having one fewer parameter should make the $2PL$ more likely to be chosen (since AIC and BIC are likelihood with a penalty for parameters).

Additional research should be conducted in expanding the types of items used. For the monotone items, this could involve expanding from the *2PL* to the *3PL*. In the proposed model of this research, there are only discrimination parameter a and difficulty parameter b . The guessing parameter c can be added to the model. Therefore, the p_m Equation 3.1 will become

$$P_m(X_{ij} = k|\theta, (a_i, b_i, c_i)) = c_i + (1 - c_i) \frac{\exp(Da_i(\theta_j - b_i))}{1 + \exp(Da_i(\theta_j - b_i))} \quad (6.1)$$

This will result in a small change for the algorithm used to calculate $\frac{\partial \log L}{\partial \varphi_i}$ in Equation 3.12 when $\varphi_i = c_i$ in E-step. It will also require adding c in the Fisher method of scoring matrix (Equation 3.17) in M-step. Then EM algorithm in Chapter 3 will bring the estimates for c_i s. Another expansion is to advance from dichotomous models to polytomous IRT models which will allow the work to be more widely applied in educational and psychological studies.

BIBLIOGRAPHY

- Andrich, D. (1988). The application of an unfolding model of the PIRT type to the measurement of attitude. *Applied Psychological Measurement*, 12, 33–51.
- Breiman, L., Friedman, J. H., Olshen, R. A., Stone, C. J. (1984). Classification and Regression Trees. Wadsworth International Group: Belmont, California.
- Birnbaum, A. (1968). Some Latent Trait Models and Their Use in Inferring an Examinees Ability. *Statistical Theories of Mental Test Scores*, In F.M Lord and M.R. Novick (eds.), 397-472, Reading, MA: Addison-Wesley.
- Bock, R.D. & Aitkin M. (1981). Marginal maximum likelihood estimation of item parameter. An application of an EM algorithm. *Psychometrika*, 46,443-459
- Bock, R.D. & Lieberman, M. (1970). Fitting a response model for n dichotomously scored items. *Psychometrika*, 35, 179-197
- Clenshaw, C. W. & Curtis, A. R. (1960). A method for numerical integration on an automatic computer. *Numerische Mathematik* 1960, Volume 2, Issue 1, pp 197-205.
- Coombs, C.H (1964). The unfolding technique in one dimension. *A theory of data* Coombs C.H. (Ed.), Ann Arbor, Michigan, Mathesis Press, 80–121.
- Crocker, L. & Algina J. (1986). Introduction to classical and modern test theory. New York: Harcourt Brace Jovanovich College Publishers
- Daniels, M. J. (1999). Item response theory Parameter Estimation Techniques. Second Edition, revised and expanded, New York: Marcel Dekker.
- Donoghue, J.R. & Allen, Nancy L. (1993). Thin versus thick matching in the Mantel-Haenszel procedure for detecting DIF *Journal of Educational Statistics*, 2, 131–154.
- Baker, F.B. (1992). Item response theory, parameter estimation techniques. New York: Marcel Dekker.

- GMokken, RJ & Lewis, C (1982). A Nonparametric Approach to the Analysis of Dichotomous Responses . *Applied Psychological Measurement*, 6 417–430
- Hambleton, R. K. (1985). Item Response Theory: Principles and Application. textit Newbury Park, CA: Sage Publications, Inc.
- Hambleton, R. K., Swaminathan, H., Rogers, H. J. (1991). Fundamentals of item response theory. Kluwer Academic Pub
- Hojtink, H (1990). A latent trait model for dichotomous choice data. *Psychometrika*, 55, 641–656.
- Junker, BW & Sijtsma, K (2000). Latent and Manifest Monotonicity. *Applied Psychological Measurement*, 24, 6581.
- Kolen, MJ & Brennan, RL (2004). Test Equating, Scaling, and Linking: Methods and Practices. 2nd edition. Springer-Verlag, New York.
- Loevinger, J. (1947). A systematic approach to the construction and evaluation of test of ability. *psychological monographs*, 61(4).
- Lord, M. F. & Novick, R. M. (1968). Statistical Theories of Mental Test Scores. Reading MA: Addison-Welsley Publishing Company
- Moisen, G. G. (2008). Classification and Regression Trees. *Ecological Informatics* p.582.
- Mokken, RJ (1971). A Theory and Procedure of Scale Analysis. De Gruyter, Berlin, Germany.
- Molenaar, I. & Sijtsma, K. (2000). User's manual MSP5 for Windows: a program for Mokken scale analysis for polytomous items. Groningen: ProGAMMA.
- Muraki, E (1992). A generalized partial credit model: Application of an EM algorithm. *Applied Psychological Measurement*, 16, 159–176.
- Post, W.J. & Snijders, T.A.B (1993). Nonparametric unfolding models for dichotomous data. *Methodika*,7,130–156.
- Post, W.J., Van Duijn, M.A.J.,& Van Baarsen, B (2001). A Single-peaked or monotone tracelines? On the choice of an IRT model for scaling data. In: Boomsma, A., Van Duijn, M.A.J.,Snijders, T.A.B. (Eds.), *Essays on item response theory*. New York, Springer Verlag.

- Roberts, J. S. & Laughlin, J. E. (1996). A unidimensional item response model for unfolding responses from a graded disagree-agree response scale. *Applied Psychological Measurement*, 20, 231–255
- Roberts, J. S., Donoghue, J. R., & Laughlin, J. E. (2000). A general item response theory model for unfolding unidimensional polytomous responses. *Applied Psychological Measurement*, 24, 3–32
- Roberts, J. S., Donoghue, J. R., & Laughlin, J. E. (2000). Characteristics of MM-L/EAP Parameter Estimates in the Generalized Graded Unfolding Model. *Applied Psychological Measurement*, 26, 192–207
- Sijtsma, K., Debets, P. & Molenaar, I. W (1990). Mokken scale analysis for polychotomous items: Theory, a computer program and an empirical application. *Quality Quantity*, 24, 173–188
- Sijtsma, K. & Molenaar, I. W (2002). Introduction to nonparametric Item Response Theory, Thousand Oaks, Sage
- Therneau, T. M., Atkinson, B., Ripley, B.D. (2006). rpart: Recursive Partitioning [<http://CRAN.R-project.org/>]
- Weeks, J (2010). plink: An R Package for Linking Mixed-Format Tests Using IRT-Based Methods, *Journal of Statistical Software*, Volume 35, Issue 12.
- Wood, D. A. (1960). Test construction: Development and interpretation of achievement tests. Columbus, OH: Charles E. Merrill Books.
- Yang, N. & Habing, B. (July, 2008) Identifying item type on mixed unfolding/monotone instruments. Paper presented at the Annual Meeting of the Psychometric Society, Durham, NH.

APPENDIX A: GAUSS-HERMITE INTEGRATION

Gauss-Hermite Integration (Clenshaw, 1960) is a numerical method for integration over the interval of $(-\infty, \infty)$. Let $f(x)$ be the function needed to be integrated, the Gauss-Hermite integration formula is

$$\int_{-\infty}^{\infty} f(x)dx = \int_{-\infty}^{\infty} \exp -x^2[\exp x^2 f(x)]dx \approx \sum_{f=1}^n w(x_f) \exp x_f^2 f(x_f) \quad (1)$$

Abscissas and Weights of Gauss-Hermite Integration

No.	Abscissas x_f	Total Weight $\exp x_f^2 f(x_f)$
1	-4	0.000119
2	-3.1111	0.002805
3	-2.2222	0.03002
4	-1.3333	0.1458
5	-.4444	0.3213
6	.4444	0.3213
7	1.3333	0.1458
8	2.2222	0.03002
9	3.1111	0.002805
10	4	0.000119

APPENDIX B: EQUATION FOR DERIVATIVE

Now we look at the derivative part of Equation $\frac{\partial P(y_i=k|v_f)}{\partial \varphi_i}$. Since we only deal with the dichotomous case, there is only one τ_i that need to be estimated (τ_i) and k can only take 0 and 1. Let

$$g_1 = \exp(-1.7\exp(al_i)(v_f - b_i)) \quad (2)$$

$$g_{001} = \exp(\exp(\gamma_i)0) = 1 \quad (3)$$

$$g_{002} = \exp(3\exp(\gamma_i)(\theta_j - \delta_i) - \exp(\gamma_i)\exp(t_i)) \quad (4)$$

$$g_{011} = \exp(\exp(\gamma_i)(\theta_j - \delta_i) - \exp(\gamma_i)\exp(t_i)) \quad (5)$$

$$g_{012} = \exp(2\exp(\gamma_i)(\theta_j - \delta_i) - \exp(\gamma_i)\exp(t_i)) \quad (6)$$

Let P_1 stand for the monotone IRF and P_0 stand for the IRF of GGUM. Then

$$f_{11} = P_1(Y_i = 1) = \frac{1}{1 + g_1} \quad (7)$$

$$f_0 = P_0(Y_i = 1) = \frac{f_{01}}{f_{00} + f_{01}} \quad (8)$$

where

$$f_{00} = g_{001} + g_{002}$$

$$f_{01} = g_{011} + g_{012}$$

$$f_{10} = P_1(Y_i = 0) = 1 - P_1(Y_i = 1) \quad (9)$$

$$P_0(Y_i = 0) = 1 - P_0(Y_i = 1) \quad (10)$$

The derivative with respect to al_i

$$\begin{aligned} \frac{\partial P(Y_i = 1|v_f)}{\partial al_i} &= \frac{\partial(\pi_i f_0(y_i = 1|v_f) + (1 - \pi_i) f_1(y_i = 1|v_f))}{\partial al_i} \\ &= \frac{\partial(1 - \pi_i) f_1(y_i = 1|v_f)}{\partial al_i} \\ &= \frac{1 - \pi_i}{(1 + g_1)^2} g_1 (-1.7(v_f - \beta)) \exp(al_i) \end{aligned}$$

and,

$$\begin{aligned}
\frac{\partial P(Y_i = 0|v_f)}{\partial al_i} &= \frac{\partial(\pi_i f_0(Y_i = 0|v_f) + (1 - \pi) f_1(Y_i = 0|v_f))}{\partial al_i} \\
&= \frac{\partial(1 - \pi_i) f_1(Y_i = 0|v_f)}{\partial al_i} \\
&= \frac{-(1 - \pi_i) \partial f_1(Y_i = 1|v_f)}{\partial al_i} \\
&= -\frac{\partial P(Y_i = 1|v_f)}{\partial al_i}
\end{aligned}$$

The derivative with respect to β_i

$$\begin{aligned}
\frac{\partial P(Y_i = 1|v_f)}{\partial \beta_i} &= \frac{\partial \pi_i f_0(Y_i = 1|v_f) + (1 - \pi_i) f_1(Y_i = 1|v_f)}{\partial \beta_i} \\
&= (1 - \pi_i) \frac{\partial f_1(Y_i = 1|v_f)}{\partial \beta_i}
\end{aligned}$$

where

$$\frac{\partial f_{11}}{\partial \beta_i} \equiv \frac{\partial f_1(Y_i = 1|v_f)}{\partial \beta_i} = \frac{1}{(1 + g_1)^2} g_1 1.7 \exp(al_i) \quad (11)$$

$$\begin{aligned}
\frac{\partial P(Y_i = 0|v_f)}{\partial \beta_i} &= -(1 - \pi_i) \frac{\partial f_1(Y_i = 1|v_f)}{\partial \beta_i} \\
&= -\frac{\partial f_{11}}{\partial \beta_i}
\end{aligned}$$

The derivative with respect to δ

$$\begin{aligned}
\frac{\partial P(Y_i = 1|v_f)}{\partial \delta_i} &= \frac{\partial \pi_i f_0(Y_i = 1|v_f) + (1 - \pi_i) f_1(Y_i = 1|v_f)}{\partial \delta_i} \\
&= \pi_i \frac{\partial f_0(Y_i = 1|v_f)}{\partial \delta_i}
\end{aligned}$$

where

$$\begin{aligned}
\frac{\partial f_0(Y_i = 1|v_f)}{\partial \delta_i} &= \frac{\partial \frac{f_{01}}{f_{01} + f_{00}}}{\partial \delta_i} \\
&= \frac{\frac{\partial f_{01}}{\partial \delta_i} f_{00} - \frac{\partial f_{00}}{\partial \delta_i} f_{01}}{(f_{00} + f_{01})^2}
\end{aligned}$$

$$\frac{\partial f_{01}}{\partial \delta_i} = -\exp(\gamma_i) g_{011} - 2\exp(\gamma_i) g_{012} \quad (12)$$

$$\frac{\partial f_{00}}{\partial \delta_i} = -3\exp(\gamma_i) g_{002} \quad (13)$$

$$\begin{aligned}\frac{\partial P(Y_i = 0|v_f)}{\partial \delta_i} &= -(1 - \pi_i) \frac{\partial f_1(Y_i = 1|v_f)}{\partial \delta_i} - \pi_i \frac{\partial f_0(Y_i = 1|v_f)}{\partial \delta_i} \\ &= -\frac{\partial P(Y_i = 1|v_f)}{\partial \delta_i}\end{aligned}$$

The derivative with respect to γ

$$\begin{aligned}\frac{\partial P(Y_i = 1|v_f)}{\partial \gamma_i} &= \pi_i \frac{\partial f_0(Y_i = 1|v_f)}{\partial \gamma_i} \\ &= \pi_i \frac{\frac{f_{01}}{f_{00}+f_{01}}}{\partial \gamma_i} \\ &= \pi_i \frac{\frac{\partial f_{01}}{\partial \gamma_i} f_{00} - \frac{\partial f_{00}}{\partial \gamma_i} f_{01}}{(f_{00} + f_{01})^2}\end{aligned}$$

where

$$\frac{\partial f_{00}}{\partial \gamma_i} = 3(v_f - \delta_i)g_{002}exp(\gamma_i) \quad (14)$$

$$\frac{\partial f_{01}}{\partial \gamma_i} = (v_f - \delta_i + exp(t_i))g_{011}exp(\gamma_i) + (2(v_f - \delta_i) + exp(t))g_{012}exp(\gamma_i) \quad (15)$$

and

$$\frac{\partial P(Y_i = 0|v_f)}{\partial \gamma_i} = -\frac{\partial P(Y_i = 1|v_f)}{\partial \gamma_i} \quad (16)$$

Derivative with respect to t_i

$$\begin{aligned}\frac{\partial P(Y_i = 1|v_f)}{\partial t_i} &= \pi_i \frac{\partial f_0(Y_i = 1|v_f)}{\partial t_i} \\ &= \pi_i \frac{\frac{f_{01}}{f_{00}+f_{01}}}{\partial t_i} \\ &= \pi_i \frac{\frac{\partial f_{01}}{\partial t_i} f_{00} - \frac{\partial f_{00}}{\partial t_i} f_{01}}{(f_{00} + f_{01})^2}\end{aligned}$$

where

$$\frac{\partial f_{00}}{\partial t_i} = 0 \quad (17)$$

$$\frac{\partial f_{01}}{\partial t_i} = exp(t_i)exp(\gamma_i)(g_{011} + g_{012}) \quad (18)$$

and

$$\frac{\partial P(Y_i = 0|v_f)}{\partial t_i} = -\frac{\partial P(Y_i = 1|v_f)}{\partial t_i} \quad (19)$$

APPENDIX C: ITEM PARAMETERS FOR CHAPTER 2

Item Parameters for 10 Monotone Items in Section 2.2

Item	1	2	3	4	5	6	7	8	9	10
<i>a</i>	-1.4	-1.4	-1.3	-1.4	-1.5	1.2	1.5	1.1	1	1.3
<i>b</i>	-0.5	-0.3	-0.2	-0.8	-1.4	-1.3	-0.4	-0.9	0.9	-0.6
<i>c</i>	0	0	0	0	0	0	0	0	0	0

Item Parameters for 10 Unfolding Items in Section 2.2

Item	11	12	13	14	15	16	17	18	19	20
α	2.1	2.1	2	2.1	2	1.7	2.2	1.7	1.5	2
<i>delta</i>	-0.5	-0.3	-0.2	-0.8	-1.4	-1.3	-0.4	-0.9	0.9	-0.6
τ_1	-0.2	-0.2	-0.2	-0.2	-0.2	-0.2	-0.2	-0.2	-0.2	-0.2

APPENDIX D: R CODE FOR CHAPTER 2

```
#R code for reading the parameters (ITEM POOL)

par3pl<-paste("a=",as.character(0.9),".txt",sep="")
par3pl<-read.table(par3pl,header=T)
par3pl<-subset(par3pl,select=c(alpha,delta,c))

for(i in 1:6)
{
temp1<-paste("a=",as.character(0.9+0.1*i),".txt",sep="")
par3pl1<-read.table(temp1,header=T)
par3pl1<-subset(par3pl1,select=c(alpha,delta,c))
par3pl<-rbind(par3pl,par3pl1)
}

for(i in 1:7)
{
temp2<-paste("a=-",as.character(0.8+0.1*i),".txt",sep="")
par3pl2<-read.table(temp2,header=T)
par3pl2<-subset(par3pl2,select=c(alpha,delta,c))
par3pl<-rbind(par3pl,par3pl2)
}

parggum<-paste("GGUMa=",as.character(0.9),".txt",sep="")
parggum<-read.table(parggum,header=T)
parggum<-subset(parggum,select=c(alpha,beta,tau0,tau1))

for(i in 1:6)
{
temp1<-paste("GGUMa=",as.character(0.9+0.1*i),".txt",sep="")
parggum1<-read.table(temp1,header=T)
parggum1<-subset(parggum1,select=c(alpha,beta,tau0,tau1))
parggum<-rbind(parggum,parggum1)
```

```

}

for(i in 1:7)
{
temp2<-paste("GGUMa=-",as.character(0.8+0.1*i),".txt",sep="")
parggum2<-read.table(temp2,header=T)
parggum2<-subset(parggum2,select=c(alpha,beta,tau0,tau1))
parggum<-rbind(parggum,parggum2)
}

#R function for items.txt which compute biserial,
#point biserial correlation etc.
items<-function(testdata,correct=T,dpct=0.27,digits=4){
n<-ncol(testdata)
nexmn<-nrow(testdata)
x<-apply(testdata,1,sum)
medx<-median(x)
lowx<-sort(x)[floor((nexmn+1)*dpct)]
highx<-sort(x)[ceiling((nexmn+1)*(1-dpct))]
pvalue<-as.numeric(apply(testdata,2,mean))
var<-as.numeric(apply(testdata,2,var)*(nexmn-1)/nexmn)
D<-rep(0,n)
PBis<-rep(0,n)
Bis<-rep(0,n)
for (i in 1:n){
if (correct==T){
xx<-x-testdata[,i]
lowx<-sort(xx)[floor((nexmn+1)*dpct)]
highx<-sort(xx)[ceiling((nexmn+1)*(1-dpct))]
}
else {xx<-x}
pu<-mean(testdata[xx >= highx,i])
pl<-mean(testdata[xx <= lowx,i])
D[i]<-pu-pl
PBis[i]<-cor(xx,testdata[,i])
Bis[i]<-sqrt(pvalue[i]*(1-pvalue[i]))*PBis[i]
/dnorm(qnorm(pvalue[i]))
}
round((10^digits)*cbind(pvalue,var,D,PBis,Bis))/(10^digits)}

#R code for Mokken scale analysis

```

```

library(mokken)
#R function check.monotonicity and check.monotonicity
check.monotonicity<-
function(X, minvi = .03, minsize = default.minsize){
  X <- check.data(X)
  N <- nrow(X)
  J <- ncol(X)
  m <- max(X) + 1
  default.minsize <- ifelse(N > 500, floor(N/10), floor(N/5))
  default.minsize <- ifelse(N <= 250, floor(N/3), default.minsize)
  default.minsize <- ifelse(N < 150, 50, default.minsize)
  if (N < minsize) stop("Sample size less than Minsize")
  # Initial computation
  R <- as.matrix(X) %*% (matrix(1,J,J) - diag(J))
  results <- list()
  # results checks per item
  I.labels <- dimnames(X)[[2]]
  if(length(I.labels)==0) I.labels <- paste("C",1:ncol(X))
  for (j in 1:J){
    violation.matrix <- matrix(0,nrow=m,ncol=10)
    dimnames(violation.matrix) <-
list(c(paste("P(X >=",1:(m-1),")",sep=""),"Total"),
  dimnames(violation.matrix)[[2]] <-
  c("#ac", "#vi", "#vi/#ac", "maxvi", "sum", "sum/#ac",
"zmax", "group", "group", "#zsig"))
    results[[j]] <- list()
    results[[j]][1] <- I.labels[j]
    sorted.R <- sort(R[,j])
    group <- max(which(sorted.R==sorted.R[minsize]))
    repeat{
      if(N - max(group) < minsize)break
      group <-
c(group,max(which(sorted.R==sorted.R[minsize+max(group)])))
    }
    group <- group[-length(group)]
    summary.matrix <- matrix(nrow = length(group)+1,ncol = 4 + 2* m)
    dimnames(summary.matrix)[[2]] <- c("Group", "Lo Score", "Hi Score", "N",
paste("F",0:(m-1)), "Mean", paste("P(X >=",1:(m-1),")",sep=""))
    summary.matrix[,1] <- 1:nrow(summary.matrix)
    summary.matrix[,4] <- c(group,N) - c(0,group)
    group <- c(sorted.R[group],max(sorted.R))
    L <- length(group)
    summary.matrix[,3] <- group
    summary.matrix[,2] <- c(min(sorted.R),group[-L]+1)
    member <- apply(1 - outer(R[,j], group, "<="),1,sum) + 1

```

```

for (i in 1:L){
  Ni <- summary.matrix[i,4]
  freq <- tabulate(X[member==i,j]+1,m)
  summary.matrix[i,5:(m+4)] <- freq
  summary.matrix[i,m+5] <- sum(freq * min(X):max(X)) / Ni
  cum.freq <- rev(cumsum(rev(freq))/Ni)
  summary.matrix[i,(m+6):(2*m+4)] <- cum.freq[2:m]
}
results[[j]][[2]] <- summary.matrix
violation.matrix[1:(m-1),1] <- L*(L-1)*.5
violation.matrix[m,1] <- L*(L-1)*.5 * (m-1)
freq <- summary.matrix[,5:(m+4)]
for (i in 1:(m-1)){
  V <- outer(summary.matrix[, (m+5+i)],summary.matrix[, (m+5+i)], "-")
  V[row(V) <= col(V)] <- 0
  V[V >= -minvi] <- 0
  violation.matrix[i,2] <- sum(ceiling(abs(V)))
  violation.matrix[i,4] <- max(abs(V))
  if(violation.matrix[i,4] > minvi){
    violation.matrix[i,5] <- sum(abs(V))
    freqd <- cbind(apply(as.matrix(freq[,1:i]),1,sum),
apply(as.matrix(freq[, (i+1):m]),1,sum))
    Z <- abs(sign(-V) * 2 * (sqrt(outer(freqd[,2]+1,freqd[,1]+1))
- sqrt(outer(freqd[,1],freqd[,2])))) /
sqrt(outer(freqd[,2],freqd[,1], "+")
+ outer(freqd[,1],freqd[,2], "+")))
    violation.matrix[i,7] <- max(Z)
    violation.matrix[i,8] <- min(col(Z) [Z==max(Z)])
    violation.matrix[i,9] <- min(row(Z) [Z==max(Z)])
    violation.matrix[i,10] <- sum(sign(Z[Z > 1.6449]))
  }
}
violation.matrix[m,2] <- sum(violation.matrix[1:(m-1),2])
violation.matrix[1:m,3]
<- violation.matrix[1:m,2]/violation.matrix[1:m,1]
violation.matrix[m,4] <- max(violation.matrix[1:(m-1),4])
violation.matrix[m,5] <- sum(violation.matrix[1:(m-1),5])
violation.matrix[1:m,6]
<- violation.matrix[1:m,5]/violation.matrix[1:m,1]
violation.matrix[m,7] <- max(violation.matrix[1:(m-1),7])
violation.matrix[m,10] <- sum(violation.matrix[1:(m-1),10])
results[[j]][[3]] <- violation.matrix
results[[j]][[4]] <- paste("Minsize = ",minsize," Minvi = ",minvi,sep="")
}
Hi <- coefH(X)$Hi
monotonicity.list <- list(results=results,I.labels=I.labels,Hi=Hi,m=m)

```

```

class(monotonicity.list) <- "monotonicity.class"
return(monotonicity.list)
}

check.data <-
function(X){
if (data.class(X) != "matrix" && data.class(X) != "data.frame")
stop("Data are not matrix or data.frame")
matrix.X <- as.matrix(X)
if (is.na(any(X))) stop("Missing values are not allowed")
if (any(mode(matrix.X)!="numeric")) stop("Data must be numeric")
if (any(matrix.X < 0)) stop("Data should be positive")
matrix.X <- matrix.X - min(matrix.X)
return(matrix.X)
}

#R function for generate simulation data set
#irf function used for 3pl model
irf<-function(theta=0,a=1,b=0,cc=0,type="3pl"){
n<-length(a)
prob<-0
if (type=="3pl"){
for (i in 1:n){
prob<-prob+cc[i]+(1-cc[i])/(1+exp(-a[i]*(theta-b[i])))
}
}
else{
for (i in 1:n){
prob<-prob+cc[i]+(1-cc)[i]*pnorm(a[i]*(theta-b[i]))
}
}
return(prob)
}

#the function to generate the scores for 12 items
irtgen<-function(nexmn=10,avec=c(1),bvec=c(0),cvec=c(0),delta,alpha,
tau0,tau1,mthet=0,sdthet=1)
{
n1<-length(avec)
n2<-length(delta)
data<-matrix(0,nrow=nexmn,ncol=n1+n2)
ability<-rnorm(nexmn,mean=mthet,sd=sdthet)
for (j in 1:nexmn)
{
for (i in 1:n1)

```

```

{
  try<-runif(1,0,1)
  if (try < irf(ability[j],avec[i],bvec[i],cvec[i]))
    {
      data[j,i]<-1
    }
}
n3=n1+1
for (i in 1:n2)
{
  try<-runif(1,0,1)

  if (try < model8(z=1,theta=ability[j],
delta=delta[i],alpha=alpha[i],
tau0=tau0[i],tau1=tau1[i]))
    {
      data[j,n1+i]<-1
    }
}
}
return(data)
}

```

#R code for Table 2.1

```

itemn<-round(runif(10,1,434))
sampleggum<-parggum[itemn,]
sample3pl<-par3pl[itemn,]
a<-sample3pl[,1]
b<-sample3pl[,2]
cc<-sample3pl[,3]
del<-sampleggum[,2]
alp<-sampleggum[,1]
tau1<-sampleggum[,4]
tau0<-sampleggum[,3]

```

```

l=length(a)+length(alp)
count<-matrix(0,nrow=1,ncol=l)
h=1000

```

```

for(i in 1:h)
{
  sampledata<-irtgen(nexmn=1000,avec=a,bvec=b,cvec=cc,delta=del,
alpha=alp,tau0=tau0,tau1=tau1)

```

```

coef<-coefH(sampledata)

```



```

sampledata2<-sampledata
trace<-NULL
for(i in 1:l)
{
if(coef$Hij[i,1]<0)
{sampledata2[,i]<-1-sampledata2[,i]
trace[i]<-i
}
}

summ<-summary(check.monotonicity(sampledata2))
for(j in 1:l)
{
if (summ[j,8]>0) {count[j]=count[j]+1}
}
}

#R code for the P-value/Biserial method
par(mfrow=c(1,2))

estimatedb<-matrix(0,ncol=n,nrow=1)
pv<-matrix(0,ncol=n,nrow=1)

for(i in 1:n)
{
model8<-function(z=1,theta,delta=del[i],alpha=al[i],tau0,tau1){
prob0<-exp(-alpha*tau0)+exp(alpha*(3*(theta-delta)-tau0))
prob1<-exp(alpha*((theta-delta)-(tau0+tau1)))+
exp(alpha*(2*(theta-delta)-(tau0+tau1)))
probt<-prob0+prob1
if(z==0)
{ prob<-prob0/probt
}
else
prob<-prob1/probt
prob
}

pvalue<-function(theta,delta=del[i],
alpha=alp[i],z=1,tau0=0,tau1=-0.2)
{
model8(theta,delta=delta,alpha=alpha,z=1,
tau0=tau0,tau1=tau1)/
sqrt(2*pi)*exp(-theta^2/2)
}
}

```

```

pv[i]<-integrate(pvalue,-30,30)[[1]]
pbisf<-function(theta,delta=del[i],alpha=alp[i],
z=1,tau0=0,tau1=-0.2)
{
theta*model8(theta,delta=delta,alpha=alpha,z=1,
tau0=tau0,tau1=tau1)/
sqrt(2*pi)*exp(-theta^2/2)
}

pbis<-integrate(pbisf,-30,30)[[1]]/(pv[i]*(1-pv[i]))

bis<-sqrt(pv[i]*(1-pv[i]))*pbis/dnorm(qnorm(pv[i]))
estimatedb[i]<--qnorm(pv[i])/bis
}
plot(estimatedb,pv,ylim=c(0,1),xlim=c(-2,2),ylab="",pch="u",xlab="estimatedb")
estimatedbm<-matrix(0,n,n)
pvm<-matrix(0,n,n)
for(i in 1:n)
{
for(j in 1:n)
{
irf<-function(theta,a=alp[i],b=del[j],cc=0){cc+(1-cc)/(1+exp(-1.7*a*(theta-b)))}
}
pvaluem<-function(theta,b=del[j],a=alp[i],cc=0)
{
irf(theta,a=a,b=b,cc=cc)/sqrt(2*pi)*exp(-theta^2/2)
}

pvm[i,j]<-integrate(pvaluem,-30,30)[[1]]
pbisfm<-function(theta,a=alp[i],b=del[j],cc=0)
{
theta*irf(theta,b=b,a=a,cc=cc)
/sqrt(2*pi)*exp(-theta^2/2)
}

pbism<-integrate(pbisfm,-30,30)[[1]]/(pvm[i,j]*(1-pvm[i,j]))
bism<-sqrt(pvm[i,j]*(1-pvm[i,j]))*
pbis/dnorm(qnorm(pvm[i,j]))
estimatedbm[i,j]<--qnorm(pvm[i,j])/bism
}
}

estimatedbms<-subset(estimatedbm,select=(-16))
pvms<-subset(pvm,select=(-16))
par(new=T)

```

```

plot(estimatedbms[10,],pvms[10,],
ylim=c(0,1),xlim=c(-2,2),
ylab="p-value",pch="m",xlab="")

abline(h=0.4)
abline(v=1)
abline(v=-1)
title("items' theory result for estimatedb vs p-value ")

#without transfer
sampledata<-irtgen(nexmn=1000,avec=a, bvec=b,cvec=cc,delta=del,
alpha=alp,tau0=tau0,tau1=tau1)
sampledata2<-sampledata

pv<-items(sampledata2)[,1]
estimatedb<--qnorm(items(sampledata2)[,1])/items(sampledata2)[,5]

n=length(alp)
lb<-n+1
up<-2*n
plot(estimatedb[1:n],pv[1:n],xlim=c(-5,5),ylim=c(0,1),pch="m",
ylab="p-value",xlab=" estimatedb")

par(new=T)
plot(estimatedb[lb:up],pv[lb:up],xlim=c(-5,5), ylim=c(0,1),
pch="u",xlab="",ylab="")
abline(h=0.4)
abline(v=-1)
abline(v=0.8)

title("items simulation result for Estimatedb vs P-value",
sub="original dataset without transfer")
plot(estimatedb,pv,xlim=c(-5,5),ylim=c(0,1),pch="m", ylab="p-value",
xlab=" estimatedb",type="n")
text(estimatedb,pv,as.character(1:40))
abline(h=0.4)
abline(v=-1)
abline(v=1)
title("items simulation result for Estimatedb vs P-value")

#transform all the items which has the negative H
coef<-coefH(sampledata)
sampledata3<-sampledata
for(i in 1:dim(sampledata)[2])
{

```

```

if(coef$Hij[i,1]<0)
{sampledata3[,i]<-1-sampledata[,i]
}
}
pv<-items(sampledata3)[,1]
estimatedb<--qnorm(items(sampledata3)[,1])/
items(sampledata3)[,5]
plot(estimatedb,pv,xlim=c(-5,5),ylim=c(0,1),
pch="m",ylab="p-value",
xlab=" estimatedb",type="n")
text(estimatedb,pv,as.character(1:20))
abline(h=0.4)
abline(v=-1)
abline(v=1)
title("items simulation result for Estimatedb vs P-value after transform")

#transfer all the decreasing score
# (decreasing monotone items)
sampledata4<-sampledata
for(i in 1:n)
{
if (a[i]<0)
{
sampledata4[,i]<-1-sampledata[,i]
}
}
pv<-items(sampledata4)[,1]
estimatedb<--qnorm(items(sampledata4)[,1])/
items(sampledata4)[,5]

n=length(alp)
lb<-n+1
up<-2*n
plot(estimatedb[1:n],pv[1:n],xlim=c(-5,5),ylim=c(0,1),pch="m",
ylab="p-value",xlab=" estimatedb")
par(new=T)
plot(estimatedb[lb:up],pv[lb:up],xlim=c(-5,5),
ylim=c(0,1),pch="u",xlab="",ylab="")

plot(estimatedb,pv,xlim=c(-5,5),ylim=c(0,1),pch="m",ylab="p-value",
xlab=" estimatedb",type="n")
text(estimatedb,pv,as.character(1:40))

abline(h=0.4)
abline(v=-1)

```

```
abline(v=1)
```

```
title("items simulation result for Estimatedb vs P-value",  
sub="only change the score for decreasing monotone items")
```

APPENDIX E: R CODE FOR THE MARGINALIZED BAYES MODAL ESTIMATION

```
#pdfs and first derivative for each parameter
#saved in C:/Na/Code/fuct of in de m and u.txt
#monotone model(2pl)
#abs(a) has the log normal distribution, let al<-log(abs(a)),
#then al has the normal distribution
# let s be the sign of the a
#then original a=s*exp(a)

#pdf of monotone
g1<-function(a,theta,beta,si)
{
  exp(-1.7*exp(a)*(theta-beta)*si)
}
f1<-function(a,theta,beta,si)
{
  1/(1+g1(a,theta,beta,si))
}

#unfolding model
g001<-function(gamma,t,theta,delta)
{
  1
}
g002<-function(gamma,t,theta,delta)
{
  exp(3*exp(gamma)*(theta-delta))
}
f00<-function(gamma,t,theta,delta)
{
  g001(gamma,t,theta,delta)+g002(gamma,t,theta,delta)
}
g011<-function(gamma,t,theta,delta)
{
  exp(exp(gamma)*(theta-delta+exp(t)))
}
g012<-function(gamma,t,theta,delta)
{
```

```

exp(2*exp(gamma)*(theta-delta)+exp(gamma)*exp(t))
}
f01<-function(gamma,t,theta,delta)
{
  g011(gamma,t,theta,delta)+g012(gamma,t,theta,delta)
}
f0<-function(gamma,t,theta,delta)
{
  f01(gamma,t,theta,delta)/(f00(gamma,t,theta,delta)+f01(gamma,t,theta,delta))
}

```

```

#first derivative of a1, transform of a
dpaf1<-function(a,theta,pi,beta,mua,sigmaa,si)
{
  if(sigmaa==0)
  {(1-pi)*(-g1(a,theta,beta,si)*(-1.7*(theta-beta))
  *exp(a)*si/(1+g1(a,theta,beta,si))^2)
  } else {
  (1-pi)*(-g1(a,theta,beta,si)*(-1.7*(theta-beta))
  *exp(a)*si/(1+g1(a,theta,beta,si))^2-(a-mua)/sigmaa^2)
  }
}

```

```

#first derivative of b
dpbf1<-function(a,theta,pi,beta,mub,sigmab,si)
{
  if (sigmab==0)
  {
  (-g1(a,theta,beta,si)*1.7*exp(a)/(1+g1(a,theta,beta,si))^2)
  *(1-pi)*si
  } else {
  (-g1(a,theta,beta,si)*1.7*exp(a)/(1+g1(a,theta,beta,si))^2
  *si-(beta-mub)/sigmab^2)*(1-pi)
  }
}

```

```

#first derivative of delta
dpdf1<-function(theta,delta,gamma,t,pi,mud,sigmad)
{
  f00<-f00(gamma,t,theta,delta)
  f01<-f01(gamma,t,theta,delta)
  g001<-g001(gamma,t,theta,delta)
  g002<-g002(gamma,t,theta,delta)
  g011<-g011(gamma,t,theta,delta)

```

```

g012<-g012(gamma,t,theta,delta)
df01=-exp(gamma)*g011-2*exp(gamma)*g012
df00=-3*exp(gamma)*g002
df0=(df01*f00-df00*f01)/(f00+f01)^2
if (sigmad==0)
{
pi*df0

} else {

pi*(df0-(delta-mud)/sigmad^2)
}
}
#first derivative of gamma, transform of alpha
dpalf1<-function(theta,delta,gamma,t,pi,mug,sigmag)
{
f00<-f00(gamma,t,theta,delta)
f01<-f01(gamma,t,theta,delta)
g001<-g001(gamma,t,theta,delta)
g002<-g002(gamma,t,theta,delta)
g011<-g011(gamma,t,theta,delta)
g012<-g012(gamma,t,theta,delta)
df00<-3*(theta-delta)*g002*exp(gamma)
df01<-((theta-delta+exp(t))*g011+(2*(theta-delta)+exp(t))*g012)*exp(gamma)
if(sigmag==0)

{
pi*(df01*f00-df00*f01)/(f00+f01)^2
} else {
pi*((df01*f00-df00*f01)/(f00+f01)^2-(gamma-mug)/sigmag^2)
}

}

#first derivative of t, transform of tau
dptf1<-function(theta,delta,gamma,t,pi,mu,sigmat)
{

f00<-f00(gamma,t,theta,delta)
f01<-f01(gamma,t,theta,delta)
g001<-g001(gamma,t,theta,delta)
g002<-g002(gamma,t,theta,delta)
g011<-g011(gamma,t,theta,delta)
g012<-g012(gamma,t,theta,delta)
df01<-exp(gamma)*(g011+g012)*exp(t)

```



```

if(sigmat==0)
{
pi*(df01*f00)/(f00+f01)^2
} else {
pi*((df01*f00)/(f00+f01)^2-(t-mut)/sigmat^2)
}
}

#EM algorithm

source("C:/Na/Code/information for ggum.txt")
library(mokken)
source("C:/Na/Code/irtgen.txt")
source("C:/Na/Code/items.txt")
source("C:/Na/Code/irtsave.txt")
source("C:/Na/Code/check.monotonicity.txt")
source("C:/Na/Code/check.data.txt")

#generate the item parameter
par3pl<-paste("C:/Na/Code/pars/a=",as.character(0.9),".txt",sep="")

for(i in 1:6)
{
temp1<-paste("C:/Na/Code/pars/a=",as.character(0.9+0.1*i),".txt",sep="")
par3pl1<-read.table(temp1,header=T)
par3pl1<-subset(par3pl1,select=c(alpha,delta,c))
par3pl<-rbind(par3pl,par3pl1)
}

for(i in 1:7)
{
temp2<-paste("C:/Na/Code/pars/a=-",as.character(0.8+0.1*i),".txt",sep="")
par3pl2<-read.table(temp2,header=T)
par3pl2<-subset(par3pl2,select=c(alpha,delta,c))
par3pl<-rbind(par3pl,par3pl2)
}

parggum<-paste("C:/Na/Code/pars/GGUMa=",as.character(0.9),".txt",sep="")
parggum<-read.table(parggum,header=T)
parggum<-subset(parggum,select=c(alpha,beta,tau0,tau1))

for(i in 1:6)
{
temp1<-paste("C:/Na/Code/pars/GGUMa=",as.character(0.9+0.1*i),".txt",sep="")

```

```

parggum1<-read.table(temp1,header=T)
parggum1<-subset(parggum1,select=c(alpha,beta,tau0,tau1))
parggum<-rbind(parggum,parggum1)
}

for(i in 1:7)
{
temp2<-paste("C:/Na/Code/pars/GGUMa=-",as.character(0.8+0.1*i),".txt",sep="")
parggum2<-read.table(temp2,header=T)
parggum2<-subset(parggum2,select=c(alpha,beta,tau0,tau1))
parggum<-rbind(parggum,parggum2)
}

itemn<-round(runif(15,1,434))
#itemn<-round(runif(20,1,434))
sampleggum<-parggum[itemn,]
sample3pl<-par3pl[itemn,]
a<-sample3pl[,1]
b<-sample3pl[,2]
cc<-sample3pl[,3]
del<-sampleggum[,2]
alp<-sampleggum[,1]
tau1<-sampleggum[,4]
tau0<-sampleggum[,3]

sampledata<-irtgen(nexmn=1000,avec=a,bvec=b,
cvec=cc,delta=del,alpha=alp,tau0=tau0,tau1=tau1)
write.table(sampledata,"C:/Na/Code/sampledata_20 w 1.7.txt")

score<-sampledata
I<-dim(score)[2]
N<-dim(score)[1]
pih<-c(rep(0,10),rep(1,(I-10)))
sampledata<-score
l<-dim(score)[2]

#initial estimates
#estimateda<-items(sampledata2)[,4]/sqrt(1-items(sampledata2)[,4]^2)
coef<-coefH(score)
sign=rep(1,I)
sampledata2<-score
for(i in 1:l)
{
if(coef$Hij[i,1]<0)
{sampledata2[,i]<-1-sampledata2[,i]
sign[i]<--sign[i]}
}

```

```

}
}
estimateda<-items(sampledata2)[,4]/sqrt(1-items(sampledata2)[,4]^2)*(-sign)/1.7
estimatedb<--qnorm(items(score)[,1])/items(score)[,5]

#estimate for tau,n is the number of examinees and l is the length of items
score<-sampledata
N<-dim(sampledata)[1]
l<-dim(sampledata)[2]
sc<-apply(sampledata,2,sum)
sc<-sc[order(sc)]
for(i in 1:l)
{
if(sc[i]<600) sn=sc[i]
}

estitau<-abs(rep(-log(sn/(N-sn)),l))

#estimate for delta, s is the score for items.
#f is the function solve for delta, given tau=estitau
estidel<-rep(0,l)
for (i in 1:l)
{
if(sc[i]< 0.6*N)
{
sn<-sc[i]
f<-function(del,tau=-estitau[i])
{
(exp(-del-tau)+exp(-2*del-tau))/
(exp(0)+exp(-del-tau)+exp(-2*del-tau)+exp(-3*del))-sn/N
}
estidel[i]<-uniroot(f,c(0,40))$root
} else
{estidel[i]<-5}
}

for(i in 1:l)
{
if (estidel[i]==0) estidel[i]=1
if (estimatedb[i]==-Inf) estimatedb[i]=-100
if (estimatedb[i]==Inf) estimatedb[i]=100
}
estidel<-sign(prcomp(sampledata)$rotation[,1])*estidel
estialp<-rep(1.5,l)

```

```

#initial estimates
s<-rep(1,I)
for(i in 1:I)
{
if (estimateda[i]<0) s[i]==-s[i]
if(estimateda[i]==0) estimateda[i]=0.01

}
al<-log(estimateda*s)
#b parameter in 2pl
bet<-estimatedb
del<-estidel
ta<-log(abs(estitau))
s=-s
#alpha in ggum
gamm<-log(estialp)

cbind(sample3pl,exp(al[1:10])*s[1:10],bet[1:10])
cbind(sampleggum,exp(ta[11:20]),exp(al[11:20]),del[11:20])

source("C:/Na/Code/fuct of in de m and u.txt")
write.table(cbind(exp(al)*s,estimatedb),"C:/Na/Code/data/akx/initial3pl.txt")
write.table(cbind(exp(gamm),del,exp(ta)),"C:/Na/Code/data/akx/initialggum.txt")
write.table(sample3pl,"C:/Na/Code/data/akx/3pl.txt")
write.table(sampleggum,"C:/Na/Code/data/akx/ggum.txt")
xk<-c(-4,-3.1111,-2.2222,-1.3333,-.4444,.4444,1.3333,2.2222,3.1111,4)
akx<-c(0.000119,0.002805,0.03002,0.1458,0.3213,
0.3213,0.1458,0.03002,0.002805,0.000119)
F<-length(xk)

parn<-matrix(0,6,I)
iterto=1
difto=1
T=1700
difto=1
gammt<-matrix(0,T,I)
delt<-matrix(0,T,I)
tt<-matrix(0,T,I)
pihh<-matrix(0,T,I)
gammt[1,]<-gamm
delt[1,]<-del
tt[1,]<-ta
pihh[1,]<-pih
alt<-matrix(0,T,I)
bett<-matrix(0,T,I)
alt[1,]<-al

```



```

for(f in 1:F)
{
for(j in 1:N)
{
nt<-pk1[j,f]*akx[f]/pl[j]
rt=nt*score[j,m]
rik[m,f]=rik[m,f]+rt
nik[m,f]=nik[m,f]+nt
}
}
}

source("C:/Na/Code/fit in in & de monotone.txt")
source("C:/Na/Code/fit in unfolding.txt")
source("C:/Na/Code/likelihood function.txt")

difto<-max(abs(ea-alt[iterto,]),abs(ebeta-bett[iterto,]),
abs(gammt[iterto,]-egamma),abs(tt[iterto,]-et),abs(delt[iterto,]-edelta))

iterto=iterto+1
pihh[iterto,]<-epi
alt[iterto,]<-ea
bett[iterto,]<-ebeta
gammt[iterto,]<-egamma
tt[iterto,]<-et
delt[iterto,]<-edelta

write.table(pihh,paste("C:/Na/Code/data/akx/pih",iterto,".txt",sep=""))
write.table(alt,paste("C:/Na/Code/data/akx/alt",iterto,".txt",sep=""))
write.table(bett,paste("C:/Na/Code/data/akx/bett",iterto,".txt",sep=""))
write.table(gammt,paste("C:/Na/Code/data/akx/gammt",iterto,".txt",sep=""))
write.table(tt,paste("C:/Na/Code/data/akx/tt",iterto,".txt",sep=""))
write.table(delt,paste("C:/Na/Code/data/akx/delt",iterto,".txt",sep=""))
write.table(cbind(exp(ea)*s,ebeta,round(epi,4)),
paste("C:/Na/Code/data/akx/3plsim",iterto,".txt",sep=""))
write.table(cbind(exp(egamma),edelta,round(epi,4)),
paste("C:/Na/Code/data/akx/ggumsim",iterto,".txt",sep=""))
}

#function in C:/Na/Code/fit in in & de monotone.txt,
#get the estimation from 2pl model

ea<-rep(0,I)
ebeta<-rep(0,I)
dif<-rep(1,I)
for(i in 1:I)

```

```

{
difab<-rep(0,2)
a<-alt[iterto,i]
beta<-bett[iterto,i]
itera<-1
si<-s[i]
while(dif[i]>0.001 & itera<T &dif[i]!="NaN")
{
infoa<-matrix(0,2,2)
infoia<-matrix(0,2,2)
da<-NULL
db<-NULL
saa<-NULL
sab<-NULL
sbb<-NULL
for(f in 1:F)
{
theta<-xk[f]
p1<-f1(a,theta,beta,si)
p0=1-p1
if (p0*p1<0.0000009)
{
da[f]=0
db[f]=0
saa[f]=0
sab[f]=0
sbb[f]=0
} else {
da[f]<-(rik[i,f]-nik[i,f]*p1)/(p1*(1-p1))*dpaf1(a,theta,pi=0,beta,mua,sigmaa=0,si)
db[f]<-(rik[i,f]-nik[i,f]*p1)/(p1*(1-p1))*dpbf1(a,theta,pi=0,beta,mub,sigmab=0,si)
saa[f]<-nik[i,f]*(1/p1+1/(1-p1))*dpaf1(a,theta,pi=0,beta,mua,sigmaa=0,si)
*dpaf1(a,theta,pi=0,beta,mua,sigmaa=0,si)
sab[f]<-nik[i,f]*(1/p1+1/(1-p1))*dpaf1(a,theta,pi=0,beta,mua,sigmaa=0,si)
*dpbf1(a,theta,pi=0,beta,mub,sigmab=0,si)
sbb[f]<-nik[i,f]*(1/p1+1/(1-p1))*dpbf1(a,theta,pi=0,beta,mub,sigmab=0,si)
*dpbf1(a,theta,pi=0,beta,mub,sigmab=0,si)
}
}
}
dla<-sum(da)-(a-mua)/sigmaa^2
dlb<-sum(db)-(beta-mub)/sigmab^2
infoa[1,1]<-sum(saa)+1/sigmaa^2
infoa[1,2]<-sum(sab)
infoa[2,1]<-infoa[1,2]
infoa[2,2]<-sum(sbb)+1/sigmab^2
d<-svd(infoa)$d
u<-svd(infoa)$u

```

```

v<-svd(infoa)$v
  dd=NULL
  for(h in 1:2)
  { if(round(d[h],5)==0) {
    dd[h]=0

    } else {dd[h]=1/d[h]}
  }
  infoia<-v%%diag(dd)%%t(u)
  difab<-infoia%%c(dla,dlb)
  addit=difab[1]
  if (addit=="NaN")
  {a=a
  } else if (addit>0.25)
  {a=a+0.25
  } else if (addit<(-0.25))
  { a=a-0.25
  } else
  {a=a+addit}

  bddit=difab[2]
  if(bddit>1)
  {beta=beta+1
  } else if (bddit<(-1))
  { beta=beta-1
  } else
  {beta=beta+bddit}
  #if(beta>5) beta=5
  #if(beta<-5) beta=-5
  dif[i]=max(abs(bddit),abs(addit))
  itera=itera+1

}
ea[i]<-a
ebeta[i]<-beta
}

#function in C:/Na/Code/fit in unfolding.txt, get estimation from ggum

edelta<-rep(0,I)
egamma<-rep(0,I)
et<-rep(0,I)
dift<-rep(1,I)
difd<-rep(1,I)
for(i in 1:I)
{

```



```

difal<-rep(0,2)
dlft<-0
delta<-delt[iterto,i]
gamma<-gammt[iterto,i]
t<-tt[iterto,i]
itert<-1

while(dift[i]>0.001 & itert<T)
{
#estimate tau first
dt<-NULL
stt<-NULL
for(f in 1:F)
{
theta<-xk[f]
p1<-f0(gamma,t,theta,delta)
p0=1-p1
if (p0*p1<0.0000009)
{

dt[f]=0
stt[f]=0
} else {
dt[f]<-(rik[i,f]-nik[i,f]*p1)/(p1*(1-p1))
*dptf1(theta,delta,gamma,t,pi=1,mut,sigmat=0)
stt[f]<-nik[i,f]*(1/p1+1/(1-p1))*dptf1(theta,delta,gamma,t,pi=1,mut,sigmat=0)
*dptf1(theta,delta,gamma,t,pi=1,mut,sigmat=0)

}
}
dlt<-sum(dt)-(t-mut)/sigmat^2
infoit<-solve(sum(stt+1/sigmat^2))
tddit<-infoit*dlt

if(tddit>1)
{t=t+1
} else if (tddit<(-1))
{ t=t-1
} else
{t=t+tddit}
dift[i]=abs(tddit)
itert=itert+1
}
#estimate gamma and beta
iteral=0

```

```

while(difd[i]>0.001 & iteral<T)
{

infoal<-matrix(0,2,2)
infoial<-matrix(0,2,2)
dd<-NULL
dal<-NULL
sdd<-NULL
sdal<-NULL
salal<-NULL
for(f in 1:F)
{
  theta<-xk[f]
  p1<-f0(gamma,t,theta,delta)
  p0=1-p1
  if (p0*p1<0.0000009)
  {
dd[f]=0
dal[f]=0

sdd[f]=0
sdal[f]=0
salal[f]=0

} else {
dd[f]<-(rik[i,f]-nik[i,f]*p1)/(p1*(1-p1))
*dpdf1(theta,delta,gamma,t,pi=1,mud,sigmad=0)
dal[f]<-(rik[i,f]-nik[i,f]*p1)/(p1*(1-p1))
*dpalf1(theta,delta,gamma,t,pi=1,mug,sigmag=0)
sdd[f]<-nik[i,f]*(1/p1+1/(1-p1))*dpdf1(theta,delta,gamma,t,pi=1,mud,sigmad=0)
*dpdf1(theta,delta,gamma,t,pi=1,mud,sigmad=0)
sdal[f]<-nik[i,f]*(1/p1+1/(1-p1))*dpalf1(theta,delta,gamma,t,pi=1,mug,sigmag=0)
*dpalf1(theta,delta,gamma,t,pi=1,mug,sigmag=0)
salal[f]<-nik[i,f]*(1/p1+1/(1-p1))*dpalf1(theta,delta,gamma,t,pi=1,mug,sigmag=0)
*dpalf1(theta,delta,gamma,t,pi=1,mug,sigmag=0)
}
}

dld<-sum(dd)-(delta-mud)/sigmad^2
dlal<-sum(dal)-(gamma-mug)/sigmag^2
infoal[1,1]<-sum(sdd)+1/sigmad^2
infoal[1,2]<-sum(sdal)
infoal[2,1]<-infoal[1,2]
infoal[2,2]<-sum(salal)+1/sigmag^2

d<-svd(infoal)$d
u<-svd(infoal)$u

```

```

v<-svd(infoal)$v
  dd=NULL
  for(h in 1:2)
  { if(round(d[h],5)==0) {
    dd[h]=0
    parc1[h]=0
  } else {dd[h]=1/d[h]}
  }
  infoial<-v%*%diag(dd)%*%t(u)
  difal<-infoial%*%c(dld,dlal)

  dddit=difal[1]
  if(dddit>1)
  {delta=delta+1
  } else if (dddit<(-1))
  { delta=delta-1
  } else
  {delta=delta+dddit}

#if (delta>6) delta=6
alddit=difal[2]
if(alddit>0.25)
  {gamma=gamma+0.25
  } else if (alddit<(-0.25))
  { gamma=gamma-0.25
  } else
  {gamma=gamma+alddit}
difd[i]=max(abs(dddit),abs(alddit))
iteral=iteral+1
}
et[i]<-t
egamma[i]<-gamma
edelta[i]<-delta
}

#function in C:/Na/Code/likelihood function.txt, used to calculated pi

#check the sign of the edelta and ea, ebeta
epi<-NULL
for(i in 1:I)
{

pif=rep(0,N)
pipf=rep(0,N)
#pif unfolding
#pipf monotone

```

```

pi<-NULL
pip<-NULL
for(j in 1:N)
{
  for(f in 1:F)
  {
for(m in 1:I)
  {
    theta<-xk[f]
    sc<-score[j,m]
    a<-ea[m]
    si<-s[m]
    beta<-ebeta[m]
    delta<-edelta[m]
    gamma<-egamma[m]
    t<-et[m]
    if(m==i)
    {
      pip[m]<-f1(a,theta,beta,si)^sc*
      (1-f1(a,theta,beta,si))^(1-sc)
      pi[m]<-f0(gamma,t,theta,delta)^sc
      *(1-f0(gamma,t,theta,delta))^(1-sc)
    } else
    {
      epih<-f1(a,theta,beta,si)*(1-pihh[itero,m])
      +f0(gamma,t,theta,delta)*pihh[itero,m]
      pi[m]<-epih^sc*(1-epih)^(1-sc)
      pip[m]<-pi[m]
    }
  }
  pif[j]<-(pif[j]+prod(pi)*akx[f])
  pipf[j]<-(pipf[j]+prod(pip)*akx[f])
}
if(pif[j]==0) pif[j]=0.0000000000000000000000000000001
if(pipf[j]==0) pipf[j]=0.0000000000000000000000000000001
}

con<-min(sum(log(pif)),sum(log(pipf)))
epi[i]<-exp(sum(log(pif))-con)/(exp(sum(log(pif))-con)+exp(sum(log(pipf))-con))
if(is.nan(epi[i])) epi[i]=1
}

# Code in C:/Na/Code/information for ggum.txt
inform<-function(theta,alpha,z=1,delta,tau0=0,tau1)

```

```

{
c=1
m=2*c+1
an<-e(alpha,z=1,theta,delta,tau0,tau1)
bn<-f(alpha,z=1,theta,delta,tau0,tau1)
gn<-g(alpha,theta,delta,tau0,tau1)
ad<-an*alpha*z
bd<-bn*(m-z)*alpha
gd<-f(alpha,z=0,theta,delta,tau0,tau1)*alpha*m+an*alpha+bn*alpha*2
pd<-((ad+bd)*gn-gd*(an+bn))/gn^2
p<-(an+bn)/gn
info<-pd^2/(p*(1-p))
info
}

#tau is a vector, and symmetric about 0. we have 3 tau's and tau[2]=0.
#that means we can only give 1 negative tau which will be
#used in the model.
e<-function(alpha=1,z=1,theta=0,delta=1,tau0=0,tau1)
{
if(z==1)
{
exp(alpha*(z*(theta-delta)-(tau0+tau1)))
}
else
{
exp(-alpha*tau0)
}
}

f<-function(alpha=1,z=1,theta=0,delta=1,tau0=0,tau1)
{
if(z==1)
{
exp(alpha*((3-z)*(theta-delta)-(tau0+tau1)))
}
else
{
exp(alpha*(3*(theta-delta)-tau0))
}
}

#c start from 0

```

```

g<-function(alpha,theta,delta,tau0=0,tau1)
{
prob0a<-exp(-alpha*tau0)
prob0b<-exp(alpha*(3*(theta-delta)-tau0))
prob1a<-exp(alpha*((theta-delta)-(tau0+tau1)))
prob1b<-exp(alpha*(2*(theta-delta)-(tau0+tau1)))
prob0a+prob0b+prob1a+prob1b
}

```

```

#Code for IRFs comparison of item been unfolding and monotone
#Used for setting 1-5 on Chapter 5
par(mfrow=c(5,4))
for(i in 1:20)
{
theta<-seq(-4,4,0.1)
a<-ea[i]
si<-s[i]
beta<-ebeta[i]
delta<-edelta[i]
gamma<-egamma[i]
t<-et[i]

p1<-f1(a,theta,beta,si)
p2<-f0(gamma,t,theta,delta)
plot(theta,p1,type='l',col='blue',ylim=c(0,1))
par(new=T)
plot(theta,p2,type='l',col='red',ylim=c(0,1))
title(paste("Item",i,sep=""))
}

```

```

par(mfrow=c(2,2))

i=4
theta<-seq(-4,4,0.1)
a<-ea[i]
si<-s[i]
beta<-ebeta[i]
delta<-edelta[i]
gamma<-egamma[i]
t<-et[i]

p1<-f1(a,theta,beta,si)
p2<-f0(gamma,t,theta,delta)
plot(theta,p1,type='l',col='blue',ylim=c(0,1))

```

```

par(new=T)
plot(theta,p2,type='l',col='red',ylim=c(0,1))
title(paste("Item",i,sep=""))

```

```

i=7
a<-ea[i]
si<-s[i]
beta<-ebeta[i]
delta<-edelta[i]
gamma<-egamma[i]
t<-et[i]

```

```

p1<-f1(a,theta,beta,si)
p2<-f0(gamma,t,theta,delta)
plot(theta,p1,type='l',col='blue',ylim=c(0,1))
par(new=T)
plot(theta,p2,type='l',col='red',ylim=c(0,1))
title(paste("Item",i,sep=""))

```

```

i=16
a<-ea[i]
si<-s[i]
beta<-ebeta[i]
delta<-edelta[i]
gamma<-egamma[i]
t<-et[i]

```

```

p1<-f1(a,theta,beta,si)
p2<-f0(gamma,t,theta,delta)
plot(theta,p1,type='l',col='blue',ylim=c(0,1))
par(new=T)
plot(theta,p2,type='l',col='red',ylim=c(0,1))
title(paste("Item",i,sep=""))

```

```

i=18
a<-ea[i]
si<-s[i]
beta<-ebeta[i]
delta<-edelta[i]
gamma<-egamma[i]
t<-et[i]

```

```

p1<-f1(a,theta,beta,si)
p2<-f0(gamma,t,theta,delta)
plot(theta,p1,type='l',col='blue',ylim=c(0,1))

```

```
par(new=T)
plot(theta,p2,type='l',col='red',ylim=c(0,1))
title(paste("Item",i,sep=""))
```


APPENDIX F: R CODE FOR THE MEAN/MEAN LINKING

```
#Mean/mean for monotone items (Setting $1$ in Chapter $5$)
linkMMm<-function(sa,sb,ea,eb){
#Performs Mean/Mean equating after applying transform to monotone items
#Reverses transform for output
#Requires the library plink to run.
#ea and eb are the vectors of estimated parameter values
#sa and sb are the vectors of true paramter values as simulatd
#ealpha, edelta, etau1 are the vectors of estimated paramger values
#salpha, sdelta, stau1 are the vectors of true parameter values as simulated
oea<-ea
oeb<-eb
rev<-sign(cor(sb,eb))
signs<-sign(sa)
ea<-ea*rev
eb<-eb*rev
dich1<-cbind(abs(sa),sb)
cat<-rep(2,length(sa))
pm<-as.poly.mod(length(sa),c("drm"),list(1:length(sa)))
x1<-as.irt.pars(dich1,cat=cat,poly.mod=pm,grp.names=c("sims"))

dich2<-cbind(abs(ea),eb)
cat<-rep(2,length(ea))
pm<-as.poly.mod(length(ea),c("drm"),list(1:length(ea)))
x2<-as.irt.pars(dich2,cat=cat,poly.mod=pm,grp.names=c("ests"))
xc<-combine.pars(list(x1,x2),
common=cbind(1:length(sa),1:length(ea)),grp.names=c("sims","ests"))
A<-attributes(plink(xc))$constants$MM[1]
B<-attributes(plink(xc))$constants$MM[2]
ra<-signs*abs(ea)/A
rb<-A*eb+B

#A and B are the linking coefficients
#sa and sb are the simulated true values
#ra and rb are the linked estimated values
#ea and eb are the unlinked estimated values
return(list(A=A,B=B,ParsM=cbind(sa,sb,ra,rb,oea,oeb)))}
linkMMm(sa,sb,ea,eb)
```

```

#Mean/mean for unfolding items (Setting $2$ in Chapter $5$)
library(lattice)
library(plink)
linkMM<-function(ea,eb,salpha,sdelta,stau1,ealpha,edelta,etau1){
oealpha<-ealpha
oedelta<-edelta
otau1<-etau1
rev<-sign(cor(sdelta,edelta))
#Don't need to reverse alpha or tau1 for the unfolding model
#as the model is symmetric
edelta<-edelta*rev
dich1<-cbind(salpha,sdelta)
cat<-rep(2,length(salpha))
pm<-as.poly.mod(length(salpha),c("drm"),list(1:length(salpha)))
x1<-as.irt.pars(dich1,cat=cat,poly.mod=pm,grp.names=c("sims"))
dich2<-cbind(c(ealpha,edelta))
cat<-rep(2,length(ealpha))
pm<-as.poly.mod(length(ealpha),c("drm"),list(1:length(ealpha)))
x2<-as.irt.pars(dich2,cat=cat,poly.mod=pm,grp.names=c("ests"))
xc<-combine.pars(list(x1,x2),common=cbind(1:length(salpha),
1:length(ealpha)),grp.names=c("sims","ests"))
A<-attributes(plink(xc))$constants$MM[1]
B<-attributes(plink(xc))$constants$MM[2]
ralpha<-ealpha/A
rdelta<-A*edelta+B
rtau1<-A*etau1
return(list(A=A,B=B,ParsU=cbind(salpha,sdelta,ralpha,
rdelta,rtau1,oealpha,oedelta,otau1)))}

vals<-read.table("C:/Users/Owner/Documents/na/setting2.txt",head=T)
ealpha<-vals$ealpha
edelta<-vals$edelta
salpha<-vals$salpha
sdelta<-vals$sdelta
ea<-vals$ea
eb<-vals$eb
tau1<-rep(-0.2,20)
etau1<-vals$etau1
linkMM(ea,eb,alpha,delta,tau1,ealpha,edelta,etau1)

#Mean/mean for unfolding items (Setting $3$ in Chapter $5$)
library(lattice)
library(plink)

```

```

linkMMm<-function(sa,sb,ea,eb,salpha,sdelta,tau1,ealpha,edelta,etau1){
  oea<-ea
  oeb<-eb
  oealpha<-ealpha
  oedelta<-edelta
  otau1<-etau1
  rev<-sign(cor(sb,eb))
  signs<-sign(sa)
  ea<-ea*rev
  eb<-eb*rev
  dich1<-cbind(abs(sa),sb)
  cat<-rep(2,length(sa))
  pm<-as.poly.mod(length(sa),c("drm"),list(1:length(sa)))
  x1<-as.irt.pars(dich1,cat=cat,poly.mod=pm,
  grp.names=c("sims"))
  dich2<-cbind(abs(ea),eb)
  cat<-rep(2,length(ea))
  pm<-as.poly.mod(length(ea),c("drm"),list(1:length(ea)))
  x2<-as.irt.pars(dich2,cat=cat,poly.mod=pm,
  grp.names=c("ests"))
  xc<-combine.pars(list(x1,x2),common=cbind(1:length(sa),
  1:length(ea)),grp.names=c("sims","ests"))
  A<-attributes(plink(xc))$constants$MM[1]
  B<-attributes(plink(xc))$constants$MM[2]
  ra<-signs*abs(ea)/A
  rb<-A*eb+B
  ralpha<-ealpha/A
  rdelta<-A*edelta+B
  rtau1<-A*etau1
  return(list(A=A,B=B,ParsM=cbind(sa,sb,ra,rb,oea,oeb),
  ParsU=cbind(salpha,sdelta,ralpha,rdelta,rtau1,oealpha,oedelta,otau1)))}
vals<-read.table("C:/Users/Owner/Documents/na/setting3.txt",head=T)
  sa<-vals$sa[1:18]
  sb<-vals$sb[1:18]
  ealpha<-vals$ealpha[19:20]
  edelta<-vals$edelta[19:20]
  salpha<-vals$salpha[19:20]
  sdelta<-vals$sdelta[19:20]
  ea<-vals$ea[1:18]
  eb<-vals$eb[1:18]
  tau1<-rep(-0.2,2)
  etau1<-vals$etau1[19:20]
linkMMm(sa,sb,ea,eb,salpha,sdelta,tau1,ealpha,edelta,etau1)

#Mean/mean for unfolding items (Setting $4$ in Chapter $5$)
library(lattice)

```

```

library(plink)
linkMMm<-function(sa,sb,ea,eb,salpha,sdelta,tau1,ealpha,edelta,etau1){
oea<-ea
oeb<-eb
oealpha<-ealpha
oedelta<-edelta
otau1<-etau1
rev<-sign(cor(sb,eb))
signs<-sign(sa)
ea<-ea*rev
eb<-eb*rev
dich1<-cbind(salpha,sdelta)
cat<-rep(2,length(salpha))
pm<-as.poly.mod(length(salpha),c("drm"),list(1:length(salpha)))
x1<-as.irt.pars(dich1,cat=cat,poly.mod=pm,grp.names=c("sims"))
dich2<-cbind(abs(ealpha),edelta)
cat<-rep(2,length(ealpha))
pm<-as.poly.mod(length(ealpha),c("drm"),list(1:length(ealpha)))
x2<-as.irt.pars(dich2,cat=cat,poly.mod=pm,grp.names=c("ests"))
xc<-combine.pars(list(x1,x2),
common=cbind(1:length(salpha),1:length(ealpha)),
grp.names=c("sims","ests"))
A<-attributes(plink(xc))$constants$MM[1]
B<-attributes(plink(xc))$constants$MM[2]
ra<-signs*abs(ea)/A
rb<-A*eb+B
ralpha<-ealpha/A
rdelta<-A*edelta+B
rtau1<-A*etau1
return(list(A=A,B=B,ParsM=cbind(sa,sb,ra,rb,oea,oeb),
ParsU=cbind(salpha,sdelta,ralpha,rdelta,rtau1,oealpha,oedelta,otau1)))}
vals<-read.table("C:/Users/Owner/Documents/na/setting4.txt",head=T)
sa<-vals$sa[1:2]
sb<-vals$sb[1:2]
ealpha<-vals$ealpha[3:20]
edelta<-vals$edelta[3:20]
salpha<-vals$salpha[3:20]
sdelta<-vals$sdelta[3:20]
ea<-vals$ea[1:2]
eb<-vals$eb[1:2]
tau1<-rep(-0.2,18)
etau1<-vals$etau1[3:20]
linkMMm(sa,sb,ea,eb,salpha,sdelta,tau1,ealpha,edelta,etau1)

#Mean/mean for mixed items with half monotone and half unfolding
#(Setting $5$ in Chapter $5$ and $40$ item simulation in Table $5.1$ and $5.2$)

```

```

library(lattice)
library(plink)
linkMM<-function(sa,sb,ea,eb,salpha,sdelta,stau1,ealpha,edelta,etau1){
oea<-ea
oeb<-eb
oealpha<-ealpha
oedelta<-edelta
otau1<-etau1
rev<-sign(cor(sb,eb))
signs<-sign(sa)
#Don't need to reverse alpha or tau1 for the
#unfolding model as the model is symmetric
ea<-ea*rev
eb<-eb*rev
edelta<-edelta*rev
dich1<-cbind(c(abs(sa),salpha),c(sb,sdelta))
cat<-rep(2,length(c(sa,salpha)))
pm<-as.poly.mod(length(c(sa,salpha)),c("drm"),list(1:length(c(sa,salpha))))
x1<-as.irt.pars(dich1,cat=cat,poly.mod=pm,grp.names=c("sims"))
dich2<-cbind(c(abs(ea),ealpha),c(eb,edelta))
cat<-rep(2,length(c(ea,ealpha)))
pm<-as.poly.mod(length(c(ea,ealpha)),c("drm"),
list(1:length(c(ea,ealpha))))
x2<-as.irt.pars(dich2,cat=cat,poly.mod=pm,grp.names=c("ests"))
xc<-combine.pars(list(x1,x2),
common=cbind(1:length(c(sa,salpha)),1:length(c(ea,ealpha))),
grp.names=c("sims","ests"))
A<-attributes(plink(xc))$constants$MM[1]
B<-attributes(plink(xc))$constants$MM[2]
ra<-signs*abs(ea)/A
rb<-A*eb+B
ralpha<-ealpha/A
rdelta<-A*edelta+B
rtau1<-A*etau1
return(list(A=A,B=B,ParsM=cbind(sa,sb,ra,rb,oea,oeb),
ParsU=cbind(salpha,sdelta,stau1,ralpha,rdelta,rtau1,oealpha,oedelta,otau1)))}
linkMM(sa,sb,ea/1.7,eb,alpha,delta,tau1,ealpha,edelta,etau1)

```

APPENDIX G: ERROR RATE REPORT AND CLASSIFICATION TREE FOR SECTION 4.3

TABLE 6.1. Error Rate with N=500 for the Original Data

	CP	nsplit	rel error	xerror	xstd
1	0.83654570	0	1.00000000	1.00604839	0.002592332
2	0.02397177	1	0.16345430	0.16354839	0.001420730
3	0.02091398	3	0.11551075	0.11620968	0.001212931
4	0.01701613	4	0.09459677	0.09525538	0.001104235
5	0.01000000	5	0.07758065	0.07840054	0.001006212

TABLE 6.2. Error Rate with N=1000 for the Original Data

	CP	nsplit	rel error	xerror	xstd
1	0.86409946	0	1.00000000	1.00524194	0.0025923436
2	0.02580645	1	0.13590054	0.13595430	0.0013050384
3	0.01614247	3	0.08428763	0.08485215	0.0010450357
4	0.01000000	5	0.05200269	0.05244624	0.0008285151

TABLE 6.3. Error Rate with N=2000 for the Original Data

	CP	nsplit	rel error	xerror	xstd
1	0.88231183	0	1.00000000	1.00489247	0.0025923482
2	0.03201613	1	0.11768817	0.11770161	0.0012202083
3	0.02094086	2	0.08567204	0.08591398	0.0010512625
4	0.01406586	3	0.06473118	0.06502688	0.0009195651
5	0.01000000	5	0.03659946	0.03690860	0.0006978026

TABLE 6.4. Error Rate with N=4000 for the Original Data

	CP	nsplit	rel error	xerror	xstd
1	0.88884409	0	1.00000000	1.00849462	0.0025922857
2	0.03506720	1	0.11115591	0.11119624	0.0011880562
3	0.02100806	2	0.07608871	0.07631720	0.0009932910
4	0.01231183	3	0.05508065	0.05510753	0.0008486953
5	0.01000000	5	0.03045699	0.03069892	0.0006374060

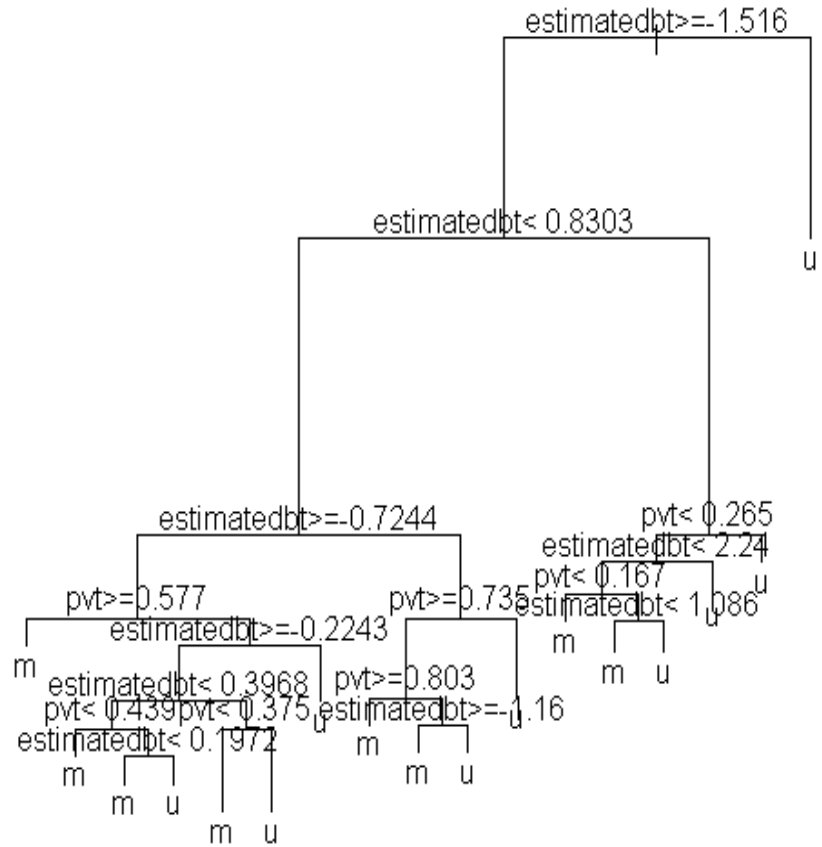


FIGURE 6.1. Classification Tree for $N = 500$ with the Transformed Data

TABLE 6.5. Error Rate with N=500 for the Transformed Data

	CP	nsplit	rel error	xerror	xstd
1	0.20723790	0	1.0000000	1.0091398	0.002592271
2	0.07010753	2	0.5855242	0.5857392	0.002359475
3	0.06575269	3	0.5154167	0.5170565	0.002270021
4	0.02327285	4	0.4496640	0.4505376	0.002165981
5	0.01367608	8	0.3334409	0.3374866	0.001941822
6	0.01176747	12	0.2712634	0.2678629	0.001765818
7	0.01158602	14	0.2477285	0.2446237	0.001698763
8	0.01000000	16	0.2245565	0.2336290	0.001665340

TABLE 6.6. Error Rate with N=1000 for the Transformed Data

	CP	nsplit	rel error	xerror	xstd
1	0.24645833	0	1.0000000	1.0052957	0.002592343
2	0.04604839	2	0.5070833	0.5091398	0.002258580
3	0.02842070	3	0.4610349	0.4653898	0.002190816
4	0.01957325	5	0.4041935	0.3869220	0.002048038
5	0.01746640	9	0.3259005	0.3230376	0.001908037
6	0.01676075	13	0.2351210	0.2803226	0.001799911
7	0.01000000	14	0.2183602	0.2281855	0.001648358

TABLE 6.7. Error Rate with N=2000 for the Transformed Data

	CP	nsplit	rel error	xerror	xstd
1	0.25169355	0	1.0000000	1.0048118	0.002592349
2	0.24694892	1	0.7483065	0.6514919	0.002429852
3	0.04557796	2	0.5013575	0.5017070	0.002247615
4	0.02544355	3	0.4557796	0.4554973	0.002174382
5	0.01897849	7	0.3341129	0.3418145	0.001951688
6	0.01851478	8	0.3151344	0.3208737	0.001902862
7	0.01676747	10	0.2781048	0.2871237	0.001818009
8	0.01102151	12	0.2445699	0.2363844	0.001673824
9	0.01000000	14	0.2225269	0.2188575	0.001618559

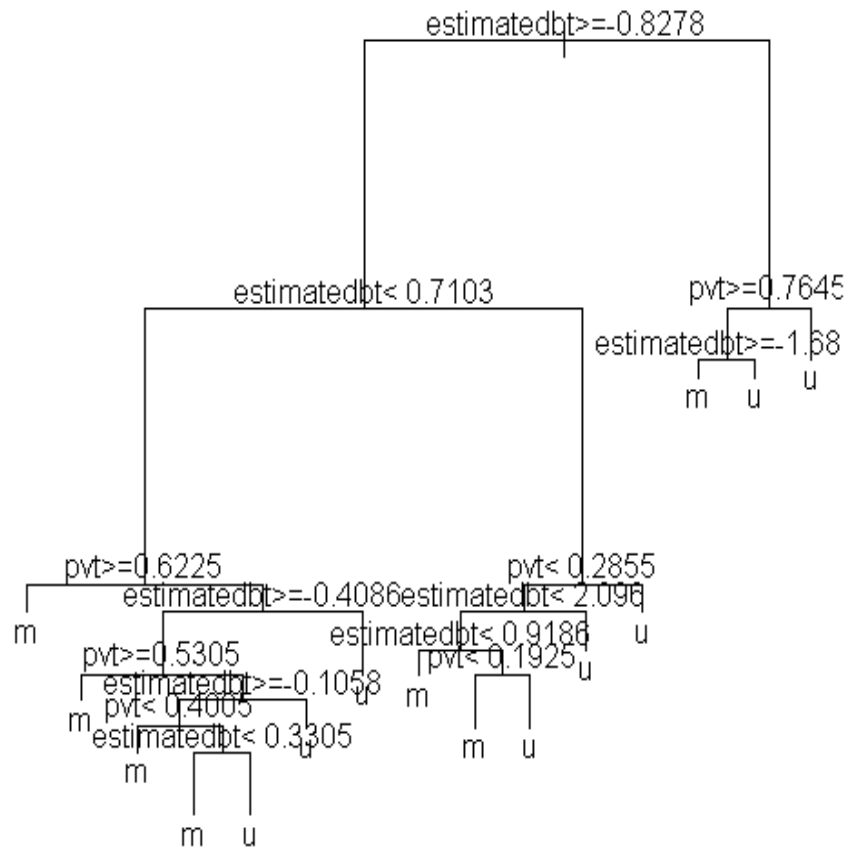


FIGURE 6.2. Classification Tree for $N = 1000$ with the Transformed Data

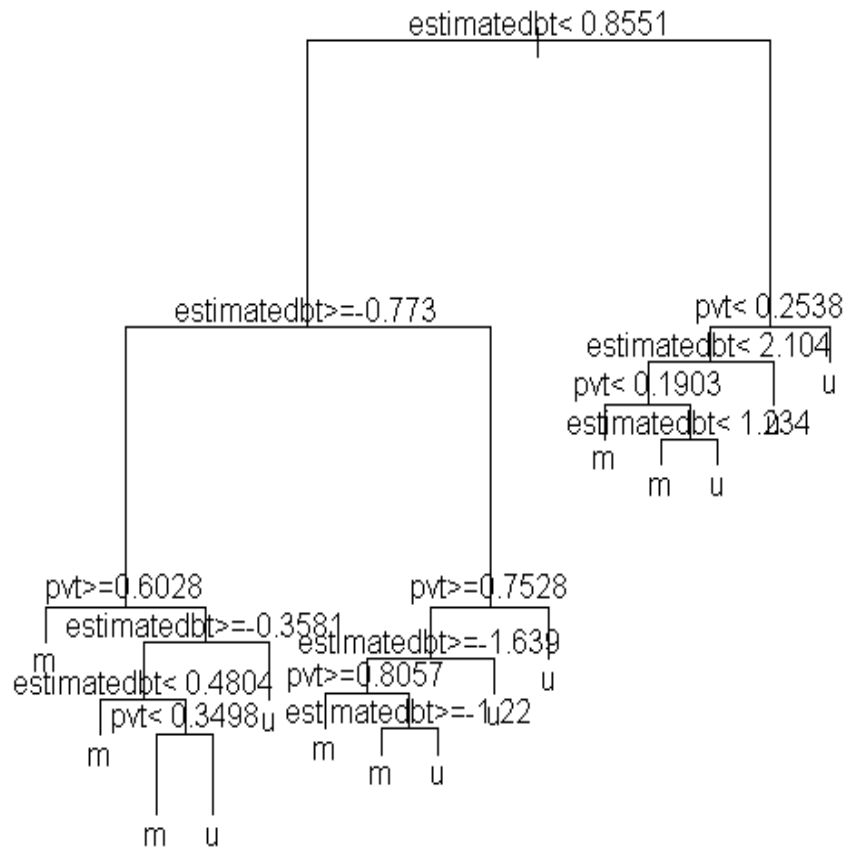


FIGURE 6.3. Classification Tree for $N = 2000$ with the Transformed Data

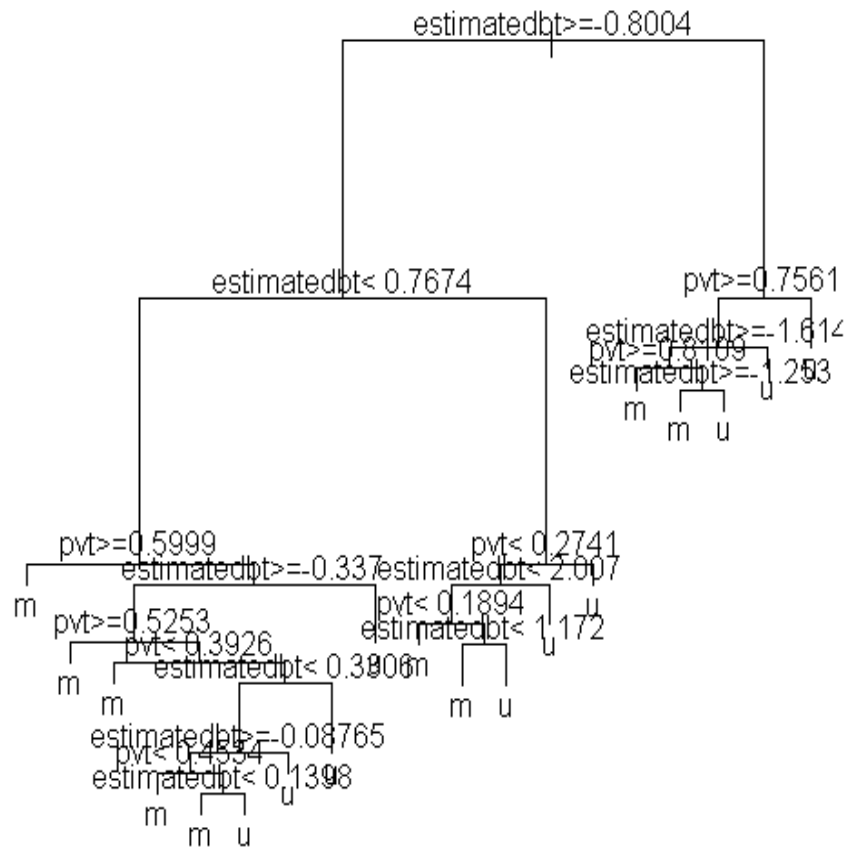


FIGURE 6.4. Classification Tree for $N = 4000$ with the Transformed Data

TABLE 6.8. Error Rate with N=4000 for the Transformed Data

	CP	nsplit	rel error	xerror	xstd
1	0.25121640	0	1.0000000	1.0071237	0.002592313
2	0.04751344	2	0.4975672	0.4982661	0.002242465
3	0.02742608	3	0.4500538	0.4507124	0.002166279
4	0.02209229	5	0.3952016	0.3891129	0.002052433
5	0.01879032	8	0.3289247	0.3258871	0.001914805
6	0.01859543	9	0.3101344	0.2863978	0.001816094
7	0.01688172	13	0.2206586	0.2254973	0.001639863
8	0.01017473	14	0.2037769	0.2062500	0.001576799
9	0.01013441	16	0.1834274	0.1916263	0.001526055
10	0.01000000	18	0.1631586	0.1732796	0.001458506