Development for New Item Bank Calibration Strategies for Computerized Adaptive Testing in Small-Scale Assessments

by

Yawei Shen

(Under the Direction of Shiyu Wang)

Abstract

Computerized adaptive testing (CAT) is a popular mode to deliver educational assessments, which gradually tailors the test for each individual examinee. Despite its great potential and benefits, CAT has limited applications in small-scale assessments. One main reason is that CAT is built on the item response theory model, which requires large sample sizes to calibrate its item parameters. This requirement is difficult to meet due to typically small classroom sizes. Furthermore, because of practical concerns such as fatigue effects of answering too many items, complex missing data would inevitably appear, which increases the difficulty to obtain accurate item parameter estimations. The overall goal of this dissertation is to develop item bank calibration strategies using small pretesting sample sizes to address these challenges. Specifically, this dissertation has three interdependent studies which proposed, refined, and evaluated the item bank calibration strategies based on incomplete calibration designs and matrix completion algorithms. Incomplete calibration designs purposely administer a subset of items to examinees to avoid fatigue effects. Matrix completion is a family of recently developed algorithms to recover a matrix with missing values and has shown potential with high missing percentages. In Study 1, 1-bit matrix completion and structured matrix completion algorithms were adapted to address the complex missing issues from four types of incomplete calibration designs with different missing mechanisms and different ways of assembling test forms. They were also adapted for binary responses in the educational assessments. Through a simulation study, a few adaptations had satisfying item difficulty parameters but overestimated item discrimination parameters. In Study 2, new item bank calibration strategies were developed based on the findings from the simulation study 1 to refine the adaptations and mitigate the overestimation for item discrimination parameters. In Study 3, the proposed strategies were evaluated in a CAT simulation and implemented in an empirical data set. The estimated abilities from the CAT using the item bank calibrated using the proposed strategies were close to those using true item parameters. In addition, the results of the empirical data implementation were comparable to those of the previous simulation studies.

Index words: [Computerized Adaptive Testing, Item Bank Calibration, Item Response Theory, Small-Assessment, Matrix Completion, Incomplete Calibration Designs]

Development for New Item Bank Calibration Strategies for Computerized Adaptive Testing in Small-Scale Assessments

by

Yawei Shen

B.S., Tianjin Universioty of Finance and Economics, China, 2014
M.A., University of Georgia, 2016
M.S., University of Georgia, 2021

A Dissertation Submitted to the Graduate Faculty of the
University of Georgia in Partial Fulfillment of the Requirements for the Degree.

Doctor of Philosophy

Athens, Georgia

2022

©2022
Yawei Shen
All Rights Reserved

Development for New Item Bank Calibration Strategies for Computerized Adaptive Testing in Small-Scale Assessments

by

Yawei Shen

Major Professor:	Shiyu Wang
Committee:	Laine Bradshaw
	Allan Cohen
	Nicole Lazar
	Laura Lu

Electronic Version Approved:

Ron Walcott
Vice Provost for Graduate Education and Dean of the Graduate School
The University of Georgia
August 2022

Dedication

To my parents, Aiju Wu and Zhiming Shen, who love, support, and believe in me unconditionally.

Acknowledgments

Words cannot express my deepest gratitude to my mentor, Dr. Shiyu Wang, for her wholehearted care, generous support, and academic guidance. She has the magic to clear my mind with a few questions and suggestions, and I can always recharge myself with confidence and courage to keep going after our conversations. Without her, my journey in graduate school would not have been so fruitful and unforgettable, and I can never achieve this far. I would like to also express my gratitude to my dissertation committee: Drs. Laine Bradshaw, Allan Cohen, Nicole Lazar, and Laura Lu, who have provided much-appreciated advice and insights to help me in completing my work. I also thank other faculty and graduate students in the Quantitative Method family, who have shown me very different theories, concepts, ideas, and techniques in this old and still developing field. I want to also thank the many staffs who have provided consistent service and help.

I must also express my gratitude to my friends: Hui Du, Xi Fang, Biyao Liang, Siyu Wan, Qian Zhang, and Jun Zhao. Without their accompany and support, I would not have had such an enjoyable journey in a foreign country and in a pandemic. I am also grateful to my fiancé, Elgin Leary, who believes that I am the smartest and the sweetest in the whole world. Last but not least, I would like to thank my family, who have always loved and supported me.

Contents

Acknowledgments v

List of Figures vii

List of Tables xv

1 Introduction 1
- 1.1 Background . . . 3
- 1.2 Overview of Existing Methods and Their Limitations. . . . 5
- 1.3 Objectives and Outline of This Dissertation . . . 6

2 Literature Review 10
- 2.1 Item Response Theory . . . 10
- 2.2 Computerized Adaptive testing . . . 14
- 2.3 Item Bank Development . . . 18
- 2.4 Missing Data Techniques . . . 25
- 2.5 Matrix Completion . . . 30

3 Study 1: Adapt Matrix Completion Algorithms in Incomplete Calibration Designs 35
- 3.1 RQ1: Adaptations for 1-Bit Matrix Completion. . . . 36
- 3.2 RQ2: Adaptations for Structured Matrix Completion. . . . 41
- 3.3 Simulation Study 1 Design . . . 46
- 3.4 Simulation Study 1 Results . . . 56

4 Study 2: Develop New Item Bank Calibration Strategies 121
- 4.1 Methods . . . 122
- 4.2 Simulation Study 2 Design . . . 131
- 4.3 Simulation Study 2 Results . . . 132

5 Study 3: Applications of the Proposed Item Bank Calibration Strategies 210
- 5.1 The CAT Simulation . . . 210
- 5.2 Real Data Demonstration . . . 233

6 Conclusion and Discussion **246**
6.1 Summary 246
6.2 Implications 252
6.3 Limitations and Future Directions 253

Bibliography **255**

Appendices **265**

A Proximal gradient descent and fast truncated singular value decomposition algorithm for 1-bit matrix completion **265**

B Determining the switching parameters of *Switch_Preferred* **266**

C Figures **273**

D Tables **311**

List of Figures

2.1 Item Characteristic Curves 11

3.1 Full Response Matrices 37
3.2 Observed and Imputed Response Matrices From Targeted Calibration Designs 38
3.3 Observed and Imputed Response Matrices From Random Calibration Designs 39
3.4 Impute Missing Responses using Structured Matrix Completion for the Targeted NEAT Design 42
3.5 Arrange Observed Response Matrix from Random NEAT Design Before Using Structured Matrix Completion 43
3.6 Imputing Missing Responses using Structured Matrix Completion for Targeted Panel Chained Design 44
3.7 Distributions of Item Parameters in Three Item Banks 48
3.8 Anchor Item Selection for NEAT Design 51
3.9 The General Procedure in the Simulation Study 55
3.10 Recovery Rates of *1bit(Sign)* and *1bit(p_tw)* in Practical Item Bank and $N = 125$ 58
3.11 Bias(b) per Item of *1bit(Sign)* and *1bit(p_tw)* in Practical Item Bank and $N = 125$ . . . 60
3.12 Bias(a) per Item of *1bit(Sign)* and *1bit(p_tw)* in Practical Item Bank and $N = 125$ . . . 61
3.13 Recovery Rates of *1bit(Sign)* and *1bit(p_tw)* Across Item Banks with $N = 125$ 62
3.14 Recovery Rates of *1bit(Sign)* and *1bit(p_tw)* Across Item Banks in Targeted NEAT Designs and $N = 125$ 63
3.15 Recovery Rates of *1bit(Sign)* and *1bit(p_tw)* in Across Item Banks Targeted PC Designs and $N = 125$ 63
3.16 RMSE(b) of *1bit(Sign)* and *1bit(p_tw)* in Across Item Banks with $N = 125$ 64
3.17 Bias(b) per Item of *1bit(Sign)* and *1bit(p_tw)* in Uniform Item Bank and $N = 125$ . . . 65
3.18 Bias(b) per Item of *1bit(Sign)* and *1bit(p_tw)* in Normal Item Bank and $N = 125$ . . . 65
3.19 RMSE(a) of *1bit(Sign)* and *1bit(p_tw)* in Across Item Banks with $N = 125$ 66
3.20 Bias(a) of *1bit(Sign)* and *1bit(p_tw)* in Uniform Item Bank and $N = 125$ 67
3.21 Bias(a) per Item of *1bit(Sign)* and *1bit(p_tw)* in Normal Item Bank and $N = 125$ 67
3.22 Bias(b) per Item of *1bit(Sign)* and *1bit(p_tw)* in Practical Item Bank and $N = 500$ . . . 70
3.23 Bias(a) per Item of *1bit(Sign)* and *1bit(p_tw)* in Practical Item Bank and $N = 500$ . . . 71
3.24 Imputed Response Matrices From *1bit(p_tw)* in Practical Item Bank 73

3.25 Observed Item Means Across Item Banks and Incomplete Calibration Designs 74
3.26 Observed Person Means Across Item Banks and Incomplete Calibration Designs 74
3.27 Item Difficulty vs. Theta 75
3.28 Item Means for *1bit(p_tw)* Compared to Full Response Matrix using Practical Item Bank with $N = 125$. 76
3.29 Difference Between *1bit(p_tw)* Imputed Matrix and Full Response Matrix using Practical Item Bank with $N = 125$. 78
3.30 Recovery Rates of Adapted Structured Matrix Completion in Practical Item Bank and $N = 125$ 80
3.31 Recovery Rates per Item of Selected Modified Structured Matrix Completion in Practical Item Bank with Sample Size of 125 81
3.32 Bias(b) Per Item in Random NEAT Designs using *SMC(p1+Sign)* and *SMC(p2+p_tw)* in Practical Item Bank with $N = 125$ 82
3.33 Bias(b) per Item of Selected Adapted Structured Matrix Completion in Practical Item Bank with Sample Size of 125 84
3.34 Bias(a) per Item of Selected Adapted Structured Matrix Completion Methods in Practical Item Bank with Sample Size of 125 86
3.35 Recovery Rates for Modified Structured Matrix Completion Across Item Banks in Targeted NEAT Designs and $N = 125$ 88
3.36 Recovery Rates of Modified Structured Matrix Completion Across Item Banks in Targeted PC Designs and $N = 125$ 88
3.37 Bias(b) per Item of Selected Adapted Structured Matrix Completion Methods in Targeted NEAT Design with Sample Size of 125 89
3.38 Bias(b) per Item of Selected Adapted Structured Matrix Completion Methods in Targeted PC Design with Sample Size of 125 90
3.39 Bias(b) per Item of Selected Adapted Structured Matrix Completion Methods in Random NEAT Design with Sample Size of 125 90
3.40 Bias(b) per Item of Selected Adapted Structured Matrix Completion Methods in Random PC Design with Sample Size of 125 91
3.41 Bias(a) per Item of Selected Modified Structured Matrix Completion in Targeted NEAT Design with $N = 125$ 91
3.42 Bias(a) per Item of Selected Modified Structured Matrix Completion in Targeted PC Design with $N = 125$ 92
3.43 Bias(a) per Item of Selected Modified Structured Matrix Completion in Random NEAT Design with $N = 125$ 92
3.44 Bias(a) per Item in Random PC Design with Sample Size of 125 93
3.45 Imputed Response Matrices From *SMC(1+p_tw)* in Practical Item Bank 98
3.46 Imputed Response Matrices From *SMC(p1+Sign)* in Practical Item Bank 99
3.47 Imputed Response Matrices From *SMC(1+p_tw)* in Practical Item Bank. 101

3.48 Imputed Response Matrices From *SMC(p1+Sign)* in Practical Item Bank. 102
3.49 Recovery Rates of All Methods in Practical Item Bank and $N = 125$ 104
3.50 RMSE(b) of Selected Methods in Practical Item Bank with $N = 125$ 106
3.51 Bias(b) of Selected Methods in Practical Item Bank with $N = 125$ 107
3.52 RMSE(b) of Selected Methods in Practical Item Bank with $N = 500$ 109
3.53 Bias(b) of Selected Methods in Practical Item Bank with $N = 500$ 110
3.54 RMSE(a) of Selected Methods in Practical Item Bank with Sample Size of 125 111
3.55 Bias(a) of Selected Methods in Practical Item Bank with Sample Size of 125 113
3.56 RMSE(a) of Selected Methods in Practical Item Bank with Sample Size of 500 114
3.57 Bias(a) of Selected Methods in Practical Item Bank with Sample Size of 500 115
3.58 Difference Between the Full Response Matrix and Imputed Response Matrices From TW in Practical Item Bank with Sample Size of 125 117

4.1 Recovery Rates Per Examinees Using *1bit(p_tw)* and *1bit(Sign)* in Practical Item Bank and $N = 125$ 123
4.2 The Difference Between Full Response Matrix and Imputed Response Matrix Using *1bit(p_tw)* in Practical Item Bank with Sample Size of 125 124
4.3 The Difference Between Full Response Matrix and Imputed Response Matrix Using *SMC(1+p_tw)* in Practical Item Bank with Sample Size of 125 125
4.4 Proposed Switching Parameters p 127
4.5 Proposed Switching Parameters $q1$ 127
4.6 Proposed Switching Parameters $q2$ 128
4.7 Heat Maps for *SMC(p2+p_tw)* Imputed Matrix using Practical Item Bank with $N = 125$. 130
4.8 Recovery Rates of the Switching Method with *1bit(p_tw)* and TW in Practical Item Bank and $N = 125$ 134
4.9 Bias(a) per Item for Switching Method with *1bit(p_tw)* and TW in Practical Item Bank and $N = 125$ 137
4.10 Bias(b) per Item for Switching Method with *1bit(p_tw)* and TW in Practical Item Bank and $N = 125$ (Free Y-Axis) 139
4.11 Bias(b) per Item for Switching Method with *1bit(p_tw)* and TW in Practical Item Bank and $N = 125$ (Zoomed-In with Fixed Y-Axis) 140
4.12 Recovery Rates per Item for Switching Method with *1bit(p_tw)* and TW Across Item Banks with $N = 125$ In Targeted NEAT Designs 140
4.13 Recovery Rates per Item for Switching Method with *1bit(p_tw)* and TW Across Item Banks with $N = 125$ in Targeted PC Designs 141
4.14 Recovery Rates per Item for Switching Method with *1bit(p_tw)* and TW Across Item Banks with $N = 125$ in Random NEAT Designs 141
4.15 Recovery Rates per Item for Switching Method with *1bit(p_tw)* and TW Across Item Banks with $N = 125$ in Random PC Designs 142

4.16 RMSE(a) for Switching Method with *1bit(p_tw)* and TW Across Item Banks with $N = 125$ . . . 143
4.17 Bias(a) per item for Switching Method with *1bit(p_tw)* and TW Across Item Banks with $N = 125$ . . . 144
4.18 RMSE(b) for Switching Method with *1bit(p_tw)* and TW Across Item Banks with $N = 125$ . . . 145
4.19 Bias(b) per item for Switching Method with *1bit(p_tw)* and TW Across Item Banks with $N = 125$ . . . 146
4.20 Recovery Rates for Switching Method with *SMC(1+p_tw)* and TW in Practical Item Bank and $N = 125$ . . . 150
4.21 Bias(a) per Item for Switching Method with *SMC(1+p_tw)* and TW in Practical Item Bank and $N = 125$ . . . 152
4.22 Bias(b) per Item for Switching Method with *SMC(1+p_tw)* and TW in Practical Item Bank and $N = 125$ (Free Y-Axis) . . . 154
4.23 Bias(b) per Item for Switching Method with *SMC(1+p_tw)* and TW in Practical Item Bank and $N = 125$ (Zoomed in) . . . 155
4.24 Recovery Rates per Item for Switching Method with *SMC(1+p_tw)* and TW Across Item Banks with $N = 125$ in Targeted NEAT Designs . . . 156
4.25 Recovery Rates per Item for Switching Method with *SMC(1+p_tw)* and TW Across Item Banks with $N = 125$ in Targeted PC Designs . . . 157
4.26 Recovery Rates per Item for Switching Method with *SMC(1+p_tw)* and TW Across Item Banks with $N = 125$ in Random NEAT Designs . . . 157
4.27 Recovery Rates per Item for Switching Method with *SMC(1+p_tw)* and TW Across Item Banks with $N = 125$ in Random PC Designs . . . 158
4.28 RMSE(a) for Switching Method with *SMC(1+p_tw)* and TW Across Item Banks with $N = 125$ . . . 159
4.29 Bias(a) per item for Switching Method with *SMC(1+p_tw)* and TW Across Item Banks with $N = 125$ . . . 160
4.30 RMSE(b) for Switching Method with *SMC(1+p_tw)* and TW Across Item Banks with $N = 125$ . . . 161
4.31 Bias(b) per item for Switching Method with *SMC(1+p_tw)* and TW Across Item Banks with $N = 125$ . . . 162
4.32 Recovery Rates of the Softened Threshold Methods in Practical Item Bank with $N = 125$. 166
4.33 Bias(a) per Item of the Softened Threshold Method in Practical Item Bank with $N = 125$. 168
4.34 Bias(b) per Item for Softened Threshold Methods in Practical Item Bank with $N = 125$ (Free Y-Axis). . . . 170
4.35 Bias(b) per Item for Softened Threshold Methods in Practical Item Bank with $N = 125$ (Zoomed In). . . . 171

4.36 Recovery Rates per Item for the Softened Threshold Methods and TW Across Item Banks with $N = 125$ In Targeted NEAT Designs 172
4.37 Recovery Rates per Item for Switching Method with *1bit(p_tw)* and TW Across Item Banks with $N = 125$ in Targeted PC Designs 172
4.38 Recovery Rates per Item for Switching Method with *1bit(p_tw)* and TW Across Item Banks with $N = 125$ in Random NEAT Designs 173
4.39 Recovery Rates per Item for Switching Method with *1bit(p_tw)* and TW Across Item Banks with $N = 125$ in Random PC Designs 173
4.40 RMSE(a) for the Softened Threshold Methods and TW Across Item Banks with $N = 125$ 174
4.41 Bias(a) per item for the Softened Threshold Methods and TW Across Item Banks with $N = 125$ 175
4.42 RMSE(b) for the Softened Threshold Methods and TW Across Item Banks with $N = 125$ 176
4.43 Bias(b) per item for the Softened Threshold Methods and TW Across Item Banks with $N = 125$ 177
4.44 Bias(b) per Item of the Softened Methods in Practical Item Bank Across Sample Sizes . 181
4.45 Recovery Rates per Item for Pooling Converting Methods in Practical Item Bank with $N = 125$. 183
4.46 Bias(a) per Item for the Pooling Method in Practical Item Bank with $N = 125$. 185
4.47 Bias(b) per Item for Pooling Converting Methods in Practical Item Bank with $N = 125$. 186
4.48 Bias(b) per Item for Pooling Converting Methods in Practical Item Bank with $N = 125$. 187
4.49 Recovery Rates per Item for the Pooling Method Across Item Banks with $N = 125$ In Targeted NEAT Designs 188
4.50 Recovery Rates per Item the Pooling Method Across Item Banks with $N = 125$ in Targeted PC Designs 189
4.51 Recovery Rates per Item for the Pooling Method Across Item Banks with $N = 125$ in Random NEAT Designs 189
4.52 Recovery Rates per Item for the Pooling Method Across Item Banks with $N = 125$ in Random PC Designs 190
4.53 RMSE(a) for the Pooling Method and TW Across Item Banks with $N = 125$ 191
4.54 Bias(a) per item for the Pooling Method, *SMC(p2+p_tw)*, and TW Across Item Banks with $N = 125$ 192
4.55 RMSE(b) for the Pooling Method Across Item Banks with $N = 125$ 193
4.56 Bias(b) per item for the Pooling Method and TW Across Item Banks with $N = 125$. 194
4.57 Recovery Rates of All Methods in Practical Item Bank with $N = 125$. 199
4.58 Recovery Rates of All Methods in Practical Item Bank with $N = 125$. 200
4.59 RMSE(a) of Selected Methods in Practical Item Bank with Sample Size of 125 202

4.60 Bias(a) Per Item of Switching Methods in Practical Item Bank Across Incomplete Calibration Designs with $N = 125$. 203
4.61 Bias(a) Per Item of Switching Methods in Practical Item Bank Across Incomplete Calibration Designs and Sample Sizes. 204
4.62 RMSE(b) of Selected Methods in Practical Item Bank with Sample Size of 125 206
4.63 Bias(b) Per Item of Selected Methods in Practical Item Bank with $N = 125$. 207
4.64 Bias(b) Per Item of Selected Methods in Practical Item Bank with $N = 125$. 208

5.1 The Flow Chart for the CAT Simulation Process 213
5.2 Correlation Plots in 40-Item Tests when $\theta \sim N(\mu, 1)$ Using Item Bank Calibrated From Targeted NEAT Designs. 215
5.3 Correlation Plots in 20-Item Tests when $\theta \sim N(\mu, 1)$ Using Item Bank Calibrated From Targeted NEAT Designs. 216
5.4 Two Examples of Extreme Estimations of θ Using *SMC(1+Switch)* in Targeted NEAT Design with Test Length of 20 218
5.5 Two Examples of Extreme Estimations of θ Using *SMC(1+Switch)* in Targeted NEAT Design with Test Length of 40 218
5.6 Correlation Plots in 40-Item Tests when $\theta \sim N(0, 1)$ Using Item Bank Calibrated From Targeted NEAT Designs. 223
5.7 Correlation Plots in 20-Item Tests when $\theta \sim N(0, 1)$ Using Item Bank Calibrated From Targeted NEAT Designs. 224
5.8 Correlation Plots in 40-Item Tests when $\theta \sim N(\mu, 1)$ Using Item Bank Calibrated From Random NEAT Designs. 225
5.9 Correlation Plots in 20-Item Tests when $\theta \sim N(\mu, 1)$ Using Item Bank Calibrated From Random NEAT Designs. 226
5.10 Histogram of Estimated θ Versus True θ with Test Length of 40 When $\theta \sim N(\mu, 1)$ Using Item Bank Calibrated in Random NEAT Designs 227
5.11 Correlation Plots in 40-Item Tests when $\theta \sim N(0, 1)$ Using Item Bank Calibrated From Random NEAT Designs. 230
5.12 Correlation Plots in 20-Item Tests when $\theta \sim N(0, 1)$ Using Item Bank Calibrated From Random NEAT Designs. 231
5.13 Histogram of Estimated θ Versus True θ with Test Length of 40 When $\theta \sim N(0, 1)$ Using Item Bank Calibration From Random NEAT Designs 232
5.14 Response Patterns 234
5.15 Distribution of Missing Percentages 235
5.16 The Difference Between the Imputed Response Matrices 237
5.17 Correlation Plots of Estimated Item Discrimination Parameters 239
5.18 Correlation Plots of Estimated Item Difficulty Parameters 242

B.1 Method 2: *Switch_Preferred* for p for Targeted NET Designs and Uniform Item Bank using $1bit(p_tw)$ . . . 267
B.2 Method 2: *Switch_Preferred* for q_1 for Targeted NET Designs and Uniform Item Bank using $1bit(p_tw)$ . . . 268
B.3 Method 2: *Switch_Preferred* for q_2 for Targeted NET Designs and Uniform Item Bank using $1bit(p_tw)$ . . . 269
B.4 *Switch_Preferred* for p, q_1, and q_2 for Targeted PC Designs and Uniform Item Bank using $1bit(p_tw)$ . . . 270
B.5 *Switch_Preferred* for p, q_1, and q_2 for random NEAT Designs and Uniform Item Bank using $1bit(p_tw)$ . . . 271
B.6 *Switch_Preferred* for p, q_1, and q_2 for Random PC Designs and Uniform Item Bank using $1bit(p_tw)$ . . . 272

C.1 Recovery Rates of *1bit(Sign)* and *1bit(p_tw)* Across Item Banks in Random NEAT Designs and $N = 125$ . . . 273
C.2 Recovery Rates of *1bit(Sign)* and *1bit(p_tw)* Across Item Banks in Random PC Designs and $N = 125$ . . . 274
C.3 Bias(b) per Item of *1bit(Sign)* and *1bit(p_tw)* in Uniform Item Bank and $N = 500$ . . . 274
C.4 Bias(b) per Item of *1bit(Sign)* and *1bit(p_tw)* in Normal Item Bank and $N = 500$ . . . 275
C.5 Bias(a) per Item of *1bit(Sign)* and *1bit(p_tw)* in Practical Item Bank and $N = 500$ . . . 276
C.6 Bias(a) per Item of *1bit(Sign)* and *1bit(p_tw)* in Normal Item Bank and $N = 500$ . . . 277
C.7 Bias(b) per Item of Selected Modified Structured Matrix Completion in Practical Item Bank with Sample Size of 125 . . . 278
C.8 Bias(a) per Item of Selected Modified Structured Matrix Completion in Practical Item Bank with Sample Size of 125 . . . 279
C.9 Recovery Rates of Modified Structured Matrix Completion Across Item Banks in Random NEAT Designs and $N = 125$ . . . 280
C.10 Recovery Rates of Modified Structured Matrix Completion Across Item Banks in Random PC Designs and $N = 125$ . . . 280
C.11 Recovery Rates of All Methods in Uniform Item Bank and $N = 125$ . . . 281
C.12 Recovery Rates of All Methods in Normal Item Bank and $N = 125$ . . . 282
C.13 Recovery Rates of All Methods in Practical Item Bank and $N = 250$ . . . 283
C.14 Recovery Rates of All Methods in Practical Item Bank and $N = 500$ . . . 284
C.15 RMSE(b) of Selected Methods in Uniform Item Bank with $N = 125$ . . . 285
C.16 RMSE(b) of Selected Methods in Normal Item Bank with $N = 125$ . . . 286
C.17 Bias(b) of Selected Methods in Uniform Item Bank with $N = 125$ . . . 287
C.18 Bias(b) of Selected Methods in Normal Item Bank with $N = 125$ . . . 288
C.19 RMSE(b) of Selected Methods in Uniform Item Bank with $N = 500$ . . . 289
C.20 RMSE(b) of Selected Methods in Normal Item Bank with $N = 500$ . . . 290
C.21 Bias(b) of Selected Methods in Uniform Item Bank with $N = 500$ . . . 291

C.22 Bias(b) of Selected Methods in Normal Item Bank with $N = 500$ 292
C.23 RMSE(a) of Selected Methods in Uniform Item Bank with Sample Size of 125 293
C.24 RMSE(a) of Selected Methods in Normal Item Bank with Sample Size of 125 294
C.25 Bias(a) of Selected Methods in Uniform Item Bank with Sample Size of 125 295
C.26 Bias(a) of Selected Methods in Normal Item Bank with Sample Size of 125 296
C.27 RMSE(a) of Selected Methods in Uniform Item Bank with Sample Size of 500 297
C.28 RMSE(a) of Selected Methods in Normal Item Bank with Sample Size of 500 298
C.29 Bias(a) of Selected Methods in Uniform Item Bank with Sample Size of 500 299
C.30 Bias(a) of Selected Methods in Normal Item Bank with Sample Size of 500 300
C.31 Recovery Rates Per Examinees Using *1bit(p_tw)* and *1bit(Sign)* in Uniform Item Bank and $N = 125$ 301
C.32 Recovery Rates Per Examinees Using *1bit(p_tw)* and *1bit(Sign)* in Normal Item Bank and $N = 125$ 302
C.33 Recovery Rates of All Methods in the Uniform Item Bank Across Incomplete Calibration Designs and Sample Sizes 303
C.34 Recovery Rates of All Methods in the Normal Item Bank Across Incomplete Calibration Designs and Sample Sizes 304
C.35 Bias(a) Per Item of Switching Methods in Uniform Item Bank Across Incomplete Calibration Designs and Sample Sizes. 305
C.36 Bias(a) Per Item of Switching Methods in Normal Item Bank Across Incomplete Calibration Designs and Sample Sizes. 306
C.37 Bias(b) Per Item of Selected Methods in Uniform Item Bank with $N = 125$. 307
C.38 Bias(b) Per Item of Selected Methods in Normal Item Bank with $N = 125$. 308
C.39 Histogram of Estimated θ Versus True θ with Test Length of 20 When $\theta \sim N(\mu, 1)$ Using Item Bank Calibrated in Random NEAT Designs 309
C.40 Histogram of Estimated θ Versus True θ with Test Length of 20 When $\theta \sim N(0, 1)$ Using Item Bank Calibrated in Random NEAT Designs 310

List of Tables

2.1 Targeted 5-Form NEAT Design . . . 22
2.2 Targeted 5-Form Panel Chained Design . . . 22
2.3 Random 5-Form NEAT Design . . . 23
2.4 Random 5-Form Panel Chained Design . . . 24

3.1 Dichotomizing Methods for An Imputed Value x_{ij} . . . 41
3.2 Converting Methods (Binary Responses To Continuous Values) . . . 45
3.3 Simulation Factors in Simulation Study 1 . . . 47
3.4 The Summary Statistics of Item Parameters in Item Banks . . . 49
3.5 Summary Statistics for Item Parameters Per Test Form in Targeted NEAT Design . . . 51
3.6 Summary Statistics for Item Parameters Per Test Form in Random NEAT Design . . . 52
3.7 Summary Statistics for Item Parameters Per Test Form in Targeted PC Design . . . 53
3.8 Summary Statistics for Item Parameters Per Test Form in Random PC Design . . . 53
3.9 Bias(b) and RMSE(b) from *1bit(Sign)* and *1bit(p_tw)* in Practical Item Bank with $N = 125$ 59
3.10 Bias(a) and RMSE(a) from *1bit(Sign)* and *1bit(p_tw)* in Practical Item Bank with $N = 125$ 60
3.11 Recovery Rates Across Sample Sizes in Targeted NEAT Designs . . . 68
3.12 Bias(b) and RMSE(b) of *1bit(Sign)* and *1bit(p_tw)* Across Sample Sizes in Practical Item Bank . . . 69
3.13 Bias(a) and RMSE(a) of *1bit(Sign)* and *1bit(p_tw)* Across Sample Sizes in Practical Item Bank . . . 71
3.14 Average Bias(b) and RMSE(b) from Eight Adapted Structured Matrix Completion Methods When $N = 125$ in Practical Item Bank . . . 82
3.15 Bias (a) and RMSE(a) from Eight Adapted Structured Matrix Completion Methods When $N = 125$ in Practical Item Bank . . . 85
3.16 Recovery Rates from Eight Modified Structured Matrix Completion Across Sample Sizes in Practical Item Bank Under Targeted NEAT Designs . . . 94
3.17 Bias(b) and RMSE(b) from Eight Modified Structured Matrix Completion Across Sample Sizes in Practical Item Bank and Targeted NEAT Designs . . . 94
3.18 Bias(b) and RMSE(b) from 8 Adaptations for Structured Matrix Completion Across Sample Sizes in Practical Item Bank and Random NEAT Designs . . . 95

3.19 Bias(a) and RMSE(a) from Eight Modified Structured Matrix Completion Across Sample Sizes in Practical Item Bank and Targeted NEAT Designs 96
3.20 Bias(a) and RMSE(a) from Eight Modified Structured Matrix Completion Across Sample Sizes in Practical Item Bank and Random NEAT Designs 96
3.21 Summarized Difference Between Imputed Response Matrices from *1bit(p_tw)* and Full Response Matrix with Regards to Proportion of Wrongly Imputed Correct Responses in Targeted NEAT Design 118
3.22 Summarized Difference Between Imputed Response Matrices from *1bit(p_tw)* and Full Response Matrix with Regards to Proportion of Wrongly Imputed Incorrect Responses in Targeted NEAT Designs 118
3.23 Summarized Difference Between Imputed Response Matrices from *1bit(p_tw)* and Full Response Matrix with Regards to Proportion of Wrongly Imputed Correct Responses in Random NEAT Design 119
3.24 Summarized Difference Between Imputed Response Matrices from *1bit(p_tw)* and Full Response Matrix with Regards to Proportion of Wrongly Imputed Incorrect Responses in Random NEAT design 119

4.1 Investigated Methods in Simulation Study 2 133
4.2 Summary Statistics for Item Parameter Estimations for Switching Method with *1bit(p_tw)* and TW in Practical Item Bank and $N = 125$ 136
4.3 Average Recovery Rates Across Sample Sizes in Practical Item Banks 148
4.4 Item Discrimination Parameter Estimations Across Sample Sizes in Practical Item Banks 148
4.5 Item Difficulty Parameter Estimations Across Sample Sizes in Practical Item Banks . . . 149
4.6 Summary Statistics for Item Parameter Estimations for Switching Method with *SMC(1+p_tw)* and TW in Practical Item Bank and $N = 125$ 151
4.7 Average Recovery Rates Across Sample Sizes in Practical Item Bank 163
4.8 Summary of Item Discrimination Parameter Estimation Across Sample Sizes in Practical Item Bank 164
4.9 Summary of Item Difficulty Parameter Estimation Across Sample Sizes in Practical Item Bank 164
4.10 Summary Statistics for Item Parameter Estimations for Softened Threshold Method with *SMC(1+p_tw)*, *SMC(p1+p_tw)* and TW in Practical Item Bank with $N = 125$. 167
4.11 Average Recovery Rates of Softened Threshold Methods Across Sample Sizes in the Practical Item Bank 178
4.12 Item Discrimination Parameters Estimations Across Sample Sizes in the Practical Item Bank 179
4.13 Item Difficulty Parameters Estimations Across Sample Sizes in Practical Item Banks . . 180
4.14 Summary Statistics for Item Parameter Estimations for the Pooling Method 184
4.15 Average Recovery Rates of the Pooling Method Across Sample Sizes in the Practical Item Bank 195

4.16 Item Discrimination Parameters Estimations of Pooling Method Across Sample Sizes in the Practical Item Bank . . . 196
4.17 Item Difficulty Parameters Estimations of the Pooling Method Across Sample Sizes in the Practical Item Bank . . . 197

5.1 Simulation Factors in Study 3 . . . 212
5.2 Number of $\hat{\theta}$ with Absolute Deviation Larger Than 5 Using Item Bank Calibrated from Targeted NEAT Designs . . . 217
5.3 Summary Statistics of $\hat{\theta}$ when $\theta \sim N(\mu, 1)$ using Item Parameters Calibrated in Targeted NEAT Design with Test Length of 40 . . . 220
5.4 Summary Statistics of $\hat{\theta}$ when $\theta \sim N(\mu, 1)$ using Item Parameters Calibrated in Targeted NEAT Design with Test Length of 20 . . . 221
5.5 Number of $\hat{\theta}$ with Deviation larger than 5 Using Item Bank Calibrated from Random NEAT Designs . . . 223
5.6 Summary Statistics of $\hat{\theta}$ when $\theta \sim N(\mu, 1)$ using Item Parameters Calibrated in Random NEAT Design with Test Length of 40 . . . 228
5.7 Test Length and Sample Size for Each Test Form . . . 234
5.8 Proportion of the Same Imputed Responses . . . 236
5.9 Correlation of Estimated Item Discrimination Parameters . . . 238
5.10 Selected Items with Different Estimated Item Discrimination Parameters . . . 240
5.11 Correlation of Estimated Item Difficulty Parameters . . . 241
5.12 Selected Items with Different Estimated Item Difficulty Parameters . . . 244

D.1 Recovery Rates of *1bit(Sign)* and *1bit(p_tw)* Across Sample Sizes in Targeted PC Designs 311
D.2 Recovery Rates of *1bit(Sign)* and *1bit(p_tw)* Across Sample Sizes in Random NEAT Designs 312
D.3 Recovery Rates of *1bit(Sign)* and *1bit(p_tw)* Across Sample Sizes in Random PC Designs 312
D.4 Bias(b) and RMSE(b) of *1bit(Sign)* and *1bit(p_tw)* Across Sample Sizes in Uniform Item Bank . . . 313
D.5 Bias(b) and RMSE(b) of *1bit(Sign)* and *1bit(p_tw)* Across Sample Sizes in Normal Item Bank . . . 313
D.6 Bias(a) and RMSE(a) of *1bit(Sign)* and *1bit(p_tw)* Across Sample Sizes in Uniform Item Bank . . . 314
D.7 Bias(a) and RMSE(a) of *1bit(Sign)* and *1bit(p_tw)* Across Sample Sizes in Normal Item Bank . . . 314
D.8 RMSE(b) from Eight Modified Structured Matrix Completion When $N = 125$ in Uniform Item Bank . . . 315
D.9 RMSE(b) from 8 Adaptations for Structured Matrix Completion When $N = 125$ in Normal Item Bank . . . 315
D.10 Bias(a) and RMSE(a) from Eight Modified Structured Matrix Completion When $N = 125$ in Uniform Item Bank . . . 316

D.11 Bias(a) and RMSE(a) from Eight Modified Structured Matrix Completion When $N = 125$ in Normal Item Bank . . . 316
D.12 Recovery Rates from Eight Modified Structured Matrix Completion Across Sample Sizes in Uniform Item Bank Under Targeted NEAT Designs . . . 317
D.13 Recovery Rates from Eight Modified Structured Matrix Completion Across Sample Sizes in Normal Item Bank Under Targeted NEAT Designs . . . 317
D.14 Bias(b) and RMSE(b) from Eight Modified Structured Matrix Completion Across Sample Sizes in Practical Item Bank and Targeted PC Designs . . . 317
D.15 Bias(b) and RMSE(b) from Eight Modified Structured Matrix Completion Across Sample Sizes in Practical Item Bank and Random PC Designs . . . 318
D.16 Bias(a) and RMSE(a) from Eight Modified Structured Matrix Completion Across Sample Sizes in Practical Item Bank and Targeted PC Designs . . . 318
D.17 Bias(a) and RMSE(a) from Eight Modified Structured Matrix Completion Across Sample Sizes in Practical Item Bank and Random PC Designs . . . 318
D.18 Summarized Difference Between Imputed Response Matrices from *SMC(1+p_tw)* and Full Response Matrix with Regards to Proportion of Wrongly Imputed Correct Responses 319
D.19 Summarized Difference Between Imputed Response Matrices from *SMC(1+p_tw)* and Full Response Matrix with Regards to Proportion of Wrongly Imputed Incorrect Responses . . . 319
D.20 Average Recovery Rates Across Sample Sizes in Uniform Item Banks . . . 320
D.21 Average Recovery Rates Across Sample Sizes in Normal Item Banks . . . 320
D.22 Average Recovery Rates Across Sample Sizes in Uniform Item Bank . . . 321
D.23 Average Recovery Rates Across Sample Sizes in Normal Item Bank . . . 321
D.24 Average Recovery Rates of Softened Threshold Methods Across Sample Sizes in the Uniform Item Bank . . . 322
D.25 Average Recovery Rates of Softened Threshold Methods Across Sample Sizes in the Normal Item Bank . . . 323
D.26 Average Recovery Rates of the Pooling Method Across Sample Sizes in the Uniform Item Bank . . . 324
D.27 Average Recovery Rates of the Pooling Method Across Sample Sizes in the Normal Item Bank . . . 325
D.28 Summary Statistics of $\hat{\theta}$ when $\theta \sim N(0, 1)$ using Item Parameters Calibrated in Targeted NEAT Design with Test Length of 40 . . . 326
D.29 Summary Statistics of $\hat{\theta}$ when $\theta \sim N(0, 1)$ using Item Parameters Calibrated in Targeted NEAT Design with Test Length of 20 . . . 327
D.30 Summary Statistics of $\hat{\theta}$ when $\theta \sim N(\mu, 1)$ using Item Parameters Calibrated in Random NEAT Design with Test Length of 20 . . . 328
D.31 Summary Statistics of $\hat{\theta}$ when $\theta \sim N(0, 1)$ using Item Parameters Calibrated in Random NEAT Design with Test Length of 20 . . . 329

D.32 Summary Statistics of $\hat{\theta}$ when $\theta \sim N(0,1)$ using Item Parameters Calibrated in Random NEAT Design with Test Length of 40 330

Chapter 1

Introduction

Computerized adaptive testing (CAT) is a popular mode to deliver educational assessments. During a CAT, if an examinee answers an item incorrectly, the next item will be easier. Similarly, if an examinee answers an item correctly, the next item will be more difficult. CAT is designed in this way so that the items are purposely directed to the examinee's ability level in real-time so that most of the information about this examinee's ability can be collected (Wainer and Mislevy, 1990). On the contrary, in a linear test, another common test mode, all the examinees answer an identical set of items that are pre-assembled by content experts. In order to test the relatively wide range of abilities for all the examinees, the linear test must contain items with a similar range of difficulties. Therefore, examinees of high abilities have to answer some simple items, and those of low abilities have to work on some difficult items. The collected responses from these items provide little information about these examinees' ability levels. The primary function of CAT is to provide estimations for examinees' ability with greater precision and shorter test length than linear tests, especially for examinees with very high or low ability levels who barely encounter items matching their ability levels on a linear test. Many researchers have found that CAT can reduce about half of the test length in average without compromising measurement quality compared to the linear test (e.g., Brown and Weiss, 1977; McBride and Martin, 1983; Moreno et al., 1984; Wise and Kingsbury, 2000; Weiss, 1982). The successful applications of CAT have been seen in several large-scale assessments, such as the National Council of State Boards of Nursing, the Armed Services Vocational Aptitude Battery

(ASVAB), Graduate Management Admission Test (GMAT), and Graduate Record Examination (GRE). According to ETS, about 1.5 million people have taken GRE between 2017 and 2020 (ETS, 2021).

In addition to these well-known large-scale assessments, CAT in fact has the potential to be used for classroom assessments and is beneficial to instructors' teaching and students' learning. For example, CAT provides accurate assessments of students' abilities, especially for examinees with very high or low levels of ability, so that teachers and students can accurately understand their teaching and learning results. In addition, since CAT significantly reduces test length and test time, teachers can spend more time on teaching and students can spend more time on learning. Furthermore, CAT provides instant results, allowing teachers to customize their teaching methods and contents in real-time. Finally, research studies show that CAT offers a better test experience and higher test-relevant motivation and engagement for students, because it adjusts the test according to the individual's ability level (Martin and Lazendic, 2018). Although it is attractive to implement CAT in classroom assessments, there are still few applications (Tao et al., 2008; Kaya and Kalender, 2018) unlike the many CAT applications in large-scale assessments. The main difference between classroom assessments and large-scale assessments is the number of examinees. Rather than millions of examinees in large-scale assessments, the average classroom sizes are very small. For example, the classroom sizes for 8^{th} and 9^{th} mathematics classes are between 20 to 35 across 11 countries (Woessmann and West, 2006). In addition, according to Schools and Staffing Survey (U.S. Department of Education, 2012), the average class sizes in Georgia are 24, 24, and 25 for primary schools, middle schools, and high schools respectively.

There are two reasons for the lack of CAT applications in small-scale assessments. The first one is the requirement for computers and other appropriate facilities, which are not always available in classrooms. The other reason is that the foundation of CAT, item response theory (IRT) model, requires large sample sizes to implement (e.g., Baker and Kim, 2004; Lord, 1980). Specifically, CAT requires an item bank consisted of a collection of items with known item parameters. These item parameters are estimated based on IRT models to capture the characteristics of items. In this way, CAT can select the best item in the item bank to match each individual's ability level. Obtaining accurate item parameter estimations based

on IRT models requires large sample sizes (e.g., Lord, 1968; Weiss and Von Minden, 2012). However, given the average classroom sizes are typically small, it is hard to collect data from large samples to calibrate the item parameters accurately in the small-scale assessments.

1.1 Background

Importance of a well-developed item bank for CAT.

A fundamental element of a CAT is an item bank which consists of a collection of items with known parameters based on IRT models (Segall, 2005). A well-developed item bank is crucial for a CAT, because the power of CAT is achieved by matching items' characteristics to examinees' ability levels. If item parameters are inaccurate, CAT will choose far easy or difficult items for examinees, and thus the efficiency of CAT will be lost. In addition, because IRT assumes a latent metric for the ability, where both items and examinees can be located, an examinee's ability estimation on that latent metric is the same though he or she takes a different set of items. With this measurement invariant property of IRT (Mellenbergh, 1989; Meredith and Millsap, 1992), CAT can provide comparable scores for examinees even they answer different sets of items. However, if the item parameters are not accurate, there will be bias in the estimation of the examinee' ability on the latent metric and thus jeopardize the fairness of CAT (Meredith, 1993).

Challenges on calibrating an item bank for small-scale assessments.

There are generally two approaches to calibrate an item bank according to whether there is a linear test before the CAT. When building a CAT to replace a linear test, the collected data from the linear test can be used to calibrate the item parameters, and an additional equating study is necessary to adjust the medium of presentation effect (Wainer and Mislevy, 1990; Thompson and Weiss, 2011). When building a completely new CAT, a large number of examinees are recruited to answer items, and the collected responses are used to estimate the item parameters. Because item bank usually contains a large number of items, it is impossible to require every examinee to answer all the items. Therefore, incomplete calibration designs are often recommended (Eggen and Verhelst, 2011; Wainer and Mislevy, 1990). The incomplete calibration designs intend to collect responses for a subset of items in the item bank from examinees, and

all the items are answered by some examinees but not all the examinees. In this dissertation, we aimed to develop a new CAT item bank without a previous linear test, and thus we focused on the second approach. The challenges of calibrating an item bank using the second approach for small-scale assessments can be summarized as follows.

The first challenge is that the large samples sizes are not always available for small-scale assessments (e.g., Welch and Frick, 1993). However, IRT requires a large sample size to ensure the accuracy of item parameter estimations. Researchers have investigated different sample sizes from 200 to more than 1,000 (e.g., Weiss and Von Minden, 2012). Şahin and Weiss (2015) mentioned that calibration with 1,000 examinees suggested by Lord (1968) is still the most common practice. De Ayala (2013) has recommended a minimum of 500 responses per item for the two-parameter logistic model, which is a popular IRT model for CAT (e.g., Makransky and Glas, 2014; Buyske et al., 1998). As introduced before, the typical classroom size is much smaller than 500.

The second challenge is the complex missing data problem caused by incomplete calibration designs. Due to practical concerns such as fatigue effects from examinees answering all the items in the item bank, incomplete calibration designs are suggested for item bank calibration. For example, Stocking (1994) suggested a CAT item bank with the size of six to eight linear tests. In this case, the missing percentages are between 83.3% (5/6) to 87.5% (7/8) if requesting examinees answer the same number of items as that in a linear test. The previous study has shown that different incomplete designs for item bank calibration typically produce missing percentages from 40% to 70% (Zhang, 2020). In addition to the high missing percentages, the missing mechanism under the incomplete calibration design is complex. In statistical analyses, an ideal missing mechanism is missing completely at random (MCAR), because many missing data estimation procedures provide unbiased estimates and recover most of the power lost due to data attrition (e.g., Schafer, 1997; Rhemtulla and Little, 2012). However, the incomplete calibration design may not produce MCAR data, because it is not reasonable to randomly assign items to examinees given some items may be too difficult or too easy for them. In this case, the data is missing not at random (MNAR). This type of missing data can reduce the statistical power, lead to biased estimates, complicate the analysis,

and produce invalid conclusions (Kang, 2013). To summarize, incomplete calibration designs lead to high proportion of missing data and MNAR. In addition to the small sample sizes in classroom assessments, the missing data may cause more troubles and threaten the item parameter estimations.

1.2 Overview of Existing Methods and Their Limitations.

To encounter the first challenge in item bank calibration with small sample sizes, researchers adopted the idea of optimal designs to calibrate item bank with a smaller number of examinees than usually recommended requirements (e.g., Berger, 1992; Chang and Lu, 2010). The key idea of optimal design in this context is to select suitable examinees, whose answers would provide maximum information for estimating the item's parameters. For example, to calibrate a difficult item, examinees with high ability levels provide more information than examinees with low ability levels. To find the appropriate examinees to answer the items, it is necessary to know examinees' ability levels. Thus, the procedures of optimal design can be summarized into two stages. In the first stage, initial estimations for both items and examinees are obtained after collecting some response data. In the second stage, optimal rules or adaptive algorithms are applied to select the most suitable examinees. In this way, the calibration efficiency is improved as smaller sample sizes are required to achieve similarly accurate item parameter estimations (e.g., Zhu, 2006). However, it is very difficult to implement optimal designs because it requires an "examinee pool" and some initial estimations for item parameters to compute the information to select examinees for items.

To address the second challenge of missing data, there are many missing data techniques that are studied in the educational measurement and psychometrics fields (e.g., Bernaards and Sijtsma, 2000; Finch, 2008; Edwards and Finch, 2018). The missing data techniques generally fall into two categories: (1) imputation and (2) maximum likelihood estimation on observed data only. The imputation related techniques, such as mean imputation, response function imputation, two-way imputation, and multiple imputations, have been studied in the contexts of examinees skipping one or more items or not reaching items in the later stage of the test, where the missing percentages are relatively low (e.g. Finch, 2008; Sijtsma and Van der Ark, 2003). The full information maximum likelihood method is used to deal with missing

data in incomplete designs for item bank calibration (Mislevy and Sheehan, 1989; Eggen and Verhelst, 2011; DeMars, 2002; Makransky and Glas, 2014). They found that marginal maximum likelihood estimation with the information of which examinee takes which test form is justified for item parameter estimations. However, this method has not been investigated in small-scale assessments.

1.3 Objectives and Outline of This Dissertation

The overall goal of this dissertation is to develop an item bank calibration strategy for small-scale assessments to address the aforementioned challenges and limitations in current methodological work. Because of the difficulty to implement the optimal design, we developed new item bank calibration strategies based on incomplete calibration designs and matrix completion algorithms for small-scale assessments. Incomplete calibration designs were used in this dissertation because of the practical concerns introduced before, and matrix completion is a family of algorithms to impute missing values in a matrix and has have shown great potential in many fields with high missing percentages (Jiang et al., 2016). The incomplete calibration designs and the matrix completion algorithms were briefly introduced here and formally discussed in the later sections.

Incomplete calibration designs. Two sets of well-studied incomplete calibration designs were investigated in this dissertation. The first was the targeted calibration design, which integrates the prior knowledge about examinees' abilities, such as grades or ages, and prior knowledge about items, such as difficulty, to assemble test forms and decide which examinees to take which test forms. In particular, two ways of assembling test forms and administering test forms to examinees using non-equivalent group with anchor test (NEAT) design and panel chained (PC) design were investigated. Though both designs were following the targeted calibration designs, they had different missing structures. For example, NEAT design had a set of anchor items and no missing responses for those items were planned. On the contrary, all items in PC designs had a pre-determined proportion of missingness. Targeted calibration designs are applicable in the small-scale assessments, because examinees who are clustered in a classroom or a grade have similar background and this background information is known and related to their ability levels. Also, content

experts and teachers have relatively good judgments about items' difficulty levels (Attali et al., 2014). In addition, it can improve the calibration efficiency by allocating specific groups of examinees to subsets of items (Eggen and Verhelst, 2011), and examines only receive items that match their ability levels. However, targeted calibration design leads to MNAR data. The second design was the random incomplete design, which produces MCAR data with a careful plan (Pokropek et al., 2011; Graham et al., 2006). In this type of design, the prior information about examinees and items are not utilized when designing the test forms or deciding which examinees to take which test form. It is especially suitable for situations where there is no prior knowledge about examinees and items. Similar to targeted calibration design, two ways of assemble test forms and administering test forms were investigated. Thus, there were four incomplete calibration designs in this dissertation in total.

Matrix completion. Matrix completion aims to recover a matrix with missing values, and Candès and Recht (2009) have shown a matrix with low-rank can be recovered from a small set of observed values. The response matrix from educational assessments (examinees by items) is a low-rank matrix, because examinees' responses depend only on their ability and item parameters. Therefore, matrix completion can be used in the field of educational measurement. With the considerations of the unique characteristics of the above-mentioned incomplete calibration designs, 1-bit matrix completion (Davenport et al., 2014) and structured matrix completion (Cai et al., 2016) were selected to investigate and build on in this study. The reason to choose 1-bit matrix completion was that it can deal with an incomplete matrix with binary data, which is directly applicable for educational assessment data that are commonly scored as correct and incorrect. Structured matrix completion was selected because it can handle MNAR, which is the case of targeted calibration designs. However, given the special characteristics of the incomplete calibration designs, there were two unique challenges to apply these two algorithms directly. Firstly, 1-bit matrix completion assumes uniform missingness, which means the probability that a response is missing is the same for all the responses in the matrix. However, targeted calibration designs do not satisfy this assumption, because the test forms and their assignments to examinees are dependent on the prior knowledge about examinees and items. In addition, though test forms were randomly assembled in the random incomplete

calibration designs, examinees answered all the items in the test form they received and missed all the other items. Thus, the missingness is not uniformly distributed in the random incomplete designs either. Secondly, the structured matrix completion is developed for a very specific structured missingness, where some rows and some columns in the matrix are completely observed. However, all the above-mentioned incomplete calibration designs do not have this specific structure. Lastly, structured matrix completion is designed for continuous matrix, while the response data in our scenario is binary.

To summarize, we proposed and evaluated new item bank calibration strategies in this dissertation. The rest of this dissertation is organized as follows. Firstly, in **chapter 2**, we reviewed and discussed the key theories and methods related to this dissertation, including item response theory, CAT, item bank development, missing data techniques, and matrix completion algorithms. Then, we conducted three interdependent studies which were documented from chapters 3 to 5. These three studies proposed, refined, and evaluated item bank calibration strategies with small pretesting sample sizes based on incomplete calibration designs and matrix completion algorithms. In **chapter 3** (Study 1), we explored and adapted both 1-bit matrix completion and structured matrix completion algorithms to address the above-mentioned two challenges of the specific missing patterns and the binary response data. The proposed adaptations were evaluated in a simulation study with various factors. The results of this simulation study revealed characteristics of the imputed response matrices of the proposed adaptations. In **chapter 4** (Study 2), we proposed three item bank calibration strategies to avoid the shortcomings of the proposed adaptations shown in the study 1. These three proposed strategies were then evaluated in a simulation study using the same design in study 1. In **chapter 5** (Study 3), we evaluated the proposed item bank calibration strategies in a CAT simulation and an empirical data set. In particular, the calibrated item parameters using the proposed item bank calibration strategies were used in a CAT simulation and the examinees' ability estimations from this CAT were evaluated. In addition, the proposed item bank calibration strategies were applied to the empirical data set collected from small-assessments, which had different missing patterns. The findings from this empirical data analysis provided insights to researchers and practitioners when

applying the proposed item bank calibration strategies. Lastly, in *chapter 6*, we summarized the main findings from chapters 3 to 5 and discussed the implications, limitations, and future directions.

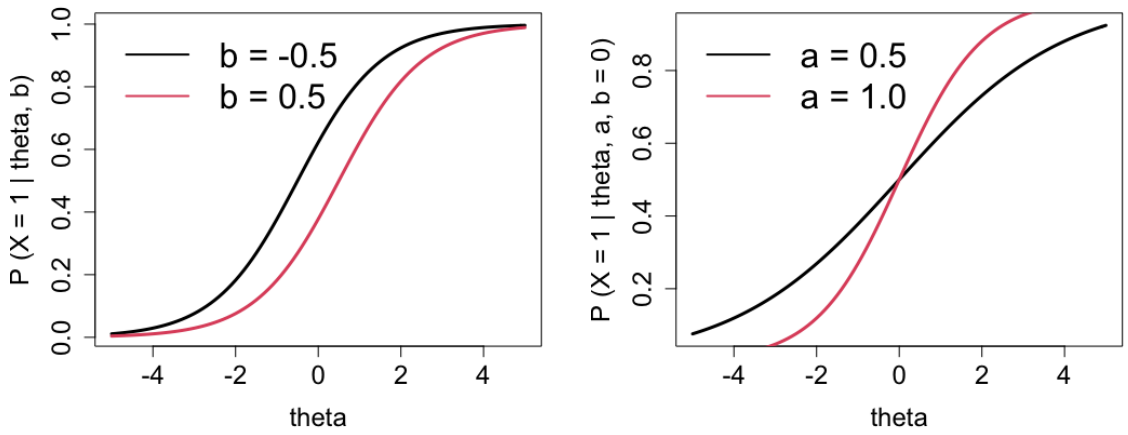
Chapter 2

Literature Review

2.1 Item Response Theory

Item response theory (IRT) is a family of mathematical models that capture the relations between examinees' response on each item in a test given their latent abilities (e.g., Baker and Kim, 2004; Lord, 1980). The key idea is that each test item is characterized by a few parameters, and each examinee is characterized by a few latent abilities. Given IRT models, the response of any examinee to any item can be probabilistically predicted. Usually, a unidimensional latent ability for examinees is assumed and denoted as θ. There are also multidimensional IRT (Reckase, 2009) where multiple latent constructs or abilities are assumed. This dissertation focuses on unidimensional IRT models. According to the number of parameters to capture item's characteristics, there are three main unidimensional IRT models: one-parameter logistic model (1PL), two-parameter logistic model (2PL), and three-parameter logistic model (3PL). The 1PL model, also known as Rasch model, uses the item difficulty (b) to describe items. Let X_{ij} represent the response from j^{th} examinee to i^{th} item, $i = 1, ..., I$ and $j = 1, ..., J$. $X_{ij} = 1$ means the answer is correct, and $X_{ij} = 0$ means the answer is incorrect. The 1PL formulates the relation using a logistic function:

$$P(X_{ij} = 1|\theta_j) = P_i(\theta_j) = \frac{1}{1 + e^{(\theta_j - b_i)}} \tag{2.1}$$



(a) 1PL (b) 2PL

Figure 2.1: Item Characteristic Curves

Figure 2.1a shows the item characteristic curves (ICC) for two items ($b_1 = -0.5$ and $b_2 = 0.5$). ICC presents the conditional probability of a correct response at each value of the ability level θ. Higher the latent ability level θ leads to larger probability of a correct response. Though many attractive features of 1PL models (Rasch, 1960; Wright and Stone, 1979), it has a restriction that ICCs must be parallel among items with different difficulty levels, and some applications have shown that it does not always provide a good fit to the data (Wainer and Mislevy, 1990). To allow for different slopes for different items, 2PL model introduces an additional item parameter, the item discrimination parameter, which is usually denoted as a. Figure 2.1b presents the ICCs for two items with the same difficulty parameter ($b = 0$) but different discrimination parameters ($a_1 = 0.5$, $a_2 = 1$) where the ICCs are not parallel. It means that the item with the red curve ($a = 1.0$) is more difficult for examinees of high ability than the item with the black curve ($a = 0.5$), though their item difficulty parameters are the same ($b = 0$). On the other hand, item with the black curve ($a = 0.5$) is more difficult for examinees of low ability than the item with the red curve ($a = 1.0$). In this way, with the same b, the item with higher discrimination parameter has more power in terms of separating examinees of low ability and examinees of high ability. The mathematical

expression for 2PL can be shown as:

$$P(X_{ij} = 1|\theta_j) = P_i(\theta_j) = \frac{1}{1 + e^{-1.7a_i(\theta_j - b_i)}} \tag{2.2}$$

With the consideration of the popularity of multiple-choice type items and the chances of guessing the item correctly, 3PL models add a third type of item parameter, the guessing parameter. This parameter builds a lower bound for the probability of getting an item correct. This study focuses on 2PL models, as 2PL models are commonly used in CAT related studies (e.g., Buyske et al., 1998; Zhu, 2006; Fink et al., 2018; Makransky and Glas, 2014).

2.1.1 Item Parameter Estimations

Without the consideration of missing values, three types of procedures are used for item parameter estimations for 2PL models: joint maximum likelihood estimations (JMLE; Hambleton and Cook, 1977), marginal maximum likelihood estimation (MMLE; Bock and Aitkin, 1981), and Bayesian approaches (e.g. Mislevy, 1986). All three methods are based on the likelihood function. The likelihood for item parameters $\mathbf{a}$, $\mathbf{b}$, and person parameters $\boldsymbol{\theta}$ given examinees' responses $\mathbf{X}$ is:

$$L(\boldsymbol{\theta}, \mathbf{a}, \mathbf{b}; \mathbf{X}) = \prod_{i=1}^{I} \prod_{j=1}^{J} P_i(\theta_j)^{x_{ij}} Q_i(\theta_j)^{1-x_{ij}} \tag{2.3}$$

where $Q_i(\theta_j) = 1 - P_i(\theta_j)$. JMLE are obtained by finding the values of each θ_j, a_i, and b_i that together maximize Equation (2.3). When using MMLE to obtain the estimations for item parameters, the likelihood function in Equation (2.3) is marginalized with respect to $f(\theta)$, where $f(\theta)$ represents the prior knowledge about the examinees's ability distribution. The marginal maximum likelihood is

$$L(\mathbf{a}, \mathbf{b}; \mathbf{X}) = \prod_{i=1}^{I} \prod_{j=1}^{J} \int P_i(\theta_j)^{x_{ij}} Q_i(\theta_j)^{1-x_{ij}} f(\theta) d(\theta) \tag{2.4}$$

More details for MMLE can be found in Bock and Aitkin (1981) and Samejima (1983). Bayesian methods multiply a prior for each of the parameters to the likelihood function in Equation (2.3) and thus obtain the posterior distributions of the parameters. The final estimations can either be the mode of the posterior distribution (Maximum a posterior Posteriori; MAP) or the mean of the posterior distribution (Expected a Posteriori; EAP).

MMLE has a major advantage over JMLE because it can estimate item parameters without having to estimate ability parameters. In addition to the benefit of computational speed, MMLE is also consistent in the number of examinees rather than JMLE. Lord (1986) pointed out that when there are only 10 or 15 item per person, JMLE will obtain biased estimates of ability parameters and thus the item parameters will not be accurately estimated. In addition, when there are over one thousand examinees and 40 or more items per person, there will be little difference between the estimates from MMLE and JMLE. In terms of Bayesian approach, because of the use of priors, it has several practical advantages over MMLE and JMLE such as the item difficulty parameter estimates never try to become infinite (Lord, 1986). However, the inappropriate choice of the prior distributions may lead to more problems than advantages. In this study, MMLE was used since our interest was on small-sample size and no prior distributions were assumed before the estimation.

2.1.2 Invariant item parameters

Using IRT models, the probability of a correct response to an item is only dependent on examinee's ability θ (Figure 2.1), and thus it is not related to whether the examinee is in a high-ability group or a low-ability group. According to the model, the item parameters remain the same no matter which group is tested, which is one of the most important properties of IRT. That is, the item parameters are invariant (Lord, 1980). In this way, it can provide comparable scores for examinees who are from different groups and answer different set of items.

IRT is the foundation for computerized adaptive testing (CAT) with its focus on items and item parameters rather than the whole tests. Item parameters are very important for CAT because they are

required to select the best items for each examinee and to compute the comparable examinees' scores for any selected set of items (e.g. Segall, 2005; Thissen et al., 2007). The details of CAT are in the next section.

2.2 Computerized Adaptive testing

Computerized adaptive testing (CAT) is a popular mode to deliver educational assessments. During a CAT, if an examinee answers an item incorrectly, the next item will be easier. Similarly, if an examinee answers an item correctly, the next item will be more difficult. CAT is designed in this way so that the items are purposely directed to the examinee's ability level in real-time so that most of the information about this examinee's ability can be collected (Wainer and Mislevy, 1990). Compared to a linear test, where all examinees answer the same set of items that have wide range of difficulties, CAT has the advantage to provide estimations for examinees' ability with higher precision and shorter test length, especially for examinees with extremely high or low ability levels. Many researchers have found that CAT can reduce about half of the test length in average without compromising measurement quality compared to the linear test (e.g., Brown and Weiss, 1977; McBride and Martin, 1983; Moreno et al., 1984; Wise and Kingsbury, 2000; Weiss, 1982).

2.2.1 Six components of CAT

To develop a CAT, six CAT components are required (Kingsbury and Zara, 1989; Segall, 2005), which are (1) measurement model; (2) item bank, (3) criterion to start the test; (4) the ability estimation method; (5) item selection algorithm; and (6) criterion to terminate the test. These six components are briefly introduced here:

Measurement model. The measurement model provides a metric to calibrate items and examinees. It is necessary for a CAT because every examinee answers different set of items in CAT that corresponds to their abilities. The most commonly adopted measurement models in CAT are IRT models (eg., Lord, 1977, 1980; Dodd et al., 1995; Van Rijn et al., 2002). In addition, other measurement models can also be

used in CAT, such as diagnostic classification models (DCMs; Rupp et al., 2010) and graphical modeling (e.g., Almond, 1995).

Item bank. The second crucial component for CAT is the item bank, which contains a collection of items with associated item parameters according to a measurement model (Segall, 2005). Many applications of CAT are built on unidimensional IRT (e.g., Lord, 1977, 1980; Dodd et al., 1995; Van Rijn et al., 2002). The process of item bank development is introduced in more details in the section 2.3.

Criterion to start the test. Usually, without any prior information, examinees are assumed with the average ability in the beginning of the test, and an item with medium difficulty is selected as the first item. If there is any prior information about the examinee, that information can be used to select the first item in the test to better match examinees' abilities.

Ability estimation method. The two most common scoring procedures used in CAT are maximum likelihood estimation (MLE) and Bayesian scoring (Weiss, 1982). The likelihood function for θ with I items is

$$L(\theta; \mathbf{X}) = \prod_{i=1}^{I} P_i(\theta)^{x_i} Q_i(\theta)^{1-x_i} \tag{2.5}$$

where $Q_i(\theta) = 1 - P_i(\theta)$. The estimate $\hat{\theta}$ is the value that maximizes the likelihood function in Equation (2.5). The Bayesian approach requires a prior distribution $f(\theta)$ for θ and generates posterior distribution $f(\theta|\mathbf{X})$:

$$f(\theta|\mathbf{X}) \propto f(\mathbf{X}|\theta) f(\theta) \tag{2.6}$$

Then the point estimate $\hat{\theta}$ can be the maximum a posteriori probability (MAP) estimate, which equals the mode of the posterior distribution. Also, $\hat{\theta}$ can be Expected A Posteriori (EAP) estimation, which refers to the expected value of the posterior probability distribution. Though the Bayesian approach suffers the potential bias from the prior distribution, it avoids the infinity estimations from MLE when all the responses were correct or incorrect, which commonly happens in the beginning of the tests. Thus, in practice, Bayesian approach is favored at the early stage of the test, and MLE is favored in the later stage of the test and the final estimations.

Item selection algorithm. The item selection algorithm defines the way to select the appropriate item for each examinee to maximize the test efficiency and satisfy non-statistical constraints, such as content balancing and item exposure control. Content balancing typically refers to a set of ancillary item-selection constraints based on content requirements and test blueprints. Exposure control requires to utilize the items in the item bank evenly. In this dissertation, we focus on maximizing the estimation efficiency. The most common item selection algorithm focusing on estimation efficiency for the IRT model is maximizing Fisher information method.

The idea of maximizing Fisher information in an adaptive test was proposed by Lord (1977). After administrating k items (k = 1,2,..) and obtaining the interim $\hat{\theta}^k$, either through MLE or Bayesian scoring, this algorithm calculates the Fisher information for all available items in the item bank given $\hat{\theta}$ and then selects the item that maximizes that Fisher information. The Fisher information for item i at ability level θ is:

$$I_i(\theta) = \frac{[P_i'(\theta)]^2}{P_i(\theta)Q_i(\theta)} \tag{2.7}$$

where $P_i(\theta)$ is defined in Equation (2.2) and $Q_i(\theta) = 1 - P_i(\theta)$.

Asymptotically, the variance of the maximum estimate $\hat{\theta}$ is the inverse of the Fisher information. Given this relationship, selecting item that maximizes Fisher information minimizes the variance of $\hat{\theta}$ and yields the most precise $\hat{\theta}$ for all ability levels (Lord, 1980; Weiss, 1982). Specifically, the Fisher information for 2PL model is presented below:

$$I_i(\theta) = a_i^2 P_i(\theta)Q_i(\theta) \tag{2.8}$$

Given this equation, the Fisher information is maximized when 1) the item difficulty is matched to the examinee's ability level, and 2) using the most discriminating items in the item bank (Lord, 1977). Thus, the accurate item parameter estimations play an crucial role in CAT.

Test termination criteria. This is the rule used to determine when to end the test, which is typically based on the number of administered items (fixed-length), or on the precision-level of the estimated trait parameter (variable-length) (Segall, 2005).

2.2.2 CAT in applications.

By delivering items that match to examinees' abilities, CAT can reduce the test length without sacrificing measurement precision. In addition, examinees will not receive items that are too difficult or too easy for them. As a theoretically sound and efficient test delivering mode, CAT has been successfully implemented in large-scale assessments, such as the National Council of State Boards of Nursing, the Armed Services Vocational Aptitude Battery (ASVAB), Graduate Management Admission Test (GMAT), and Graduate Record Examination (GRE). According to ETS, about 1.5 million people have taken GRE between 2017 and 2020 (ETS, 2021).

There are a limited number of studies on implementing CAT in small-scale assessments, though CAT in fact has the potential to be used for classroom assessments and is beneficial to instructions' teaching and students' learning. The primary benefits of CAT in the small-scale assessment include providing accurate ability estimations with short tests in real-time and more engaging test experiences (Martin and Lazendic, 2018). Despite the above-mentioned advantages, CAT is rarely seen in small-scale assessments like classrooms (Kaya and Kalender, 2018; Tao et al., 2008). The main reason is the complexity and rigour of the commonly adopted measurement model, IRT models, which requires large sample sizes to calibrate. However, small-scale assessments, such as classroom assessments, do not usually have that many examinees. By avoiding using the IRT model as the measurement model, Welch and Frick (1993) and Tao et al. (2008) proposed to build CAT for small-scale assessments using Sequential Probability Ratio Test (SPRT) model, which only provides the examinee's mastery result rather than a continuous score from IRT models. In addition, Chang et al. (2019) proposed to use non-parametric item selection algorithm for a CAT built on cognitive diagnostic models, which provides mastery results for multiple attributes or skills of interests. Both types of studies focus on using alternative measurement models or item selection algorithms to avoid calibrating item parameters based on IRT models. They provide a categorical estimation, pass/fail or mastery/non-mastery, which is easier to estimate but provide less information than a continuous score that IRT model provides. No study has addressed the challenges of building a CAT in small-scale assessment based on IRT models.

2.3 Item Bank Development

In CAT, examinees are administered individualized tests. These tests consist of different sets of items drawn from an item bank. The item bank is a collection of items with associated item parameters according to a selected measurement model, such as 2PL model (Segall, 2005). Generally, depending on whether the item bank is a mix of old and new items or completely new items, there are two approaches to develop an item bank. When there is a mix of old and new items, the new items can be "seeded" into the current operational tests (Thompson and Weiss, 2011). If the current operational test is a CAT and the new items are selected to administer on the fly, this process is also referred to as online calibration (e.g., Wainer and Mislevy, 1990; Ban et al., 2001; Jones and Jin, 1994; Chang and Lu, 2010). During online calibration, the responses to the new items are not used to estimate examinees' final ability estimations, but to estimate their item parameters. One additional step of putting the new items to the existing metric is required unless the plan is to build a new metric (Thompson and Weiss, 2011). When developing a completely new item bank, there are generally four steps: (1) given the test specifications or requirements, create sufficient items following minimal procedures of writing, reviewing, and revising items; (2) perform an initial pretesting for the items and conduct basic item analysis; (3) select the appropriate items and create additional items if needed; (4) evaluate the calibrated item bank via simulation studies (Prestwood et al., 1985; Flaugher and Meier, 1990). There are research focused on each step of constructing an item bank. For example, van der Linden et al. (2000) proposed to use the technique of integer programming to optimize the design of item bank. Veldkamp and van der Linden (2009) took one step further and proposed to use integer programming model for the assembly of the shadow tests in a series of CAT simulations to identify the item bank specifications. In addition, Krass and Williams (2003) focused on statistical methods and used direct marginal maximum likelihood estimation and a non-parametric approach to estimate item parameters. They found that the non-parametric approach performed better only when the examinees' ability levels followed a standard normal distribution. Furthermore, since an item bank usually contains a large number of items, it is impossible to require each examinee to answer all the items. Thus, it is usually

recommended to use incomplete calibration designs to collect responses in the pretesting step (Lord, 1980; Eggen and Verhelst, 2011; Wainer and Mislevy, 1990; Thompson and Weiss, 2011). In addition to the literature concentrating on one step of the item bank development, many researchers and practitioners documented their implementation of developing an item bank for specific subjects or constructs, such as Armed Services Vocational Aptitude Battery (ASVAB Prestwood et al., 1985),the burden of headaches (Bjorner et al., 2003), depression symptoms (Fliege et al., 2005), and pain (Petersen et al., 2016).

In this dissertation, we aimed to build a completely new CAT without any Paper & Pencils tests nor existed item bank. In particular, we focused on the pretesting step and intended to use incomplete calibration designs to collect responses and use various imputation methods to improve item parameter estimations. Before reviewing the incomplete calibration designs, the missing mechanisms are briefly introduced first. After introducing the incomplete calibration designs, methods to handle the missingness when estimating item parameters are presented in the section 2.4.

2.3.1 Missing mechanisms

Rubin (1976) presented the most common missing mechanisms: missing completely at random (MCAR), missing at random (MAR), and missing not at random (MNAR). The simple definitions for them are listed below:

- MCAR: the cause of missingness is completely unrelated to any of the observed or missing variables.
- MAR: the cause of missingness is related to the observed variables but not the missing values themselves.
- MNAR: the cause of missingness is directly related to the vales that are missing.

In statistical analyses, MCAR and MAR are considered as ignorable (Schafer, 1997), which indicates that we can ignore the missingness of the data when we impute or augment missing data. MCAR is especially preferred because any modern missing data estimation procedure provides unbiased estimates and recovery much of the power lost due to data attrition. When data is MAR, the missing values are

predictable from the observed variables, and thus the lost information can be recovered to a degree (e.g., Rhemtulla and Little, 2012). MNAR is non-ignorable, thus, to separate MAR from MNAR is important; however, when the relations between whether a value is missing and the missing value on the variable are unknown, there are neither statistical tests or visual techniques to distinguish between MAR and MNAR (e.g., McKnight et al., 2007; Van Buuren, 2012). Some researches indicate some ways of distinguishing MAR and MNAR by making subjective judgements based on logic and sound understanding of the study designs (e.g., McKnight et al., 2007).

2.3.2 Incomplete calibration designs

Incomplete calibration designs purposely collect responses for a subset of items in the item bank from examinees in a way that all the items are answered by some examinees but not all the examinees (Mislevy and Sheehan, 1989; Eggen and Verhelst, 2011; Lord, 1980). An important benefit from incomplete calibration design is that the reduced fatigue effects from shorter testing time and less burden for examinees may produce more valid data (Little and Rhemtulla, 2013). Given how to use the prior knowledge about items such as difficulty and the prior knowledge about examinees' abilities such as grades or ages, there are three types incomplete calibration designs: multistage testing designs, targeted testing designs, and random incomplete designs. Multistage testing designs shared similar ideas with multistage testing (Lord, 1971; Wainer and Mislevy, 1990; Hendrickson, 2007). The prior knowledge about items are used to assemble test forms with different difficulty levels. All examinees take the routing test form, which is of medium difficulty. Given examinees' performance on the routing test form, they will receive either a difficult or an easy test form in the next stage. Thus, no prior knowledge about examinees is required. In this dissertation, we focused on targeted testing designs and random incomplete designs, which are introduced in detail below.

Targeted Testing Designs. In the targeted testing designs, the prior knowledge about items and examinees are utilized to assemble test forms and assign different test forms to examinees. The prior knowledge about examinees is usually related to their ability levels, such as their grades or ages. Similarly, the prior

knowledge of the items is related to item difficulty, such as the judgements from content experts or teachers (Attali et al., 2014). By allocating specific group of examinees to a specific subset of items, targeted testing designs can improve the efficiency of item bank calibration (Eggen and Verhelst, 2011). That is, targeted calibration designs lead to accurate item parameter estimations with smaller sample sizes. However, we have not found any study that has specifically investigated the amount of the efficiency from targeted calibration designs. One possible reason is that the main purpose of the targeted calibration design is to reduce test time for examinees rather than improve the calibration efficiency. For example, Eggen and Verhelst (2011) demonstrated this process by calibrating 9 items with Rasch model with 2000 examinees from two sub-populations. In addition, Mislevy and Sheehan (1989) demonstrated to calibrate 10 items for 3PL model using data from three groups of examinees with sample sizes of 900, 1089, and 1159 respectively. In addition, the missing mechanism of targeted calibration design is complicated. Mislevy and Sheehan (1989) showed that MAR does not hold for the targeted calibration design when estimating item parameters, while Mislevy and Wu (1996) showed that MAR holds when estimating examinees' ability parameters. In this dissertation, two types of targeted calibration designs based on non-equivalent group with anchor test (NEAT) design (von Davier et al., 2004) and panel chained (PC) design (Graham et al., 2001; Pokropek et al., 2011) are investigated.

Targeted Non-Equivalent Group with Anchor Test (NEAT) Design. NEAT design has been widely used since at least the early 1940s (von Davier et al., 2004). The most common NEAT design involves two item sets (A and B) and another set of common items (X), which is referred to as anchor items. This design produces two test forms, and each test form is consist of the anchor item set and one of the two item sets. Two groups of examinees take two test forms respectively.

NEAT design can also be generalized to more than two test forms. For example, Table 2.1 showed a 5-form NEAT design under the idea of the targeted testing design. The item bank is divided into 6 item sets including one anchor item set X. The other five sets of items are ranked from the easiest to the hardest, and they are assigned to examinees with the lowest to the highest ability levels. The anchor items can be selected as a representation for the whole item bank (Zhang, 2020).

Table 2.1: Targeted 5-Form NEAT Design

Test Form	X (Anchor Items)	A (Easiest)	B	C	D	E (Hardest)	Test-takers
1	✓					✓	G_1 (Highest ability)
2	✓				✓		G_2
3	✓			✓			G_3
4	✓		✓				G_4
5	✓	✓					G_5 (Lowest ability)

Targeted Panel chained (PC) design. Panel chained design has been used in educational assessments (Samsudin et al., 2020; Lord, 1980) and in longitudinal analysis to reduce the cost and minimize examinees' attrition over time (Graham et al., 2001; Pokropek et al., 2011). Table 2.2 presents a PC design for five groups of examinees with five test forms that are constructed from four item sets. Under the idea of targeted testing design, each group of examinees receive two adjacent item sets that are close to their ability levels except for the examinees with the highest or the lowest ability level. Examinees with the highest ability will answer the hardest item set, and examinees with the lowest ability will answer the easiest item set.

Table 2.2: Targeted 5-Form Panel Chained Design

Test Form	A (Easiest)	B	C	D (Hardest)	Test-takers
1				✓	G_1 (Highest ability)
2			✓	✓	G_2
3		✓	✓		G_3
4	✓	✓			G_4
5	✓				G_5 (Lowest ability)

Random incomplete designs. In the random incomplete designs, items are randomly assigned to different test forms in a way that test forms are parallel in content and difficulty with same number of items. No prior knowledge of examinees are used to decide which test form to administer to which examinee. Random incomplete designs have been used in large-scale assessments. For example, in the National Assess-

ment of Educational Progress (NAEP), examinees are randomly assigned a subset of the items (Donahue, 1999). When the missingness is well planned and the process of causing missingness is under control, it generates MCAR data, so there are no serious concerns about statistical analysis for the incomplete data (Pokropek et al., 2011; Rhemtulla and Little, 2012). To achieve the MCAR, Rhemtulla and Little (2012) recommended to break up any sub-scales and make sure items between sets are maximally correlated, and Rhemtulla and Hancock (2016) suggested to assign test forms to examinees at the individual level to ensure having an MCAR type of data. In this study, the NEAT design and PC design introduced above were also planned to generate MCAR data without using the prior knowledge about items and examinees to assemble the test forms or decide which group of examinees take which test form.

Random NEAT design. Table 2.3 presents the random NEAT design. Items are divided into 6 sets including an anchor item set X. The remaining items are assigned to five sets randomly and the item sets have similarly average difficulty levels and discrimination levels. Each test form is composed of the anchor item set and one of the five item sets, and each test form is administered to examinees randomly. The prior knowledge about examinees is not used during the administration of the test.

Random PC design. Table 2.4 presents the random PC design, which has the similar structure to the targeted PC design. The difference is that items are divided into four sets randomly in random PC design without using any prior knowledge about items. Each test form is composed of either one or two item sets, and each test form is administered to examinees randomly.

Table 2.3: Random 5-Form NEAT Design

Test Form	X (Anchor Items)	A (Parallel)	B	C	D	E (Parallel)	Test-takers
1	✓					✓	Randomly selected
2	✓				✓		Randomly selected
3	✓			✓			Randomly selected
4	✓		✓				Randomly selected
5	✓	✓					Randomly selected

Table 2.4: Random 5-Form Panel Chained Design

Test Form	A (Parallel)	B	C	D (Parallel)	Test-takers
1				✓	Randomly selected
2			✓	✓	Randomly selected
3		✓	✓		Randomly selected
4	✓	✓			Randomly selected
5	✓				Randomly selected

2.3.3 Sequential optimal designs

To improve the calibration efficiency, many studies have investigated the idea of optimal designs aiming to select best examinees to minimize the variance of the item parameters of interest. In other words, the objective is to select examinees with certain ability levels that maximize some functions of Fisher information of the item parameters. In the literature of optimal designs, the common optimal criteria are D-optimality, E-optimality, and A-optimality, which maximizes the determinant, maximizes the minimum of the eigenvalue, and minimizes the trace of the inverse of the information matrix, respectively. All of these criteria have been studied in the context of item calibration (e.g., Buyske et al., 1998; Lu, 2014; van der Linden and Ren, 2015; Chang and Lu, 2010). For example, Chang and Lu (2010) adopted the two-point design (Kalish and Rosenberger, 1978; Mathew and Sinha, 2001) which used the D-optimal criterion for a logistic regression model. This two-point design selects equal number of examinees at $F_{17.6}$ and $F_{82.4}$, where F_p denotes the p^{th} percentile of a logistic distribution. That means the design points should be evenly put at $\theta_1 = (-1.5434/a) + b$ and $\theta_2 = (1.5434/a) + b$ if using 2PL model. However, the best design points are dependent on both person and item parameters, which are not known in many cases.

To address the issue of lack of knowledge about examinees and items, sequential optimal designs have been investigated in a few studies (Ali and Chang, 2014; Kingsbury, 2009; Zhu, 2006). Basically, this type of design assumes a well-calibrated item bank and has two stages. In the first stage, the pretest items are randomly administered to examinees at the seeding location of the test, where the rest of the test

is administered based on certain adaptive algorithm using the established operational item bank. After certain amount of the random trials, the initial estimations for pretest item parameters are obtained. In the second stage, the pretest items are administered to examinees based on optimal designs using the initial estimations of item parameters and the interim estimations for examinees' ability. Kingsbury (2009) used 500 responses per pertest item to calibrate the pretest items using 1PL model in the second stage. In addition, Zhu (2006) used at least 300 examinees to calibrate item parameters using the 2PL model with known initial item parameter estimations. Only two publications about sequential optimal designs for item bank calibration have not assumed an well-established item bank (Makransky and Glas, 2014; Fink et al., 2018). Specifically, Makransky and Glas (2014) proposed a continuous updating calibration strategy where items are administered randomly up to a given number of examinees, and these items are then calibrated and become eligible as operational items to the remaining examinees. However, at least 1000 examinees were required to achieve an ability precision level similarly to the results with a completely known item bank. Fink et al. (2018) included equating process into the continuous calibration strategy to maintain the scale in the item pool across test cycles with the consideration for the possibility of item parameter drift for linking items. Their simulation study showed that 450 examinees were required for 2PL model after the 1st stage of obtaining initial item parameter estimations.

2.4 Missing Data Techniques

There are many methods for dealing with missing data, and they fall into two categories in general: (1) use the observed data only and (2) impute the missing data. Among methods that only use the observed data, full information maximum likelihood (Anderson, 1957; Schafer and Graham, 2002; Lord, 1955; Beale and Little, 1975; Dempster et al., 1977) is claimed to be state-of-art method by previous studies (Schafer, 1997). Under the umbrella of the imputation methods, there are two sub-categories: single imputation and multiple imputation. These three methods are reviewed in this section.

2.4.1 Full information maximum likelihood method

Full information maximum likelihood method (FIML) uses the maximum likelihood algorithm with all available data to estimate parameters. Specifically, examinees with missing data are only ignored for the parameters where they have missing responses, and are included to estimate other parameters where they have observed responses. That is to say, if i^{th} examinee's response for item j is missing, this missing response would be ignored when estimating item j's parameters or estimating examinee i's parameters, and i^{th} examinee's responses for other items are kept to estimate other parameters. In this section, the statistical details of FIML with missing data in the context of 2PL model is presented. Let $P_j(\theta_i)$ represents the probability of the correct answer from examinee i with latent ability θ to item j, and $Q_j(\theta_i) = 1 - P_j(\theta_i)$.

Firstly, the log-likelihood function for 2PL model for examinee i can be presented as

$$logL_i = logL(\theta_i, a_j, b_j | X_i) = \sum_{j=1}^{J_i} X_{ij} P_j(\theta_i) + (1 - X_{ij}) Q_j(\theta_i) \tag{2.9}$$

where J_i is the number of items answered by examinee i, $J_i \leq J$, and J is the total number of items in the test or item bank. From Equation (2.9), the log-likelihood function for examinee i with missing values only depends on the observed J_i responses and ignores the parameters with missing values, which implies that the log-likelihood functions are different for each examinee with different missing responses in the data. The log-likelihood over all examinees is the summation of log-likelihood functions as follows:

$$logL = \sum_{i=1}^{I} logL_i \tag{2.10}$$

where I is the total number of examinees. The parameters of interests are then identified by maximizing the likelihood function in Equation (2.10).

As the default missing data technique in most IRT software programs (Edwards and Finch, 2018), FIML is used in various conditions. Previous research showed that FIML tends to yield unbiased parameter estimates when the missing mechanism is either MCAR or MAR in the context of latent variables for

continuous data with the multivariate normal distribution (Graham et al., 2007; Enders and Bandalos, 2001). However, with categorical data, Enders (2001) has shown that the standard errors of the estimates can be negatively biased.

2.4.2 Single imputation methods

Generally, imputation techniques replace missing values with the most plausible values, which can be generated or estimated using information from the observed values of the variable itself and other variables. Single imputation methods produce a single replacement value for every missing value of the variable. Researchers have studied many single imputation methods in the field of educational assessments including treating the missing responses as incorrect, fractionally correct, overall mean imputation, item mean imputation, person mean imputation, corrected item mean substitution imputation, response function imputation, model based imputation, and two-way imputation (e.g., Bernaards and Sijtsma, 1999; Huisman, 1999; Bernaards and Sijtsma, 2000; Sijtsma and Van der Ark, 2003; Finch, 2008). Among these methods, many studies agreed that treating missing responses as incorrect led to biased estimates (e.g., Lord, 1974, 1982; De Ayala et al., 2001; Finch, 2008), because its strong assumption that missing response is due to not knowing the correct response is often not true. There are many different causes for skipped or missed responses, such as lack of motivation or anxiety. Thus, treating missing responses as incorrect is not recommended. In addition, many studies have found that no one single imputation technique is superior to others generally (e.g., Finch, 2008; Smits et al., 2002; Bernaards and Sijtsma, 2000). However, Sijtsma and Van der Ark (2003) showed that two-way imputation is robust with smaller sample sizes, which is the scenario of interest in this study. Thus, the two-way imputation method is introduced here.

Bernaards and Sijtsma (2000) proposed two-way (TW) imputation after their personal communication with D. B. Rubin. The idea is based on the two-way ANOVA models. In particular, the imputed value is a combination of the item effect, the person effect, and the overall effect. Before introducing TW, a few new notations are introduced first. Let I_j be the number of observed responses from item j, $I_j < I$, and their indices are collected in $A_{(j)}$. Similarly, J_i is the number of observed responses from examinee i,

$J_i < J$, and their indices are collected in $A_{(i)}$. In addition, $OM = \frac{1}{n}\sum_{i\in A_{(i)}}\sum_{j\in A_{(j)}} X_{ij}$, which is the overall mean across all observed responses, $IM_j = \frac{1}{I_j}\sum_{i\in A_{(j)}} X_{ij}$ and $PM_i = \frac{1}{J_i}\sum_{j\in A_{(i)}} X_{ij}$ are the average score for item j and the average score for examinee i over all the observed responses respectively. Using the TW, the missing response of item j from examinee j is computed as

$$TW_{ij} = PM_i + IM_j - OM. \tag{2.11}$$

where TW_{ij} is not a binary value, and thus the final binary response is drawn from a Bernoulli distribution with TW_{ij} as the parameter.

2.4.3 Multiple imputation

Different from single imputation, multiple imputation (MI; e.g., Schafer, 1997) generates multiple replacement values for every missing value and thus yield multiple complete data sets before analyses. In this way, MI accounts for the inherent uncertainty in sampling from a population by introducing a degree of randomness to the imputations and has shown to provide greater accuracy than simple imputation methods (Schafer and Graham, 2002). Generally, MI consists of three phases: (1) the imputation phase, (2) the analysis phase, and (3) the pooling phase. The imputation phase requires an assumption about the probability model underlying the data that incorporates information from all of the measured variables. For binary response data, the theoretically suitable multinomial distribution is impractical because the saturated log-linear model is severely degraded with just a few variables (Schafer, 1997). Finch (2008) pointed out that the multinomial model was impossible for more than 10 items. However, researchers have shown that the normal-based probability model yield acceptable results (Schafer, 1997; Finch, 2008). Given the assumption about the probability model, the imputed values can be drawn from it with multiple trials and thus multiple complete data sets are created. With the results from the imputation phase, the analysis phase analyzes every created complete data set and results in several sets of parameter estimates and

standard errors. Lastly, the pooling phase generates a single point estimate using the multiple estimates from the analysis phase.

Multivariate Imputation by Chained Equations (MICE). Recently, multivariate imputation by chained equations (MICE) have drawn attentions of researchers in the educational assessment field (Edwards and Finch, 2018; Xiao and Bulut, 2020). MICE, which is also known as fully conditional specification or sequential regression multiple imputation, is based on multiple imputation and relies on separate marginal distributions for each variable in the data set, and thus is more flexible than multiple imputation (Raghunathan et al., 2001; Van Buuren and Groothuis-Oudshoorn, 2011; Van Buuren and Oudshoorn, 1999). MICE builds on the assumption that missing data are MAR. Azur et al. (2011) illustrated MICE with seven general steps of the procedure:

Step 1: A simple imputation, such as mean imputation, is performed for every missing value in the data set.

Step 2: The imputed values from step 1 are set back to missing for the 1^{st} variable that has missing values.

Step 3: The 1^{st} variable is treated as the dependent variable and regressed on the other variables. This regression model can be logistic regression for the binary variable.

Step 4: The missing values for the 1^{st} variable are then replaced with the predictions from the regression model in step 3.

Step 5: Steps 2-4 are repeated for each variable that has missing values. One iteration is done when missing values in all the variables are imputed once.

Step 6: Multiple iterations are repeated, and the imputations are updated at each iteration.

Step 7: The entire imputation process (step 1 - 6) is repeated to generate multiple imputed data sets.

In addition to using parametric regression model in step 3, other non-parametric models, such as classification and regression trees (CART) and random forest (RF), can be used to predict the missing values.

CART constructs a predictive model in the form of decision trees by splitting the sample into several sub-groups using a set of predictors in multiple steps. In the first step, CART searches all possible ways to split the samples based on some predictors and select the split that minimizes the homogeneity in the resulting sub-groups with respect to the predicted variable. In the next steps, CART repeats this process and splits the sup-groups until further splits do not have sufficient gains in sub-group's homogeneity in terms of the predicted variable. RF is an extension of CART. Rather than just one tree in CART, RF creates a number of decision trees using bootstrap resampling of both observations and predictors from the original data. CART and RF are non-parametric methods and do not depend on any assumptions. Edwards and Finch (2018) investigated MI, MICE, MICE(CART), and MICE(RF), which yield data that have better estimation for both item difficulty and item discrimination parameters than FIML. In addition, Xiao and Bulut (2020) found that the MICE and RFI have similar estimation for person parameters and FIML lead to best estimations under most conditions. Both studies found that MICE(CART) and MICE(RF) led to very similar results. Xiao and Bulut (2020) recommended to use MICE-CART especially for large volumes of data because it has lower computational cost.

2.5 Matrix Completion

In the language of machine learning, to recovery a matrix with incomplete sampling of its entries is a matrix completion problem. Matrix completion has been applied in a wide range of applications including collaborative filtering system identification, sensor localization, and rank aggregation (Davenport et al., 2014). One famous application is known as "Netflix problem". In this setting, an incomplete matrix is observed where each entry represents a rating for a particular user on a particular movie. In this matrix, many entries are missing, because any user only rates a subset of all available movies on Netflix. The goal is to predict the missing ratings according to observed ratings so that Netflix can provide personalized recommendations. Candès and Recht (2009) have shown that a $n \times n$ matrix with rank r can be recovered from a small set of observed entries under suitable conditions: (1) at least one observation per row and one observation per column; (2) the singular vectors need to be sufficiently spread to minimize the number of

observations needed to recover a low-rank matrix. In addition, Candès and Recht (2009) showed it can be done by solving a simple convex optimization problem.

In the educational assessment, an examinee's responses are dependent on their achievement or ability and the items' characteristics. Especially, in the unidimensional item response models, only one ability is required for responses to the test to meet the assumption of local independence. Though many studies have argued about multiple abilities are required to respond to items, the dimensionalities are still small. For example, the Program for International Student Assessment (PISA) in 2003 measured mathematics, reading, science and problem solving with a main focus on mathematics. The mathematics domain is differentiated into four sub-dimensions including space and shape, change and relationship, uncertainty, and quantity (OECD, 2003; Schulz, 2005). That is, there are only 7 dimensions of abilities while there are thousands of students and testing questions. Thus, to recovery an incomplete matrix from an educational assessment is a low rank problem, and matrix completion is justifiable.

Though the promise of applying matrix completion to educational assessments, there are two unique challenges. Firstly, most matrix completion methods are developed for continuous data (e.g., Candès and Tao, 2010; Gross, 2011; Recht et al., 2010; Candes and Plan, 2011), while the educational assessment data are binary. A few studies investigated this issue. The most popular one is the 1-bit matrix completion (Davenport et al., 2014) for binary data assuming a underlying continuous low-rank matrix. Their purpose is to recover the underlying low-rank matrix given the observed binary response matrix. Thus, how to produce a recovered binary response is still unsolved. Secondly, most matrix completion methods have an assumption of MAR. In particular, 1-bit matrix completion assumes uniformly sampling, which means the probability of the entry (i, j) being observed is the number of observed cases over the number of total cases. However, if using targeted testing designs, sequential optimal designs, or adaptive designs, the missing patterns are complex and may not satisfy this assumption. Cai and Zhou (2013) proposed structured matrix completion which is designed for non-uniform sampling cases. However, structured matrix completion is for continuous data.

In this section, both 1-bit matrix completion and structured matrix completion are briefly reviewed. The notations are introduced first. Let the matrix $\mathbf{X}$ be the observed response matrix with I rows and J columns, where each row represents an examinee and each column represents an item. The observed entries are denoted as X_{ij} and $(i,j) \in \Omega$, where Ω is the complete set of N entries in $\mathbf{X}$ with observed responses, $N < I \times J$. Thus, the missing entries are denoted as X_{ij} and $(i,j) \notin \Omega$. The full matrix including the actual values for the missing responses is denoted as $\mathbf{X}_{full}$.

2.5.1 1-bit matrix completion

The name of "1-bit matrix completion" is from the field of 1-bit compressed sensing theories, where only the signs of the measurements are observed. The matrix completion problem is more challenging when the observed matrix is binary (Davenport et al., 2014). Though it seems natural to treat the binary observations, 0 and 1, as continuous and then dichotomize the imputed matrix $\hat{\mathbf{X}}$ into two categories, it is actually impossible to recover the complete matrix. For example, suppose the binary observations $\mathbf{X}_{ij}$ are from a latent continuous matrix $\mathbf{M}_{ij}$, where $(i,j) \in \Omega$. $\mathbf{X}_{ij}$ is 1 if $\mathbf{X}_{ij}$ is positive and $\mathbf{X}_{ij}$ is 0 otherwise. Even for a matrix $\mathbf{M} = uv^T$ with rank 1, where u and v are column vectors, one can freely modify the magnitude of the elements of u and v without changing the sign patterns of $\mathbf{M}$. That is, there is no unique solution. In addition, the sign function is not convex, and thus it is impossible to locate the best solution via optimization algorithms.

Davenport et al. (2014) proposed to add noises to the observations and thus solved the 1-bit matrix completion problem. Rather than observed a continuous matrix $\mathbf{M}$, a binary matrix $\mathbf{X}$ is observed, and $\mathbf{X}$ depends on $\mathbf{M}$ in a way as:

$$X_{ij} = \begin{cases} +1 & \text{with probability of } f(M_{ij}) \\ -1 & \text{with probability of } 1 - f(M_{ij}) \end{cases}, \quad (i,j) \in \Omega \tag{2.12}$$

where $f(x)$ can be logistic regression model, $f(x) = \frac{e^x}{1+e^x}$, or probit regression model, $f(x) = 1 - \Phi(-x/\sigma)$, where Φ is the cumulative distribution function of a standard Gaussian.

To approximate either $\mathbf{M}$ or $f(\mathbf{M})$, Davenport et al. (2014) proposed to maximize the log-likelihood function subject to a set of convex constraints, which can be formulated as

$$\hat{M} = \arg\max_{\mathbf{Y}} \mathcal{L}_{\Omega,\mathbf{X}}(\mathbf{Y}) \quad \text{subject to } ||Y||_* \leq \alpha\sqrt{rIJ}. \tag{2.13}$$

where $||Y||_* \leq \alpha\sqrt{rIJ}$ is a relaxation of the condition where each entry of $\mathbf{M}$ is bounded in magnitude by α and rank$(\mathbf{M}) \leq r$. In addition, $\mathcal{L}_{\Omega,\mathbf{X}}$ is the log of the likelihood function, which can be expressed as:

$$\mathcal{L}_{\Omega,\mathbf{X}}(\mathbf{Y}) = \sum_{(i,j)\in\Omega} (\mathbf{1}_{X_{ij}=1} log(f(Y_{ij})) + \mathbf{1}_{X_{ij}=1} log(1 - f(Y_{ij}))) \tag{2.14}$$

Thus, the objective function for binary data can be reformulated as:

$$\text{minimize} - \mathcal{L}_{\Omega,\mathbf{X}} + \lambda||Y||_* \tag{2.15}$$

where the first term aims to achieve the maximum likelihood, and the second term aims to add penalty to high rank matrix. Note that $||\mathbf{Y}||_*$ denotes the nuclear norm of $\mathbf{Y}$, which is the sum of the singular values, and λ is user-defined constant. The algorithm of using proximal gradient descent and fast truncated singular value decomposition to find the solution for Equation (2.15) are shown in the Appendix A.

2.5.2 Structured matrix completion

Cai et al. (2016) investigated conditions where a subset of rows and a subset of the columns of an approximately low rank matrix are observed. With certain row and column permutations, the observed matrix

$\mathbf{M}$ can be re-arranged into a block form:

$$\begin{array}{cc} m_2 & J-m_2 \\ \begin{pmatrix} \mathbf{M}_{(11)} & \mathbf{M}_{(12)} \\ \mathbf{M}_{(21)} & \mathbf{M}_{(22)} \end{pmatrix} & \begin{array}{c} m_1 \\ I-m_1 \end{array} \end{array} \tag{2.16}$$

where $\mathbf{M}_{(11)} \in \mathbb{R}^{m_1 \times m_2}$, $\mathbf{M}_{(12)} \in \mathbb{R}^{m_1 \times (J-m_2)}$, and $\mathbf{M}_{(21)} \in \mathbb{R}^{(I-m_1)\times m_2}$ are observed, and $\mathbf{M}_{(22)} \in \mathbb{R}^{(I-m_1)\times(J-m_2)}$ is missing. Cai et al. (2016) used Schur complement to recover $\mathbf{M}_{(22)}$ when $\mathbf{M}$ is exactly low rank as shown below:

$$\mathbf{M}_{(22)} = \mathbf{M}_{(21)}\mathbf{V}\mathbf{\Sigma}^{-1}\mathbf{U}^T\mathbf{M}_{(12)}, \tag{2.17}$$

where the SVD of $\mathbf{M}_{(11)}$ is $\mathbf{M}_{(11)} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$, and $\mathbf{U} \in \mathbb{R}^{I\times r}$, $\mathbf{\Sigma} \in \mathbb{R}^{r\times r}$,and $\mathbf{V} \in \mathbb{R}^{J\times r}$. When $\mathbf{M}$ is approximately low rank, the authors proposed to transform the observed blocks using the singular value decomposition (SVD) and remove some unimportant rows and columns using threshold rules before applying Schur complement. When the rank r is unknown, Cai et al. (2016) proposed to firstly estimate $\hat{r}$, and recovery $\mathbf{M}_{(22)}$ following Equation (2.17). Let $\mathbf{M}_{\cdot 1} = (\mathbf{M}_{11}^T, \mathbf{M}_{21}^T)^T$ and $\mathbf{M}_{1\cdot} = (\mathbf{M}_{11}, \mathbf{M}_{12})$, the process to estimated $\hat{r}$ can be summarized as below:

1. Calculate the SVD for $\mathbf{M}_{\cdot 1}$ and $\mathbf{M}_{1\cdot}$, $\mathbf{M}_{\cdot 1} = \mathbf{U}^{(1)}\mathbf{\Sigma}^{(1)}\mathbf{V}^{(1)T}$, $\mathbf{M}_{1\cdot} = \mathbf{U}^{(2)}\mathbf{\Sigma}^{(2)}\mathbf{V}^{(2)T}$.

2. Move the significant factors of $\mathbf{M}_{\cdot 1}$ and $\mathbf{M}_{1\cdot}$ to the front by rotating the columns of $\mathbf{M}_{\cdot 1}$ and rows of $\mathbf{M}_{1\cdot}$ based on SVD. That is, compute $\mathbf{Z}_{11}$, $\mathbf{Z}_{12}$, and $\mathbf{Z}_{21}$ by:

$$\mathbf{Z}_{11} = \mathbf{U}^{(2)T}\mathbf{M}_{11}\mathbf{V}^{(1)}, \quad \mathbf{Z}_{12} = \mathbf{U}^{(2)T}\mathbf{M}_{12}, \quad \mathbf{Z}_{12} = \mathbf{M}_{21}\mathbf{V}^{(1)}.$$

3. Find the largest $\hat{r}$ to satisfy that $\mathbf{Z}_{11}[1:\hat{r}, 1:\hat{r}]$, which is the first $\hat{r}$ columns and $\hat{r}$ rows of $\mathbf{Z}_{11}$, is nonsingular and the determinant of $\mathbf{Z}_{21}[1:\hat{r}, 1:\hat{r}]\mathbf{Z}_{11}^{-1}[1:\hat{r}, 1:\hat{r}]$ is smaller than 2. This is suggested from both theoretical analysis and empirical analysis.

Chapter 3

Study 1: Adapt Matrix Completion Algorithms in Incomplete Calibration Designs

Given the characteristics of the above-mentioned four incomplete calibration designs: targeted NEAT, random NEAT, targeted PC, and random PC designs, there are some problems if directly use 1-bit matrix completion and structured matrix completion to impute the missing responses. In particular, 1-bit matrix completion has the uniformly missing assumption, which is not satisfied by the four incomplete designs of interest. In addition, structured matrix completion requires continuous input values, provides continuous imputed values, and requires a specific missing structures, which are not satisfied in the four incomplete calibration designs. To solve these problems, two research questions were investigated in this study:

RQ1 How to deal with the non-uniformly missingness from incomplete calibration designs for 1-bit matrix completion?

RQ2 How to make the structured matrix completion applicable to the incomplete calibration designs and binary response matrices?

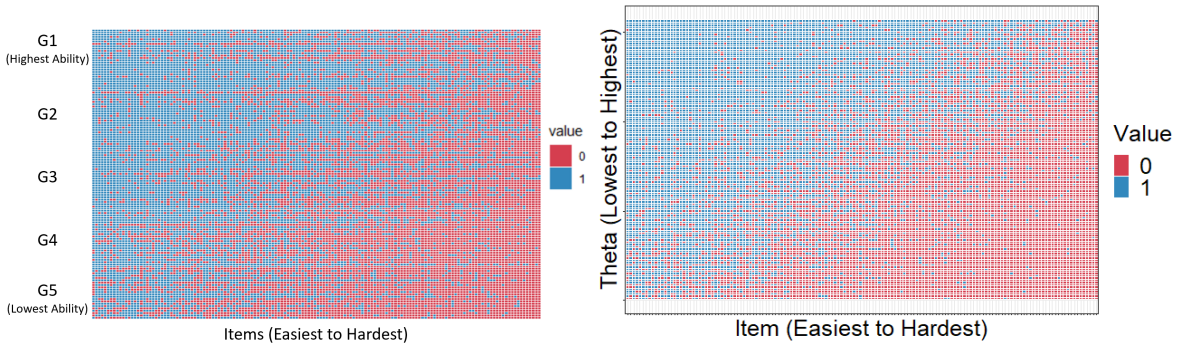
In the next sections, the methods of adapted 1-bit matrix completion and structured matrix completion to answer these two research questions were first presented. Then, their performances were evaluated in a simulation study and presented next.

3.1 RQ1: Adaptations for 1-Bit Matrix Completion.

As reviewed in section 2.5.1, 1-bit matrix completion assumes that the missingness is uniform. In other words, the probability of missing a value is the same across all the entries in the matrix. However, the four incomplete calibration designs do not meet this assumption. For example, in the targeted testing design, examinees of high abilities answer difficult items, and examinees of low abilities answer easy items. Thus, the probability of a missing response to an easy item from a high-ability examinee is much higher than a missing response to a difficult item from the same examinee. Given this particular design, the assumption of uniformly missing is not satisfied, which jeopardizes the performance of 1-bit matrix completion.

To demonstrate the concern of violating the uniform missing assumption, a pilot simulation was conducted to check the imputed response patterns using 1-bit matrix completion in the four incomplete calibration designs. Firstly, a full response matrix was simulated where all the items were answered by all the examinees. Figure 3.1a presents this full response matrix after ordering the items by their difficulties on the x-axis and ordering examinees by their ability groups on the y-axis. This figure shows a gradual transition from the upper left corner with the majority of correct responses (blue mosaics) to the lower right corner with the majority of incorrect responses (red mosaics), which was expected. This pattern is even more obvious if ordering examinees by their individual abilities (Figure 3.1b). In addition, without ordering items by difficulty on the x-axis, even ordering examinees by their ability groups, the gradual transition was disappeared (Figure 3.1c).

Secondly, the incomplete response matrices from the targeted NEAT and targeted PC designs were generated using the full response matrix in Figure 3.1a. Given their specific designs (Tables 2.1 and 2.2), the responses at the planned missing locations were masked out as missing. The response patterns of the targeted NEAT and targeted PC designs are shown in Figures 3.2a and 3.2b, respectively. In the observed



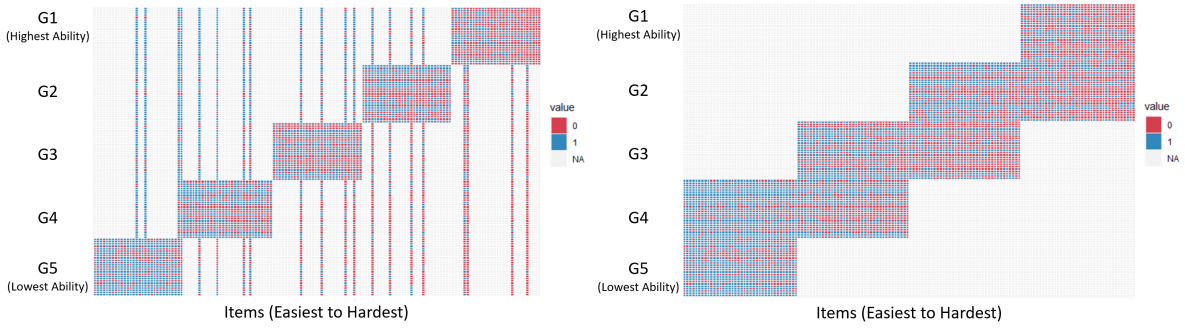
(a) Items Ordered by Difficulty

(b) Both Items and Examinees are Ordered

(c) Items Not Ordered

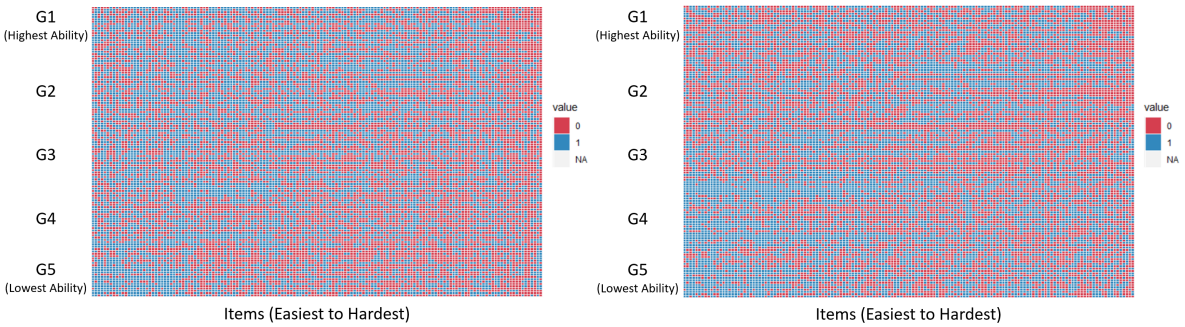
Figure 3.1: Full Response Matrices

matrix under targeted NEAT designs, the observed block for the group 5, where examinees had the lowest abilities and answered the easiest test form, has more blue mosaics than the observed block for group 1, where examinees had the highest abilities and answered the hardest test form. That is to say, the examinees of the lowest ability group might have higher raw scores than the examinees of the highest ability group in the targeted NEAT designs. The similar pattern is also shown in targeted PC designs. The imputed response matrices using 1-bit matrix completion for targeted NEAT and targeted PC designs without considering the violated uniformly sampling assumption are presented in Figures 3.2c and 3.2d. In these two figures, the blue and red mosaics are evenly distributed across items and examinees. Thus, if ignoring the violation of uniform sampling assumption, the transitioning pattern shown in Figure 3.1a can not be recovered and thus the performance of the 1-bit matrix completion is not satisfying for the targeted NEAT and PC designs.



(a) Targeted NEAT Design

(b) Targeted PC Design

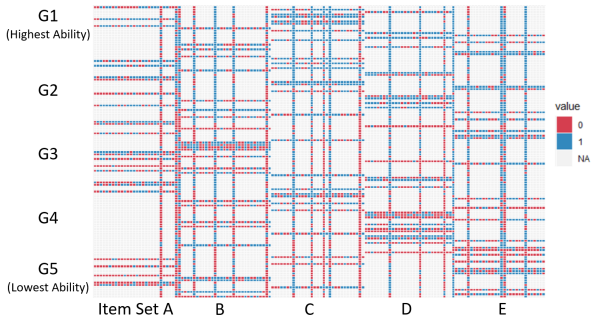


(c) Recovered Response Matrix from Targeted NEAT Design

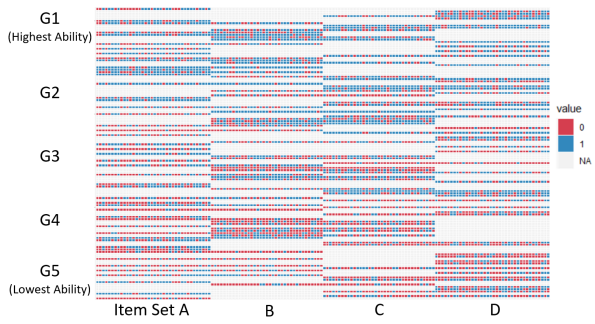
(d) Recovered Response Matrix from Targeted PC Design

Figure 3.2: Observed and Imputed Response Matrices From Targeted Calibration Designs

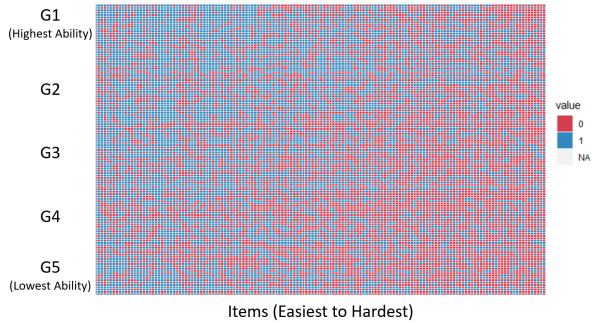
Thirdly, the incomplete response matrices from the random NEAT and random PC designs were generated using the full response matrix in Figure 3.1c and are presented in Figures 3.3a and 3.3b respectively. Using the random calibration designs, the observed matrices did not show the same pattern as the targeted designs. Using 1-bit matrix completion, the corresponding imputed matrices are presented in Figures 3.3c and 3.3d. In these two figures, the items are ordered from the easiest to the hardest. If the original full response matrix is well-recovered, the imputed response matrices should have the gradual transition shown in Figure 3.1a where high-ability groups of examinees answered easy items correctly and low-ability groups of examines answer difficult items incorrectly. However, the imputed matrices show a pattern where the blue and red mosaics are evenly distributed. Thus, the non-uniformly sampling is not satisfied in random NEAT and random PC designs either, and the 1-bit matrix completion can not be directly implemented.



(a) Random NEAT Design



(b) Random PC Design



(c) Recovered Response Matrix from Random NEAT Design

(d) Recovered Response Matrix from Random PC Design

Figure 3.3: Observed and Imputed Response Matrices From Random Calibration Designs

Using 1-bit matrix completion, the imputed values $\hat{\mathbf{M}}$ in Equation (2.13) are continuous, and they are dichotomized using zero as the cut-off point in common practice. That is, if the imputed value $\hat{\mathbf{M}}_{ij}$ is larger than zero, the final binary imputed value is 1. Otherwise, if the imputed value $\hat{\mathbf{M}}_{ij}$ is larger than zero, the final binary imputed value is 0. As shown in the pilot simulation study, the performances of the 1-bit matrix completion are jeopardized due to the violation of the uniform sampling assumption for all four incomplete calibration designs. To take into account of the characteristics of the incomplete calibration designs, rather than using a universal cut-off point, we proposed to use different cut-off points for imputed values. This idea was inspired from the two-way imputation method (Bernaards and Sijtsma, 2000).

Let x_{ij} represent the continuous imputed value for the response of the j^{th} examinee and the i^{th} item. The proposed cut-off point for each imputed value x_{ij} was computed in two steps. In the first step, the imputed value was transformed into the range between 0 and 1 using the logistic transformation. This transformed value is denoted as x^*_{ij} and $x^*_{ij} = \frac{e^x_{ij}}{1+e^x_{ij}}$, where $x^*_{ij} \in (0, 1)$. In the second step, the x^*_{ij} was compared to a threshold TW_{ij} and then dichotomized. If using PC designs, TW_{ij} was computed following Equation (2.11) with an extra step of linear transformation to make TW_{ij} to be in the range of 0 and 1. Specifically, $TW_{ij} = PM_j + IM_i - OM$, where PM_j was the observed person mean for the j^{th} examinee, IM_i was the observed item mean for the i^{th} item, and OM was the observed overall mean. If x^*_{ij} was larger than $1 - TW_{ij}$, the imputed value was replaced as 1, and 0 otherwise. In this way, effects from both person and item were considered when dichotomizing the continuous imputed values. If using NEAT designs,the person mean PM_j was computed only using the anchor items rather than all the items in the test form. Because anchor items covered various difficulty levels and were answered by all the examinees, the person means from the anchor items can better reflect the relative person effects across examinees. On the contrary, especially in the targeted designs, because examinees only answered items that matched their abilities, their person means might be similar and close to 0.5 across examinees, though their true abilities were different. This dichotomizing method was denoted as *p_tw* and was summarized in Table 3.1. The traditional way of using zero as the cut-off point was denoted as *Sign*.

Table 3.1: Dichotomizing Methods for An Imputed Value x_{ij}

Method	Threshold	Rule	Binary output
Sign (Common Practice)	0	$x_{ij} > 0$	1
		$x_{ij} \leq 0$	0
p_tw (PC Designs)	$TW_{ij} = PM_j^a + IM_i^b - OM^c$	$x_{ij}^{*e} > (1 - TW_{ij}^{*d})$	1
		$x_{ij}^{*} \leq (1 - TW_{ij}^{*})$	0
p_tw (NEAT Designs)	$TW_{ij} = PM_{aj}^f + IM_i - OM$	$x_{ij}^{*} > (1 - TW_{ij}^{*})$	1
		$x_{ij}^{*} \leq (1 - TW_{ij}^{*})$	0

a. PM_j is the observed person mean.
b. IM_i is the observed item mean.
c. OM is the observed overall mean.
d. TW_{ij}^* is constrained to the range of 0 to 1, $TW_{ij}^* = \frac{TW_{ij} - min(TW_{ij})}{max(TW_{ij}) - min(TW_{ij})}$.
e. x_{ij}^* was the logistic transformation for x_{ij} $x^* = \frac{e_{ij}^x}{1+e_{ij}^x}$
f. PM_{aj} is the observed person mean on anchor items only.

3.2 RQ2: Adaptations for Structured Matrix Completion.

In order to implement the structured matrix completion for our scenario, there are three main adaptations. The first one is to arrange the observed matrices from incomplete calibration designs to meet the specific structural missing pattern required in the structured matrix completion. The second adaptation is to convert binary responses into continuous inputs for structured matrix completion. The last one is to dichotomize the continuous imputed values to binary responses. These three adaptations are introduced sequentially.

3.2.1 Arrange the observed matrices from incomplete calibration designs.

The structured matrix completion requires that some columns and some rows of the incomplete matrix are completely observed; however, the four incomplete calibration designs do not satisfy this requirement. To solve this issue, we proposed to impute the missing values in multiple steps. The main principle is to permute and rearrange the observed matrix so that a subset of the observed matrix would meet this

specific requirement of structured matrix completion. The detailed permutation and rearrangement in each step for the four incomplete designs are described below.

Targeted NEAT Design. For the targeted NEAT design, the observed matrix was fully imputed after five steps as shown in Figure 3.4. In Figure 3.4, the blue block represents the observed responses, and the white block represents the missing responses. In step 1, the columns of anchor items, which were answered by all examinees, were permutated to the left of the response matrix (denoted as $\mathbf{M}_{11}$ and $\mathbf{M}_{21}$ in Figure 3.4). The rows of examinees from group 1 were permutated to the top of the response matrix (denoted as $\mathbf{M}_{11}$ and $\mathbf{M}_{12}$). In this step, the missing responses of group 2 to 5 to the item set E (denoted as $\mathbf{M}_{22}$) were imputed using the structured matrix completion. In step 2, the rows of group 2 were permutated to the top of the response matrix, and the missing responses of groups 1, 3, 4 and 5 to the item set D (denoted as $\mathbf{M}_{22}$ on the right side of Figure 3.4) were computed. This process was repeated until all the missing entries in the response matrix were filled, which would be finished in five steps.

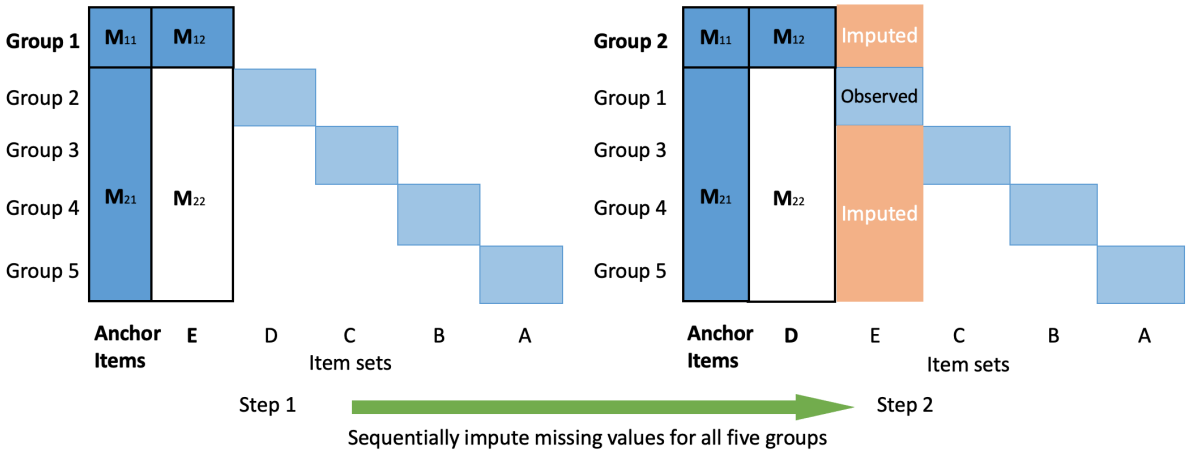


Figure 3.4: Impute Missing Responses using Structured Matrix Completion for the Targeted NEAT Design

Random NEAT design. Prior to the five steps of rearranging and imputing the response matrix from the targeted NEAT design, one extra step was conducted for the response matrix from the random NEAT design. That is, the random NEAT design was firstly rearranged into the similar structure of targeted NEAT design as shown in Figure 3.5. For example, rather than an observed blue block of examinees in

group 1 and item set E in the targeted NEAT design, the observed blue block in the random NEAT design, $\mathbf{M}_{12}$ in Figure 3.5, was from examinees who answered item set E. These examinees were randomly selected from the whole sample. Then the similar five steps were performed to the rearranged response matrix from the random NEAT design to impute all the missing values.

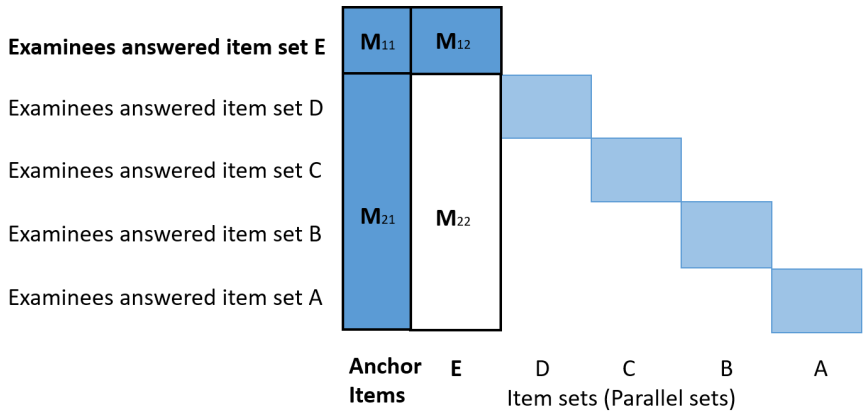
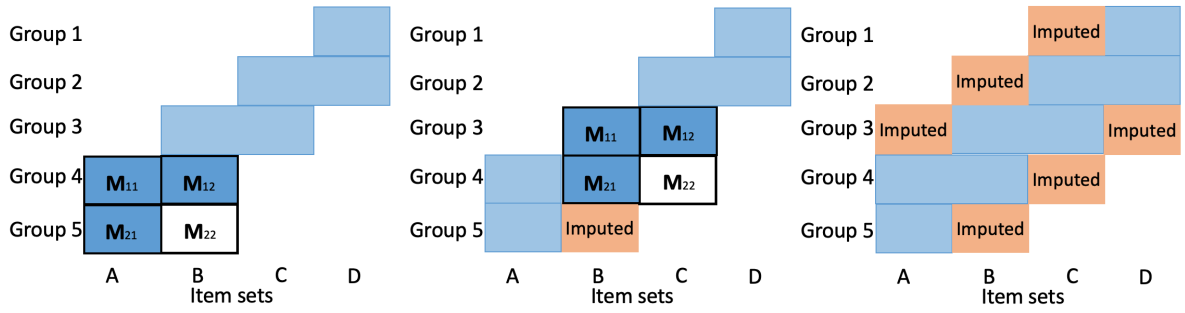


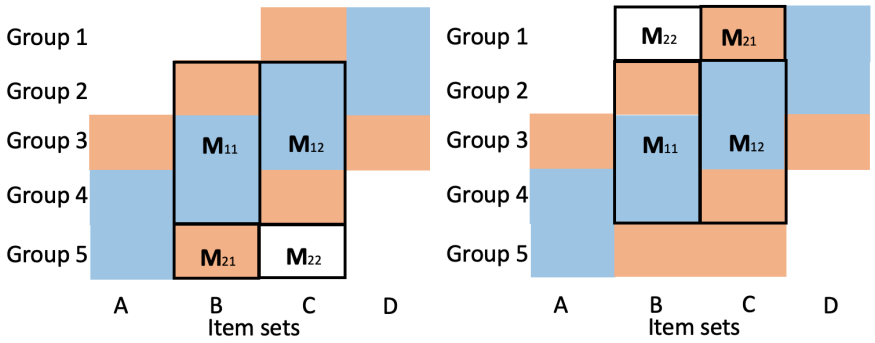
Figure 3.5: Arrange Observed Response Matrix from Random NEAT Design Before Using Structured Matrix Completion

Targeted PC design. For the targeted PC design, there were three rounds of imputation along with permutations and rearrangements of the observed response matrices. Each round has a few steps which is shown in Figure 3.6. In the first round, every two adjacent observed blocks on the diagonal of the response matrix were treated as one mini response matrix. For example, in the response matrix on the left side of Figure 3.6a, the observed responses from groups 4 and 5 on item sets A and B were used to impute the missing responses of group 5 to item set B ($\mathbf{M}_{22}$). The procedure was repeated until the imputed response matrix on the right side of Figure 3.6a was obtained after 6 steps. Note that the imputed values were continuous. In the round 2, the imputed values from the round 1 and the observed responses were used together to impute the remaining missing responses. For example, in order to obtain the missing responses for group 5 to the item set C ($\mathbf{M}_{22}$ in the response matrix on the left side of Figure 3.6b), the observed responses from groups 2, 3, and 4 (shown as the blue blocks) along with the imputed values for group 2, 4 and 5 (shown as the orange blocks) were treated as known. Similarly, the missing responses for group 1 to the item set B were obtained ($\mathbf{M}_{22}$ in the response matrix on the right side of Figure 3.6b). In

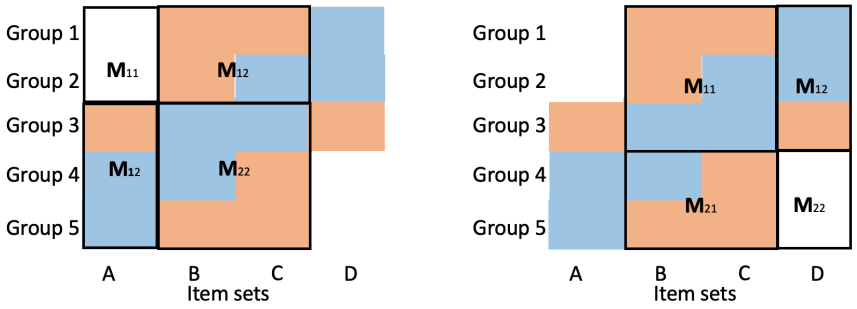
the last round, the missing responses from group 1 and 2 to the item set A and the missing responses from group 4 and 5 to item set D were imputed (Figure 3.6c).



(a) Round 1



(b) Round 2



(c) Round 3

Figure 3.6: Imputing Missing Responses using Structured Matrix Completion for Targeted Panel Chained Design

Random PC design. For random PC designs, a prior step of organizing the random PC design into the structure of the targeted PC design was conducted first. Then the three rounds of rearrangements introduced for the targeted PC design were used to imputed missing values.

3.2.2 Convert binary responses into continuous inputs for structured matrix completion.

Different from the 1-bit matrix completion, the structured matrix completion is developed for continuous matrix. Thus, the observed binary response matrix needs to be converted into a continuous matrix before using the structured matrix completion. In this study, we proposed four different converting methods which were summarized in Table 3.2. The first converting method simply changed 0 to -1 and kept 1 as 1. The next three methods were based on the item parameter estimations and ability estimations from the observed data using full information maximum likelihood estimation. In particular, the first step of the three methods was to estimate item parameters $\hat{a}$ and $\hat{b}$, and examinee' ability parameter $\hat{\theta}$ from the observed response matrices. With these estimations, the probability of correctness for each missing response can be computed based on the 2PL model following Equation 2.2. The probabilities of correctness, denoted as $p0$, were used as the second converting method to produce continuous input values for structured matrix completion. Note that $p0$ is in the range between 0 and 1. In the next two converting methods, $p0$ was transformed into a scale from negative values to positive values using two different ways. In the first way, the continuous values, $p1$, were obtained by subtracting the mean of $p0$ such that the center of $p1$ was set at 0. In the second way, the continuous values, $p2$, were obtained by standardizing $p0$ such that the center of $p2$ was 0, and its standard deviation was 1.

Table 3.2: Converting Methods (Binary Responses To Continuous Values)

Method (Notation)	Continuous input
1	$0 \to -1$ $1 \to 1$
$p0$	$p_{0_{ij}} = (1 + e^{-1.7\hat{a}_i(\hat{\theta}_j - \hat{b}_i)})^{-1^*}$
$p1$	$p_{1_{ij}} = p_{0_{ij}} - \bar{p_0}$
$p2$	$p_{2_{ij}} = (p_{0_{ij}} - \bar{p_0})/sd(p_0)$

* $\hat{a}_i$ and $\hat{b}_i$ are estimated discrimination and difficulty parameters for item i. $\hat{\theta}_j$ is the estimated ability for examinee j.

3.2.3 Dichotomize the continuous output from structured matrix completion.

The output from structured matrix completion is a continuous matrix. The two dichotomizing methods, *Sign* and *p_tw*, introduced in section 3.1 (Table 3.1) were used to obtain a final binary response matrix from structured matrix completion.

3.3 Simulation Study 1 Design

In the simulation study 1, the proposed adaptations for 1-bit matrix completion and structured matrix completion were evaluated under different incomplete calibration designs, item banks, and sample sizes. The general settings for the simulation are introduced in this section and the results are presented in next section.

3.3.1 General Settings in the Simulation Study 1

For 1-bit matrix completion, there were two dichotomizing methods, *Sign* and *p_tw*. For structured matrix completion, there were eight methods of adaptations which were combinations of four converting methods and two dichotomizing methods. The total of ten methods were evaluated under various conditions and compared to *FIML*, multiple imputation with two-way imputation, MICE with CART, and MICE with RF which were treated as four baselines. Three factors were of interests in this simulation study: item banks, sample sizes, and incomplete calibration designs. Specifically, each method was evaluated under 3 different item banks, 3 different sample sizes, and 4 incomplete calibration designs. Thus, there were 36 conditions for each of the 14 methods. These factors were summarized in the Table 3.3 with another three factors with only one level.

Table 3.3: Simulation Factors in Simulation Study 1

Factors	Levels
Measurement model	2-parameter logistic model
Item bank size	150 items
Item parameters	$a \sim Unif[-0.4, 2], b \sim Unif[-2, 2]$
	$a \sim logN(0, 0.5), b \sim N(0, 1)$
	Based on a real item bank
Ability parameters	$N(\boldsymbol{\mu},\ 1), \boldsymbol{\mu} = [-1, -0.5, 0, 0.5, 1]$
Sample size	125, 25 per group
	250, 50 per group
	500, 100 per group
Incomplete calibration design	Targeted NEAT design
	Targeted PC design
	Random NEAT design
	Random PC design
Baseline	Multiple imputation using two-way Imputation
	FIML
	MICE(CART)
	MICE(RF)

Item bank. The number of items was fixed at 150 as a small or medium-sized item bank. The item parameters were generated from three different sets of distributions. The first and second item banks were generated given the common practices in the psychometric simulation studies. For the first item bank, the discrimination parameter a were drawn from Uniform $(0.4, 2.0)$, and the difficulty parameter b were drawn from Uniform $(-2.0, 2.0)$ (e.g., Kirisci et al., 2001; Kohli et al., 2015). This item bank was denoted as the Uniform item bank. In the second item bank, the discrimination parameter a were simulated from

$logN(0,\ 0.5)$, and the difficulty parameter b were simulated from $N(0,\ 1)$ (e.g., Lee, 2011; Suh et al., 2012; Orlando and Thissen, 2000). In addition, if the simulated a parameter was larger than 2.5 or smaller than 0.2, it would be excluded and regenerated to be in the range of 0.2 to 2.5. This item bank was denoted as the Normal item bank. For these two item banks, to avoid the situation where items with low (high) difficulty parameters only had either large (small) discrimination parameters, items with different values of difficulty parameter b were paired with similar discrimination parameter a following the design by Zhang (2020). The third item bank was simulated based on a practical item bank that was studied in Wang et al. (2016). The distributions of a and b parameters from the three item banks were presented in Figure 3.7. The summary statistics for the item parameters were documented in Table 3.4. Averagely, Normal item bank had relatively easy items with high discriminating power.

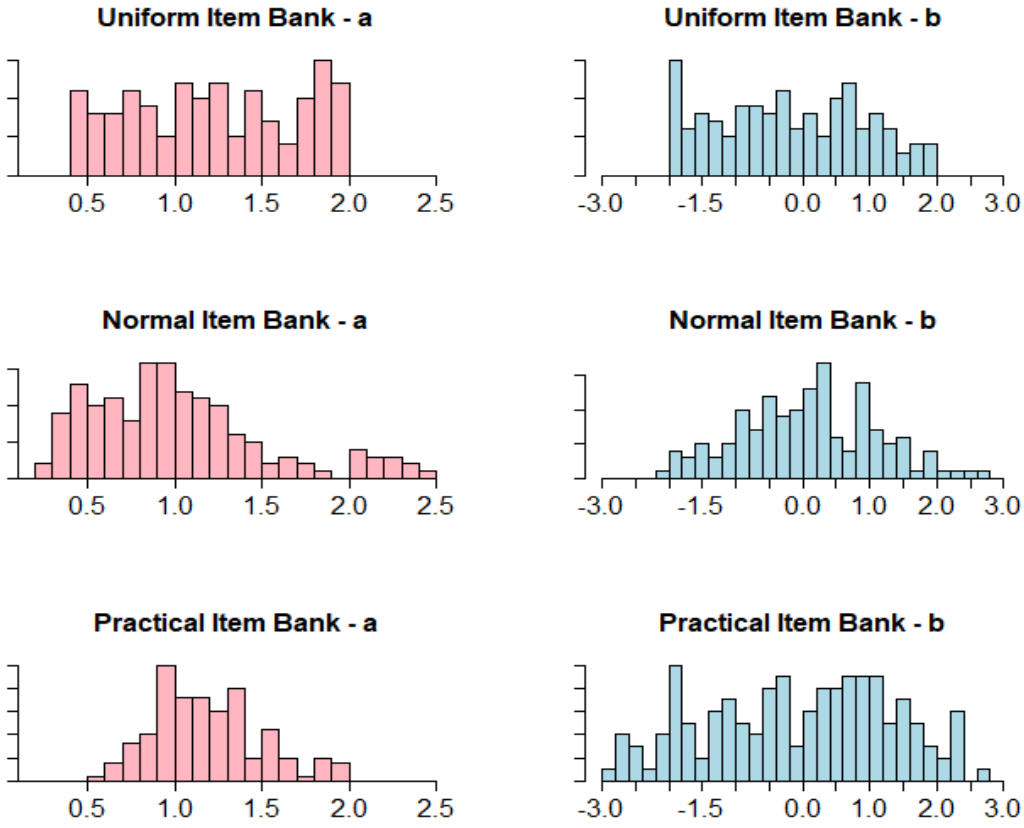


Figure 3.7: Distributions of Item Parameters in Three Item Banks

Table 3.4: The Summary Statistics of Item Parameters in Item Banks

Item Bank	a	b
Uniform	0.750	0.000
	(0.355)	(1.429)
Normal	1.219	-0.206
	(0.480)	(1.110)
Practical	1.015	0.191
	(0.504)	(1.131)

Note. The numbers in parentheses are standard deviations.

Sample sizes. Three levels of sample sizes were studied: 125, 250, and 500. Previous studies suggested using 500 examinees per item to calibrate item parameters in 2PL model (De Ayala, 2013; Ban et al., 2002). The focus of this study is to calibrate item parameters with small sample sizes, and thus two smaller sample sizes, 125 and 250, were chosen to evaluate the performance of the proposed methods. Given the survey that the average classroom seizes are between 20 to 35 (Woessmann and West, 2006; U.S. Department of Education, 2012), the smallest sample size per group was determined as 25, which led to five groups to form 125 examinees. In addition, for the total sample sizes of 250 and 500, there were 50 and 100 examinees per group if having five groups. It is less common for a classroom with 100 students, however, because of similar teaching and learning plans, some classrooms have students that share similar ability levels. Thus, it is possible to have four classrooms of 25 students to be combined as one group of 100 examinees.

Examinees' ability parameter. In this study, examinees' abilities θ were simulated from 5 different normal distributions to represent five sub-populations following Zhang (2020). Examinees' abilities were from five normal distributions $N(-1, 1)$, $N(-0.5, 1)$, $N(0, 1)$, $N(0.5, 1)$, and $N(1, 1)$.

Incomplete calibration designs. In this study, four incomplete calibration designs introduced in Section 2.3.2 were investigated. These four incomplete calibration designs were presented next.

Targeted NEAT design. The targeted NEAT design followed Zhang (2020) and was presented in Table 2.1 in literature review chapter. In particular, the number of the anchor items was fixed at 20, and the

anchor items were selected to represent the whole item bank as shown in Figure 3.8. Specifically, all the items in the item bank were divided into five equal-sized strata after ordering item difficulty parameters from the smallest to the largest. If the item difficulty parameters followed uniform distribution, 4 items were selected within each strata with different item discrimination parameters. Among the 4 items, 2 of them had item discrimination parameters smaller than their median, and the other 2 items had item discrimination parameter larger than their median. The similar procedure was used to select anchor items when item difficulty parameters followed a normal distribution. The only difference was that more anchor items were selected in the middle strata than the two end strata to reflect the item difficulty parameter distribution in this item bank. For the practical item bank, since the distribution of the item difficulty parameters is relatively uniform (Figure 3.7), the anchor items were selected in the same way as the uniform item bank. The rest of the items in the item bank were assigned to five item sets according the item difficulty parameters. Each item set with the anchor item set together were constructed as one test form. For example, the test form 1, including the anchor item set X and the item set A, was the hardest form and thus assigned to the group 1 of the examinees with the highest ability. Similarly, the group 2 of the examinees got test form 2. The summary statistics for item parameters by test forms in three item banks were presented in Table 3.5. From this table, the item difficulty parameters were increasing from Form 1 to Form 5, while item discrimination parameters were very similar across five test forms. In this design, given each test form had 46 items, the missing percentage was 69.3%.

Random NEAT design. The second incomplete calibration design in this simulation study was the random NEAT design (Table 2.3), which differed from the targeted NEAT designs in two ways. Firstly, in the random NEAT design, item sets A to E were parallel with similar difficulty levels. The summary statistics for item parameters by test forms in the random NEAT design were presented in Table 3.6. From this table, both item discrimination parameters and item difficulty parameters were similar across five test forms. Secondly, examinees received the test forms randomly at the individual level. That is to say, examinees in the same ability group could received different test forms.

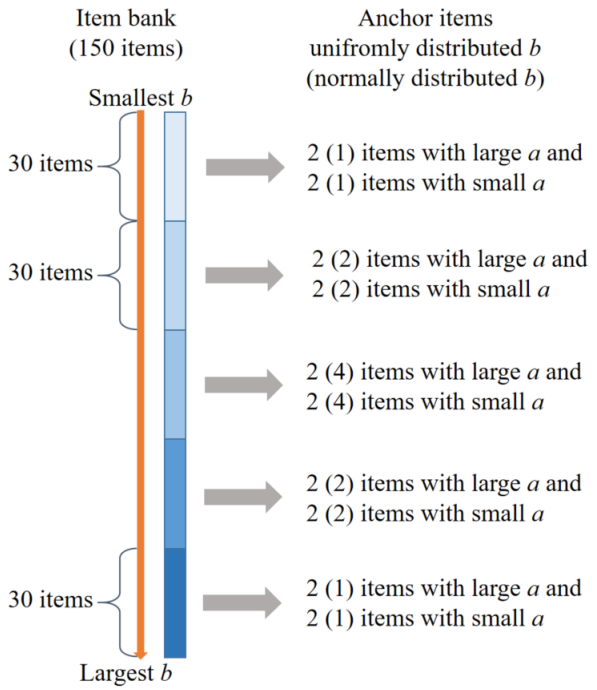


Figure 3.8: Anchor Item Selection for NEAT Design

Table 3.5: Summary Statistics for Item Parameters Per Test Form in Targeted NEAT Design

		Form 1	Form 2	Form 3	Form 4	Form 5
Uniform	a	1.215	1.215	1.217	1.225	1.213
		(0.481)	(0.562)	(0.455)	(0.482)	(0.471)
	b	-1.069	-0.602	-0.206	0.243	0.684
		(1.074)	(0.823)	(0.715)	(0.799)	(1.057)
Normal	a	1.099	0.996	1.005	0.942	1.057
		(0.460)	(0.482)	(0.395)	(0.543)	(0.566)
	b	-0.721	-0.235	0.107	0.435	0.926
		(0.960)	(0.616)	(0.508)	(0.582)	(0.916)
Practical	a	1.088	1.151	1.176	1.262	1.241
		(0.310)	(0.274)	(0.280)	(0.307)	(0.298)
	b	-1.230	-0.581	0.036	0.507	1.029
		(1.253)	(0.805)	(0.630)	(0.761)	(1.139)

Note. The numbers in parentheses are standard deviations.

Table 3.6: Summary Statistics for Item Parameters Per Test Form in Random NEAT Design

		Form 1	Form 2	Form 3	Form 4	Form 5
Uniform	a	1.163	1.348	1.200	1.174	1.251
		(0.463)	(0.456)	(0.496)	(0.462)	(0.513)
	b	0.027	-0.206	-0.354	-0.195	-0.267
		(1.155)	(1.070)	(1.045)	(0.963)	(1.220)
Normal	a	0.983	1.052	1.094	0.936	0.948
		(0.487)	(0.456)	(0.523)	(0.389)	(0.374)
	b	0.095	0.182	0.134	-0.040	0.111
		(1.108)	(1.007)	(0.929)	(1.075)	(1.015)
Practical	a	1.172	1.207	1.240	1.096	1.218
		(0.282)	(0.282)	(0.341)	(0.258)	(0.289)
	b	-0.001	0.061	-0.203	0.049	-0.147
		(1.186)	(1.149)	(1.459)	(1.332)	(1.338)

Note. The numbers in parentheses are standard deviations.

Targeted PC design. In this design, all items were divided into four item sets according to their item difficulty parameters. As shown in Table 2.2, item set A had the easiest items. Item set B had harder questions than A but easier questions than C and D. In particular, test form 1 had item set D, which was the hardest and assigned to group 1 of examinees with the lowest ability. Test form 2 had item sets C and D, which was easier than test form 1 and thus assigned to group 2 of examinees. The summary statistics of the item parameters in each test form were presented in Table 3.7. In addition, the missing rate in this design was 60.0%, which was similar to the targeted and random NEAT design.

Random PC design. The random PC designs (Table 2.4) were similar to targeted PC design, but all the items were randomly assigned to the test forms, and thus the test forms had similar difficulties. Table 3.8 shows the summary statistics for the item parameters per test form. It can be seen that both item discrimination parameters and item difficulty parameters were similar across test forms. Similar to random NEAT design, examinees received the test forms randomly at the individual level.

Table 3.7: Summary Statistics for Item Parameters Per Test Form in Targeted PC Design

		Form 1	Form 2	Form 3	Form 4
Uniform	a	1.210	1.232	1.234	1.202
		(0.500)	(0.524)	(0.464)	(0.448)
	b	-1.644	-0.646	0.256	1.211
		(0.256)	(0.258)	(0.304)	(0.390)
Normal	a	1.115	1.001	0.979	0.964
		(0.503)	(0.373)	(0.542)	(0.578)
	b	-1.208	-0.236	0.392	1.362
		(0.437)	(0.205)	(0.222)	(0.482)
Practical	a	1.021	1.119	1.244	1.352
		(0.320)	(0.222)	(0.278)	(0.317)
	b	-1.915	-0.533	0.588	1.620
		(0.493)	(0.352)	(0.268)	(0.464)

Note. The numbers in parentheses are standard deviations.

Table 3.8: Summary Statistics for Item Parameters Per Test Form in Random PC Design

		Form 1	Form 2	Form 3	Form 4
Uniform	a	1.194	1.172	1.249	1.263
		(0.472)	(0.464)	(0.477)	(0.519)
	b	0.101	-0.237	-0.306	-0.385
		(1.134)	(1.054)	(0.993)	(1.224)
Normal	a	1.027	1.057	1.048	0.931
		(0.607)	(0.447)	(0.534)	(0.415)
	b	0.166	0.112	-0.005	0.036
		(1.037)	(1.044)	(0.998)	(0.989)
Practical	a	1.161	1.245	1.137	1.195
		(0.285)	(0.343)	(0.323)	(0.294)
	b	-0.012	-0.153	-0.053	-0.029
		(1.154)	(1.511)	(1.470)	(1.434)

Note. The numbers in parentheses are standard deviations.

Baselines. The first baseline was the multiple imputation with two-way imputation to replace the missing responses. Multiple imputation provides better accuracy than simple imputation in previous studies (e.g., Schafer and Graham, 2002). In addition, two-way imputation, which was a simple imputation, was robust with small sample size (Sijtsma and Van der Ark, 2003). Thus, the two-way imputation based multiple imputation was chosen as a baseline. In particular, given the imputed values TW_{ij} using Equation (2.11), five integers were drawn from Bernoulli distribution with TW_{ij} as the parameter. The final binary response is the rounded average of the five integers. The second baseline is to use FIML to calibrate item parameters directly as it is the most commonly used method (Finch, 2008). Last two baselines were the Multivariate Imputation by Chained Equations (MICE) with classification and regression trees (CART) and random forest imputation (RF), which have drawn attention of researchers in the educational assessment field (e.g., Xiao and Bulut, 2020).

3.3.2 General Procedures in the Simulation Study 1

The purpose of this simulation study was to evaluate the proposed adaptations to 1-bit matrix completion and structured matrix completion under different incomplete calibration design in various conditions, such as different item banks and sample sizes. For each of the 36 conditions introduced in the previous section, ten imputation methods based on matrix completion with additional four baselines were investigated. The general procedure consisted with four steps (Figure 3.9): (1) generating full response matrices ($\mathbf{Y}_{full}$) given the simulated item parameters in the item bank, examinees' abilities, and sample sizes; (2) masking out some responses as missing responses given the incomplete calibration designs to obtain the incomplete response matrices ($\mathbf{Y}_{obs}$); (3) imputing the missing values to obtain the imputed binary response matrix from each imputation method ($\mathbf{Y}_{imp}$); and (4) calibrating item parameters using $\mathbf{Y}_{imp}$ (and $\mathbf{Y}_{obs}$ for FIML method) and obtaining $\hat{\mathbf{a}}$ and $\hat{\mathbf{b}}$.

This study was conducted in $flexMIRT$ and R (R Core Team, 2021). Specifically, $flexMIRT$ was used to estimate item parameters and ability parameters with multiple group calibration and marginal maximum likelihood estimation (MMLE) as Mislevy and Sheehan (1989) and Eggen and Verhelst (2011)

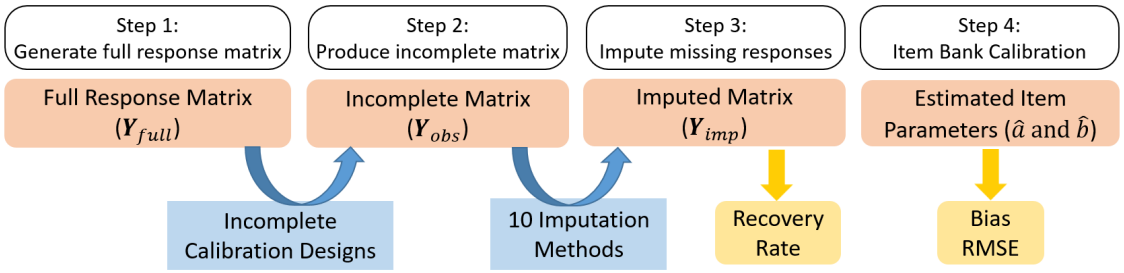


Figure 3.9: The General Procedure in the Simulation Study

have shown it is justified in incomplete calibration designs. Note that when responses for one item were all incorrect or correct, this item was removed before estimating item parameters because of the infinite estimation of MMLE. The rest of this simulation study was conducted in R. The structured matrix completion was conducted using $smc.FUN()$ function in the $StructureMC$ package in R (Liu et al., 2015). The R codes for 1-bit matrix completion were adopted and modified from the github (Avouac and Bjaï, 2020). MICE package (Van Buuren and Groothuis-Oudshoorn, 2011) was used for two MICE based baselines. All conditions were replicated 50 times. Among the 50 replications, the item parameters were fixed and the ability parameters were regenerated each time. The evaluation criteria were introduced next.

Evaluation criteria. To evaluate the performance of proposed adaptations for 1-bit matrix completion and structured matrix completion, two sets of evaluation criteria were checked. The first set was to evaluate the accuracy of the imputed and dichotomized responses via average recovery rates (RR) over R replications, which was computed as

$$RR = \frac{\sum_{r=1}^{R} n_{cr}/n_{mis}}{R}$$

where n_{cr} is the number of correctly imputed responses in the r^{th} replication and n_{mis} is the number of missing responses. Recovery rates evaluate the imputation methods directly. The higher the recovery rates, the more correctly imputed responses from that imputation method.

The second set was to evaluate the item parameter estimation with bias and root mean square of error (RMSE). Over R replications, the bias and RMSE can be computed with the following formula

$$Bias = \frac{\sum_{r=1}^{R}(\hat{\pi}_r - \pi)}{R}$$
$$RMSE = \sqrt{\frac{\sum_{r=1}^{R}(\hat{\pi}_r - \pi)^2}{R}}$$

where π is the true value for the parameter and $\hat{\pi}_r$ is the estimation for the parameter at r^{th} replication. The estimation for item parameters is another way to evaluate how the proposed imputation methods improve the item bank calibration. Since the estimated item parameters would be directly used in the future CAT applications, it is crucial that the imputation methods would also provide accurate estimations of item parameters.

3.4 Simulation Study 1 Results

In this simulation, the ten proposed imputation methods were evaluated. Two of them were based on 1-bit matrix completion and paired with two dichotomizing methods, *p_tw* and *Sign*, which were denoted as *1bit(p_tw)* and *1bit(Sign)*. The other eight methods were based on structured matrix completion with four converting methods, *1*, *p0*, *p1*, and *p2*, and two dichotomizing methods, *p_tw* and *Sign*. These eight methods were denoted as *SMC(1+p_tw)*, *SMC(p0+p_tw)*, *SMC(p1+p_tw)*, *SMC(p2+p_tw)*, *SMC(1+Sign)*, *SMC(p0+Sign)*, *SMC(p1+Sign)*, and *SMC(p2+Sign)*. In addition, the four baselines in this study: FIML, two-way imputation, MICE with CART, and MICE with RF, were denoted as *FIML*, *TW*, *MICE(CART)*, and *MICE(RF)*. Note that FIML did not involve any imputation and thus recovery rate was not applicable as an evaluation criterion. The simulation results were presented in three sections. The first section focuses on the two adaptations for 1-bit matrix completion. Their recovery rates, bias(b), RMSE(b), bias(a), and RMSE(a) were presented and the potential reasons to explain the observed patterns

were investigated. The second section focuses on the eight adaptations for structured matrix completion. Lastly, the results from all methods, including baseline methods, were compared and discussed.

3.4.1 Results of Adapted 1-bit Matrix Completion.

In the simulation, the two modifications for 1-bit matrix completion, *1bit(Sign)* and *1bit(p_tw)*, were evaluated in 36 conditions from three item banks, three sample sizes, and four incomplete calibration designs. In particular, their recovery rates, estimations for item parameters a and b in the practical item bank and sample size of 125 were presented first. Then, the recovery rates and the estimations for item parameters in the other two item banks and two sample sizes were examined to evaluate the influence of item banks and sample sizes.

Recovery rates from *1bit(Sign)* and *1bit(p_tw)* in practical item bank with $N = 125$.

Figure 3.10 presents the recovery rate per item from *1bit(Sign)* and *1bit(p_tw)* in the practical item bank with sample size of 125. The items are ordered from the easiest to the hardest on the x-axis. Generally, *1bit(p_tw)* produced higher recovery rates than *1bit(Sign)* in all four incomplete calibration designs. The recovery rates from *1bit(Sign)* were slightly higher than 0.5, which was close to the random guess for binary responses. In addition, *1bit(p_tw)* provided higher recovery rates for easy and difficult items than medium-difficulty items. Furthermore, incomplete calibration designs showed some influence on *1bit(p_tw)* but not on *1bit(Sign)*. Specifically, the recovery rates of *1bit(p_tw)* were lower in the targeted PC designs than the other three incomplete calibration designs for most of the items, and the recovery rates across the other incomplete calibration designs were similar. The potential reasons for these observations were discussed after presenting the results of item parameter estimations and the influence of item banks and sample sizes.

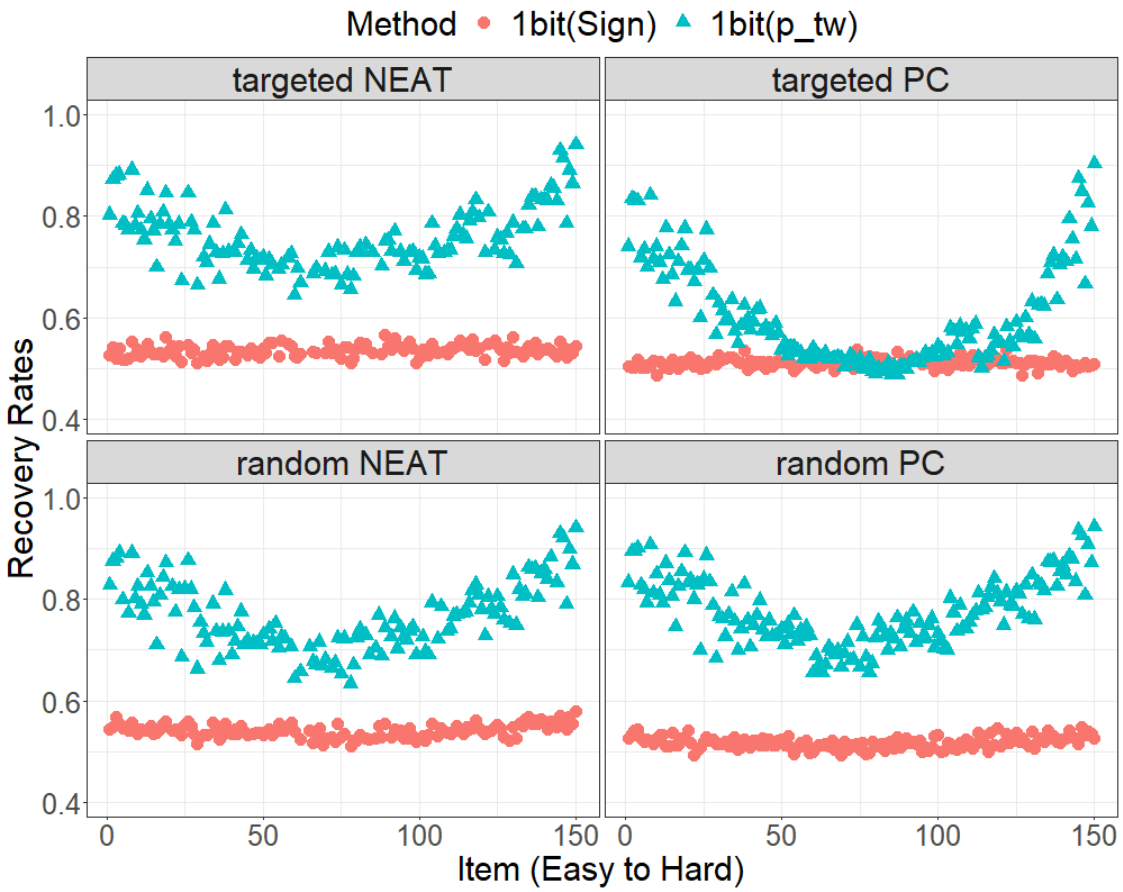


Figure 3.10: Recovery Rates of *1bit(Sign)* and *1bit(p_tw)* in Practical Item Bank and $N = 125$

Bias(b) and RMSE(b) from *1bit(Sign)* and *1bit(p_tw)* in practical item bank with $N = 125$.

From Table 3.9, *1bit(p_tw)* had bias(b) closer to zero and smaller RMSE(b) than *1bit(Sign)* across incomplete calibration designs. In particular, the RMSE(b) of *1bit(Sign)* were very large. This finding is consistent with previous observation that *1bit(p_tw)* had higher recovery rates than *1bit(Sign)*. In addition, the RMSE(b) of *1bit(p_tw)* were smaller in the targeted NEAT and targeted PC designs than in random NEAT and random PC designs.

The average RMSE(b) and bias(b) are summarised statistics and some nuances are hidden. Figure 3.11 shows bias(b) per item under four incomplete calibration designs. Items were ordered from the easiest to the hardest on the x-axis. In targeted NEAT and targeted PC designs, *1bit(p_tw)* showed some systematic bias(b) according to test forms but their bias(b) were still closer to zero than *1bit(Sign)* for most items. In

random NEAT and random PC designs, *1bit(p_tw)* had bias(b) close to zero for medium-difficulty items and large bias(b) for easy and difficult items. The large bias(b) of *1bit(p_tw)* for easy and difficult items in random NEAT and random PC designs explained larger average RMSE(b) in these two designs than random NEAT and random PC designs (Table 3.9. In addition, *1bit(sign)* produced bias(b) with large variations in all four incomplete calibration designs.

Table 3.9: Bias(b) and RMSE(b) from *1bit(Sign)* and *1bit(p_tw)* in Practical Item Bank with $N = 125$

Design	Method	Bias(b)	RMSE(b)
Targeted NEAT	1bit(Sign)	-1.901	28.307
	1bit(p_tw)	0.077	0.667
Targeted PC	1bit(Sign)	-0.257	11.840
	1bit(p_tw)	0.059	0.824
Random NEAT	1bit(Sign)	0.510	16.690
	1bit(p_tw)	0.091	2.221
Random PC	1bit(Sign)	-1.874	40.349
	1bit(p_tw)	-0.401	4.944

Bias(a) and RMSE(a) from *1bit(Sign)* and *1bit(p_tw)* in practical item bank with $N = 125$.

From Table 3.10, bias(a) from *1bit(Sign)* were negative in all incomplete calibration designs, while bias(a) from *1bit(p_tw)* were positive. The magnitude of bias(a) and RMSE(a) of *1bit(Sign)* were smaller than *1bit(p_tw)*. In addition, the bias(a) and RMSE(a) of *1bit(p_tw)* were smaller in the targeted PC and random PC designs than in targeted NEAT and random NEAT designs. In particular, *1bit(p_tw)* had the smallest bias(a) and RMSE(a) in targeted PC designs. These observations were in contrast to the recovery rates and estimations for item difficulty parameters.

To investigate the estimation for the discrimination perimeters at the item-level, Figure 3.12 shows the bias(a) per item with $N = 125$ using the practical item bank. *1bit(Sign)* underestimated a for most of the items in all incomplete calibration designs, while *1bit(p_tw)* overestimated a for most of the items with different patterns in different incomplete calibration designs. Specifically, in targeted NEAT designs, bias(a) from *1bit(p_tw)* showed a convex shape, where bias(a) for easy and difficult items were larger than bias(a) for medium-difficulty items. On the contrary, in random PC designs, bias(a) from *1bit(p_tw)* had

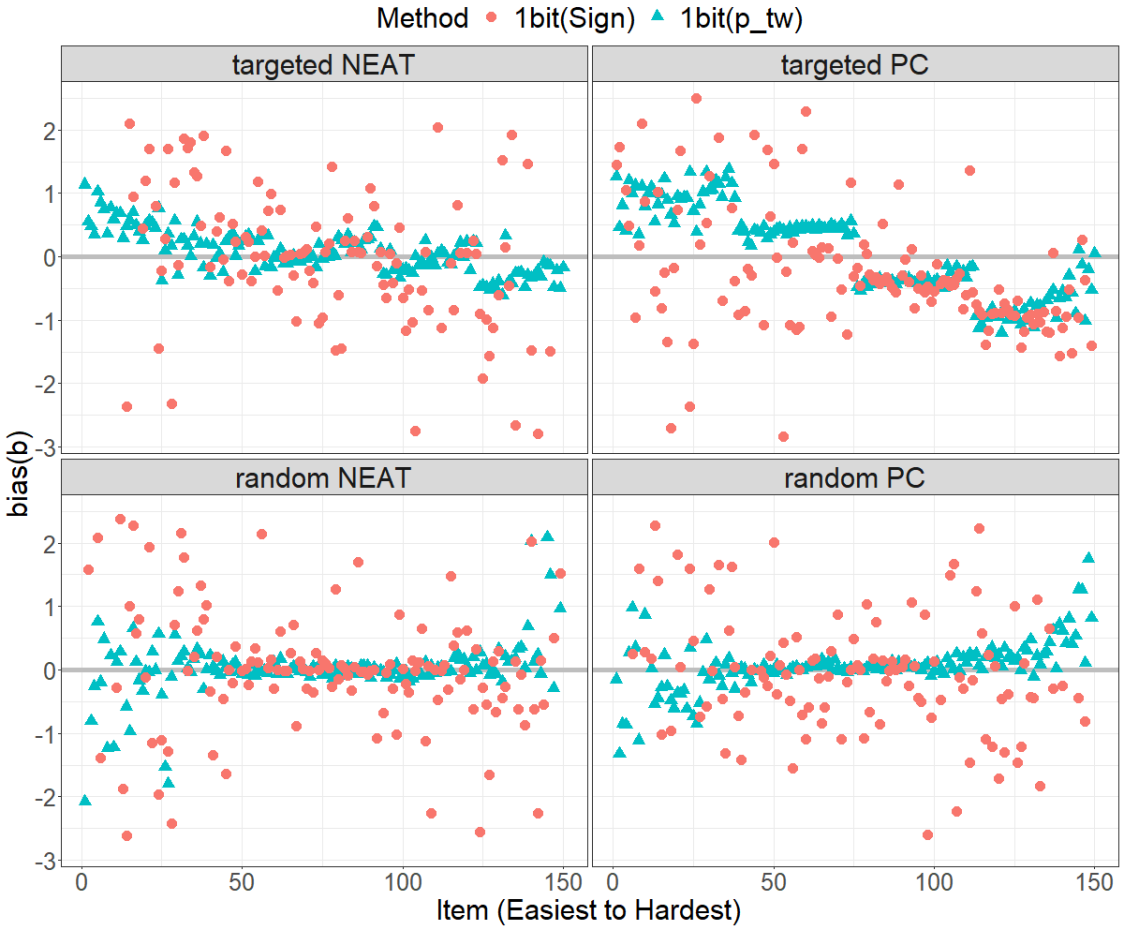


Figure 3.11: Bias(b) per Item of *1bit(Sign)* and *1bit(p_tw)* in Practical Item Bank and $N = 125$

Table 3.10: Bias(a) and RMSE(a) from *1bit(Sign)* and *1bit(p_tw)* in Practical Item Bank with $N = 125$

Condition	Method	Bias(a)	RMSE(a)
Targeted NEAT	1bit(Sign)	-0.894	0.989
	1bit(p_tw)	2.024	3.087
Targeted PC	1bit(Sign)	-0.691	0.826
	1bit(p_tw)	0.606	0.818
Random NEAT	1bit(Sign)	-0.732	0.825
	1bit(p_tw)	2.330	4.018
Random PC	1bit(Sign)	-0.826	1.125
	1bit(p_tw)	1.121	1.549

a concave shape, where bias(a) for easy and difficult items were smaller than that for medium difficulty items. In addition, bias(a) of *1bit(p_tw)* from targeted PC and random NEAT designs were similar across item difficulties. Lastly, across the four incomplete calibration designs, *1bit(p_tw)* had the smallest bias(a) in targeted PC designs.

Figure 3.12: Bias(a) per Item of *1bit(Sign)* and *1bit(p_tw)* in Practical Item Bank and $N = 125$

Influence from item banks on *1bit(Sign)* and *1bit(p_tw)*.

The findings of *1bit(Sign)* and *1bit(p_tw)* in the practical item bank were also shown in the other two item banks. This conclusion was based on all three evaluation criteria. Firstly, in terms of recovery rates, *1bit(p_tw)* also had higher recovery rates than *1bit(Sign)* in uniform and normal item banks (Figure 3.13). In this Figure, *1bit(p_tw)* had higher recovery rates than *1bit(Sign)* in all three item banks. In addition, *1bit(p_tw)* had recovery rates varied similarly given different incomplete calibration designs across three

item banks, and its recovery rates were the lowest in targeted PC designs. Furthermore, recovery rates from *1bit(p_tw)* were higher for easy and difficult items than for medium-difficulty items across item banks. For example, Figure 3.14 shows the recovery rate per item from three item banks when using targeted NEAT designs with $N = 125$. Though recovery rates of *1bit(p_tw)* had larger variations across items in the uniform and normal item banks than in the practical item bank, they were still higher for easy and difficult items than for medium-difficulty items in general. This pattern was also shown in the targeted PC design (Figure 3.15) where the recovery rates were the lowest among four incomplete calibration designs. The same pattern was shown in random NEAT and random PC designs (see Figures C.1 and C.2 in Appendix).

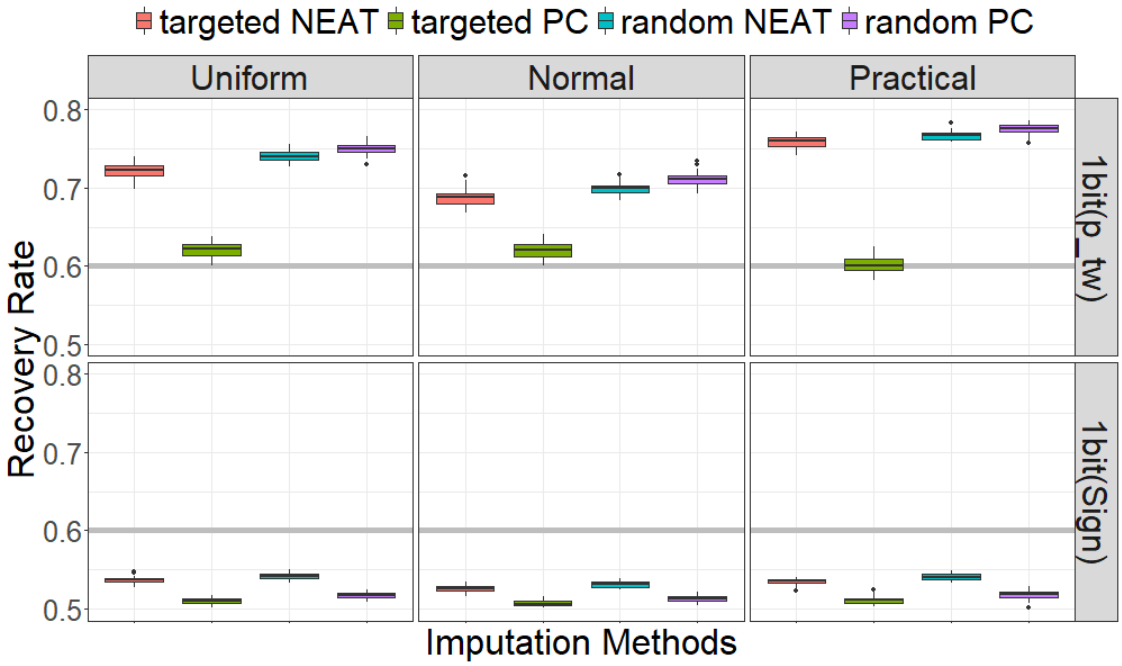


Figure 3.13: Recovery Rates of *1bit(Sign)* and *1bit(p_tw)* Across Item Banks with $N = 125$

Secondly, the results from RMSE(b) and bias(b) were also similar across three item banks. *1bit(p_tw)* had much smaller RMSE(b) and smaller variations than *1bit(Sign)* across all three item banks (Figure 3.16). In addition, Figures 3.17 and 3.18 showed the bias(b) per item in the uniform and normal item banks respectively. Similar to the observations in the practical item bank, bias(b) from *1bit(p_tw)* showed systematic bias pattern given the test forms in the targeted NEAT and targeted PC designs. In addition,

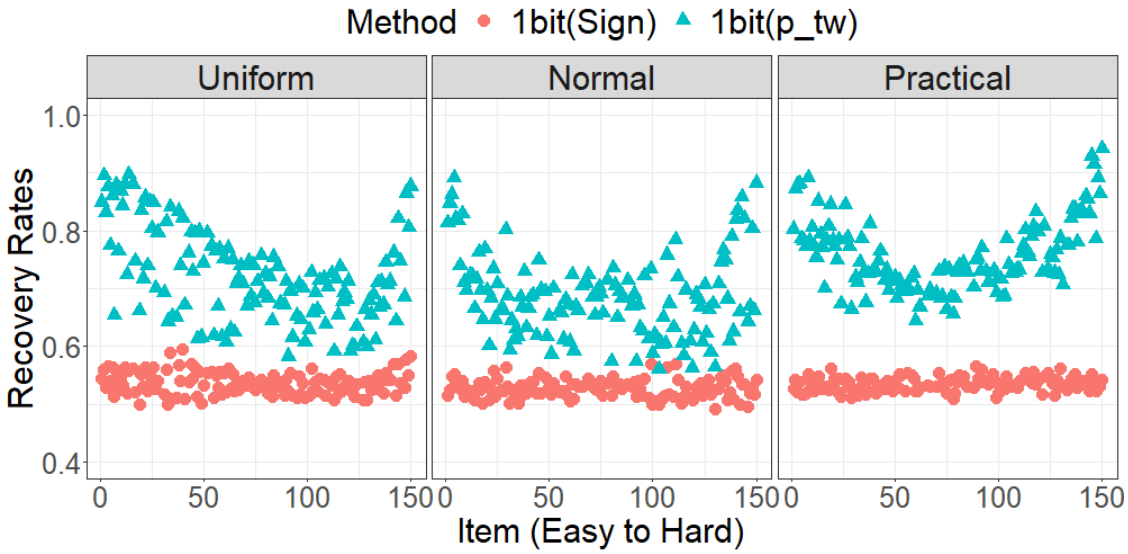


Figure 3.14: Recovery Rates of *1bit(Sign)* and *1bit(p_tw)* Across Item Banks in Targeted NEAT Designs and $N = 125$

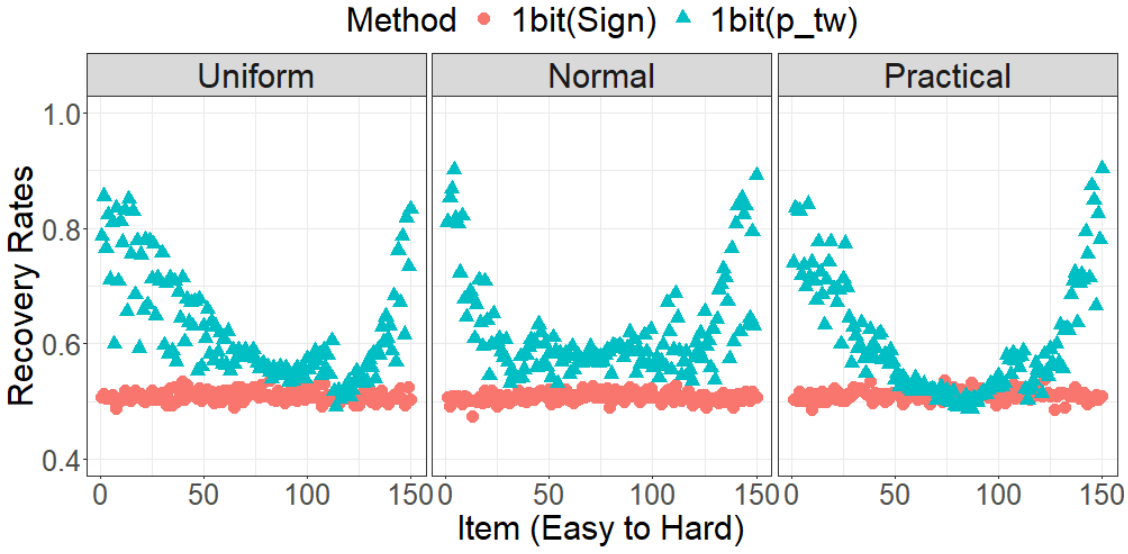


Figure 3.15: Recovery Rates of *1bit(Sign)* and *1bit(p_tw)* in Across Item Banks Targeted PC Designs and $N = 125$

1bit(p_tw) had bias(b) closer to zero for medium-difficulty items and larger bias(b) for easy and difficult items.

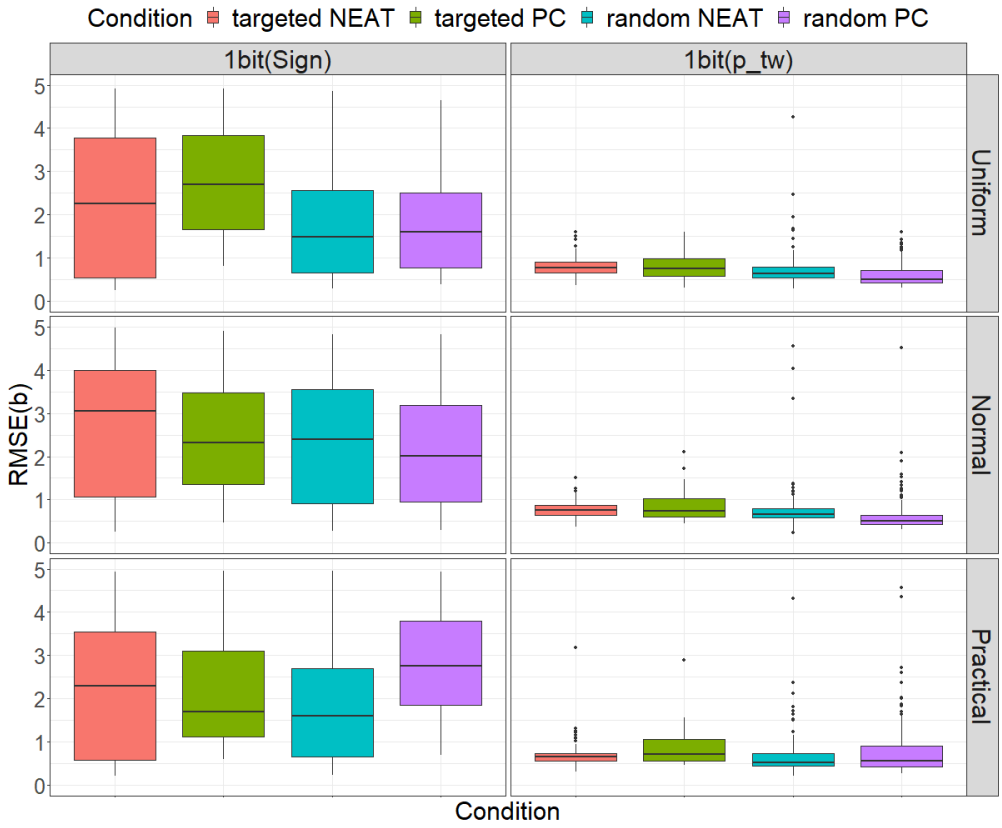


Figure 3.16: RMSE(b) of *1bit(Sign)* and *1bit(p_tw)* in Across Item Banks with $N = 125$

Lastly, the RMSE(a) and bias(a) from *1bit(p_tw)* and *1bit(Sign)* were similar across item banks. RMSE(a) from *1bit(p_tw)* were larger in targeted NEAT and random NEAT designs than that in targeted PC and random PC designs across three item banks across three item banks (Figure 3.19). In addition, Figures 3.20 and 3.21 present the bias(a) per item in the uniform and normal item banks respectively. Similar observations in the practical item bank were shown in the uniform and normal item banks. For example, *1bit(Sign)* underestimated a for most of the items, and *1bit(p_tw)* overestimated a for most of the items. In addition, bias(a) of *1bit(p_tw)* showed a convex shape in the targeted NEAT designs, and bias(a) of *1bit(p_tw)* presented a relatively flat with a concave shape in the other three incomplete calibration designs.

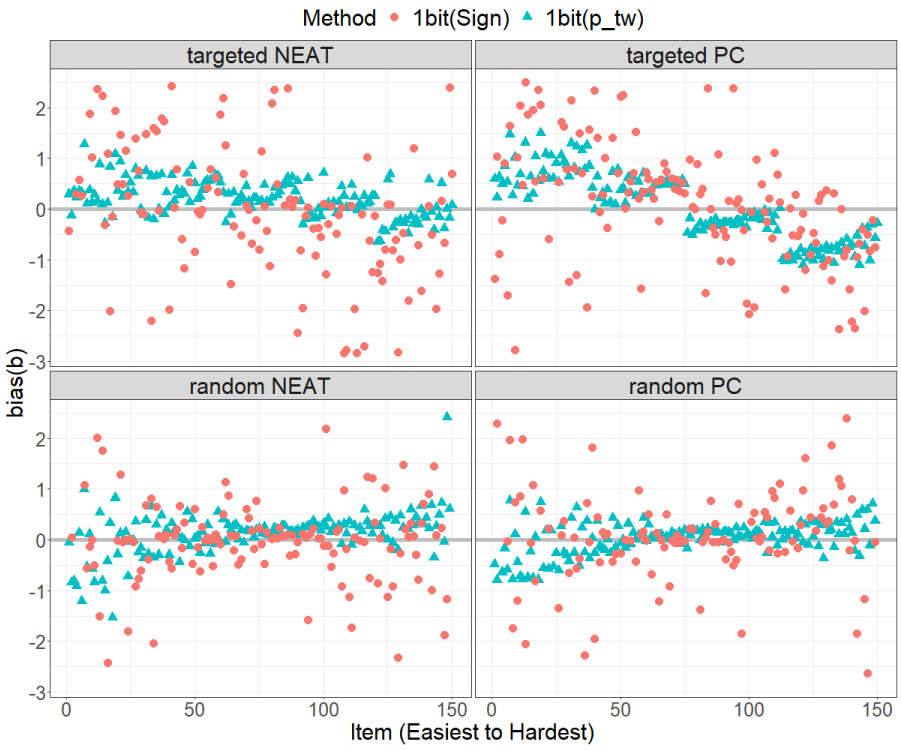


Figure 3.17: Bias(b) per Item of *1bit(Sign)* and *1bit(p_tw)* in Uniform Item Bank and $N = 125$

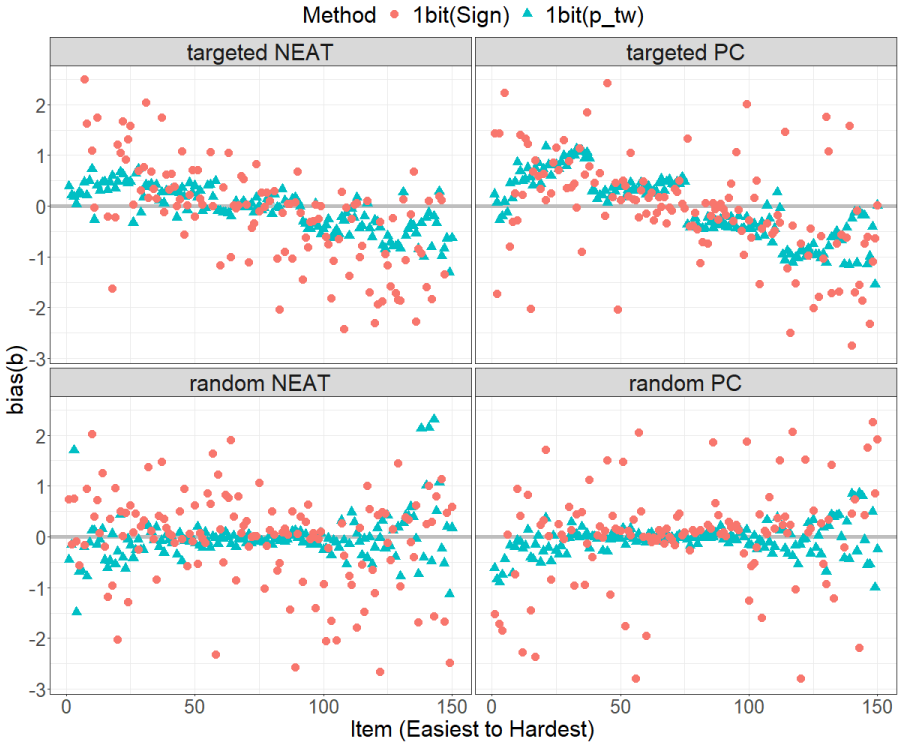


Figure 3.18: Bias(b) per Item of *1bit(Sign)* and *1bit(p_tw)* in Normal Item Bank and $N = 125$

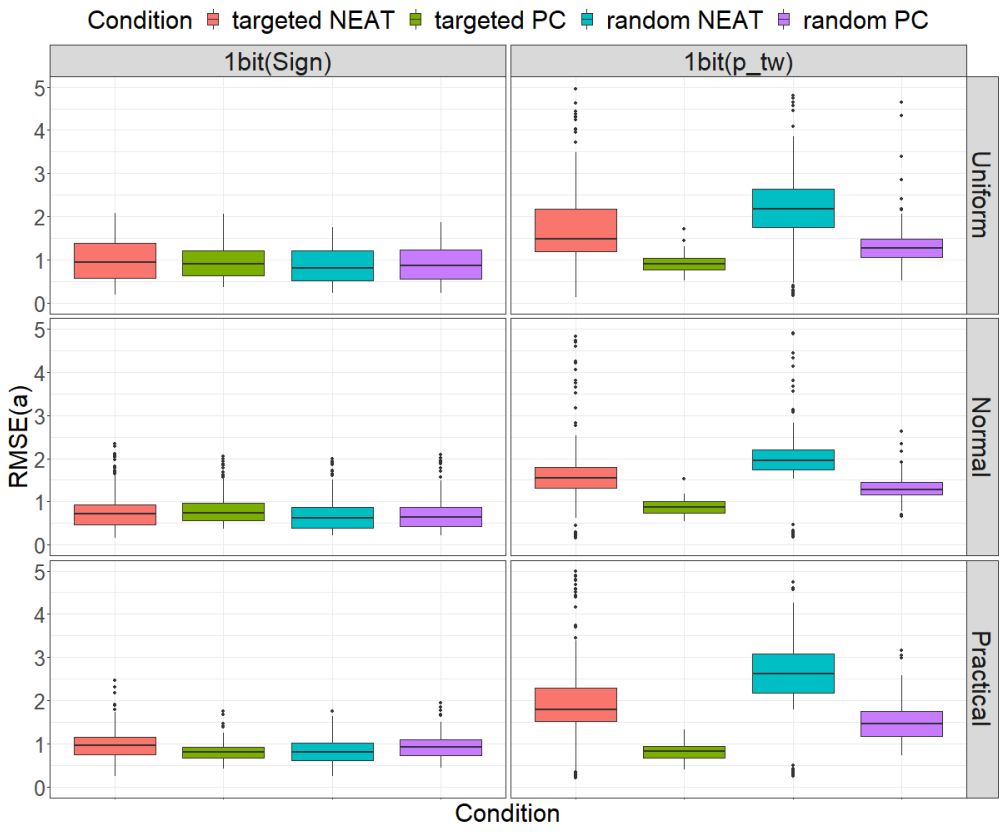


Figure 3.19: RMSE(a) of *1bit(Sign)* and *1bit(p_tw)* in Across Item Banks with $N = 125$

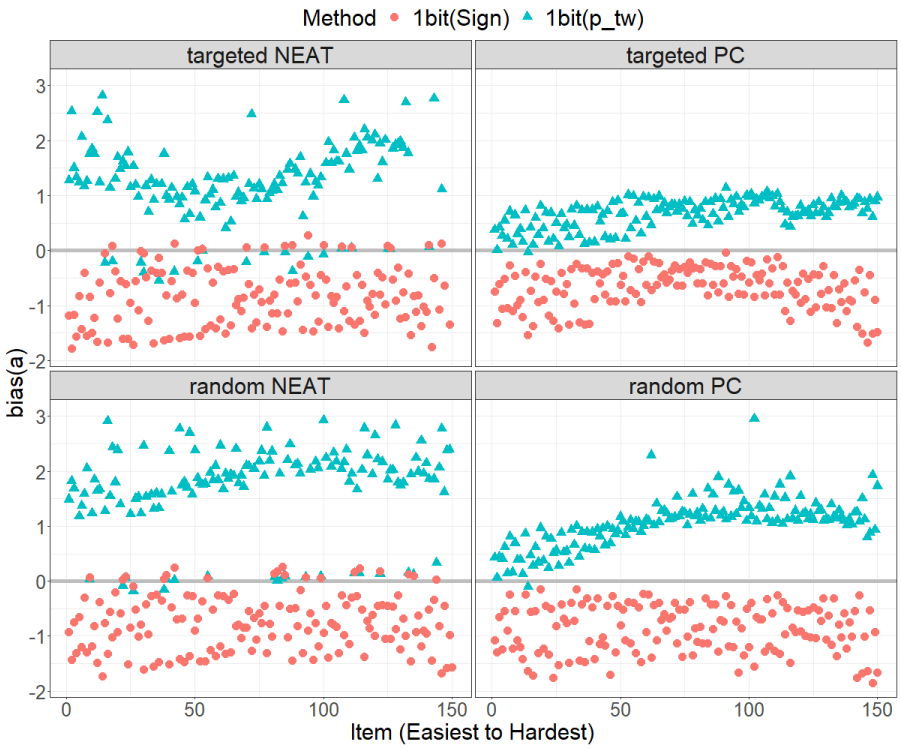


Figure 3.20: Bias(a) of *1bit(Sign)* and *1bit(p_tw)* in Uniform Item Bank and $N = 125$

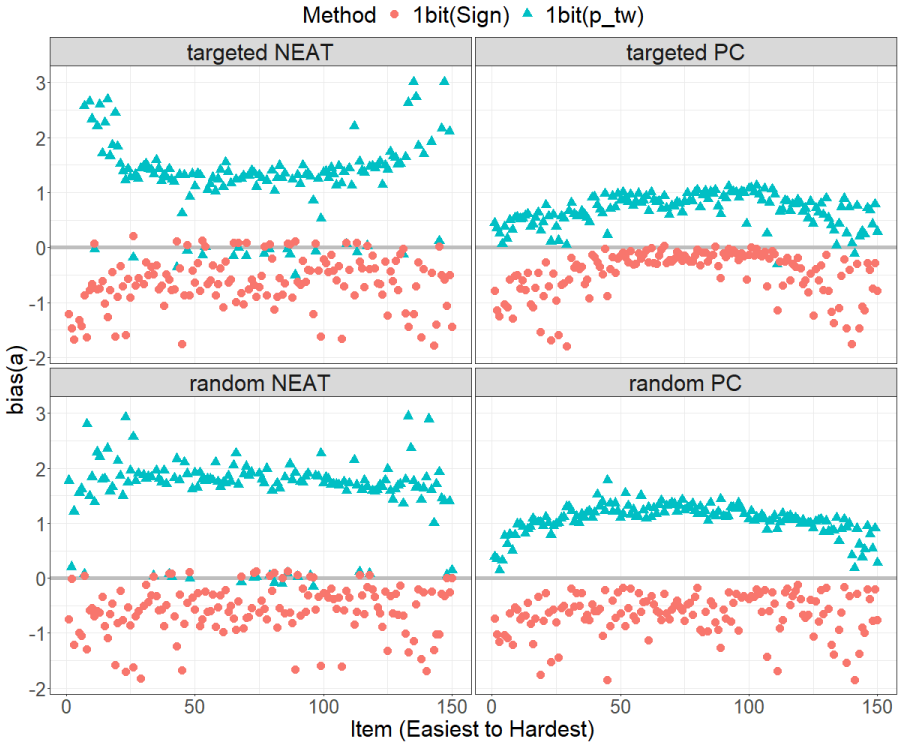


Figure 3.21: Bias(a) per Item of *1bit(Sign)* and *1bit(p_tw)* in Normal Item Bank and $N = 125$

Influence from sample sizes on *1bit(Sign)* and *1bit(p_tw)*.

The results from *1bit(Sign)* and *1bit(p_tw)* were robust in terms of sample sizes. This conclusion was also built on findings from recovery rates, estimations for discrimination parameters, and estimations for difficulty parameters. Firstly, the average recovery rates for *1bit(p_tw)* and *1bit(Sign)* were compared. Table 3.11 shows the results in the targeted NEAT designs. From this table, as sample sizes increased, the average of recovery rates of both *1bit(p_tw)* and *1bit(Sign)* slightly increased, and their standard deviations slightly decreased. This pattern was also shown in the other three incomplete calibration designs (Tables D.1, D.2 and D.3 in the Appendix).

Table 3.11: Recovery Rates Across Sample Sizes in Targeted NEAT Designs

	Method	N = 125	N = 250	N = 500
Uniform	1bit(p_tw)	0.722 (0.010)	0.723 (0.007)	0.726 (0.006)
	1bit(Sign)	0.537 (0.004)	0.548 (0.003)	0.573 (0.003)
Normal	1bit(p_tw)	0.687 (0.010)	0.691 (0.006)	0.695 (0.005)
	1bit(Sign)	0.527 (0.004)	0.535 (0.003)	0.555 (0.003)
Practical	1bit(p_tw)	0.759 (0.008)	0.763 (0.005)	0.767 (0.003)
	1bit(Sign)	0.535 (0.004)	0.546 (0.003)	0.570 (0.003)

Note. The numbers in parentheses are standard deviations.

The influence of sample sizes on bias(b) and RMSE(b) from *1bit(Sign)* and *1bit(p_tw)* were also investigated. Table 3.12 presented the average bias(b) and RMSE(b) across three sample sizes in the practical item bank. From this table, RMSE(b) from *1bit(p_tw)* decreased as sample sizes increased, while RMSE(b) from *1bit(Sign)* did not. In addition, bias(b) from both adaptations with $N = 500$ in practical item bank were presented in Figure 3.22 and showed similar patterns to that with $N = 125$ (Figure 3.11). The main difference was that the range of bias(b) from *1bit(p_tw)* was smaller when $N = 500$ than $N = 25$,

especially in the random NEAT and random PC designs. Similar findings were also observed in uniform and normal item banks (see Tables D.4, D.5, and Figures C.3, C.4 in the Appendix).

Note that *1bit(Sign)* had very large bias(b) and RMSE(b) in the random NEAT design with $N = 250$ and in the targeted NEAT design with $N = 500$. It was due to a few extreme estimations in a few replications. For example, when using the random NEAT design with $N = 250$, *1bit(Sign)* had an extreme estimation for an easy item ($b = -2.461$) with $\hat{b} = 503.478$ in one replication, and it had an extreme estimation for a difficult item ($b = 2.636$) with $\hat{b} = 1070.606$ in another replication. After removing these two extreme estimations, the bias(b) was 0.126, and the RMSE(b) was 5.763. Similarly, after removing seven extreme estimations with absolute deviation of b larger than 1000 in the targeted NEAT design with $N = 500$, *1bit(Sign)* had bias(b) of 0.200 and RMSE(b) of 10.300.

Table 3.12: Bias(b) and RMSE(b) of *1bit(Sign)* and *1bit(p_tw)* Across Sample Sizes in Practical Item Bank

		N = 125		N = 250		N = 500	
Design	Method	Bias(b)	RMSE(b)	Bias(b)	RMSE(b)	Bias(b)	RMSE(b)
Targeted NEAT	1bit(Sign)	-1.901	28.307	0.223	14.414	-19.764	161.881
	1bit(p_tw)	0.077	0.667	0.115	0.517	0.110	0.461
Targeted PC	1bit(Sign)	-0.257	11.840	-0.662	6.519	1.323	18.702
	1bit(p_tw)	0.059	0.824	0.051	0.758	0.027	0.720
Random NEAT	1bit(Sign)	0.510	16.690	-45.390	330.118	0.630	10.364
	1bit(p_tw)	0.091	2.221	0.030	0.508	0.035	0.319
Random PC	1bit(Sign)	-1.874	40.349	-3.681	42.391	1.430	21.868
	1bit(p_tw)	-0.401	4.944	-0.031	0.470	-0.035	0.358

Lastly, sample sizes did not influence bias(a) and RMSE(a) from *1bit(p_tw)* and *1bit(Sign)*. For example, Table 3.13 presents the average bias(a) and RMSE(a) across three sample sizes in the practical item bank, and their results were similar across sample sizes. In particular, the average bias(a) and RMSE(a) of *1bit(p_tw)* slightly decreased as sample sizes increased in all incomplete calibration designs. This pattern was also shown for *1bit(Sign)* except in the random PC designs where its bias(a) and RMSE(a) slightly increased as the sample sizes increased. In addition, Figure 3.23 presents bias(a) per item in practical item bank when

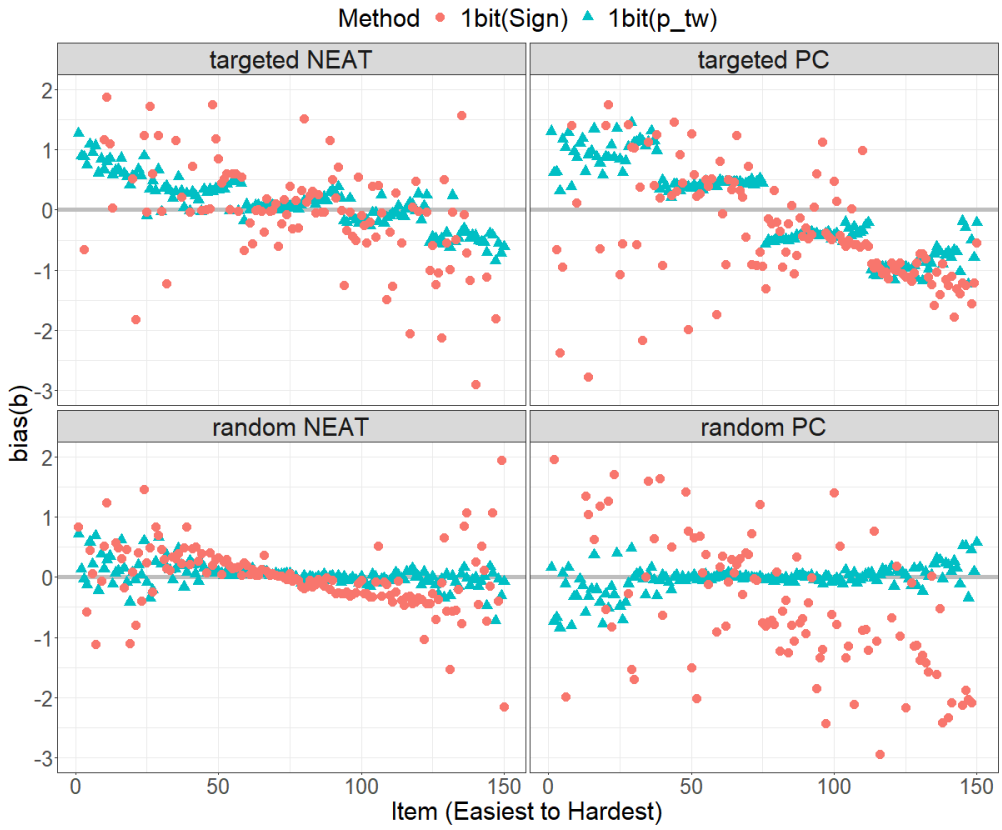


Figure 3.22: Bias(b) per Item of *1bit(Sign)* and *1bit(p_tw)* in Practical Item Bank and $N = 500$

$N = 500$, which showed a similar pattern to the figure for $N = 125$ (Figure 3.12). Similar findings were also observed in uniform and normal item banks (Tables D.6, D.7, and Figures C.5, C.6).

Table 3.13: Bias(a) and RMSE(a) of *1bit(Sign)* and *1bit(p_tw)* Across Sample Sizes in Practical Item Bank

		N = 125		N = 250		N = 500	
Design	Method	Bias(a)	RMSE(a)	Bias(a)	RMSE(a)	Bias(a)	RMSE(a)
Targeted NEAT	1bit(Sign)	-0.894	0.989	-0.850	0.941	-0.768	0.859
	1bit(p_tw)	2.024	3.087	1.752	2.130	1.758	1.922
Targeted PC	1bit(Sign)	-0.691	0.826	-0.714	0.791	-0.795	0.858
	1bit(p_tw)	0.606	0.818	0.587	0.696	0.521	0.596
Random NEAT	1bit(Sign)	-0.732	0.825	-0.610	0.719	-0.438	0.553
	1bit(p_tw)	2.330	4.018	1.799	2.114	1.679	1.771
Random PC	1bit(Sign)	-0.826	1.125	-0.863	0.901	-1.050	1.234
	1bit(p_tw)	1.121	1.549	1.038	1.194	0.996	1.099

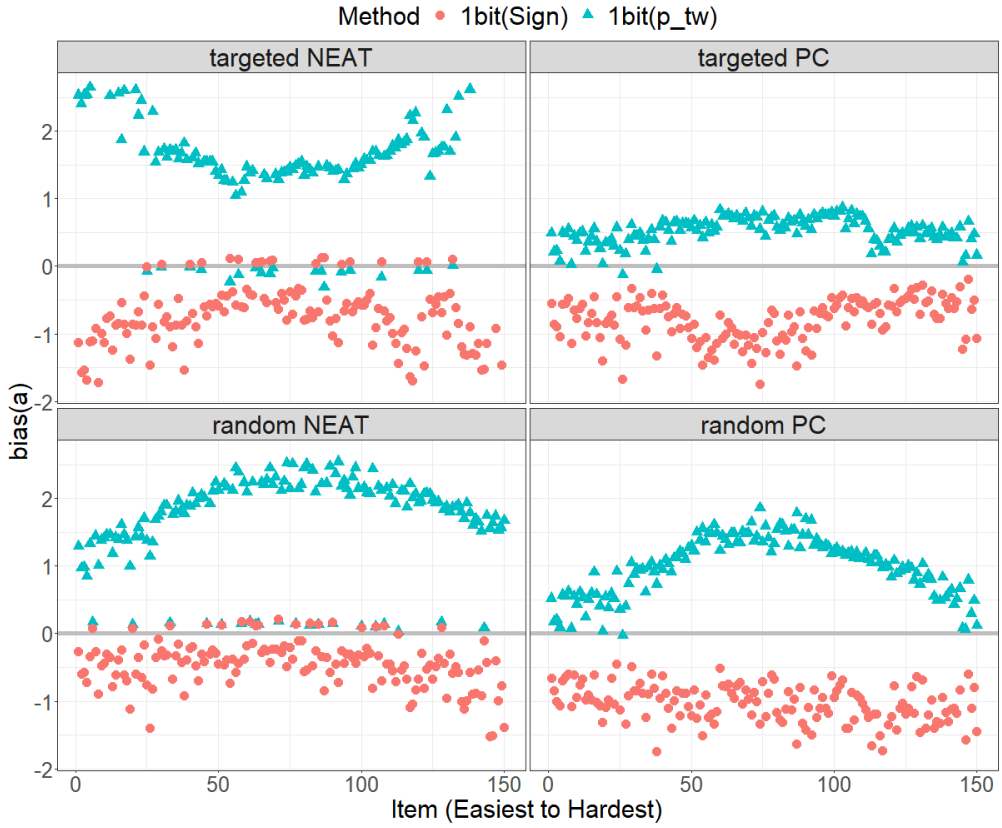


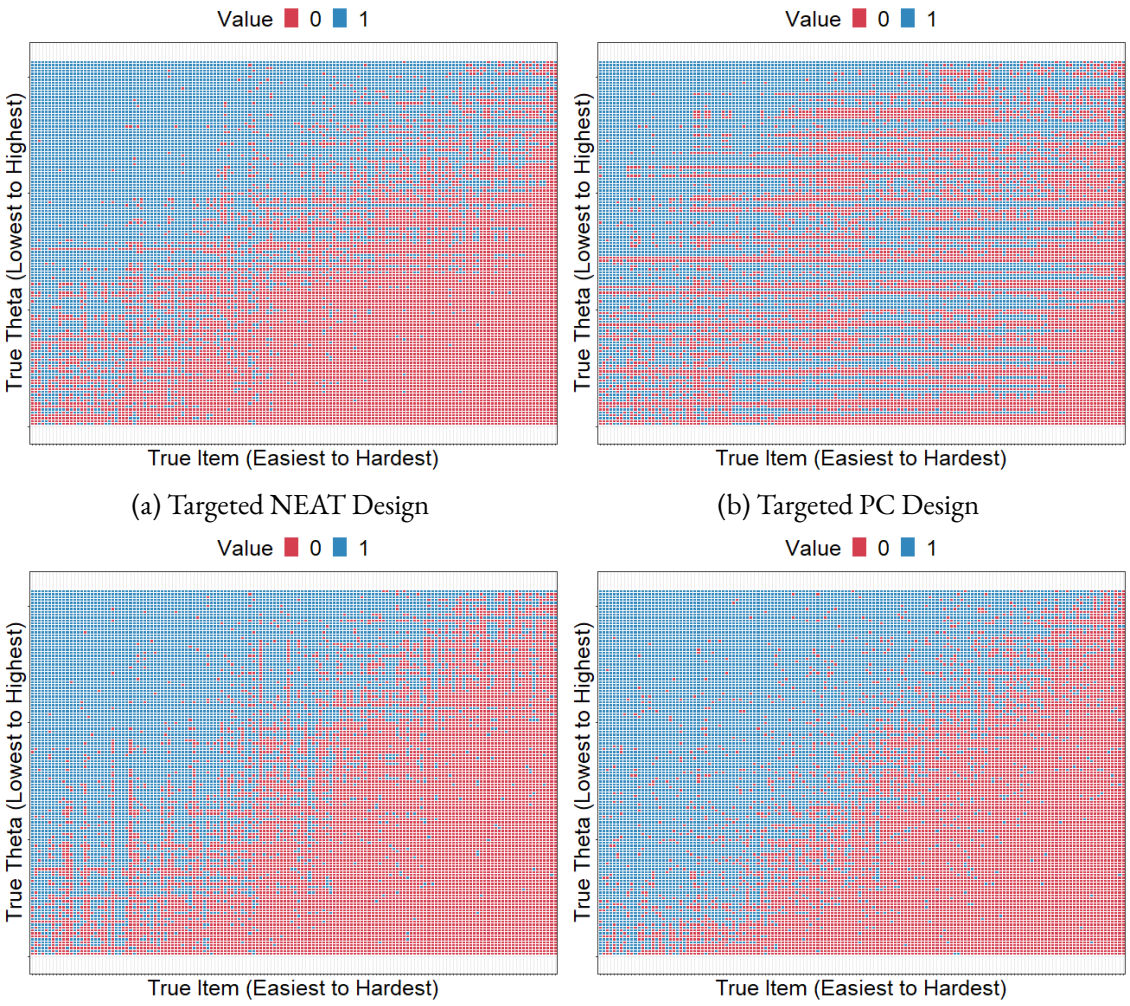
Figure 3.23: Bias(a) per Item of *1bit(Sign)* and *1bit(p_tw)* in Practical Item Bank and $N = 500$

Discussion about *1bit(Sign)* and *1bit(p_tw)*.

From previous findings, *1bit(p_tw)* had higher recovery rates, better bias(b), and worse bias(a) than *1bit(Sign)*. In addition, *1bit(p_tw)* had low recovery rates, large bias(b) and RMSE(b), and small bias(a) and RMSE(a) in targeted PC designs. To understand the reasons behind these observations, a series of investigations were conducted, including computing the item means and person means of the simulated incomplete response matrices, visualizing the imputed response matrices, and visualizing the difference between imputed response matrices.

Imputed response matrices from *1bit(p_tw)* recovered the characteristics of the full response matrices in the targeted NEAT, random NEAT, and random PC designs.

1bit(p_tw) had higher recovery rates than *1bit(Sign)* in all four incomplete calibration designs, three item banks, and three sample sizes. Remember that *1bit(Sign)* produced the imputed response matrices with evenly distributed correct and incorrect responses regardless item difficulties or examinees' ability levels (Figures 3.2c, 3.2d, 3.3c, and 3.3d). Thus, it was not surprising that *1bit(Sign)* resulted in recovery rates slightly larger than 0.5. On the contrary, Figures 3.24a, 3.24b, 3.24c, and 3.24d present the imputed response matrices using *1bit(p_tw)* in four incomplete calibration designs, respectively. The items were ordered from the easiest to the hardest on the x-axis, and the examinees are ordered by their abilities from the lowest to the highest. In the targeted NEAT, random NEAT, and random PC designs, the imputed response matrices of *1bit(p_tw)* had the gradual transitions from more blue mosaics (incorrect responses) to more red mosaics (correct responses) from top left corner to the bottom right corner of the imputed response matrices, which was similar to the full response matrix shown in Figure 3.1b.



(a) Targeted NEAT Design

(b) Targeted PC Design

(c) Random NEAT Design

(d) Random PC Design

Figure 3.24: Imputed Response Matrices From *1bit(p_tw)* in Practical Item Bank

1bit(p_tw) did not recover the characteristics of the full response matrices in the targeted PC designs.

Though *p_tw* significantly improved the imputation results of 1-bit matrix completion than *Sign* in the targeted NEAT, random NEAT and random PC designs, it did not work in the targeted PC designs. As shown in Figure 3.24b, the imputed response matrix did not show the gradual transition pattern of the full response matrix. The advantage of *1bit(p_tw)* over *1bit(Sign)* is from the cut-off point TW_{ij} which considers both item and person effects. In particular, the observed item means and person means were used to capture the item and person effects. Recall that item mean was computed as the average score for each item across the observed responses, and person mean was computed as the average score for

each examinee across the observed responses. Figures 3.25 and 3.26 present the item means and person means across incomplete calibration designs in the practical item bank with $N = 125$. From Figure 3.25, item means were larger for easier items and lower for harder items in all incomplete calibration designs, and thus they were good indicators for item difficulties. From Figure 3.26, person means were larger as examinees' abilities increase in random NEAT and random PC designs. The person means were similar across examinees with medium abilities in targeted NEAT design. In targeted PC designs, the person means did not strictly increase as θ increased for examinees of medium ability. Thus, person means from the targeted PC designs were not good indicators for examinees with medium ability levels in targeted PC designs. Thus, the thresholds of TW for these examinees would not reflect their person effects, which led to even distribution of correct and incorrect responses for those examinees.

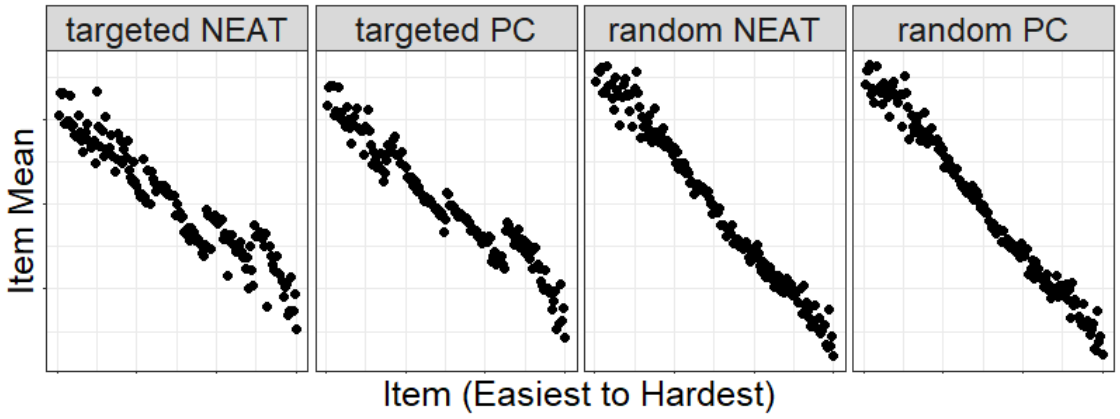


Figure 3.25: Observed Item Means Across Item Banks and Incomplete Calibration Designs

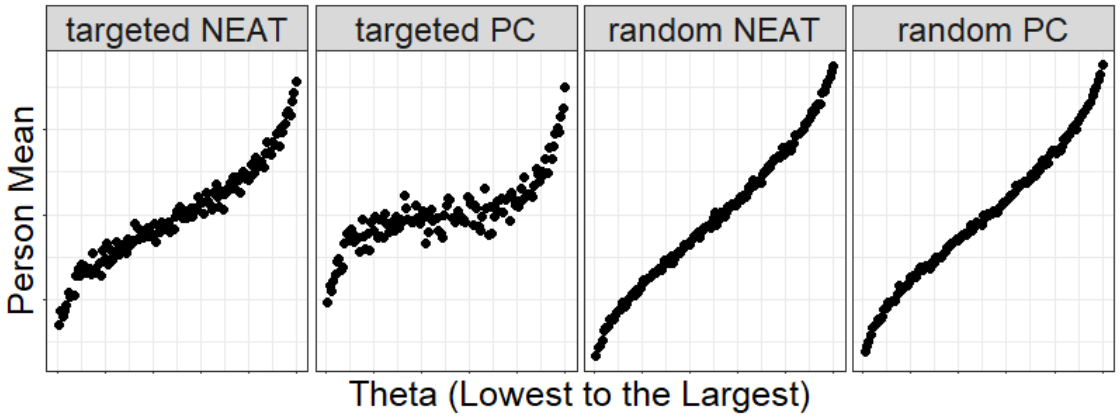


Figure 3.26: Observed Person Means Across Item Banks and Incomplete Calibration Designs

1bit(p_tw) provided large bias(b) for easy and hard items.

1bit(p_tw) had high recovery rates for the easy and difficult items (e.g., Figure 3.10). However, bias(b) for those items were larger than the medium-difficulty items especially in random NEAT and random PC designs. One possible reason was that there were few examinees of low and high θ providing informative responses to calibrate easy and difficult items. Figure 3.27 presents the histograms of item difficulties from the practical item bank and simulated θ from one replication, and there were only a few examinees had θ that were close to the easiest and hardest items. Thus, though *1bit(p_tw)* produced higher recovery rates for easy and difficult items than medium-difficult items, bias(b) for those items are still larger.

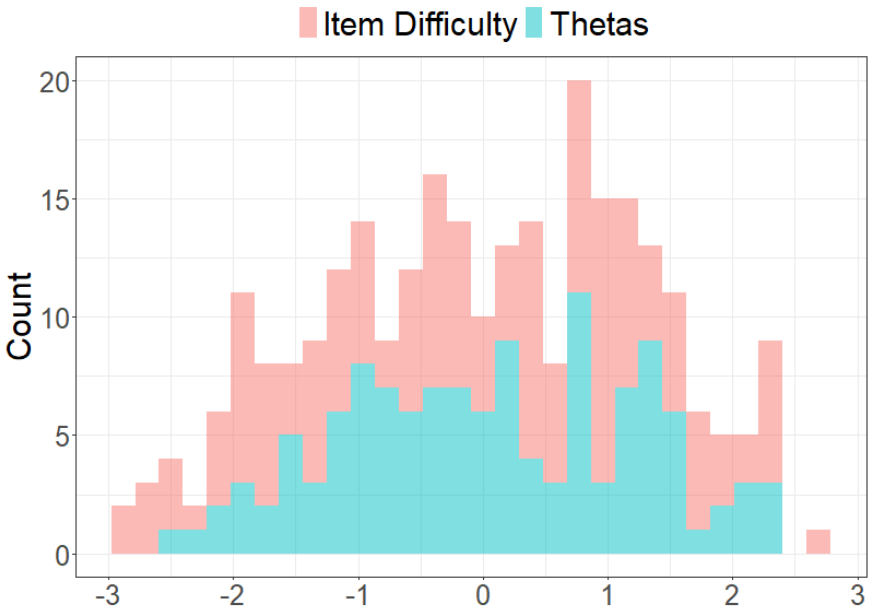
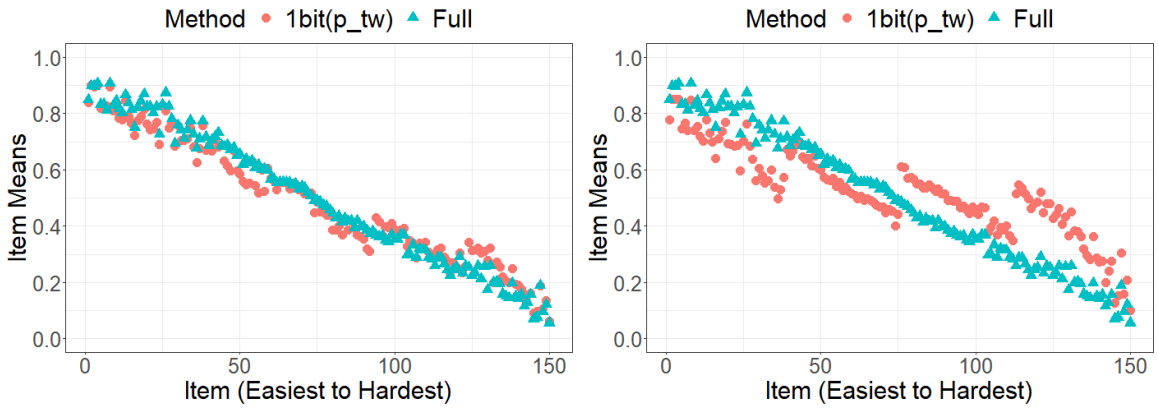


Figure 3.27: Item Difficulty vs. Theta

Furthermore, 1bit(p_tw) overestimated b for easy items and underestimated b for hard items in the targeted NEAT and targeted PC designs (Figure 3.11). Another possible reason for this pattern was the characteristics of the imputed response matrix. *1bit(p_tw)* imputed more incorrect responses for easy items and more correct responses for difficult items than the original full response matrix. Figure 3.28 presents the item means from *1bit(p_tw)* imputed response matrix and that from full response matrix, where the item means for easy items were lower than the full response matrix, and the item means for hard items

were higher. In particular, this characteristic was more obvious in targeted PC design than targeted NEAT designs, which explains the severer overestimations for easy items and underestimations for hard items in targeted PC designs.



(a) Targeted NEAT Designs

(b) Targeted PC Designs

Figure 3.28: Item Means for *1bit(p_tw)* Compared to Full Response Matrix using Practical Item Bank with $N = 125$.

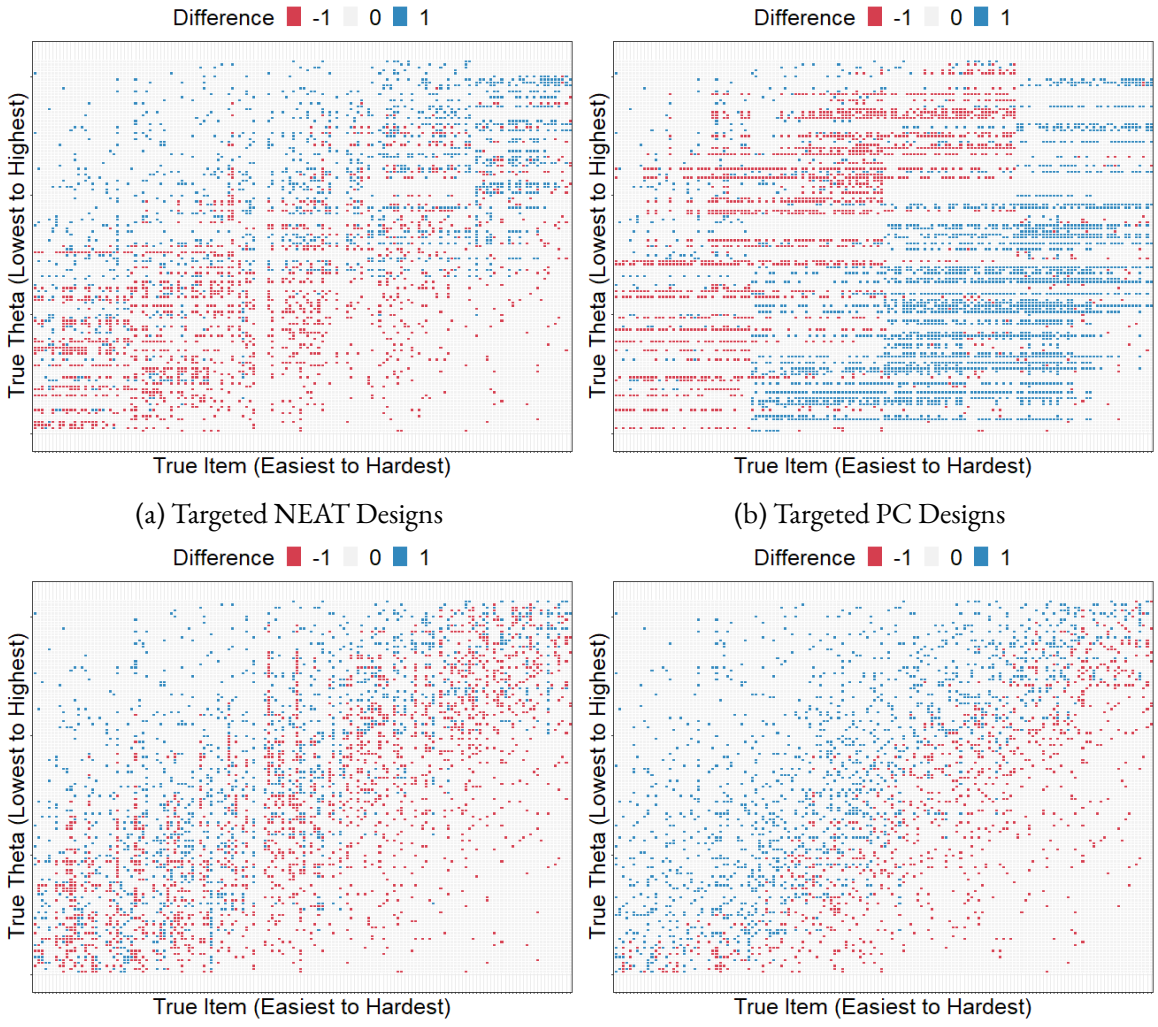
1bit(p_tw) had worse RMSE(a) and bias(a) than *1bit(Sign)*.

As demonstrated before, *1bit(p_tw)* better recovered the response matrix than *1bit(Sign)*. It was then expected that *1bit(p_tw)* had higher recovery rates and better estimations of item parameters than *1bit(Sign)*. However, *1bit(p_tw)* overestimated a and had larger RMSE(a) than *1bit(Sign)*. The difference between the imputed response matrix using *1bit(p_tw)* and the full response matrix using practical item bank with $N = 125$ was presented in Figure 3.29. The items were ordered by difficulty on x-axis, and the examinees were ordered by θ on the y-axis. The grey mosaics indicated the imputed responses were the same as the full response matrix. A red mosaic (-1) indicated that *1bit(p_tw)* imputed an incorrect response while the original response was correct. On the contrary, the blue mosaic (1) indicated that *1bit(p_tw)* imputed a correct response while the original response was incorrect. From this Figure, *1bit(p_tw)* imputed more incorrect responses to examinees of low abilities especially for easy items (more red mosaics on the bottom left corner of the matrix), and more correct responses to examinees of high abilities especially for difficult items (more blue mosaics on the upper right corner of the matrix). Recall that the definition of the

item discrimination parameter is the maximum slope for the item characteristic curve, which is when the probability of correctness is 0.5. That is to say, if the item discrimination parameter is overestimated, the chance for an examinee of low ability to answer that item correctly is lower and the chance for an examinee of high ability to answer that item correctly is higher. This pattern was shown in Figure 3.29 and thus explained the overestimation for the item discrimination parameters using *1bit(p_tw)*.

Similar patterns were also shown in the random NEAT and random PC designs (Figures 3.29c and 3.29d). However, different from the targeted NEAT designs, the imputed response matrix of *1bit(p_tw)* in the random NEAT and the random PC designs had a boundary between blue and red mosaics along the diagonal of the matrix. That is, both item difficulties and examinees' abilities had equal impacts on the imputed response matrix of *1bit(p_tw)* in the random NEAT and the random PC designs for all examinees and items, while they did not in the targeted NEAT designs. It was expected because the person means and item means from the random NEAT and the random PC designs were more correlated examinees' abilities and item difficulties than in the targeted NEAT designs (Figures 3.25 and 3.26). This explained the different patterns of bias(a) in these three incomplete calibration designs.

The imputed response matrix was very different from the full response matrix in the targeted PC design. In Figure 3.29b, the difference between the imputed response matrix and the full response matrix showed a clear structure aligned with the targeted PC design. In addition, this difference response matrix was different from the other three incomplete calibration designs. Thus, it was not surprising that *1bit(p_tw)* had bias(a) closer to zero in targeted PC design than the other three incomplete calibration designs, though its bias(b) was relative large with clear systematic pattern.



(a) Targeted NEAT Designs

(b) Targeted PC Designs

(c) Random NEAT Designs

(d) Random PC Designs

Figure 3.29: Difference Between *1bit(p_tw)* Imputed Matrix and Full Response Matrix using Practical Item Bank with $N = 125$.

3.4.2 Results of Adapted Structured Matrix Completion.

The results of eight adaptations for structured matrix completions, *SMC(1+p_tw)*, *SMC(p0+p_tw)*, *SMC(p1+p_tw)*, *SMC(p2+p_tw)*, *SMC(1+Sign)*, *SMC(p0+Sign)*, *SMC(p1+Sign)*, and *SMC(p2+Sign)* were summarized in this section. The results in the practical item bank with the sample size of 125 were firstly presented. In particular, their recovery rates and estimations for item difficulty and discrimination parameters were

presented in details. Then, the results in the other two item banks and two sample sizes were presented to examine the influence of item banks and sample sizes.

Recovery rates from adapted structured matrix completion methods in the practical item bank with $N = 125$.

Figure 3.30 presents the box plots for the recovery rates of all eight adapted structured matrix completion across four incomplete calibration designs in the practical item bank with the sample size of 125. The recovery rates of the eight adaptations showed similar patterns in the targeted NEAT, random NEAT, and random PC designs. In particular, *SMC(p1+p_tw)* had the highest recovery rate among the eight adaptations. *SMC(1+p_tw)*, *SMC(1+Sign)*, and *SMC(p0+p_tw)* had similarly high recovery rates. Across these three incomplete calibration designs, when fixing the converting methods, *p_tw* had higher recovery rates than *Sign*. This finding was consistent with adapted 1-bit matrix completion. Furthermore, when fixing the dichotomizing method as *p_tw*, *1*, *p0* and *p1* had better performance than *p2*. In addition, the recovery rates were generally lower with wider ranges in the targeted PC designs than in the other three incomplete calibration designs. Among the eight adaptations, *SMC(p1+p_tw)* and *SMC(p2+p_tw)* had relative high recovery rates in the targeted PC designs.

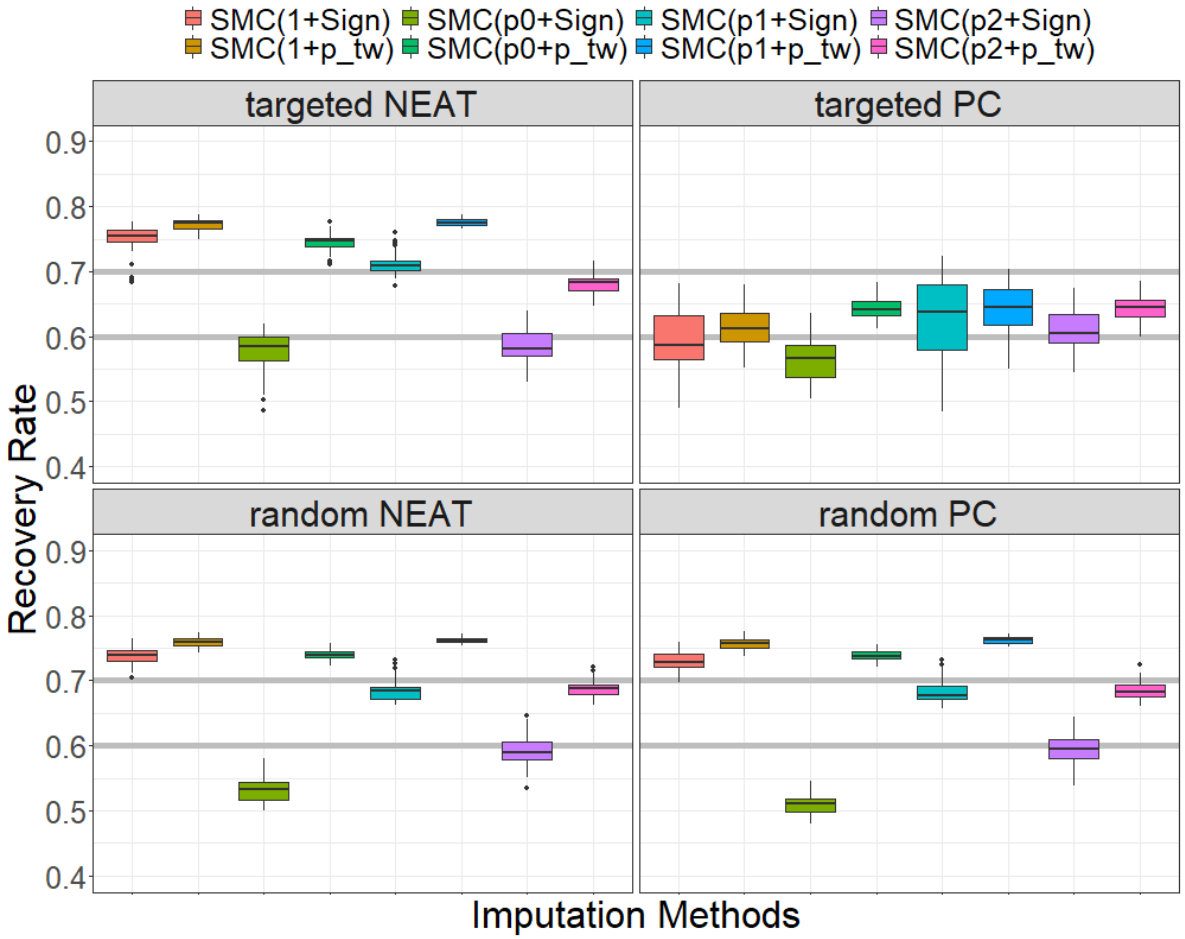
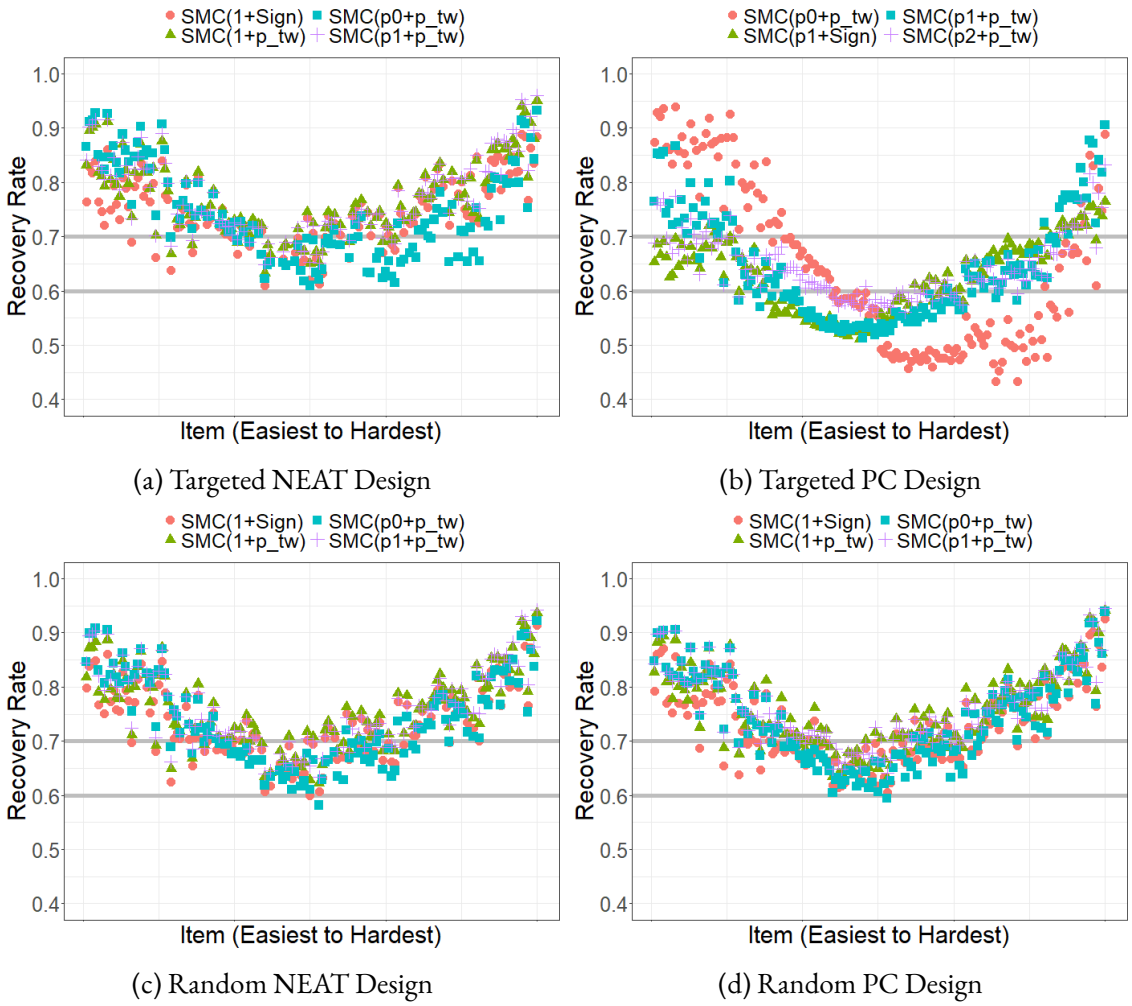


Figure 3.30: Recovery Rates of Adapted Structured Matrix Completion in Practical Item Bank and $N = 125$

To further investigate the different performances among the eight adapted structured matrix completion methods at the item-level, the recovery rates per item from the top four adaptations were presented in Figure 3.31. Generally, the four adaptations in targeted NEAT, random NEAT, and random PC designs presented similar recovery rates pattern. That is, their recovery rates for easy and difficult items were higher than those for medium-difficulty items. In addition, *SMC(po+p_tw)* provided higher recovery rates for easy items and lower recovery rates for hard items than the other three adaptations in targeted PC design (Figure 3.31b).



(a) Targeted NEAT Design

(b) Targeted PC Design

(c) Random NEAT Design

(d) Random PC Design

Figure 3.31: Recovery Rates per Item of Selected Modified Structured Matrix Completion in Practical Item Bank with Sample Size of 125

Bias(b) and RMSE(b) from adapted structured matrix completion methods in practical item bank with $N = 125$.

The average of bias(b) and RMSE(b) of all items were first summarized in Table 3.14. In the targeted NEAT, targeted PC, and random PC designs, the adaptation that produced the smallest bias(b) also produced relatively small RMSE(b), though its RMSE(b) was not the smallest. In the random NEAT designs, *SMC(p1+Sign)* had the smallest bias(b) of 0.017, but it had a large RMSE(b) of 9.025. Figure 3.32 presents its bias(b) per item along with *SMC(p2+p_tw)*, which had the smallest RMSE(b) in the random NEAT designs. The items were ordered from the easiest to the hardest on the x-axis. *SMC(p1+Sign)* had

a few extreme estimations for very easy and very difficult items, and *SMC(p2+p_tw)* produced bias(b) close to zero for most of the items. Thus, though *SMC(p1+Sign)* has smaller bias(b), *SMC(p2+p_tw)* was considered better in terms of item difficulty parameter estimations in the random NEAT designs.

Table 3.14: Average Bias(b) and RMSE(b) from Eight Adapted Structured Matrix Completion Methods When $N = 125$ in Practical Item Bank

	Targeted NEAT		Targeted PC		Random NEAT		Random PC	
Method	Bias(b)	RMSE(b)	Bias(b)	RMSE(b)	Bias(b)	RMSE(b)	Bias(b)	RMSE(b)
1+Sign	0.565	9.156	0.485	5.954	0.275	4.465	0.165	3.745
1+p_tw	**0.062**	1.155	**0.074**	1.277	0.122	2.422	**-0.065**	2.457
p0+Sign	-0.889	40.678	2.322	11.592	0.758	44.754	3.198	16.696
p0+p_tw	-0.726	2.132	-7.001	54.459	-2.219	23.759	1.542	23.918
p1+Sign	0.084	10.779	0.993	18.687	**0.017**	9.025	0.572	8.302
p1+p_tw	0.122	1.047	0.951	7.391	0.534	5.686	-0.673	9.178
p2+Sign	0.094	3.338	0.229	1.778	-0.148	6.402	0.411	**0.794**
p2+p_tw	0.172	**1.045**	0.237	**0.947**	0.189	**1.189**	0.418	0.865

Note. The bold numbers indicated the bias(b) that is close to 0 or the smallest RMSE(b) in each incomplete calibration design.

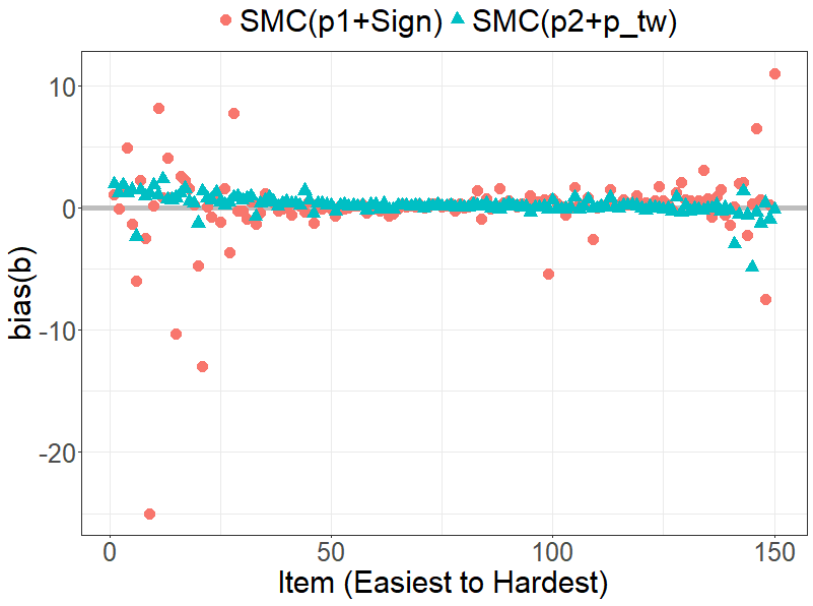


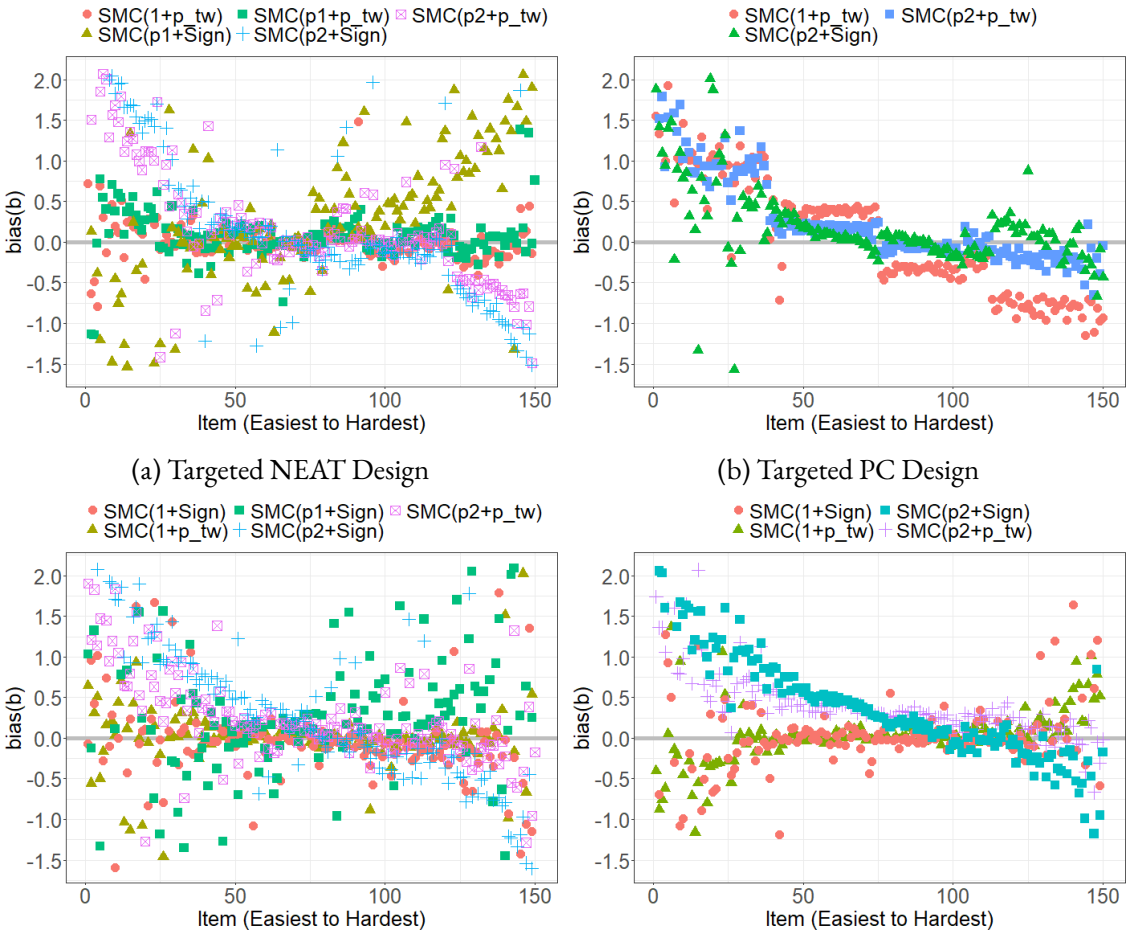
Figure 3.32: Bias(b) Per Item in Random NEAT Designs using *SMC(p1+Sign)* and *SMC(p2+p_tw)* in Practical Item Bank with $N = 125$

To further compare the estimated item difficult parameters from the adapted structured matrix completion methods at the item-level, their bias(b) per item were presented in Figure 3.33. The selected adaptations had either small bias(b) or RMSE(b) under each incomplete calibration design given Table 3.14. Note that the y-axis was fixed in the range of -1.5 to 2, and thus the extreme values were not shown in this set of figures. The corresponding figures including all values were presented in Figures C.7 in the Appendix.

Bias(b) of these selected adapted structured matrix completion showed different patterns in different incomplete calibration designs. In the targeted NEAT designs, *SMC(1+p_tw)* and *SMC(p1+p_tw)* had bias(b) closer to zero for most items than the other three adaptations (Figure 3.33a). In the targeted PC designs, all three selected adaptations overestimated b for easy items, and *SMC(1+p_tw)* and *SMC(p2+p_tw)* also underestimated b for difficult items (Figure 3.33b). The bias(b) from all three adaptations showed systematic patterns corresponding to the designed test forms. Among the five selected adaptations in the random NEAT designs, *SMC(1+Sign)* and *SMC(1+p_tw)* had bias(b) closer to zero than the other three adaptations for most items (Figure 3.33c). In addition, their bias(b) had smaller variations for items of medium difficulty than the other items. In the random PC designs, the bias(b) from *SMC(p2+Sign)* and *SMC(p2+p_tw)* decreased as item difficulties increased (Figure 3.33d), which was similar to the patterns in the targeted PC designs. On the contrary, bias(b) from *SMC(1+Sign)* and *SMC(1+p_tw)* increased as item difficulty increased. Though *SMC(1+Sign)* and *SMC(1+p_tw)* had bias(b) closer to zero for medium-difficulty items, they had a few extreme estimations for b for very easy and difficult items (Figure C.7d in the Appendix). To summarize, *SMC(1+p_tw)* produced relatively good estimations for item difficulty parameters in all four incomplete calibration designs.

Bias(a) and RMSE(a) from adapted structured matrix completion methods in practical item bank with $N = 125$.

Table 3.15 presents the average bias(a) and RMSE(a) over replications in practical item bank with $N = 125$. Same as the findings in Table 3.14, the adaptations with the smallest bias(a) were not always



(a) Targeted NEAT Design

(b) Targeted PC Design

(c) Random NEAT Design

(d) Random PC Design

Figure 3.33: Bias(b) per Item of Selected Adapted Structured Matrix Completion in Practical Item Bank with Sample Size of 125

with the smallest RMSE(a). The large RMSE(a) for adaptations with small bias(a) was due to the extreme estimations for a few items.

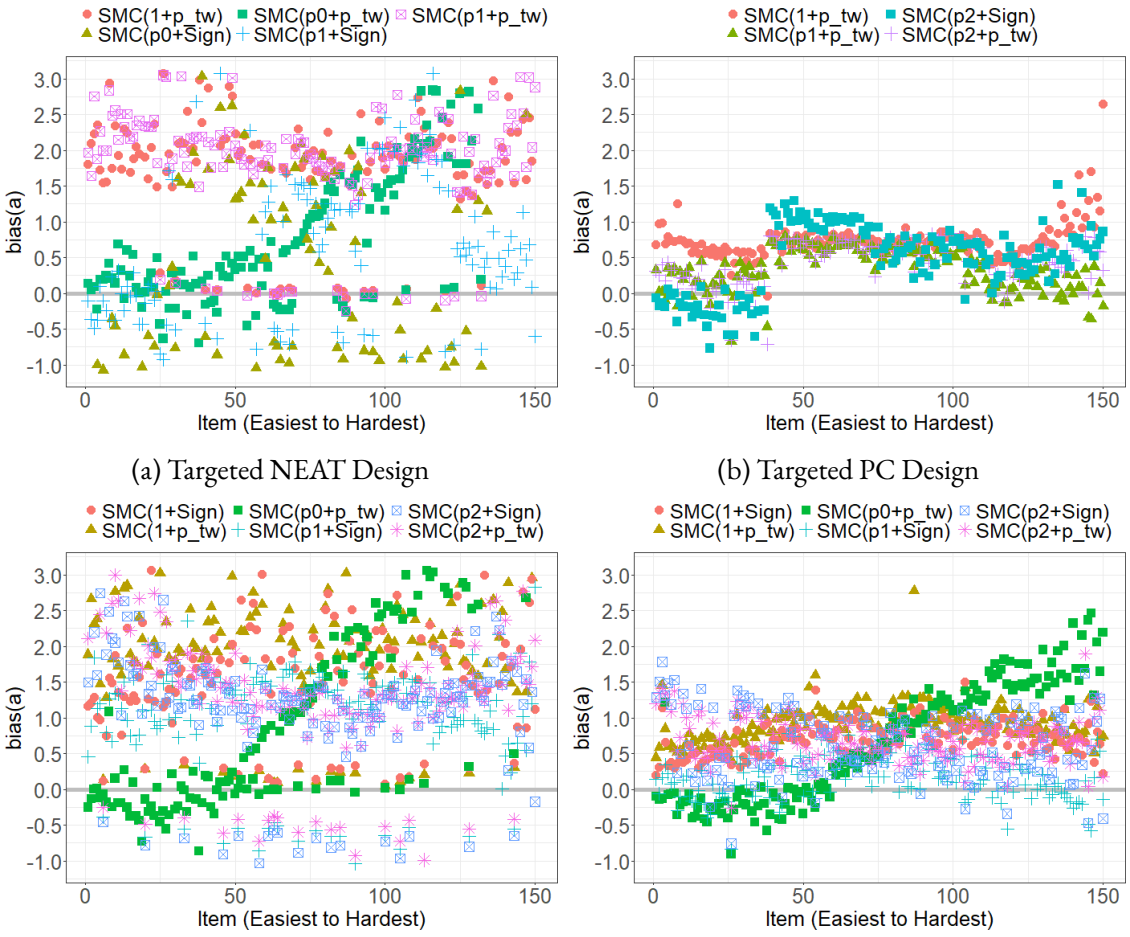
Table 3.15: Bias (a) and RMSE(a) from Eight Adapted Structured Matrix Completion Methods When $N = 125$ in Practical Item Bank

	Targeted NEAT		Targeted PC		Random NEAT		Random PC	
Method	Bias(a)	RMSE(a)	Bias(a)	RMSE(a)	Bias(a)	RMSE(a)	Bias(a)	RMSE(a)
1+Sign	1.664	3.844	0.639	1.595	2.274	6.852	0.705	1.353
1+p_tw	1.857	2.917	0.736	1.051	2.208	4.090	0.931	1.462
p0+Sign	**1.124**	8.207	1.281	3.221	1.511	10.392	**-0.030**	4.767
p0+p_tw	1.475	**2.241**	0.621	1.203	1.473	2.412	0.676	1.239
p1+Sign	1.466	4.057	0.807	3.484	**0.981**	2.076	0.174	**0.777**
p1+p_tw	1.828	2.650	**0.331**	**0.651**	1.718	2.740	0.608	1.082
p2+Sign	2.178	5.023	0.459	1.426	1.075	**2.049**	0.619	0.908
p2+p_tw	1.833	3.276	0.445	0.743	1.364	2.277	0.685	0.960

Note. The bold numbers indicated the bias(a) that is closest to 0 or the smallest RMSE(a) in each incomplete calibration design.

Bias(a) per item of the selected adapted structured matrix completion methods were presented in the Figure 3.34. These selected methods either had small bias(a) or RMSE(a) in the Table 3.15. In addition, *SMC(1+p_tw)* and *SMC(1+Sign)* were also plotted in random NEAT and random PC designs. Though their bias(a) and RMSE(a) were not among the few smallest in random NEAT and random PC designs, their bias(b) and RMSE(b) were relatively good. Note that the range for y-axis in this set of figures was fixed from -1 to 2.5 to visualize the nuances among the selected methods. The corresponding figures without fixing the y-axis were shown in Figure C.8 in Appendix.

In the targeted NEAT designs, the bias(a) from *SMC(p0+p_tw)* was close to zero for easy items and increased as items' difficulties increased (Figure 3.34a). In addition, *SMC(1+p_tw)* and *SMC(p1+p_tw)* had similar and relatively large bias(a) for most of items. Though *SMC(p1+Sign)* showed smaller bias(a) in this figure, it had extreme estimations for some items (Figure C.8a). In targeted PC design, the bias(a) from four selected adaptations had similar systematic patterns of bias(a) per item according to the designed test forms, and *SMC(p1+p_tw)* had bias(a) that were closest to zero (Figure 3.34b). In the random NEAT



(a) Targeted NEAT Design

(b) Targeted PC Design

(c) Random NEAT Design

(d) Random PC Design

Figure 3.34: Bias(a) per Item of Selected Adapted Structured Matrix Completion Methods in Practical Item Bank with Sample Size of 125

designs, no systematic bias(a) was shown for most adaptations, while bias(a) from *SMC(p0+p_tw)* slightly increased as item difficulties increased (Figure 3.34c), which was similar in the targeted NEAT design. In addition, *SMC(p1+Sign)*, *SMC(p2+Sign)*, and *SMC(p2+p_tw)* overestimated a for many items and also underestimated a for some items. *SMC(1+Sign)* and *SMC(1+p_tw)* overestimated most of the items. Bias(a) from *SMC(p0+p_tw)* in the random PC designs increased as item difficulties increased (Figure 3.34d). This pattern of *SMC(p0+p_tw)* was also shown in targeted NEAT and random NEAT designs. To summarize, the best adaptations in terms of estimations for item discrimination parameters were different under different incomplete calibration designs. In particular, *SMC(p1+p_tw)* had relatively good estimations for discrimination parameters in targeted NEAT and targeted PC designs, and *SMC(p1+Sign)* had relatively good estimations in random NEAT and random PC designs.

Influence from item banks on adapted structured matrix completion methods.

Item banks did not show much influence on the performance of the eight adaptations for structured matrix completion in terms of recovery rates, estimations for item difficulty and discrimination parameters. Firstly, recovery rates from the 8 adaptations in the uniform and normal item banks had the same patterns as those in practical item banks. For example, Figures 3.35 presents the recovery rates in the targeted NEAT designs across three item banks. *SMC(1+p_tw)* and *SMC(p1+p_tw)* had the largest recovery rates in all three item banks, and the recovery rates of all adaptations in practical item bank were generally higher than the other two item banks. The above two patterns were also shown in random NEAT and random PC designs (Figures C.9 and C.10 in the Appendix). In addition, the recovery rates of the eight adaptations showed similar patterns across item banks in targeted PC designs where *SMC(p0+p_tw)*, *SMC(p1+Sign)*, and *SMC(p1+p_tw)* (Figure 3.36). In addition, their recovery rates were similar in all three item banks.

Secondly, bias(b) and RMSE(b) from the uniform and normal item banks were similar to those from the practical item banks. The average bias(b) and RMSE(b) across items for uniform and normal item banks were presented in Tables D.8 and D.9 in the Appendix. The best adaptations in the uniform and normal item banks were selected given those two tables under each incomplete calibration design. The bias(b) per item from best candidate adaptations in the targeted NEAT designs were presented in Figure

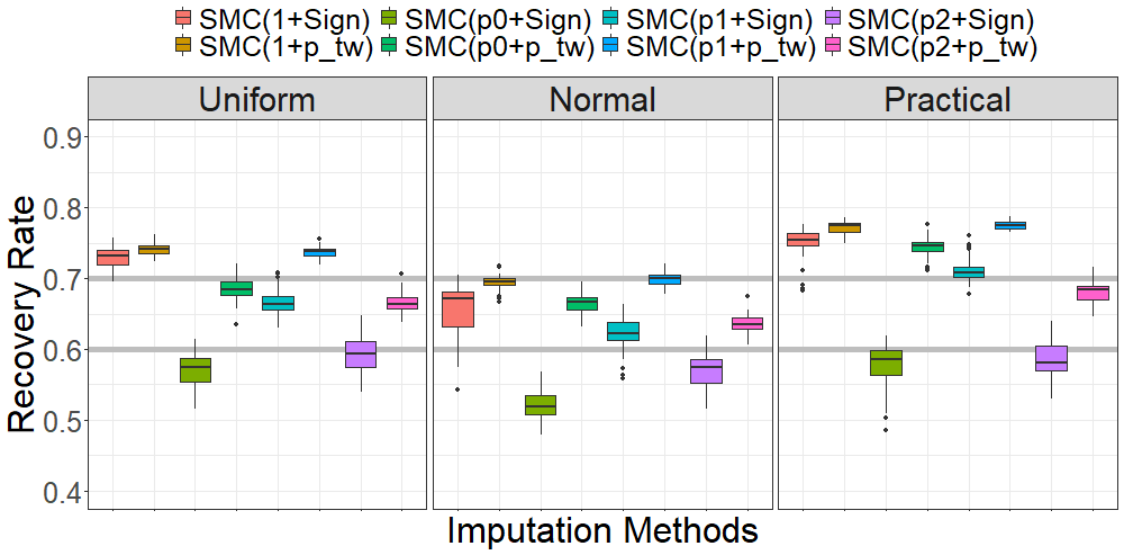


Figure 3.35: Recovery Rates for Modified Structured Matrix Completion Across Item Banks in Targeted NEAT Designs and $N = 125$

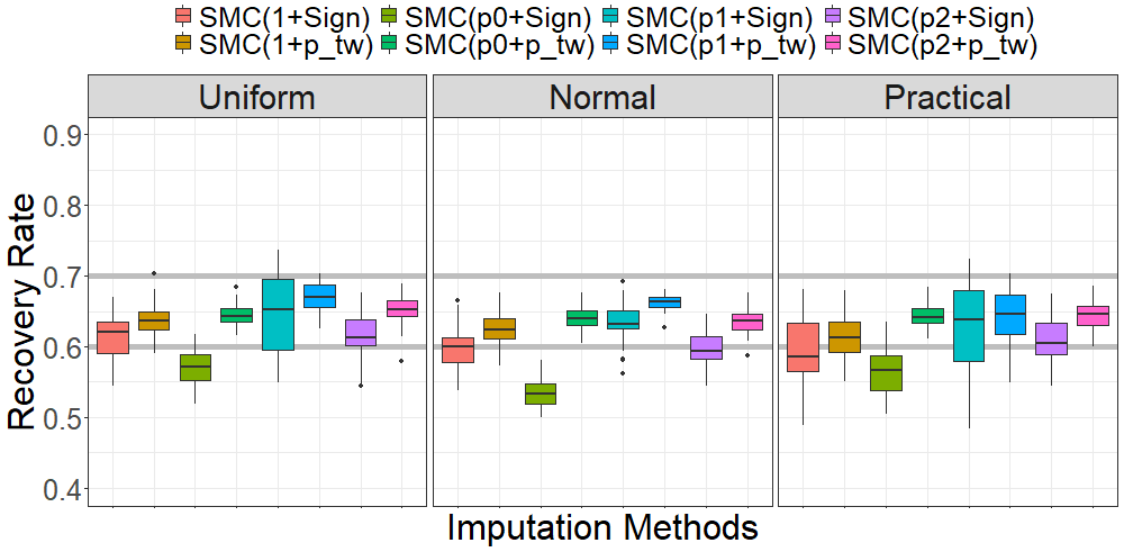
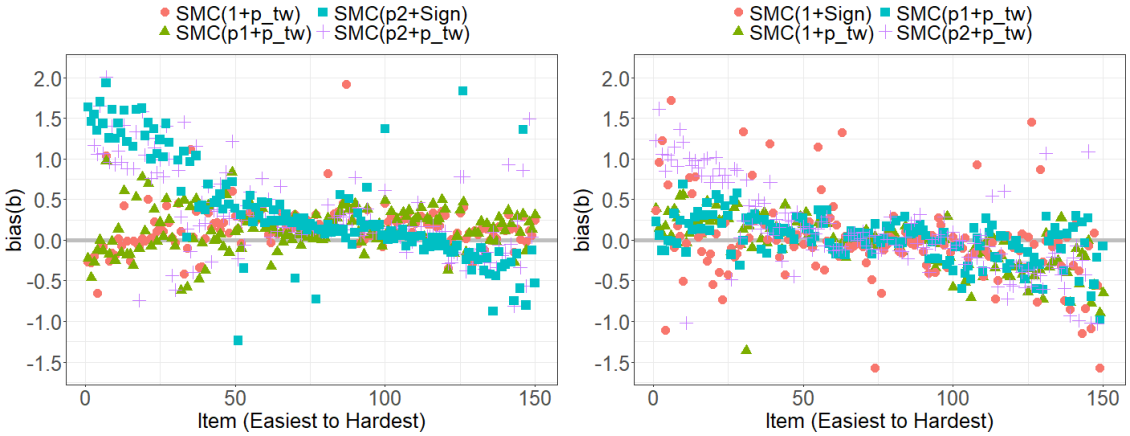


Figure 3.36: Recovery Rates of Modified Structured Matrix Completion Across Item Banks in Targeted PC Designs and $N = 125$

3.37. Similarly to the results in practical item bank (Figure 3.33a), *SMC(p1+p_tw)* had the bias(b) close to zero in both uniform and normal item banks. In the targeted PC designs, though the patterns of bias(b) from the best adaptations were different across three item banks (Figures 3.33b and 3.38), they all showed systematic bias(b) aligned with the designed test forms. In addition, *SMC(p1+p_tw)* had the bias(b) that were the closet to zero among the selected adaptations. In the random NEAT designs, *SMC(1+p_tw* had the best bias(b) in both uniform and normal item banks (Figures 3.39), which was consistent with the findings in the practical item bank. In addition, similar results were also shown in random PC designs (Figure 3.40).

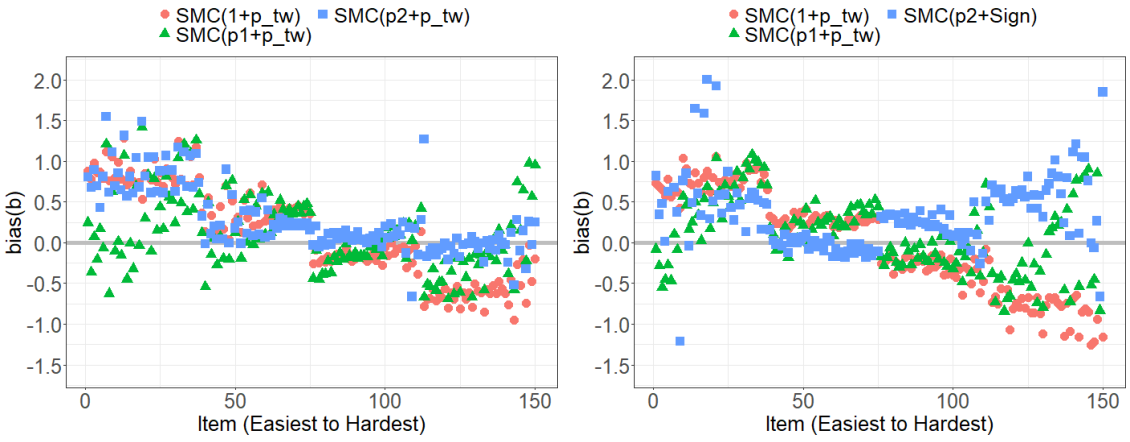


(a) Uniform Item Bank

(b) Normal Item Bank

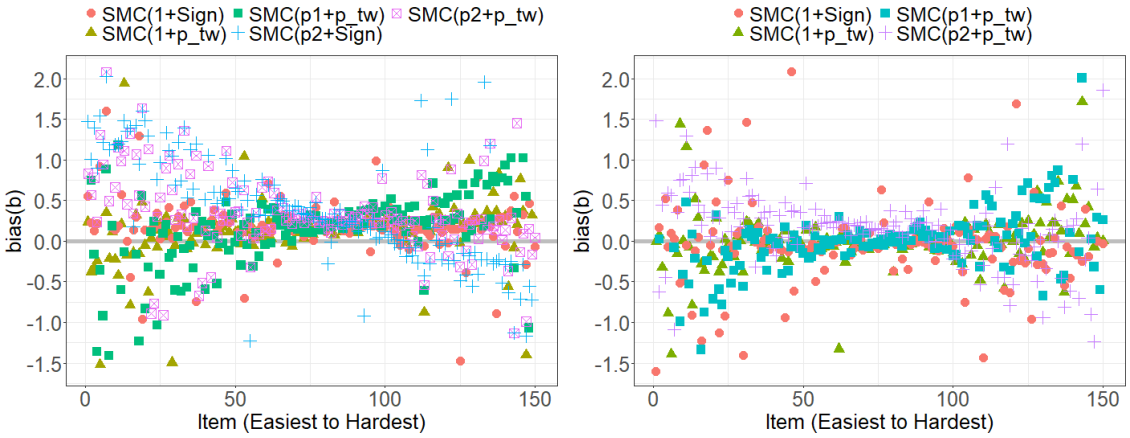
Figure 3.37: Bias(b) per Item of Selected Adapted Structured Matrix Completion Methods in Targeted NEAT Design with Sample Size of 125

Lastly, the bias(a) and RMSE(a) were similar across item banks. The average bias(a) and RMSE(a) from all eight adaptations for structured matrix completion are presented in Tables D.10 and D.11 in the Appendix. Similar to the investigation for practical item bank, the bias(a) per item from the selected adaptations from uniform and normal item banks were plotted for four incomplete calibration designs respectively (Figures 3.41 to 3.44). In targeted NEAT and targeted PC designs, *SMC(p1+p_tw)* provided slightly better bias(a) than the other selected adaptations in all three item banks (Figures 3.34a, 3.34b, 3.41, and 3.42). In random NEAT and random PC designs, *SMC(p1+Sign)* had bias(a) closest to zero for most of items in all three item banks (Figure 3.34c, 3.34d, 3.43, and 3.44).



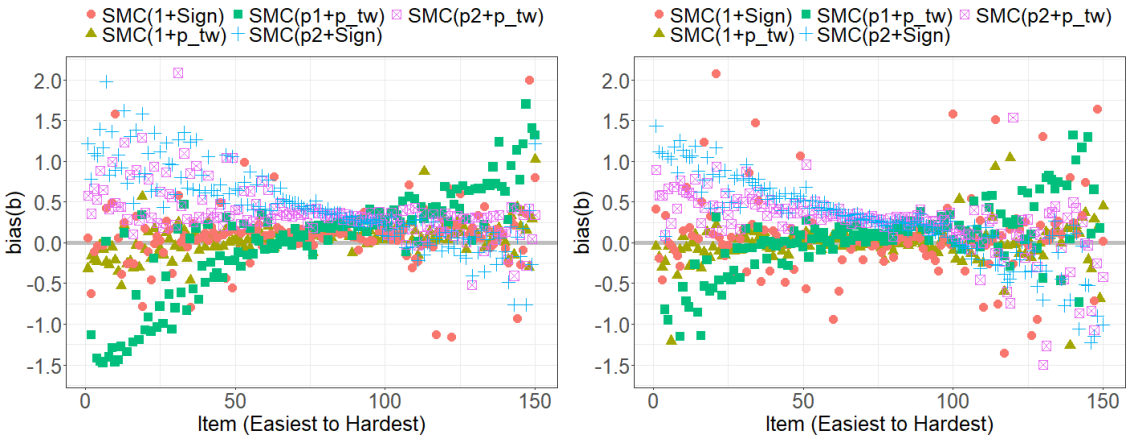
(a) Uniform Item Bank (b) Normal Item Bank

Figure 3.38: Bias(b) per Item of Selected Adapted Structured Matrix Completion Methods in Targeted PC Design with Sample Size of 125



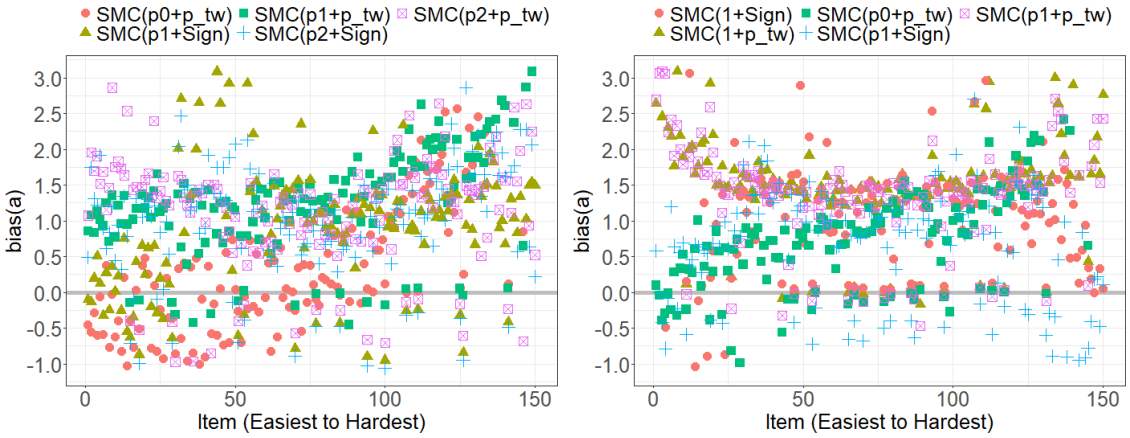
(a) Uniform Item Bank (b) Normal Item Bank

Figure 3.39: Bias(b) per Item of Selected Adapted Structured Matrix Completion Methods in Random NEAT Design with Sample Size of 125



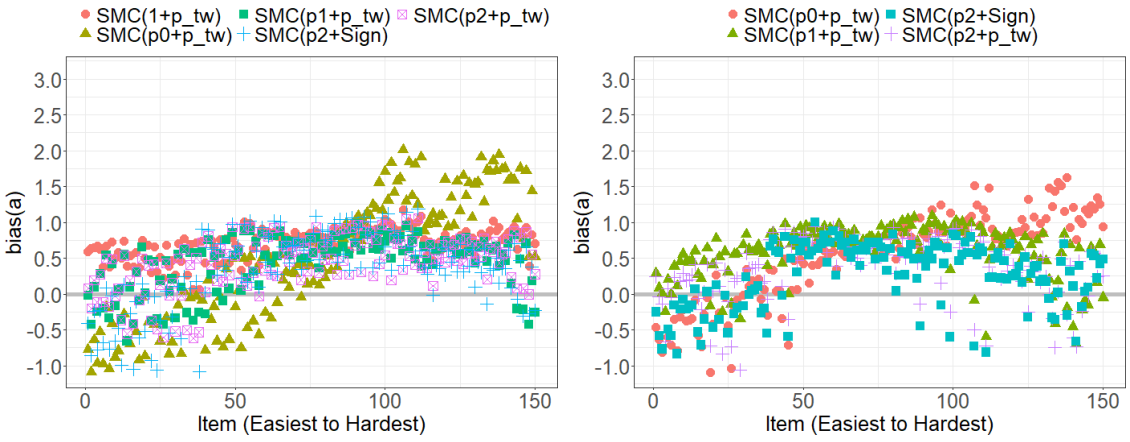
(a) Uniform Item Bank (b) Normal Item Bank

Figure 3.40: Bias(b) per Item of Selected Adapted Structured Matrix Completion Methods in Random PC Design with Sample Size of 125



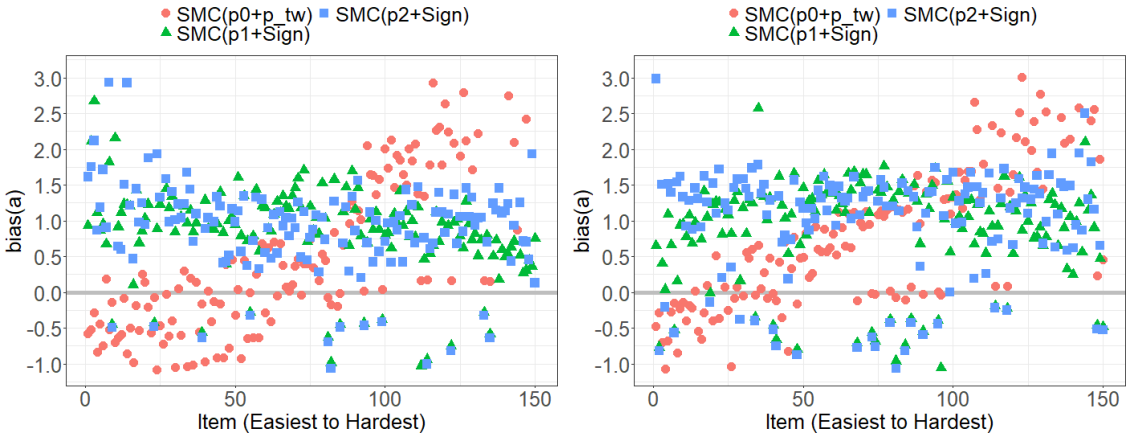
(a) Uniform Item Bank (b) Normal Item Bank

Figure 3.41: Bias(a) per Item of Selected Modified Structured Matrix Completion in Targeted NEAT Design with $N = 125$



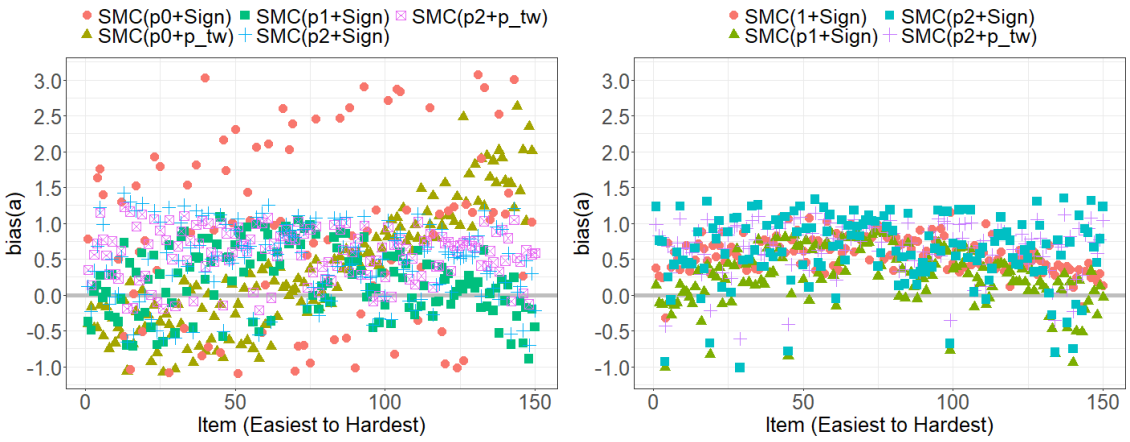
(a) Uniform Item Bank (b) Normal Item Bank

Figure 3.42: Bias(a) per Item of Selected Modified Structured Matrix Completion in Targeted PC Design with $N = 125$



(a) Uniform Item Bank (b) Normal Item Bank

Figure 3.43: Bias(a) per Item of Selected Modified Structured Matrix Completion in Random NEAT Design with $N = 125$



(a) Uniform Item Bank

(b) Normal Item Bank

Figure 3.44: Bias(a) per Item in Random PC Design with Sample Size of 125

Influence from sample sizes on adapted structured matrix completion methods.

The sample sizes had little influence on the recovery rates, estimation for item difficulty and discrimination parameters using the modified structured matrix completion. Firstly, the recovery rates increased slightly as sample sizes increased in some cases. For example, *SMC(1+p_tw)*, *SMC(1+Sign)*, and *SMC(p1+p_tw)* had increasing average recovery rates from $N = 125$ to $N = 500$ in practical item bank under targeted NEAT designs (Table 3.16). On the contrary, *SMC(p0+Sign)* had decreasing recovery rates across the three sample sizes. Furthermore, other adaptations had recovery rates very similar across different sample sizes, such as *SMC(p0+p_tw)*. The observations that most modified structured matrix completion had similar recovery rates across sample sizes were also shown in the uniform and normal item banks (Tables D.12 and D.13 in the Appendix). Generally, the modified structured matrix completion were robust to sample sizes, which was a good property for small-sample calibration scenario.

Secondly, the estimations for item difficulty parameters presented similar patterns across sample sizes. The RMSE(b) was generally decreasing as sample sizes increased for most conditions, but the bias(b) is not strictly decreasing. For example, *SMC(1+p_tw)* and *SMC(p1+p_tw)*, the two adaptations with good estimations for item difficulty parameters, had similar bias(b) across three sample sizes, and their

Table 3.16: Recovery Rates from Eight Modified Structured Matrix Completion Across Sample Sizes in Practical Item Bank Under Targeted NEAT Designs

Method	N = 125	N = 250	N = 500
SMC(1+p_tw)	0.773	0.782	0.785
SMC(1+Sign)	0.750	0.772	0.777
SMC(p0+p_tw)	0.745	0.746	0.745
SMC(p0+Sign)	0.579	0.576	0.571
SMC(p1+p_tw)	0.775	0.781	0.785
SMC(p1+Sign)	0.712	0.718	0.716
SMC(p2+p_tw)	0.681	0.690	0.686
SMC(p2+Sign)	0.586	0.604	0.596

RMSE(b) decreased as sample size increased in practical item bank with targeted NEAT designs (Table 3.17). In addition, *SMC(1+Sign)* and *SMC(1+p_tw)*, the two modified method with good estimations for item difficulty parameters in the random NEAT design, had similar bias(b) across sample sizes (Table 3.18). *SMC(1+Sign)* had large RMSE(b) when $N = 125$ and $N = 250$, which was due to a few extreme estimations for easy and hard items (Figure C.7c in Appendix). The results from targeted PC and random PC designs were the same (Table D.14and D.15 in the Appendix).

Table 3.17: Bias(b) and RMSE(b) from Eight Modified Structured Matrix Completion Across Sample Sizes in Practical Item Bank and Targeted NEAT Designs

	N = 125		N = 250		N = 500	
Method	Bias(b)	RMSE(b)	Bias(b)	RMSE(b)	Bias(b)	RMSE(b)
SMC(1+Sign)	0.565	9.156	0.277	3.723	0.933	10.768
SMC(1+p_tw)	0.062	1.155	0.044	0.405	0.063	0.327
SMC(p0+Sign)	-0.889	40.678	-9.056	117.856	3.895	43.35
SMC(p0+p_tw)	-0.726	2.132	-0.706	1.114	-0.766	1.026
SMC(p1+Sign)	0.084	10.779	-0.008	3.574	0.306	1.636
SMC(p1+p_tw)	0.122	1.047	0.077	0.401	0.085	0.319
SMC(p2+Sign)	0.094	3.338	0.309	2.355	0.33	1.078
SMC(p2+p_tw)	0.172	1.045	0.31	0.657	0.393	0.659

Table 3.18: Bias(b) and RMSE(b) from 8 Adaptations for Structured Matrix Completion Across Sample Sizes in Practical Item Bank and Random NEAT Designs

	N = 125		N = 250		N = 500	
Method	Bias(b)	RMSE(b)	Bias(b)	RMSE(b)	Bias(b)	RMSE(b)
SMC(1+Sign)	0.275	4.465	-0.273	5.438	-0.061	0.96
SMC(1+p_tw)	0.122	2.422	0.018	0.552	0.022	0.327
SMC(p0+Sign)	0.758	44.754	1.179	15.202	0.241	14.948
SMC(p0+p_tw)	-2.219	23.759	-1.652	4.167	-1.731	3.507
SMC(p1+Sign)	0.017	9.025	-0.059	5.820	-2.479	20.248
SMC(p1+p_tw)	0.534	5.686	-2.936	24.471	-0.137	0.893
SMC(p2+Sign)	-0.148	6.402	1.066	7.620	0.350	1.042
SMC(p2+p_tw)	0.189	1.189	0.240	0.558	0.308	0.511

Lastly, the estimations for item discrimination parameters presented similar patterns across sample sizes. Bias(a) and RMSE(a) were generally decreasing as sample size increased. For example, *SMC(1+p_tw)* and *SMC(p1+p_tw)*, the two adaptations provided good estimations for item discrimination parameters in the practical item bank and targeted NEAT design, had both bias(a) and RMSE(a) decreasing as sample sizes increased (Table 3.19). In addition, Table 3.20 shows that *SMC(p1+Sign)*, the best adaptation in terms of estimating item discrimination parameters, had both bias(a) and RMSE(a) decreasing as sample size increased. The results from targeted PC and random PC designs were similar (Tables D.16 and D.17 in the Appendix).

Discussion about the adapted structured matrix completion methods.

From previous sections, there were three major observations. Firstly, the eight adaptations performed differently in the targeted PC design than the other three incomplete calibration designs. Specifically, they had generally lower recovery rates and worse estimations for item difficulty parameters in the targeted PC designs than in the other three incomplete calibration designs. This finding was also shown when using *1bit(p_tw)*. To demonstrate this finding, the imputed response matrices of *SMC(1+p_tw)* and *SMC(p1+Sign)*, which provided relatively accurate estimations for item difficulty parameters and item discrimination parameters, were selected and presented (Figure 3.45 and 3.46). Note that the items

Table 3.19: Bias(a) and RMSE(a) from Eight Modified Structured Matrix Completion Across Sample Sizes in Practical Item Bank and Targeted NEAT Designs

	N = 125		N = 250		N = 500	
Method	Bias(a)	RMSE(a)	Bias(a)	RMSE(a)	Bias(a)	RMSE(a)
SMC(1+Sign)	1.664	3.844	1.342	1.89	1.204	1.552
SMC(1+p_tw)	1.857	2.917	1.645	1.915	1.608	1.741
SMC(p0+Sign)	1.124	8.207	2.282	16.029	0.297	5.448
SMC(p0+p_tw)	1.475	2.241	1.295	1.754	1.231	1.578
SMC(p1+Sign)	1.466	4.057	0.628	1.721	0.501	1.244
SMC(p1+p_tw)	1.828	2.650	1.511	1.724	1.505	1.626
SMC(p2+Sign)	2.178	5.023	1.572	2.681	1.556	2.296
SMC(p2+p_tw)	1.833	3.276	1.55	2.075	1.546	1.847

Table 3.20: Bias(a) and RMSE(a) from Eight Modified Structured Matrix Completion Across Sample Sizes in Practical Item Bank and Random NEAT Designs

	N = 125		N = 250		N = 500	
Method	Bias(a)	RMSE(a)	Bias(a)	RMSE(a)	Bias(a)	RMSE(a)
SMC(1+Sign)	2.274	6.852	1.254	1.761	1.133	1.458
SMC(1+p_tw)	2.208	4.090	1.549	1.876	1.420	1.577
SMC(p0+Sign)	1.511	10.392	0.608	9.540	0.331	9.502
SMC(p0+p_tw)	1.473	2.412	1.100	1.603	1.016	1.385
SMC(p1+Sign)	0.981	2.076	0.605	1.091	0.546	0.924
SMC(p1+p_tw)	1.718	2.740	1.212	1.511	1.079	1.261
SMC(p2+Sign)	1.075	2.049	0.971	1.335	0.962	1.217
SMC(p2+p_tw)	1.364	2.277	1.218	1.506	1.232	1.426

were ordered by their difficulties on x-axis and examinees were ordered by their θ on the y-axis. The imputed response matrices of *SMC(1+p_tw)* and *SMC(p1+Sign)* showed a gradual transition from the left top corner to the bottom right corner in the targeted NEAT, random NEAT, and random PC designs, which was similar to the original full response matrix. On the contrary, the imputed response matrices of *SMC(1+p_tw)* in the targeted PC designs presented a pattern where the correct responses and incorrect responses were relatively distributed across all the items (Figure 3.45b). In addition, *SMC(p1+Sign)* presented a transition in targeted PC design (Figure 3.46b); however, the top left corner, which were responses of easy items and examinees of highest abilities, were imputed with incorrect responses. Though *SMC(1+p_tw)* and *SMC(p1+Sign)* did not fully recover the structured missingness in the targeted PC designs, their imputed response matrices were more close to the full response matrix than the imputed response matrix of *1bit(p_tw)* in the targeted PC designs (Figure 3.24b).

Secondly, *SMC(1+p_tw)* and *SMC(p1+p_tw)*, the two adaptations provided good estimations for item difficulty parameters, had a few extreme estimations for easy and difficult items. Figure 3.27 in previous section showed that there were not enough examinees of very low and high θ to match the very easy and difficult items in the item bank.

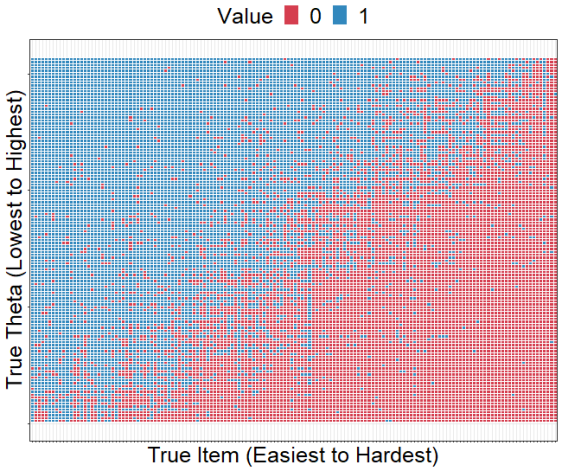
Lastly, *SMC(1+p_tw)* produced accurate estimations for item difficulty parameters but not for item discrimination parameters. On the contrary, *SMC(p1+Sign)* produced more accurate estimations for item discrimination parameters but not for item difficulty parameters. To investigate this observation, their imputed response matrices were compared with the full response matrix respectively. The response patterns from the targeted PC designs were not presented because either of these two adaptations successfully recovered the pattern of the full response matrix as demonstrated before.

The difference between the imputed response matrix of *SMC(1+p_tw)* and the full response matrix were presented in Figure 3.47. In targeted NEAT designs, *SMC(1+p_tw)* imputed more incorrect responses (red mosaics) to the lower right triangle and more correct responses (blue mosaics) to the upper left triangle (Figure 3.47a). There were similar blue and red mosaics, indicating wrongly imputed responses, around the diagonal area. It was because those items' difficulties matched examinees' ability, and

(a) Targeted NEAT Design

(b) Targeted PC Design

(c) Random NEAT Design

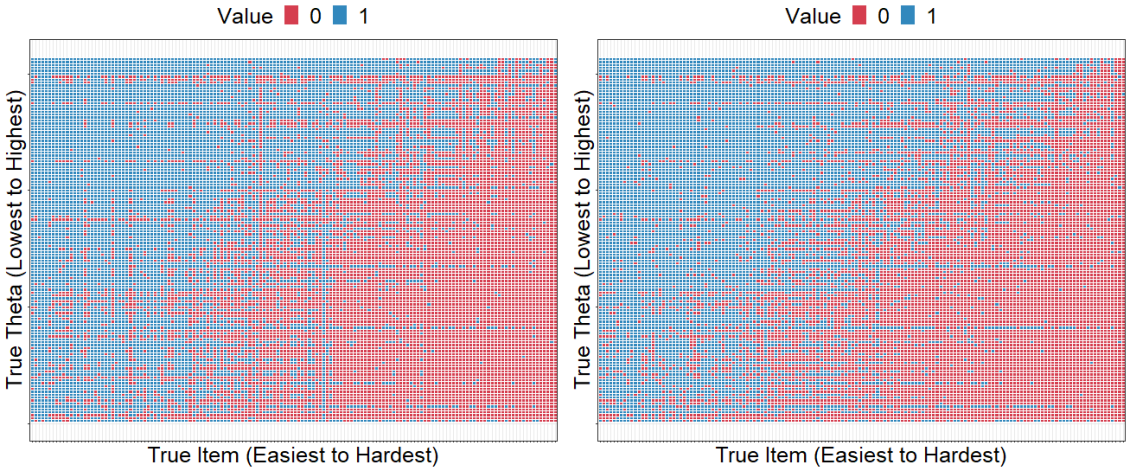


(d) Random PC Design

Figure 3.45: Imputed Response Matrices From *SMC(1+p_tw)* in Practical Item Bank

(a) Targeted NEAT Design

(b) Targeted PC Design



(c) Random NEAT Design

(d) Random PC Design

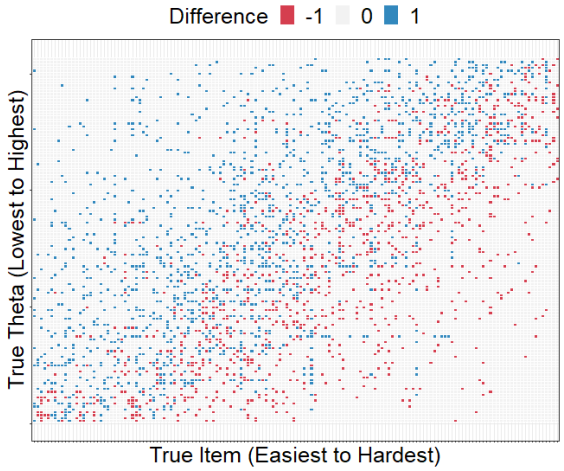
Figure 3.46: Imputed Response Matrices From *SMC(p1+Sign)* in Practical Item Bank

there were more uncertainty of whether those responses were correct or incorrect. The imputed response matrices from *SMC(1+p_tw)* showed similar patterns in random NEAT and random PC designs (Figures 3.47b and 3.47c). The separation between correct and incorrect responses given examinees' ability and item difficulty of *SMC(1+p_tw)* was slightly more clear than that of *1bit(p_tw)*.

The imputed response matrices from *SMC(p1+Sign)* showed different patterns. From Figure 3.48, the imputed response matrices had more wrongly imputed responses than those from *SMC(1+p_tw)*. In addition, the blue and the red mosaics were more evenly presented in the response matrices. Thus, *SMC(p1+Sign)* had better item discrimination parameter estimations than *SMC(1+p_tw)* and worse item difficulty parameter estimations. In addition, this pattern was also shown in the imputed response matrices of *1bit(Sign)* but closer to the full response matrix. Hence, the structured missing patterns of the incomplete calibration designs had less impact on structured matrix completion paired with *p1* than 1-bit matrix completion.

(a) Targeted NEAT Design

(b) Random NEAT Design

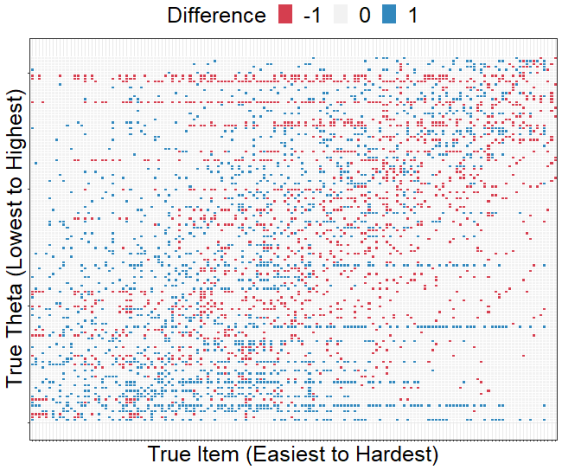


(c) Random PC Design

Figure 3.47: Imputed Response Matrices From *SMC(1+p_tw)* in Practical Item Bank.

(a) Targeted NEAT Design

(b) Random NEAT Design



(c) Random PC Designs

Figure 3.48: Imputed Response Matrices From *SMC(pr+Sign)* in Practical Item Bank.

3.4.3 Results of Adapted 1-bit Matrix Completion, Adapted Structured Matrix Completion, and Baselines.

From the previous sections, between the two adaptations for 1-bit matrix completion, *1bit(p_tw)* provided higher recovery rates and better estimation for item difficulty parameters than *1bit(Sign)* across three item banks, three sample sizes, and four incomplete calibration designs. On the contrary, *1bit(Sign)* had better RMSE(a) and bias(a) than *1bit(p_tw)*. In addition, the eight adaptations for structured matrix completion had different performance given incomplete calibration designs. Generally, *SMC(1+Sign)*, *SMC(1+p_tw)*, and *SMC(p1+p_tw)* had better performance than the other adaptations in four incomplete calibration designs with regard to recovery rates, item difficulty parameter estimations, and item discrimination parameter estimations. To further examine the performance of the modified matrix completion imputation methods, all ten of them were compared together along with four baselines, *FIML*, *TW*, *MICE(CART)*, and *MICE(RF)*, under the three evaluation criteria. Note that *FIML* was not evaluated by recovery rates because there was no imputation involved in this method.

Recovery rates among all methods.

Similarly to the previous sections, all the methods were firstly evaluated under the condition with the practical item bank and $N = 125$, and their performance in the other two item banks and sample sizes were examined next. The box plot of the average recovery rates from all 13 imputation methods in Practical item bank with sample size of 125 was presented in Figure 3.49.

Comparing the adapted 1-bit matrix completion methods with the adapted structured matrix completion methods, *SMC(1+Sign)*, *SMC(1+p_tw)*, and *SMC(p1+p_tw)*, had higher recovery rates than *1bit(p_tw)* in targeted NEAT designs, while *1bit(p_tw)* had highest recovery rates in random NEAT and random PC designs. In addition, their recovery rates were lower in the targeted PC designs than in the other three incomplete calibration designs.

Comparing the adapted matrix completion methods with the baselines, some adapted matrix completion imputation methods, such as *1bit(p_tw)*, *SMC(1+p_tw)*, and *SMC(p1+p_tw)*, had higher recovery

rates than all three baselines in the targeted NEAT, random NEAT, and random PC designs, In the targeted PC designs, the recovery rates from the adapted matrix completion methods were similar to the results from baselines.

Among the three baselines, two MICE based imputation methods had relatively low recovery rates in all four incomplete calibration designs, except for *MICE(RF)* that had high recovery rates in targeted PC design. In addition, *TW* showed relatively high recovery rates in random NEAT and PC designs. This indicated that *TW* was more suitable when the missing mechanism was MCAR.

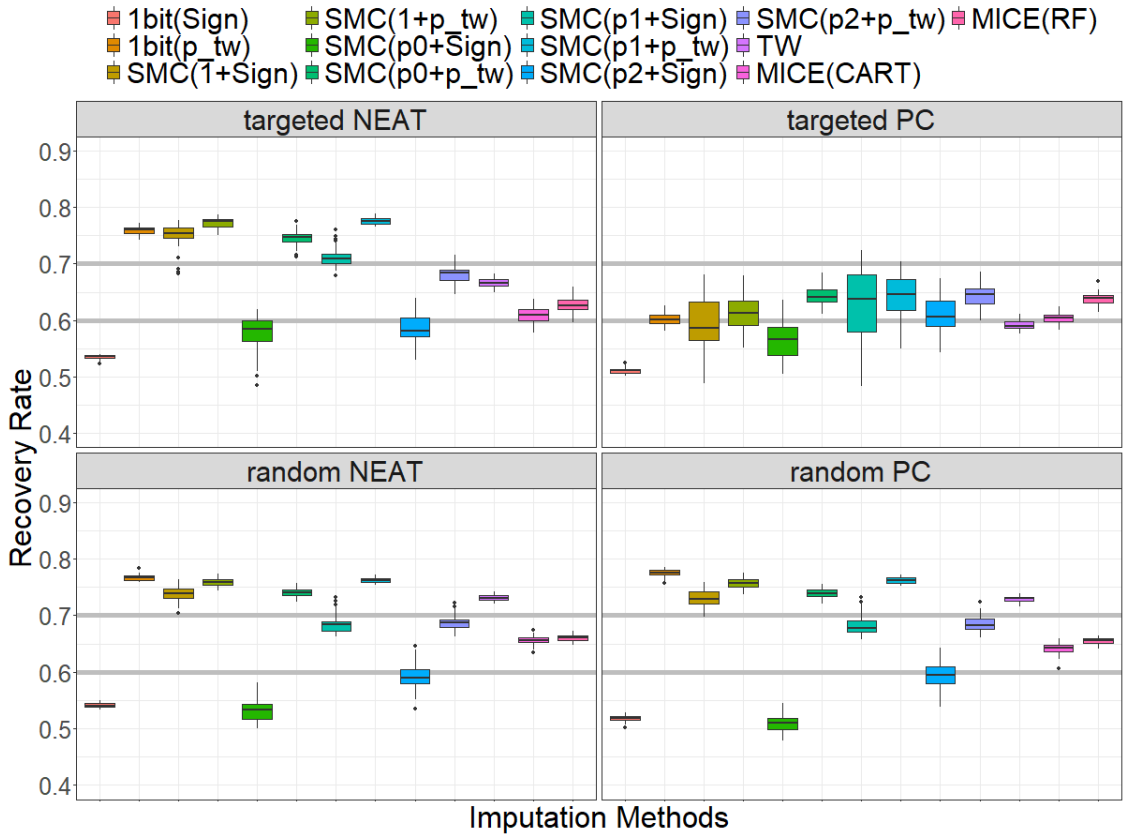


Figure 3.49: Recovery Rates of All Methods in Practical Item Bank and $N = 125$

The above findings were also shown in the uniform and normal item banks and with $N = 250$ and $N = 500$ (see Figures C.11 to C.34 in the Appendix). To summarize, *1bit(p_tw)*, *SMC(1+p_tw)*, and *SMC(p1+p_tw)* had the highest recovery rates in the targeted NEAT, random NEAT, and random PC designs among all 13 imputation methods. Their recovery rates in the targeted PC designs were not

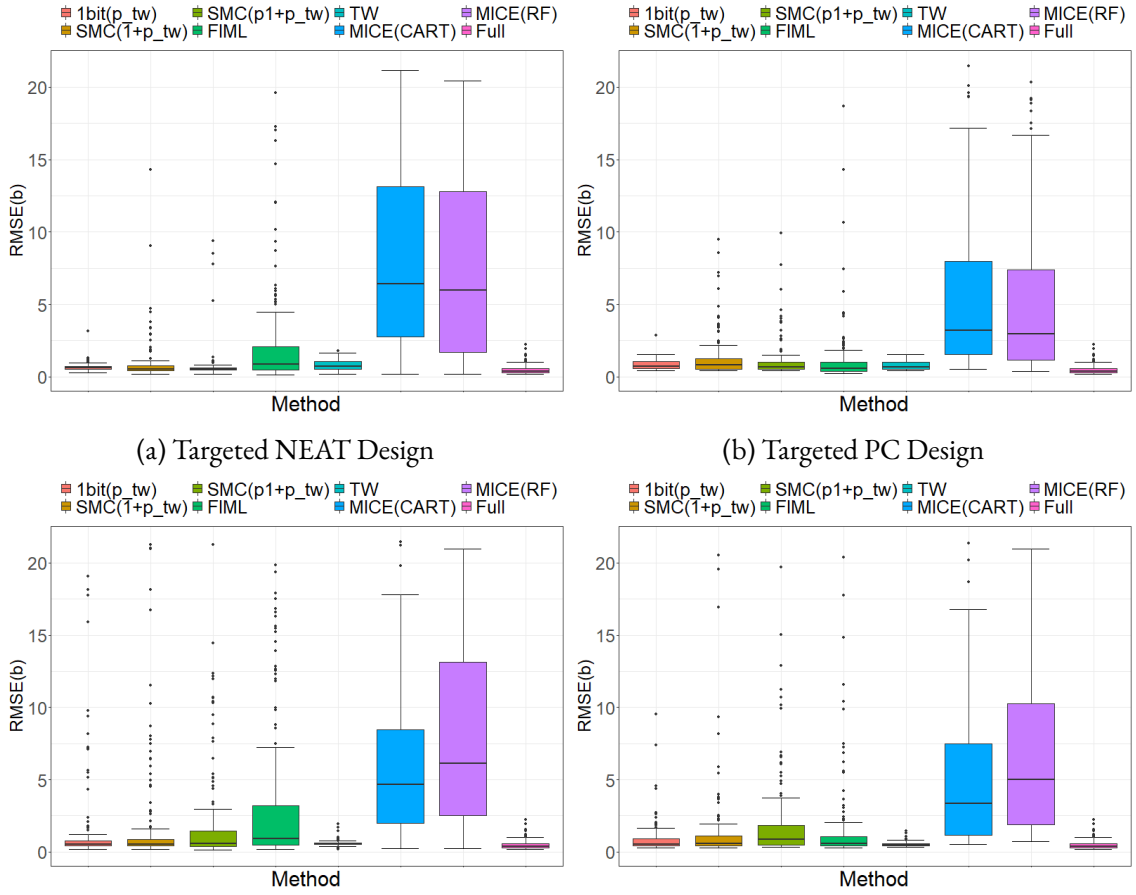
satisfying. Among the three baseline methods, *TW* had relatively high recovery rates in random NEAT and random PC designs.

Bias(b) and RMSE(b) among all methods.

From the previous sections, *1bit(p_tw)* had better bias(b) and RMSE(b) than *1bit(Sign)*, and *SMC(1+p_tw)* and *SMC(p1+p_tw)* had better performance in bias(b) and RMSE(b) than the other adapted structured matrix completion methods. In this section, these adaptations were compared to the four baselines: *FIML*, *TW*, *MICE(CART)*, and *MICE(RF)*. In addition, their bias(b) and RMSE(b) were also compared with those from the full response matrix, which was denoted as *Full* and treated as the best scenario. Specifically, the average RMSE(b) and bias(b) per item from the selected methods were compared under four incomplete calibration designs using practical item bank and $N = 125$ first. Their performance were evaluated in the other two item banks and sample sizes. Lastly, the best adaptations or baseline methods were identified.

Figure 3.50 is the box plot for RMSE(b) of the best adaptations for 1-bit matrix completion and structured matrix completion, all the baselines, and the best scenario in the four incomplete calibration designs. From this Figure, RMSE(b) of *Full* were very small across all four incomplete calibration designs. The average RMSE(b) of *1bit(p_tw)*, *SMC(1+p_tw)*, and *SMC(p1+p_tw)* were small, but they had some extreme estimations especially in the random NEAT and random PC designs. Note that *1bit(p_tw)* had similar RMSE(b) with *Full* in the targeted NEAT designs. Among the baselines, *MICE(CART)* and *MICE(RF)* resulted in the largest RMSE(b) in all four incomplete designs. *FIML* had relatively large RMSE(b) in the targeted NEAT and random NEAT designs and had relatively small RMSE(b) in the targeted PC and random PC designs. It also had some extreme estimations in all four incomplete calibration designs. *TW* had the smallest RMSE(b) among four baselines, and even had smaller RMSE(b) than *Full* in random NEAT and random PC designs.

To further evaluate the performance of *1bit(p_tw)*, *SMC(1+p_tw)*, *SMC(p1+p_tw)* and the best baseline *TW*, their bias(b) per item were computed under four incomplete calibration designs (Figure 3.51). In targeted NEAT design, *1bit(p_tw)* and *SMC(1+p_tw)* had bias(b) closer to zero than *TW* for easy



(a) Targeted NEAT Design

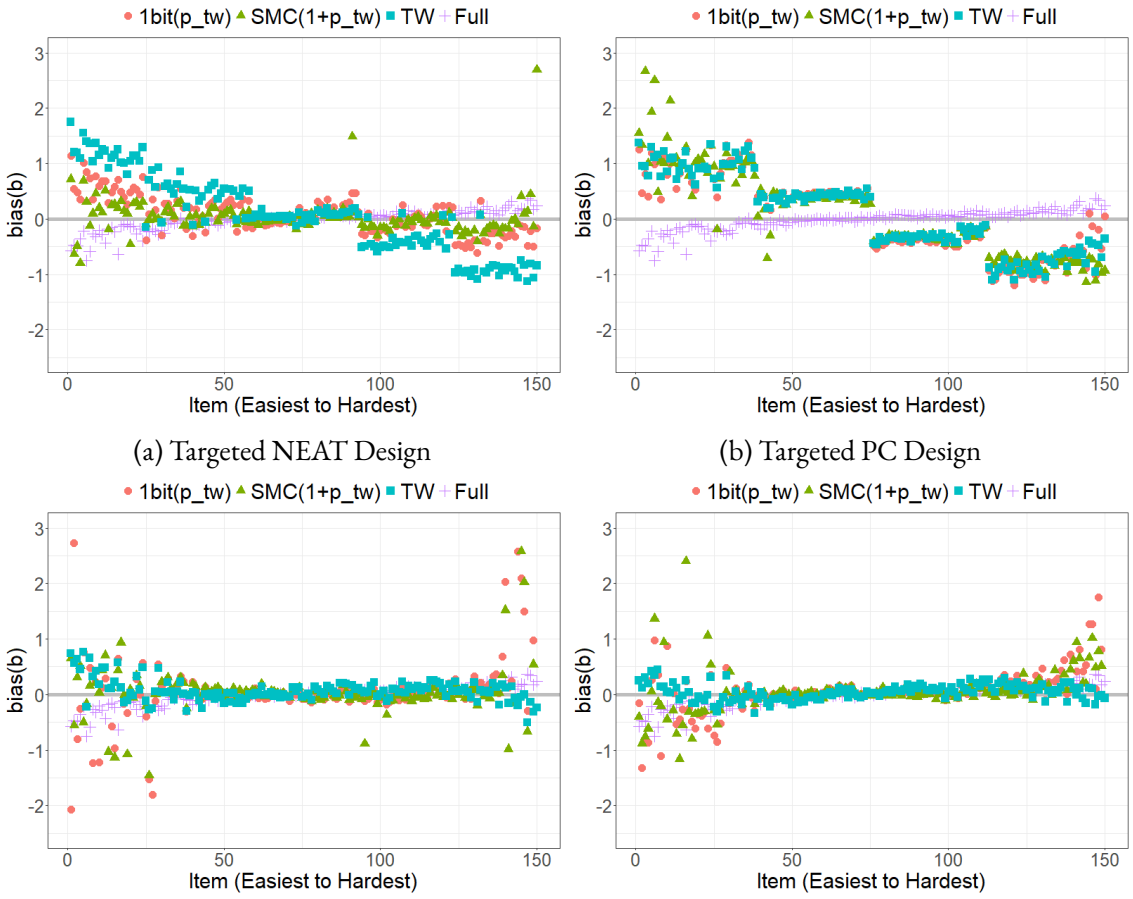
(b) Targeted PC Design

(c) Random NEAT Design

(d) Random PC Design

Figure 3.50: RMSE(b) of Selected Methods in Practical Item Bank with $N = 125$

and difficult items. In addition, *SMC(1+p_tw)* had better bias(b) than *1bit(p_tw)*. In targeted PC designs, *SMC(1+p_tw)*, *1bit(p_tw)* and *TW* had similar bias(b) across all the items and presented systematic patterns given the designed test forms. In random NEAT and random PC designs, *1bit(p_tw)* and *SMC(1+Sign)* had larger bias(b) than *TW* only for easy and difficult items. In addition, the bias(b) of full response matrix were relatively similar to the bias(b) of the selected imputation methods except in targeted PC design. This observation was consistent with the previous findings that *1bit(p_tw)*, *SMC(1+p_tw)*, and *SMC(p1+p_tw)* did not recover the characteristics of the full response matrix if using the targeted PC designs. To summarize, *1bit(p_tw)*, *SMC(1+p_tw)*, *SMC(p1+p_tw)* improved bias(b) in targeted NEAT and PC designs than the best baseline *TW*, while they produced large bias(b) for easy and difficult items in random NEAT and random PC designs.



(a) Targeted NEAT Design

(b) Targeted PC Design

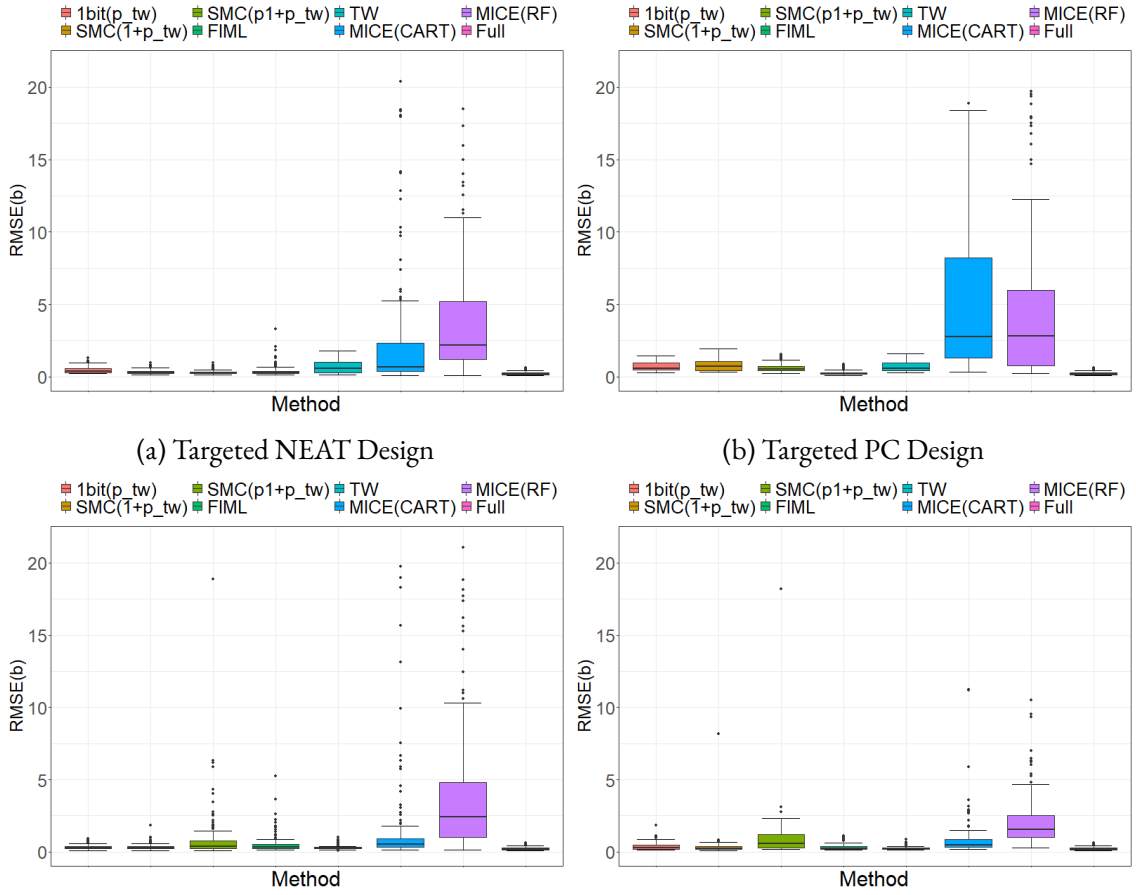
(c) Random NEAT Design

(d) Random PC Design

Figure 3.51: Bias(b) of Selected Methods in Practical Item Bank with $N = 125$

The above findings of RMSE(b) and bias(b) were also shown in the uniform and normal item banks (see Figures C.15 and C.18 in the Appendix). In addition, the RMSE(b) of all the methods decreased as sample sizes increased. For example, Figure 3.52 presents the RMSE(b) of the selected methods in practical item bank with $N = 500$. In particular, there were less extremely estimated item difficulty parameters of the adapted matrix completion methods when $N = 500$ than $N = 125$. Furthermore, bias(b) of the selected methods were also smaller with less variations when $N = 500$ than $N = 125$ in the practical item bank (Figure 3.53). The impact of sample sizes on RMSE(b) and bias(b) were also shown in the other two item banks (see Figures C.19 to C.22 in the Appendix).

To summarize, *1bit(p_tw)* and *SMC(1+p_tw)* had the best item difficulty parameter estimations in the targeted NEAT designs. *TW* had the best item difficulty parameter estimations especially for easy and difficulty items in the random NEAT and random PC designs. In addition, across all the methods, larger the sample sizes, the better item difficulty parameter estimations.



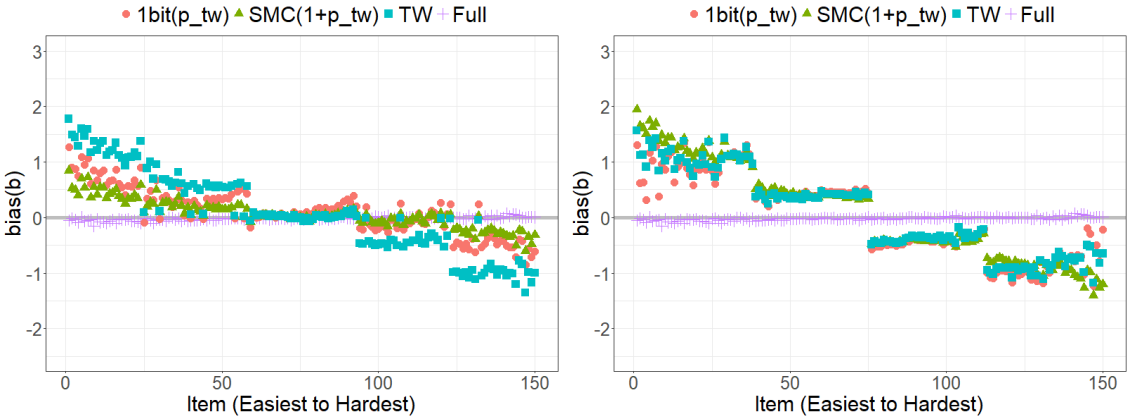
(a) Targeted NEAT Design

(b) Targeted PC Design

(c) Random NEAT Design

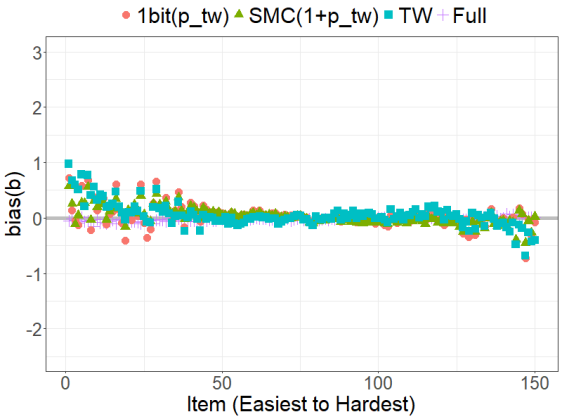
(d) Random PC Design

Figure 3.52: RMSE(b) of Selected Methods in Practical Item Bank with $N = 500$

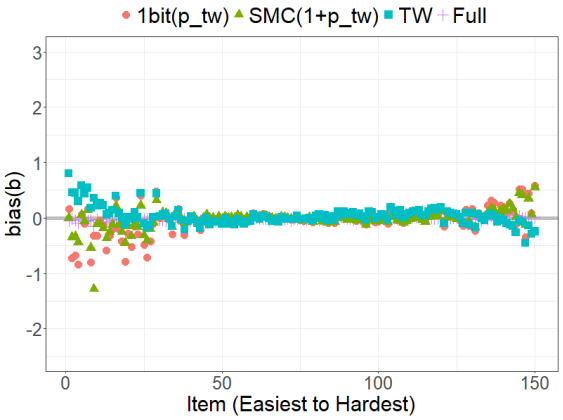


(a) Targeted NEAT Design

(b) Targeted PC Design



(c) Random NEAT Design

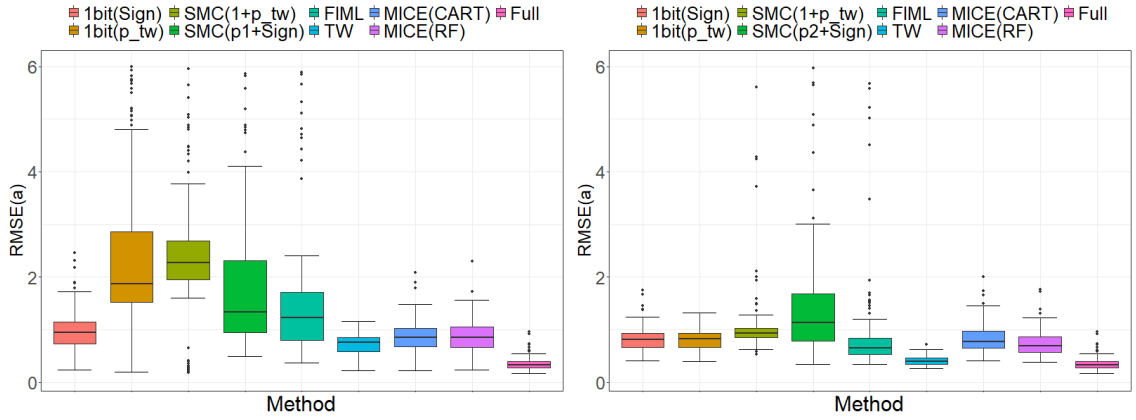


(d) Random PC Design

Figure 3.53: Bias(b) of Selected Methods in Practical Item Bank with $N = 500$

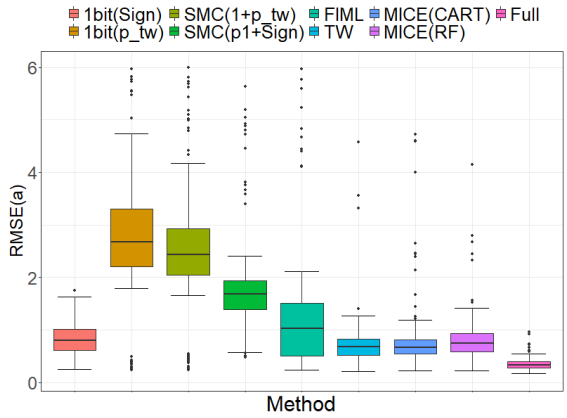
Bias(a) and RMSE(a) among all methods.

From previous sections, *1bit(Sign)* provided better estimation for item discrimination parameters than *1bit(p_tw)*, and *SMC(p1+Sign)* provided better estimations than the other adapted structured matrix completions in the targeted NEAT, random NEAT, and random PC designs. *SMC(p2+Sign)* provided the best estimations in the targeted PC designs. In this sections, these adapted matrix completion methods were compared to the baselines and *Full*.

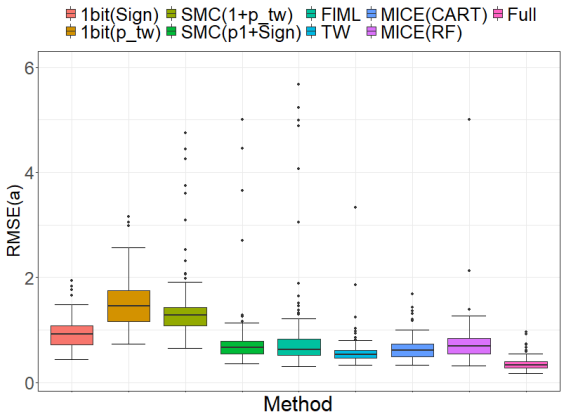


(a) Targeted NEAT Design

(b) Targeted PC Design



(c) Random NEAT Design



(d) Random PC Design

Figure 3.54: RMSE(a) of Selected Methods in Practical Item Bank with Sample Size of 125

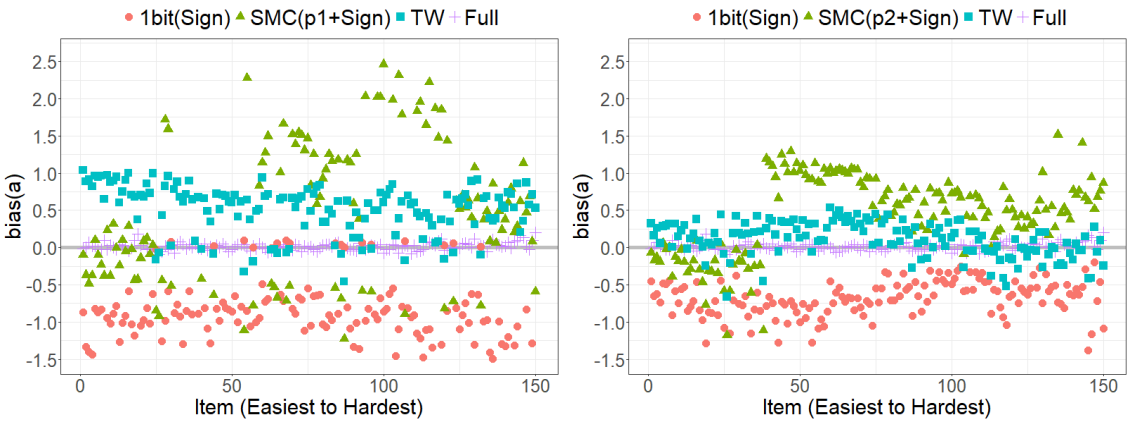
Figure 3.54 shows the box plot of RMSE(a) from the above-mentioned methods in the practical item bank with $N = 125$. *1bit(p_tw)* and *SMC(1+p_tw)* were added as they provided good item difficulty parameter estimations. From this set of figures, RMSE(a) from *Full* were small across all incomplete calibration designs. *1bit(p_tw)* and *SMC(1+p_tw)*, though had small RMSE(b), resulted in large

RMSE(a) in the targeted NEAT, random NEAT, and random PC designs. Specifically, *SMC(1+p_tw)* had smaller RMSE(a) than *1bit(p_tw)* in these three incomplete calibration designs. In addition, *1bit(Sign)* and *SMC(p1+Sign)* had smaller RMSE(a) than *1bit(p_tw)* and *SMC(1+p_tw)* but slightly larger than the baselines including *MICE(CART)*, *MICE(RF)*, and *TW*. The other baseline, *FIML*, though had relatively small average RMSE(a), had some extreme estimations. Among the four baselines, *TW* had the smallest RMSE(a) in all four incomplete calibration designs. All methods had larger RMSE(a) then *Full*.

To further investigate the performance of the above-mentioned imputation methods, their bias(a) per item were presented in Figure 3.55. *TW* had bias(a) closer to zero than *1bit(Sign)* and *SMC(1+Sign)* in the targeted NEAT, targeted PC, and random NEAT designs. In the random PC designs, *SMC(p1+Sign)* had bias(a) closer to zero for easy and difficult items than *TW* and slightly larger bias(a) for items with medium difficulties.

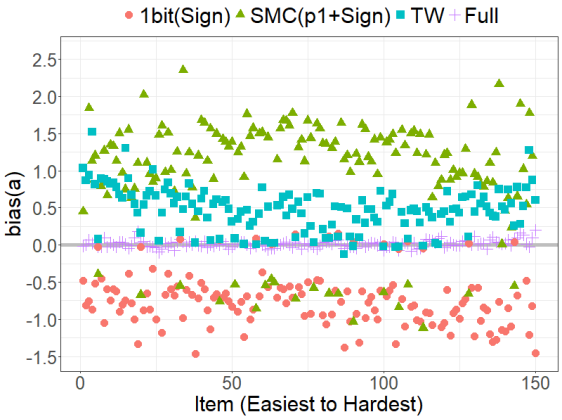
The above findings of RMSE(a) and bias(a) were also shown in the uniform and normal item banks (see Figures C.23 and C.26 in the Appendix). In addition, the RMSE(a) of all the methods decreased as sample sizes increased. For example, Figure 3.56 presents the RMSE(a) of the selected methods in practical item bank with $N = 500$. Furthermore, in the practical item bank, bias(a) of *SMC(p1+Sign)* slightly smaller when $N = 500$ than $N = 125$ in the targeted PC and random NEAT designs (Figure 3.53). Bias(a) of the other methods were similar when $N = 500$ and $N = 125$. The impact of sample sizes on RMSE(b) and bias(b) were also shown in the other two item banks (see Figures C.27 to C.30 in the Appendix).

To summarize, in the targeted NEAT, targeted PC, and random NEAT designs, though *1bit(Sign)* and *SMC(p1+Sign)* had better estimated item discrimination parameters than *1bit(p_tw)* and *SMC(1+p_tw)*, their performance were not as good as *TW* and other baselines. In random PC designs, *SMC(p1+Sign)* had similar estimated item discrimination parameters with *TW*.

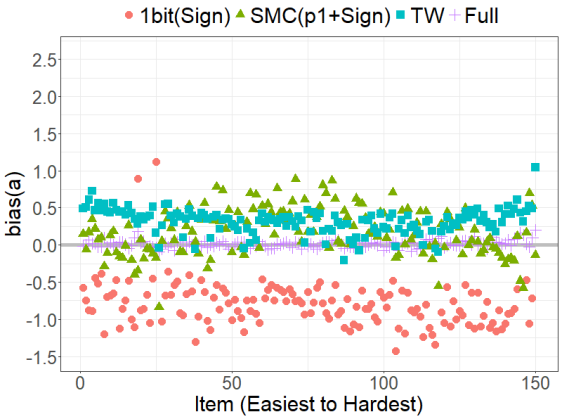


(a) Targeted NEAT Design

(b) Targeted PC Design

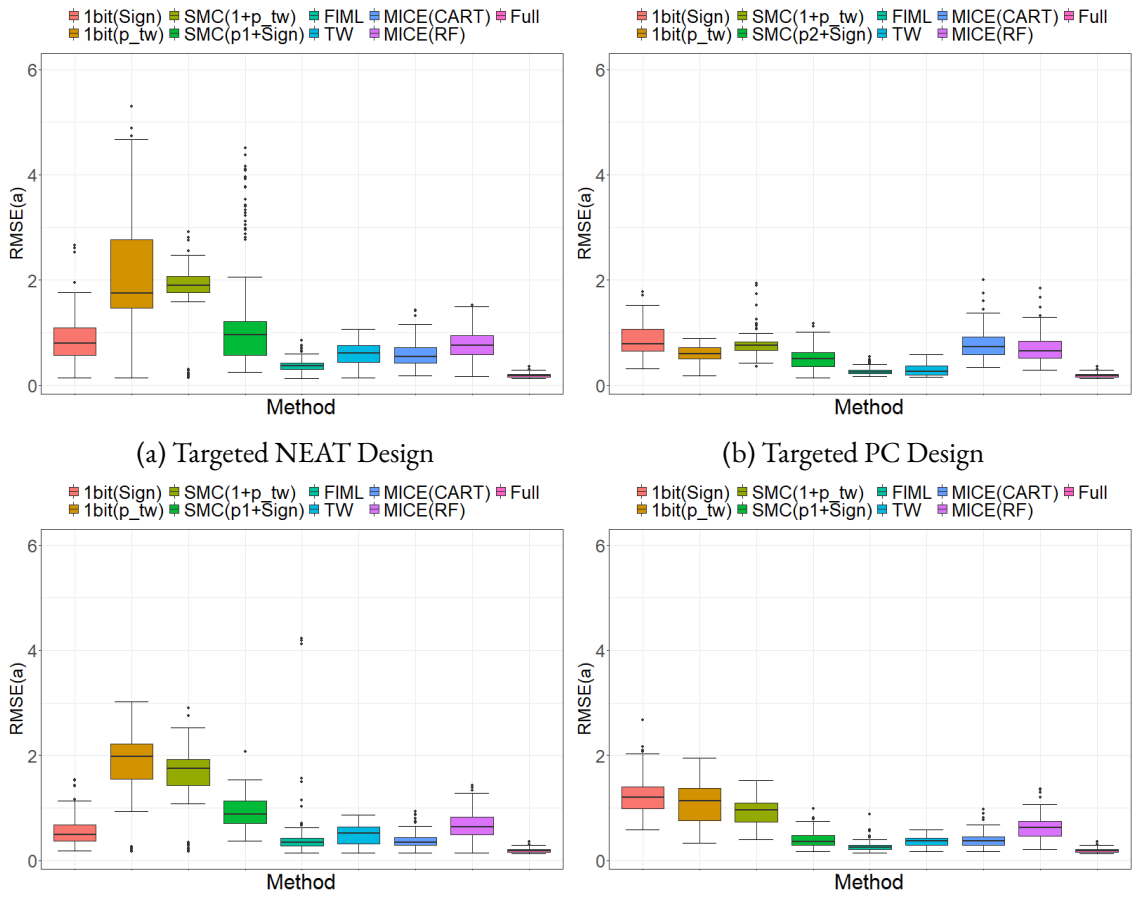


(c) Random NEAT Design



(d) Random PC Design

Figure 3.55: Bias(a) of Selected Methods in Practical Item Bank with Sample Size of 125



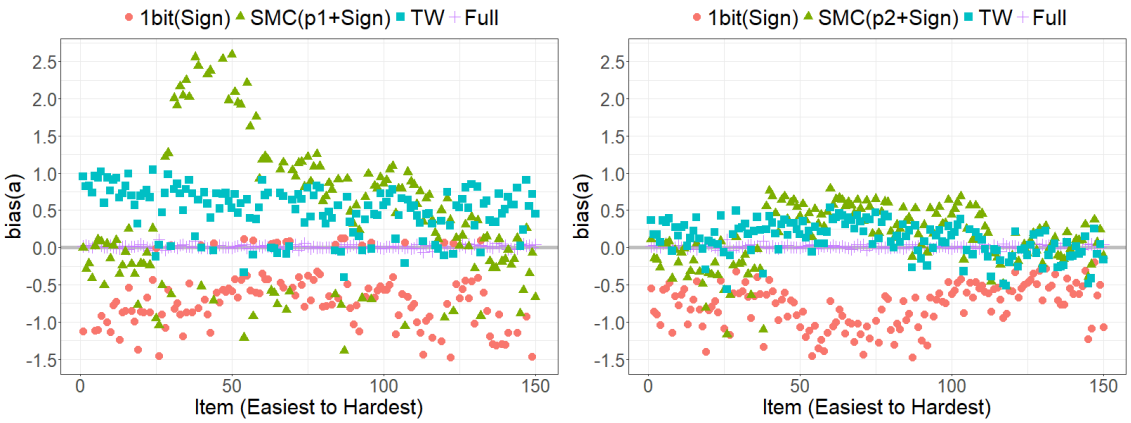
(a) Targeted NEAT Design

(b) Targeted PC Design

(c) Random NEAT Design

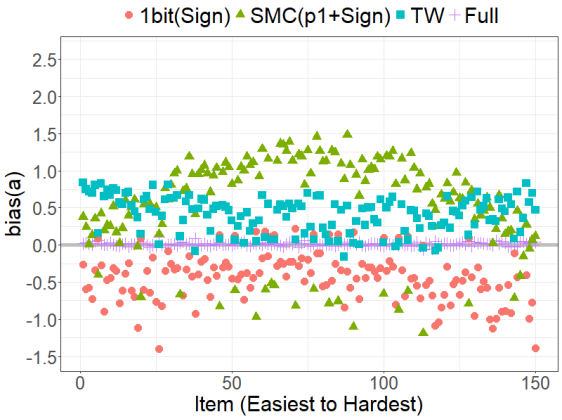
(d) Random PC Design

Figure 3.56: RMSE(a) of Selected Methods in Practical Item Bank with Sample Size of 500

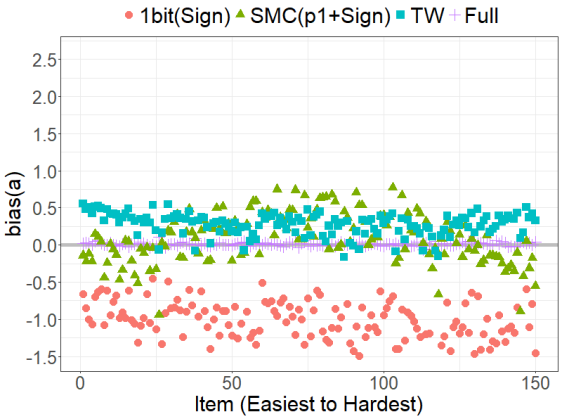


(a) Targeted NEAT Design

(b) Targeted PC Design



(c) Random NEAT Design



(d) Random PC Design

Figure 3.57: Bias(a) of Selected Methods in Practical Item Bank with Sample Size of 500

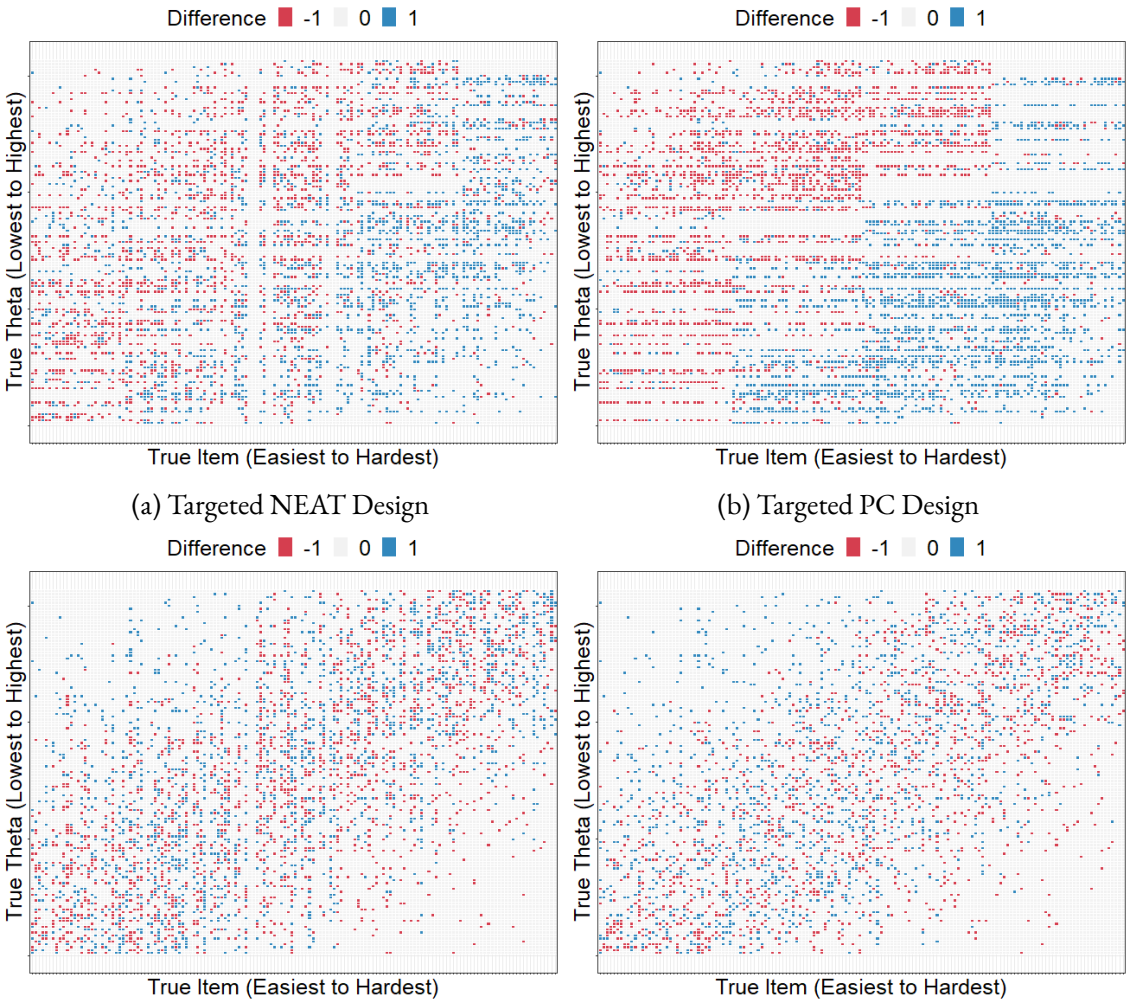
Discussion for all methods investigated in this simulation study.

Among the four baseline methods, *TW* achieved the best performance and sometimes were even better than the modified matrix completion methods. For example, in the targeted NEAT designs, *1bit(p_tw)* and *SMC(1+p_tw)* provided better estimations for item difficulty parameters than *TW* but not for item discrimination parameters. A series of investigations were conducted to further evaluate *TW* with the proposed adaptations for 1-bit matrix completion and structured matrix completion.

Figure 3.58a presents the difference between the *TW* imputed response matrix with the original full response matrix. Compare to the corresponding matrix using *1bit(p_tw)* (Figure 3.29a) and *SMC(1+p_tw)* (Figure 3.47a), *TW* had more colorful mosaics, which is expected because its recovery rates were lower. In addition, though there was also a visible pattern of the red and blue mosaics from *TW*, the red and blue mosaics were more evenly distributed across test forms and examinees than *1bit(p_tw)* and *SMC(1+p_tw)*. This indicates that *TW* did not separate examinees and items clearly given their abilities and item difficulties as clearly as *1bit(p_tw)* and *SMC(1+p_tw)*. Thus, it was expected that *1bit(p_tw)* and *SMC(1+p_tw)* had better estimations for item difficulty parameters.

To better quantify the observations from Figures 3.58a and 3.29a, Tables 3.21 and 3.22 summarized the difference between *1bit(p_tw)* and full response matrix by dividing items into five groups given item difficulties and dividing examinees into five groups given their abilities. In the Table 3.21, the number in each cell represents the proportion of wrongly imputed correct responses (blue mosaics). For example, 0.033 in item 1 - 30 and group 1 indicates that imputed 3.3% of the responses in that block were imputed as correct responses using *1bit(p_tw)* but their original responses are incorrect. It is bold because it is larger than the corresponding value from *TW* imputation. From this table, *1bit(p_tw)* wrongly imputed more correct responses mainly for upper left triangle of the response matrix (easy items with examinees of high abilities) than *TW*, and it imputed less correct responses for lower right triangle of the response matrix (hard items with examinees of low abilities).

On the contrary, Table 3.22 documents the proportion of wrongly imputed incorrect responses (red mosaics in the heat maps). From this table, *1bit(p_tw)* wrongly imputed more incorrect responses for



(a) Targeted NEAT Design

(b) Targeted PC Design

(c) Random NEAT Design

(d) Random PC Design

Figure 3.58: Difference Between the Full Response Matrix and Imputed Response Matrices From *TW* in Practical Item Bank with Sample Size of 125

lower right triangle of the response matrix, and less incorrect responses for upper left triangle. This pattern indicates that *1bit(p_tw)* imputed a response matrix where examinees of low abilities have higher possibility to answer an item incorrectly and examinees of high abilities have higher possibility to answer an item correctly. Thus, it explained that *1bit(p_tw)* overestimated item discrimination parameters more severely than *TW* imputation. *SMC(1+p_tw)* presents similar patterns (Tables D.18 and D.19 in the Appendix).

Table 3.21: Summarized Difference Between Imputed Response Matrices from *1bit(p_tw)* and Full Response Matrix with Regards to Proportion of Wrongly Imputed Correct Responses in Targeted NEAT Design

	item 1 - 30	item 31 - 60	item 61 - 90	item 91 - 120	item 121 - 150
Group 1 (Highest θ)	**0.033**	**0.051**	**0.094**	**0.145**	**0.192**
Group 2	**0.065**	**0.102**	**0.141**	**0.200**	0.182
Group 3	**0.087**	**0.118**	0.100	0.112	0.070
Group 4	**0.083**	0.083	0.042	0.030	0.013
Group 5 (Lowest θ)	0.045	0.031	0.005	0.001	0.000

Note. The bold number indicates that number is larger than the corresponding value from *TW*.

Table 3.22: Summarized Difference Between Imputed Response Matrices from *1bit(p_tw)* and Full Response Matrix with Regards to Proportion of Wrongly Imputed Incorrect Responses in Targeted NEAT Designs

	item 1 - 30	item 31 - 60	item 61 - 90	item 91 - 120	item 121 - 150
Group 1 (Highest θ)	0.000	0.004	0.019	0.033	0.036
Group 2	0.007	0.044	0.087	0.102	**0.075**
Group 3	0.064	0.148	**0.140**	**0.126**	**0.072**
Group 4	0.185	**0.211**	**0.146**	**0.089**	**0.048**
Group 5 (Lowest θ)	**0.207**	**0.174**	**0.083**	**0.039**	**0.019**

Note. The bold number indicates that number is larger than the corresponding value from *TW*.

In the random NEAT and random PC designs, *TW* had relatively high recovery rates, its estimation for item difficulty parameters were similar to *1bit(p_tw)* and *SMC(1+p_tw)*, and it had better estimation for item discrimination parameters. The difference between its imputed response matrices and the full response matrix were shown in Figures 3.58c and 3.58d. To further investigate the reason that *1bit(p_tw)*

provided worse item parameter estimation than *TW*, its summarized difference between imputed response matrix and full response matrix under random NEAT design is presented in Tables 3.23 and 3.24. The pattern was similar to that in targeted NEAT designs.

Table 3.23: Summarized Difference Between Imputed Response Matrices from *1bit(p_tw)* and Full Response Matrix with Regards to Proportion of Wrongly Imputed Correct Responses in Random NEAT Design

	item 1 - 30	item 31 - 60	item 61 - 90	item 91 - 120	item 121 - 150
Group 1 (Highest θ)	0.028	**0.046**	**0.087**	**0.136**	0.140
Group 2	**0.058**	**0.103**	**0.142**	0.126	0.045
Group 3	**0.093**	**0.151**	0.110	0.043	0.006
Group 4	**0.132**	0.143	0.043	0.006	0.000
Group 5 (Lowest θ)	0.139	0.064	0.007	0.000	0.000

Note. The bold number indicates that number is larger than the corresponding value from *TW*.

Table 3.24: Summarized Difference Between Imputed Response Matrices from *1bit(p_tw)* and Full Response Matrix with Regards to Proportion of Wrongly Imputed Incorrect Responses in Random NEAT design

	item 1 - 30	item 31 - 60	item 61 - 90	item 91 - 120	item 121 - 150
Group 1 (Highest θ)	0.000	0.001	0.020	0.085	0.156
Group 2	0.001	0.017	0.092	**0.183**	**0.143**
Group 3	0.013	0.077	**0.157**	**0.153**	**0.084**
Group 4	0.064	**0.173**	**0.151**	**0.090**	**0.045**
Group 5 (Lowest θ)	**0.184**	**0.173**	**0.083**	**0.035**	**0.016**

Note. The bold number indicates that number is larger than the corresponding value from *TW*.

3.4.4 Summary of Simulation Study 1 Results

In the simulation study 1, the proposed ten adaptations for 1-bit matrix completion and structured matrix completion were evaluated under various conditions. The findings suggested that the item banks and sample sizes did not have much influence on their performance, but incomplete calibration designs did. In particular, the proposed adaptations were able to recover the structured missingness from the targeted

NEAT, random NEAT, and random PC designs, but not for targeted PC designs. In addition, sample sizes had slight impact. Specifically, item parameter estimations were slightly improved as sample sizes increased.

The proposed dichotomizing method, *p_tw*, can handle the violation of uniformly missing assumptions of 1-bit matrix completion. It produced an imputed response matrix that was similar to the full response matrix, and thus improved the recovery rates and estimations for b parameters significantly. In addition, a few proposed adapted structured matrix completion methods, such as *SMC(1+p_tw)* and *SMC(p1+p_tw)*, also had better recovered the characteristics of the full response matrix than *SMC(1+Sign)* and *SMC(p1+Sign)*, which led to higher recovery rates and better estimations for item difficulty parameters. However, these proposed adaptations overestimated item discrimination parameters. By comparing the imputed response matrix and the full response matrix, these adapted matrix completion methods, such as *1bit(p_tw)* and *SMC(1+p_tw)*, imputed more correct responses to easy items from examinees of high abilities (upper left triangle of the response matrix) and imputed more incorrect responses to hard items from examinees of low abilities (bottom right triangle of the response matrix). In this way, though they improved recovery rates and the estimation for item difficulty parameters, they were not capable of recovering item discrimination parameters with high accuracy because of the more extreme response patterns.

The best adaptations, *1bit(p_tw)*, *SMC(1+p_tw)*, and *SMC(p1+p_tw)*, had higher recovery rates than all the baselines. They also provided the most accurate estimations for item difficulty and discrimination parameters in targeted NEAT designs. However, *TW* provided similarly good estimations for item difficulty parameters and even more accurate estimations for item discrimination parameters than the adapted matrix completion methods in random NEAT and random PC designs. The findings of this study inspired us to improve the item bank calibration by eliminating the extreme response patterns from *1bit(p_tw)*, *SMC(1+p_tw)*, and *SMC(p1+p_tw)*, which were introduced in the next study.

Chapter 4

Study 2: Develop New Item Bank Calibration Strategies

The purpose of Study 2 was to develop new item bank calibration strategies based on the adapted matrix completion methods investigated in Study 1. The results of simulation study 1 indicated that both 1-bit and structured matrix completion with appropriate adaptations provided more accurately imputed responses than baselines. In particular, *1bit(p_tw)*, *SMC(1+p_tw)*, and *SMC(p1+p_tw)* recovered the features of the full response matrix if using targeted NEAT, random NEAT, and random PC designs. In addition, they provided more accurate estimations of item difficulty parameters than the baselines. However, they overestimated item discrimination parameters, because these adapted matrix completion methods imputed more correct responses to easy items for examinees of high abilities and more incorrect responses to difficult items for examinees of low abilities after examining their response patterns.

The overall goal of this study was to mitigate the extreme response patterns found in the adapted matrix completion methods to further improve the item calibration results. Specifically, three methods, switching method, softened threshold method, and pooling method were proposed and then evaluated in a simulation study.

4.1 Methods

4.1.1 Switching method

The principle of switching method was to switch extreme patterns, such as mostly incorrect or correct responses, in the imputed response matrices in study 1. In study 1, we found that the best adaptations, such as *1bit(p_tw)* and *SMC(1+p_tw)*, imputed more correct responses to examinees of high abilities on easy items, and more incorrect responses to examinees of low abilities on difficult items. These methods resulted in overestimation for the item discrimination parameters. Thus, the degree of overestimation would be reduced if some correct responses of examinees of high ability and easy items were switched to incorrect ones, and some incorrect responses of examinees of low ability and hard items were switched to correct ones.

To execute this idea, three sets of switching parameters were necessary: (1) the proportion of responses to be switched p, (2) the proportion of selected responses to be switched from correct to incorrect responses q_1, and (3) the proportion of selected responses to be switched from correct to incorrect responses q_2. These three parameters should be determined based on recovery rates, items' difficulties and examinees' abilities. That is to say, the proportion of responses to be switched, p, should be small when the recovery rates are high, and thus the correctly imputed responses have the small chance to be selected and switched. In addition, q_1 should be large for easy items from examinees of high abilities, q_2 should be large for hard items from examinees of low abilities. In this way, the extreme response patterns would be reduced.

The challenge of the switching method was to determine these three parameters. In this study, We used the results of study 1 as reference to propose a set of switching parameters that were applied to all conditions including four incomplete calibration designs, three item banks, and three sample sizes. The procedure of determining these switching parameters were introduced next.

Determining p parameter. To determine p, the general pattern of recovery rates in all conditions was considered. Especially, easy and hard items had higher recovery rates than medium-difficulty items (e.g., Figures 3.14, 3.15, C.1, and C.2). In addition, examinees of high and low ability had higher recovery rates

than examinees of medium ability (Figures 4.1, C.31, and C.32). Thus, the choices for p followed this pattern. Specifically, p should be small both from responses of either easy items and high-ability examinees and from responses of hard items and low-ability examinees, because these two groups of responses were most correctly imputed already as observed in study 1. On the contrary, p should be large for from responses of easy items and low-ability examinees and from responses of hard items and high-ability examinees. These two types of responses had relatively low chance to be imputed correctly as observed in study 1.

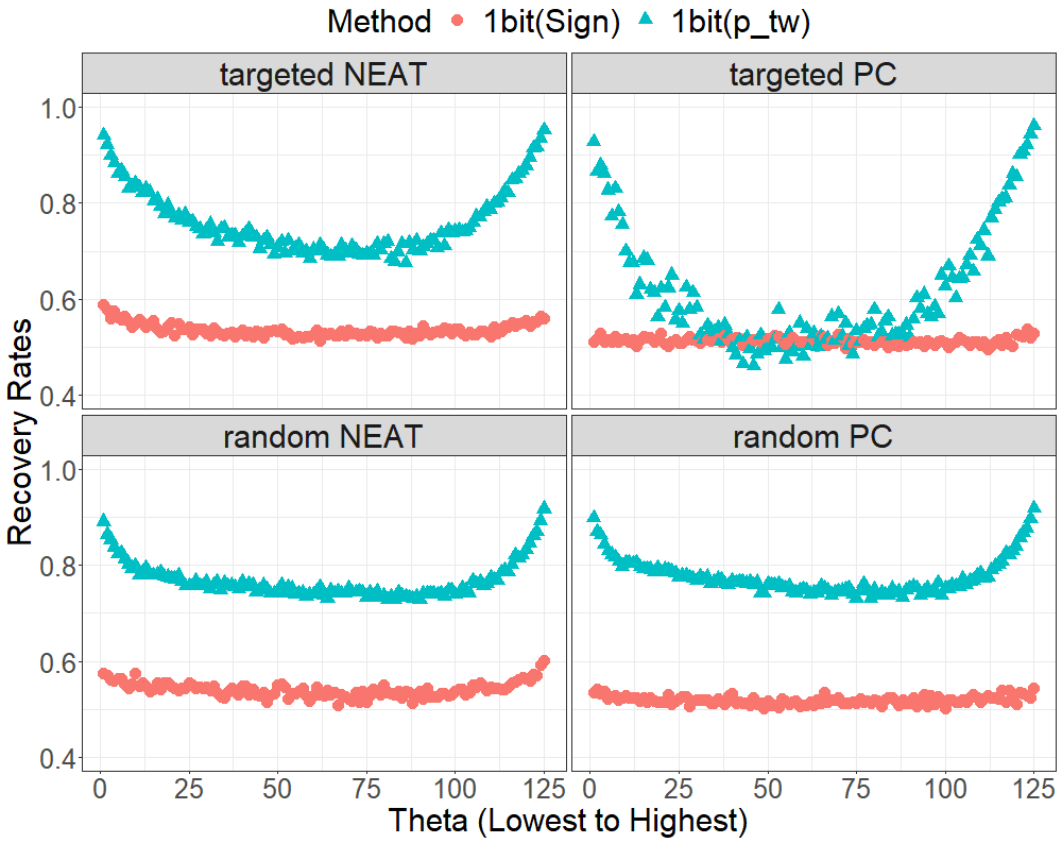
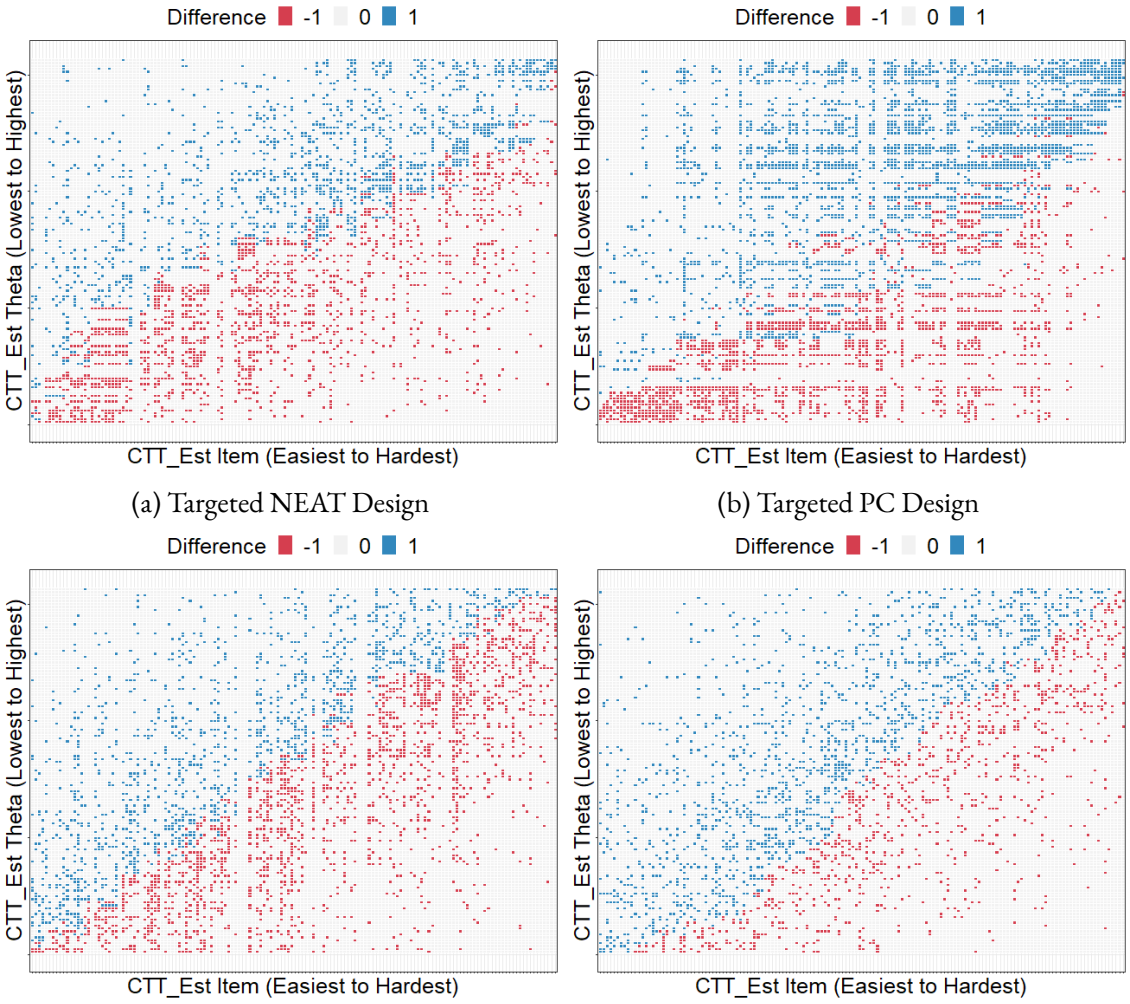


Figure 4.1: Recovery Rates Per Examinees Using *1bit(p_tw)* and *1bit(Sign)* in Practical Item Bank and $N = 125$

The above-mentioned pattern for p was dependent on the item difficulty and examinee's ability, which were unknown in reality. In this study, we proposed to use the imputed item and person scores as empirical indicators for item difficulty and examinee's ability. Especially, item score was the total number of correct responses for that item among all examinees, and person score was the total number of correct responses for that examinee among all items. For example, Figure 4.2 presents the difference between the heat maps

of full response matrix and $1bit(p_tw)$ imputed response matrix after ordered with the item scores and person scores. They were similar to the heat maps that were ordered by the true item and ability parameters (Figure 3.29). From this set of Figures, the extreme response pattern was shown in all four incomplete calibration designs. Similar patterns were also shown in $SMC(1 + p_tw)$ (Figure 4.3). Thus, ordering the imputed response matrix using item scores and person scores would work for the purpose of setting switching parameters.



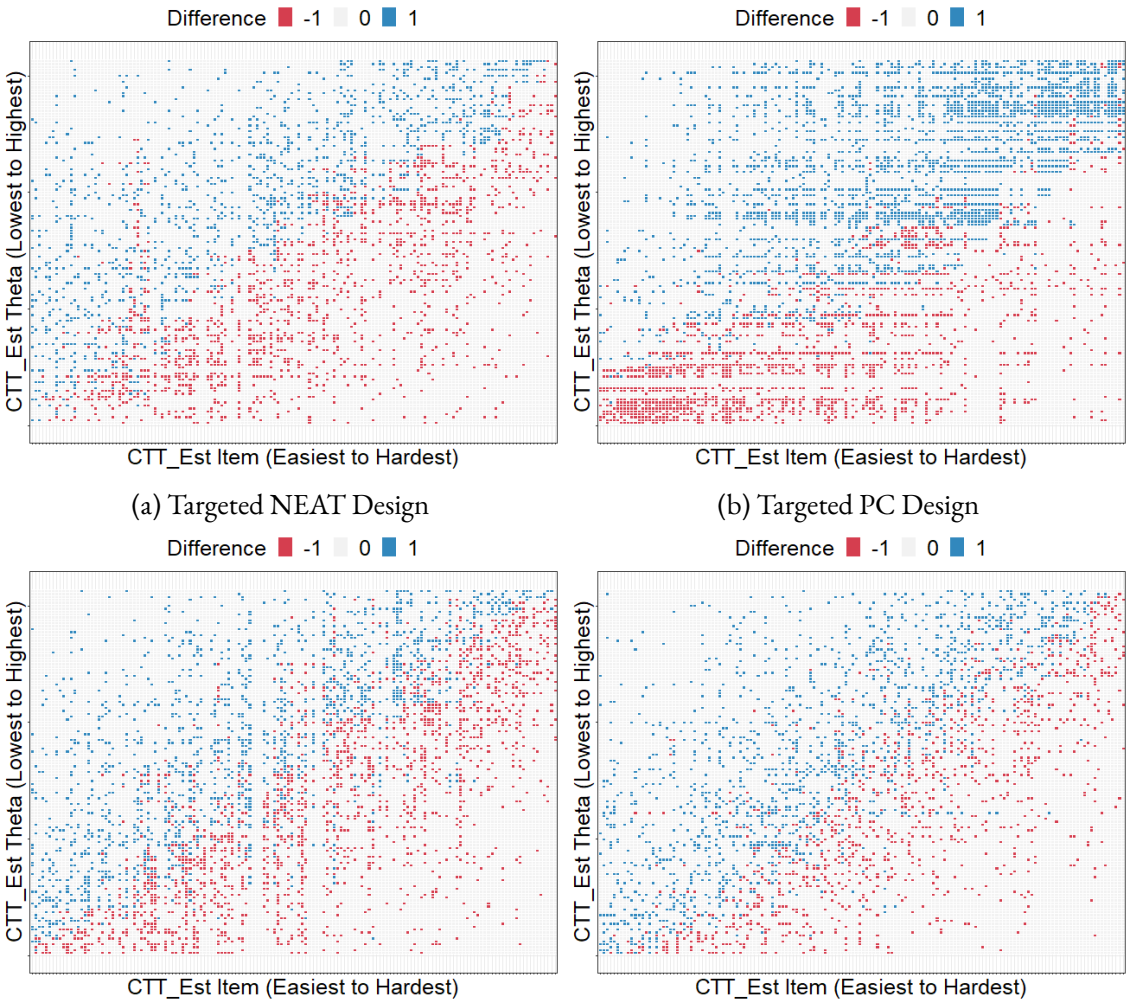
(a) Targeted NEAT Design

(b) Targeted PC Design

(c) Random NEAT Design

(d) Random PC Design

Figure 4.2: The Difference Between Full Response Matrix and Imputed Response Matrix Using *1bit(p_tw)* in Practical Item Bank with Sample Size of 125



(a) Targeted NEAT Design

(b) Targeted PC Design

(c) Random NEAT Design

(d) Random PC Design

Figure 4.3: The Difference Between Full Response Matrix and Imputed Response Matrix Using *SMC(1+p_tw)* in Practical Item Bank with Sample Size of 125

Given the observations of recovery rates from study 1, two sets of p were proposed for NEAT and PC designs respectively (Figure 4.4). The two sets of p followed the same pattern. That is, p was large when item difficulties and examinees' abilities matched on the diagonal, and p was small when items were either much harder or easier than examinees' abilities. The only difference between p in NEAT and PC designs was due to the different number of designed test forms. NEAT designs had five test forms and hence five item sets. These five item sets were ordered by the item scores, and these p values were used for both targeted NEAT and random NEAT designs. Note that item sets in Figure 4.4a were different from the test forms in either targeted NEAT or random NEAT designs. The test forms in the targeted NEAT designs were constructed with anchor items and item sets based on the true item difficulty, and the test forms in the random NEAT designs were constructed with anchor items with rest of the items assigned to test forms randomly. PC design has four test forms and hence four item sets. These p values were used for both targeted PC and random PC designs. Note that these item sets were ordered by item scores, and therefore they were not the same as the test forms in either targeted PC or random PC designs. Lastly, the groups of examinees in Figure 4.4 were ordered by the person scores, and thus they were different from the groups in the four incomplete calibration designs. The groups were given the true ability parameters in the targeted designs or assigned randomly in random designs.

Determining q_1 and q_2 parameters.

To determine q_1 and q_2, the results of recovery rates from study 1 were also used as the reference. For example, Figures 4.2 and 4.3 indicate that the *1bit(p_tw)* and *SMC(1+p_tw)* produced more incorrectly imputed correct responses for the combination of hard items and high-ability examinees. That is, q_1 should follow a general pattern where it should be large in this combination of θ and item difficulty and it should be small for the opposite combinations. The proposed q_1 were shown in Figure 4.5, which reflected this pattern in both NEAT and PC designs. On the contrary, q_2 should follow a general pattern where it should be large for responses of easy items and examinees of high ability. The proposed $q2$ were shown in Figure 4.6.

p% of responses are selected to switch	Item Set 1 (Easy)	Item Set 2	Item Set 3	Item Set 4	Item Set 5 (Hard)
Group 5 (Large theta)	0.05	0.05	0.15	0.15	0.25
Group 4	0.05	0.15	0.15	0.25	0.15
Group 3	0.15	0.15	0.25	0.15	0.15
Group 2	0.15	0.25	0.15	0.15	0.05
Group 1 (Small theta)	0.25	0.15	0.15	0.05	0.05

(a) NEAT Designs

p% of responses are selected to switch	Item Set 1 (Easy)	Item Set 2	Item Set 3	Item Set 4 (Hard)
Group 5 (Large theta)	0.05	0.05	0.15	0.25
Group 4	0.05	0.15	0.25	0.25
Group 3	0.15	0.25	0.25	0.15
Group 2	0.25	0.25	0.15	0.05
Group 1 (Small theta)	0.25	0.15	0.05	0.05

(b) PC Designs

Figure 4.4: Proposed Switching Parameters p

q1% of the selected right responses are switched to wrong	Item Set 1 (Easy)	Item Set 2	Item Set 3	Item Set 4	Item Set 5 (Hard)
Group 5 (Large theta)	0.9	0.9	0.9	0.7	0.5
Group 4	0.9	0.9	0.7	0.5	0.1
Group 3	0.9	0.7	0.5	0.1	0
Group 2	0.7	0.5	0.1	0	0
Group 1 (Small theta)	0.5	0.1	0	0	0

(a) NEAT Designs

q1% of the selected right responses are switched to wrong	Item Set 1 (Easy)	Item Set 2	Item Set 3	Item Set 4 (Hard)
Group 5 (Large theta)	0.9	0.9	0.8	0.6
Group 4	0.9	0.8	0.6	0.4
Group 3	0.8	0.6	0.4	0
Group 2	0.6	0.4	0	0
Group 1 (Small theta)	0.4	0	0	0

(b) PC Designs

Figure 4.5: Proposed Switching Parameters $q1$

q2% of the selected wrong responses are switched to right	Item Set 1 (Easy)	Item Set 2	Item Set 3	Item Set 4	Item Set 5 (Hard)
Group 5 (Large theta)	0	0	0	0.1	0.5
Group 4	0	0	0.1	0.5	0.7
Group 3	0	0.1	0.5	0.7	0.9
Group 2	0.1	0.5	0.7	0.9	0.9
Group 1 (Small theta)	0.5	0.7	0.9	0.9	0.9

(a) NEAT Designs

q2% of the selected wrong responses are switched to right	Item Set 1 (Easy)	Item Set 2	Item Set 3	Item Set 4 (Hard)
Group 5 (Large theta)	0.0	0.0	0.0	0.4
Group 4	0.0	0.0	0.4	0.6
Group 3	0.0	0.4	0.6	0.8
Group 2	0.4	0.6	0.8	0.9
Group 1 (Small theta)	0.6	0.8	0.9	0.9

(b) PC Designs

Figure 4.6: Proposed Switching Parameters $q2$

4.1.2 Softened threshold method

The second method, the softened threshold method, was less straightforward than the switching method. This method proposed a less restricted threshold than *p_tw* by pulling *p_tw* closer to 0.5. In study 1, we found that *p_tw* recovered the full response matrix but showed an extreme pattern, which produced overestimated item discrimination parameters. On the contrary, *Sign* produced evenly distributed correct and incorrect responses among imputed values, which led to worse estimations for difficulty parameters b than *p_tw* and underestimations for discrimination parameters a. The extreme patterns from *p_tw* would probably be reduced and thus the estimation of item discrimination parameters would be improved, if pulling the cutoff points of *p_tw* closer to 0.5. Note that, after logistic transform, the continuous outputs from matrix completion were in the scale of 0 to 1, and 0.5 would be the equivalent cutoff point of the *Sign* method. Specifically, for item i and examinee j, a softened threshold, $p_tw(soft)_{ij}$ for its imputed value x^*_{ij} on the logistic scale, $x^*_{ij} \in (0, 1)$, was proposed following the formula:

$$p_tw(soft)_{ij} = TW_{ij} + \frac{0.5 - TW_{ij}}{2}, \tag{4.1}$$

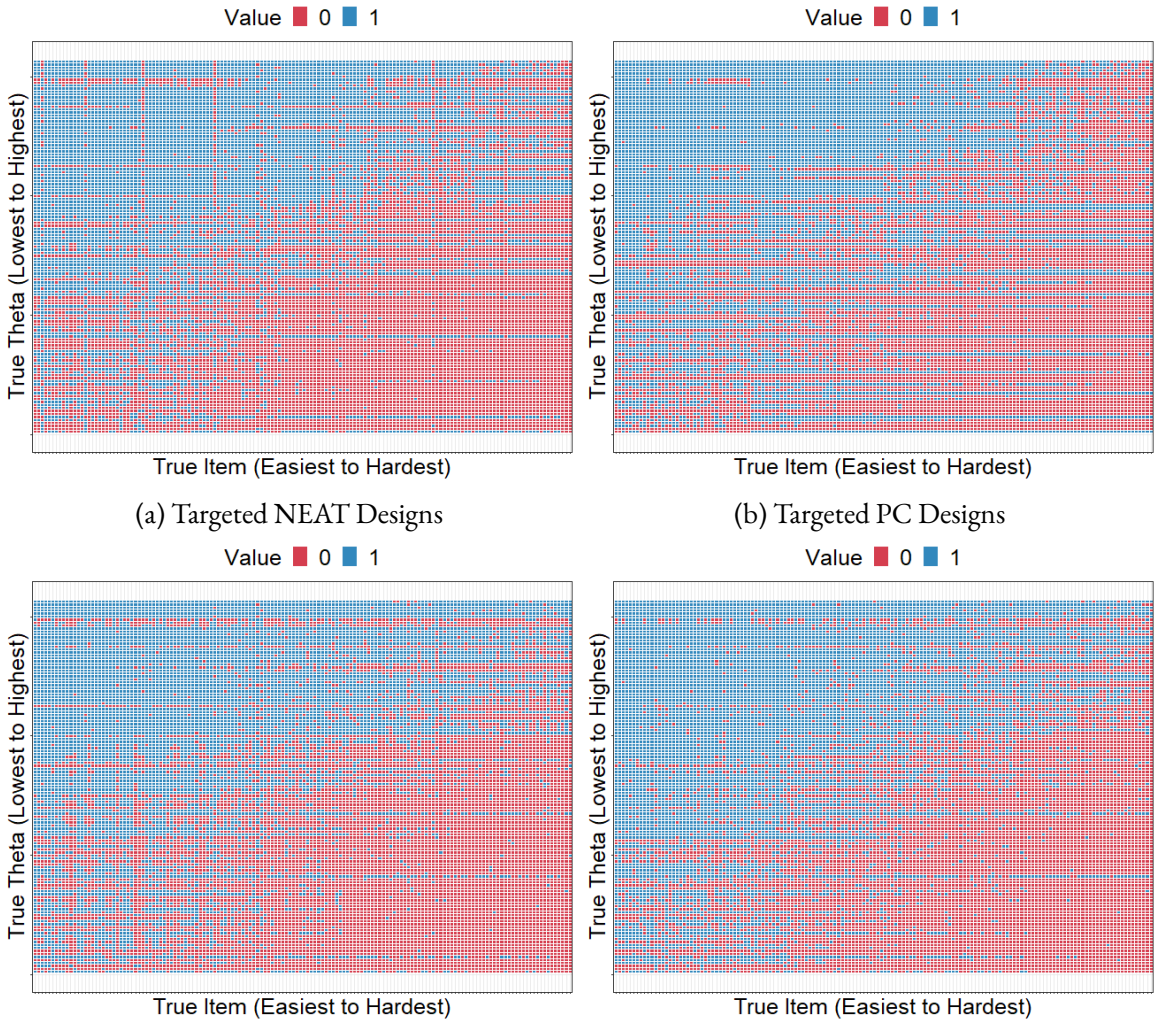
where $TW_{ij} = PM_j + IM_i - OM$ and was presented in Table 3.1. The final binary imputed value would be 1 if $x^*_{ij} > (1 - pool_{ij})$ and 0 otherwise.

To demonstrate Equation 4.1, let's assume a response from an examinee of high ability to an easy item is missing. In this case, the item mean IM_i is large because of the low item difficulty, and person mean PM_j is also large because of the high ability. The cutoff point of p_tw, which is $1 - TW_{ij} = 1 - (PM_j + IM_i - OM)$, would be very small. Thus, the imputed response has very high chance to be larger than $1 - TW_{ij}$ and be dichotomized as correct. However, if using this softened cutoff point, the cutoff point $1 - p_tw(soft)_{ij}$ would be larger than $1 - TW_{ij}$ and closer to 0.5, and the chance of a correct imputation is reduced. For example, if $TW_{ij} = 0.9$ with $1 - TWij = 0.1$, the corresponding $p_tw(soft)_{ij} = 0.9 + \frac{0.5-0.9}{2} = 0.7$ with $1 - p_tw(soft)_{ij} = 0.3$. Thus, the imputed value has to be larger than 0.3 to be dichotomized as a correct response, which is less possible than using p_tw with the cutoff point of 0.1.

4.1.3 Pooling converting method

The third method, pooling converting methods, was designed for structured matrix completion only. We found that among the four continuous structured matrix completion, *1* and *p1* recovered the characteristics of the full response matrix. When paired with *p_tw* as dichotomizing methods, they showed more extreme response patterns than the full response matrix. In addition, *SMC(p2+p_tw)* produced less extreme response patterns (Figure 4.7) than *SMC(1+p_tw)* and *SMC(p1+p_tw)*. Thus, the extreme response patterns would be reduced if pooling the imputed responses from *SMC(1+p_tw)*, *SMC(p1+p_tw)*, and *SMC(p2+p_tw)*. Note that *SMC(p0+p_tw)* was not included, because it showed a systematic pattern for bias(a) where bias(a) increased as item difficulty increased.

Let M_1, M_2, and M_3 represent the imputed response matrices from binary, p1, and p2, and M_{final} represent the final response matrix after pooling. In particular, this method follows four steps:



(a) Targeted NEAT Designs

(b) Targeted PC Designs

(c) Random NEAT Designs

(d) Random PC Designs

Figure 4.7: Heat Maps for *SMC(p2+p_tw)* Imputed Matrix using Practical Item Bank with $N = 125$.

1. Compute the point-biserial correlation for each item in all three imputed response matrices.
2. If any item has point-biserial correlation smaller than 0, all the responses for that item would be replaced with missing values.
3. Randomly select one imputed response from $M_{1(i,j)}$, $M_{2(i,j)}$, and $M_{3(i,j)}$ to fill in $M_{final(i,j)}$. If all three candidates, $M_{1(i,j)}$, $M_{2(i,j)}$, and $M_{3(i,j)}$, are missing because of negative point-biserial correlation, either 0 or 1 would be randomly assigned to $M_{final(i,j)}$.
4. The above three steps are repeated until all cells in M_{final} are filled.

4.2 Simulation Study 2 Design

The purpose of this simulation study was to evaluate the proposed three strategies based on the modified 1-bit matrix completion and structured matrix completion in study 1. In particular, the switching method was applied to *1bit(p_tw)* and *SMC(1+p_tw)*. These two adaptations were selected because they showed high recovery rates, good estimations for item difficulty parameters, and overestimation for item discrimination parameters. In addition to the proposed set of switching parameters (Figures 4.4, 4.5, and 4.6), denoted as *Switch_General*, another set of switching parameters were introduced in this simulation. This set of switching parameters used the results of study 1, and customized the switching parameters for each incomplete calibration design and item bank to achieve the maximum improvement. This second set of switching parameters, denoted as *Switch_Preferred*, was treated as the upper baseline, which showed the greatest amount of the improvements when some prior information were available. Although the *Switch_General* also used the simulation results from study 1, the switching parameters were determined to reflect only the general pattern and therefore not fit any design in particular. The results of *Switch_Preferred* would show the potential improvements of switching method in realistic situations. The procedure of selecting the switching parameters of *Switch_Preferred* was presented in the Appendix B. In addition, the softened threshold method was applied to *SMC(1+p_tw)* and *SMC(p1+p_tw)*. It is not used for *1bit(p_tw)*, because the imputed continuous values from 1-bit matrix completion were very close

to 0.5 on the logistic scale. In this case, the softened threshold would not make much difference. Using the example in the section 4.2, though softened threshold pulls the cutoff point from 0.1 to 0.3 and decreases the chance for an imputed value to be dichotomized as a correct response, the response would still be imputed as correct if the imputed continuous value is 0.45 on the logistic scale. Generally, we found that all the imputed values from 1-bit matrix completion were very close to 0.5 on the logistic scale in study 1, and thus softened threshold method was not applied to *1bit(p_tw)*. Lastly, since 1-bit matrix completion method did not require a converting method, the pooling the converting method was not used for 1-bit matrix completion either.

Thus, in this simulation study, there were 7 methods investigated: switching method using general switching parameters for *1bit(p_tw)* and *SMC(1+p_tw)*, which were denoted as *1bit(Switch_General)* and *SMC(1+Switch_General)*, switching method using item bank and incomplete calibration design specific switching parameters for *1bit(p_tw)* and *SMC(1+p_tw)*, which were denoted as *1bit(Switch_Preferred)* and *SMC(1+Switch_Preferred)*, softened threshold method for *SMC(1+p_tw)* and *SMC(p1+p_tw)*, denoted as *SMC(1+Soft)* and *SMC(p1+Soft)*, and pooling method for *SMC(1+p_tw)*, *SMC(p1+p_tw)*, and *SMC(p2+p_tw)*, which is denoted as *SMC(pool+p_tw)*. Their recovery rates, bias, and RMSE for item parameter estimations were evaluated with their corresponding adaptations study 1, and the best baseline, *TW*. Thus, 15 methods, in total, were evaluated in this study. The 14 structured matrix completion based methods were summarized in Table 4.1. The general settings and procedures in this simulation study were the same as in study 1. The details can be found in Table 3.3 and Figure 3.9 in study 1.

4.3 Simulation Study 2 Results

4.3.1 Results from the switching method.

The switching method was applied to both *1bit(p_tw)* and *SMC(1+p_tw)*, and their results were presented separately. In particular, the recovery rates, bias(a), RMSE(a), and RMSE(b) from *1bit(Switch_General)* and *1bit(Switch_Preferred)* were first summarized. Their performance was compared to *1bit(p_tw)* to eval-

Table 4.1: Investigated Methods in Simulation Study 2

Proposed methods	Investigated in study 2	Corresponding method in study 1
Switching Methods	1bit(Switch_General) 1bit(Switch_Preferred) SMC(1+Switch_General) SMC(1+Switch_Preferred)	1bit(p_tw) SMC(1+p_tw)
Softened Threshold method	SMC(1+Soft) SMC(p1+Soft)	SMC(1+p_tw) SMC(p1+p_tw)
Pooling converting methods	SMC(pool+p_tw)	SMC(1+p_tw) SMC(p1+p_tw) SMC(p2+p_tw)

uate whether the item calibration was improved from study 1 and compared to TW to evaluate whether they achieved better performance than the best baseline in study 1. The results of *SMC(1+Switch_General)* and *SMC(1+Switch_Preferred)* were then presented following a similar outline.

Results from *1bit(Switch_General)* and *1bit(Switch_Preferred)* compared to *1bit(p_tw)* and *TW*.

In this section, the results of *1bit(Switch_General)* and *1bit(Switch_Preferred)* from practical item bank with $N = 125$ were presented first. The influence from sample sizes and item banks were then investigated.

Recovery rates in the practical item bank with $N = 125$.

The recovery rates of *1bit(Switch_General)* and *1bit(Switch_Preferred)* were similar to each other, and their recovery rates were slightly lower than *1bit(p_tw)* (Figure 4.8). That is, both *1bit(Switch_General)* and *1bit(Switch_Preferred)* were not capable of improving the recovery rates. In addition, they had similar recovery rate per item with TW except in targeted NEAT designs, where TW has the lowest recovery rates.

Item discrimination parameter estimations in the practical item bank with $N = 125$.

The average bias(a) and RMSE(a) for item discrimination parameters were summarized in Table 4.2. The bold numbers indicate that the switching method had better results than *1bit(p_tw)*. The bias(a) and

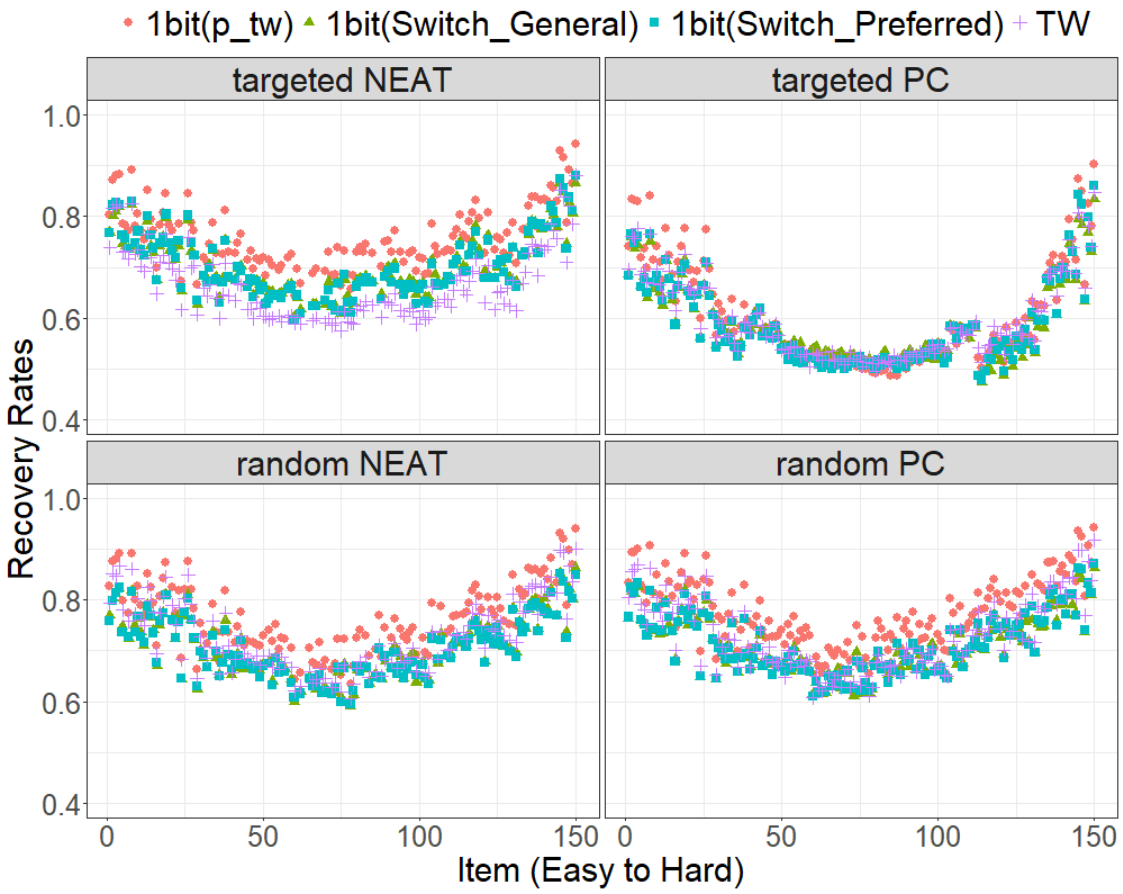


Figure 4.8: Recovery Rates of the Switching Method with *1bit(p_tw)* and *TW* in Practical Item Bank and $N = 125$

RMSE(a) of *1bit(Switch_General)* and *1bit(Switch_Preferred)* were similar. *1bit(Switch_Preferred)* had slightly smaller bias(a) and RMSE(a) than *1bit(Switch_General)* only in the targeted NEAT and random NEAT designs, which was not expected as the switching parameters of *1bit(Switch_Preferred)* were customized to each of the four incomplete calibration designs. This indicated that *Switch_Preferred* did not have advantage over *Switch_General* with the known information from the previous simulation study. In addition, both *1bit(Switch_General)* and *1bit(Switch_Preferred)* had smaller bias(a) and RMSE(a) than *1bit(p_tw)* in all four incomplete calibration designs, and their bias(a) and RMSE(a) were smaller than *TW* in the targeted NEAT, random NEAT, and random PC designs. Their bias(a) and RMSE(a) were slightly larger than those of *TW* in the targeted PC design.

To further evaluate their performance, Figure 4.9 presents their bias(a) per item along with *1bit(p_tw)* and *TW*. From this figure, *1bit(Switch_General)* and *1bit(Switch_Preferred)* were similar to each other and had much smaller bias(a) for almost all the items than *1bit(p_tw)* and similar to *TW* especially in targeted NEAT, random NEAT, and random PC designs. In targeted PC designs, bias(a) from *1bit(Switch_General)* and *1bit(Switch_Preferred)* were smaller than *1bit(p_tw)* only for the medium-difficulty items. In addition, *1bit(Switch_General)* and *1bit(Switch_Preferred)* produced more negative bias(a) for very easy and difficult items in random NEAT and random PC designs.

Item difficulty parameter estimations in the practical item bank with $N = 125$.

From Table 4.2, *1bit(Switch_General)* and *1bit(Switch_Preferred)* had smaller bias(b) and RMSE(b) than *1bit(p_tw)* only in random PC designs. They had slightly smaller or similar bias(b) than *1bit(p_tw)* in targeted NEAT and targeted PC designs. *1bit(Switch_General)* had much larger bias(b) and RMSE(b) than *1bit(p_tw)* in random NEAT designs, which was due to a few extreme estimations. For example, *1bit(Switch_General)* produced $\hat{b} = 69469.358$ for the item with $b = -2.671$ in one replication. After removing that one extreme estimation, the average of bias(b) and RMSE(b) were 0.049 and 2.751 respectively, and they were close to *1bit(p_tw)*. Compared to *TW*, both *1bit(Switch_General)* and *1bit(Switch_Preferred)* had similar or larger bias(b) and RMSE(b).

Table 4.2: Summary Statistics for Item Parameter Estimations for Switching Method with *1bit(p_tw)* and *TW* in Practical Item Bank and $N = 125$

Condition	Method	Bias(a)	RMSE(a)	Bias(b)	RMSE(b)
Targeted NEAT	1bit(p_tw)	2.024	3.087	0.077	0.667
	1bit(Switch_General)	**0.378**	**0.753**	0.080	0.873
	1bit(Switch_Preferred)	**0.238**	**0.653**	**0.074**	1.364
	TW	0.515	0.723	0.072	0.783
Targeted PC	1bit(p_tw)	0.606	0.818	0.059	0.824
	1bit(Switch_General)	**-0.027**	**0.455**	**0.020**	0.891
	1bit(Switch_Preferred)	**-0.110**	**0.544**	**0.020**	1.032
	TW	0.111	0.413	0.077	0.792
Random NEAT	1bit(p_tw)	2.330	4.018	0.091	2.221
	1bit(Switch_General)	**0.137**	**0.611**	9.313	68.228
	1bit(Switch_Preferred)	**0.121**	**0.582**	0.512	4.593
	TW	0.477	0.741	0.072	0.585
Random PC	1bit(p_tw)	1.121	1.549	-0.401	4.944
	1bit(Switch_General)	**0.064**	**0.573**	**-0.186**	**3.661**
	1bit(Switch_Preferred)	**0.115**	**0.564**	**0.009**	**1.447**
	TW	0.318	0.581	0.044	0.544

Note. The bold numbers indicated better item parameter estimations from switching method than *1bit(p_tw)*.

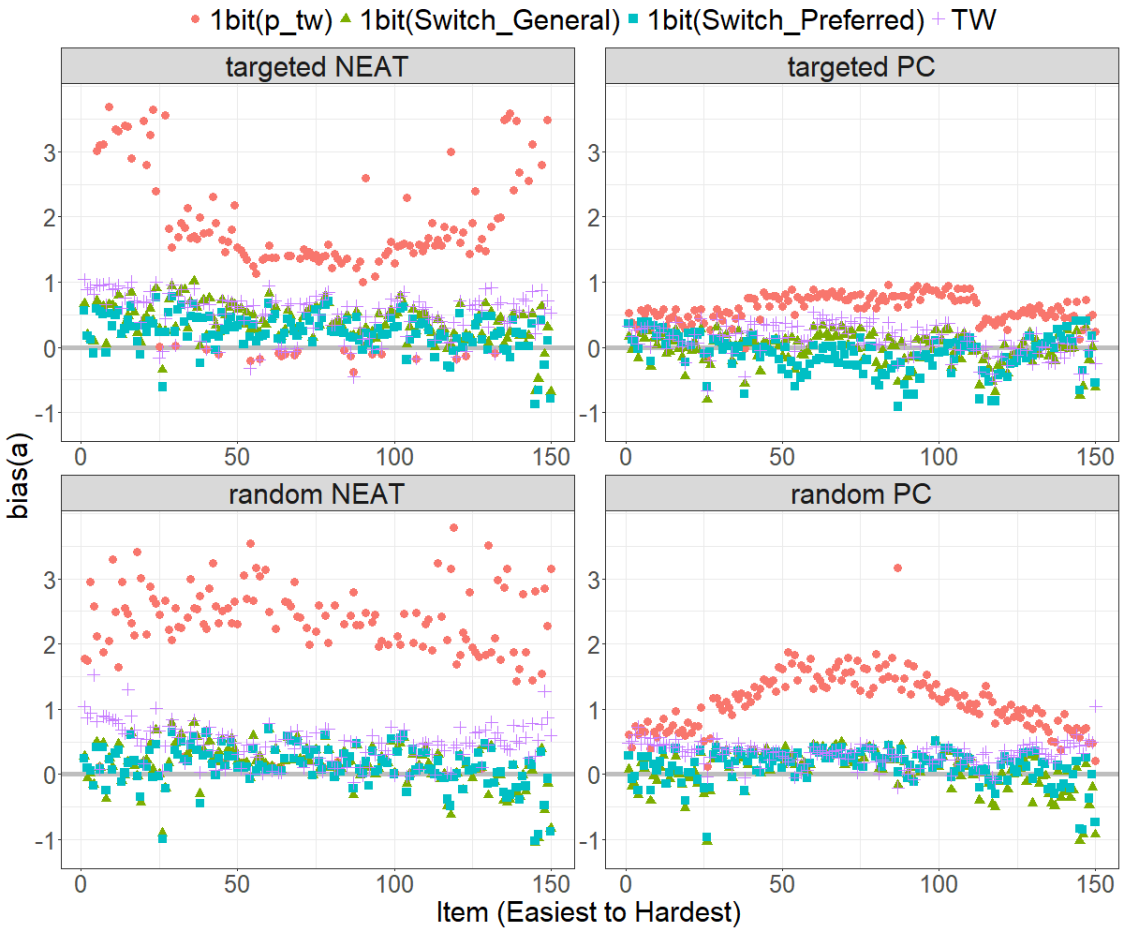


Figure 4.9: Bias(a) per Item for Switching Method with *1bit(p_tw)* and *TW* in Practical Item Bank and $N = 125$

To further examine their performance in estimating item difficulty parameters, their bias(b) per item were plotted in Figure 4.10 along with *1bit(p_tw)* and *TW*. This set of figures did not fix the range of the y-axis, and therefore the extreme estimations were shown. *1bit(Switch_General)* had one serious outliers in random NEAT designs, which was mentioned before. *1bit(Switch_Preferred)* and *1bit(Switch_General)* had a few extreme estimations in random PC designs, but they were smaller than *1bit(p_tw)*. In addition, Figure 4.11 zooms in with a fixed y-axis in the range of -2 to 2 to better visualize the bias(b) among methods. From this set of figures, the performance of *1bit(Switch_General)* and *1bit(Switch_Preferred)* were different across the four incomplete calibration designs. In targeted NEAT designs, *1bit(Switch_General)* and *1bit(Switch_Preferred)* had bias(b) closer to zero than *TW* for easy and difficult items, while they had a few large negative bias(b) for easy items and large positive bias(b) for difficult items. In random NEAT and random PC designs, *1bit(Switch_General)* and *1bit(Switch_Preferred)* overestimated *b* slightly more than *TW* for the hard items, and they underestimated *b* more for the easy items. To summarize, the switching method did not sacrifice the estimations of item difficulty parameter from *1bit(p_tw)*. It produced similar item difficulty parameters in the targeted NEAT and targeted PC designs, but they led to a few more extreme estimations in the random NEAT and random PC designs. Lastly, *1bit(Switch_Preferred)* did not have advantages over *1bit(Switch_General)* in terms of estimating item difficulty parameters.

Influence of item banks on *1bit(Switch_General)* and *1bit(Switch_Preferred)*.

The results of *1bit(Switch_General)* and *1bit(Switch_Preferred)* from the uniform and normal item banks were similar to the findings from the practical item bank. Firstly, the recovery rates were similar across item banks in all four incomplete calibration designs (Figures 4.12, 4.13, 4.14, and 4.15). Across three item banks, in the targeted NEAT design, both *1bit(Switch_General)* and *1bit(Switch_Preferred)* had recovery rates slightly lower than *1bit(p_tw)* but slightly higher than *TW*. In targeted PC designs, *1bit(Switch_General)* and *1bit(Switch_Preferred)* had similar recovery rates with the other two methods. In random NEAT and random PC designs, *1bit(Switch_General)* and *1bit(Switch_Preferred)* had similar recovery rates with *TW* but slightly worse than *1bit(p_tw)*.

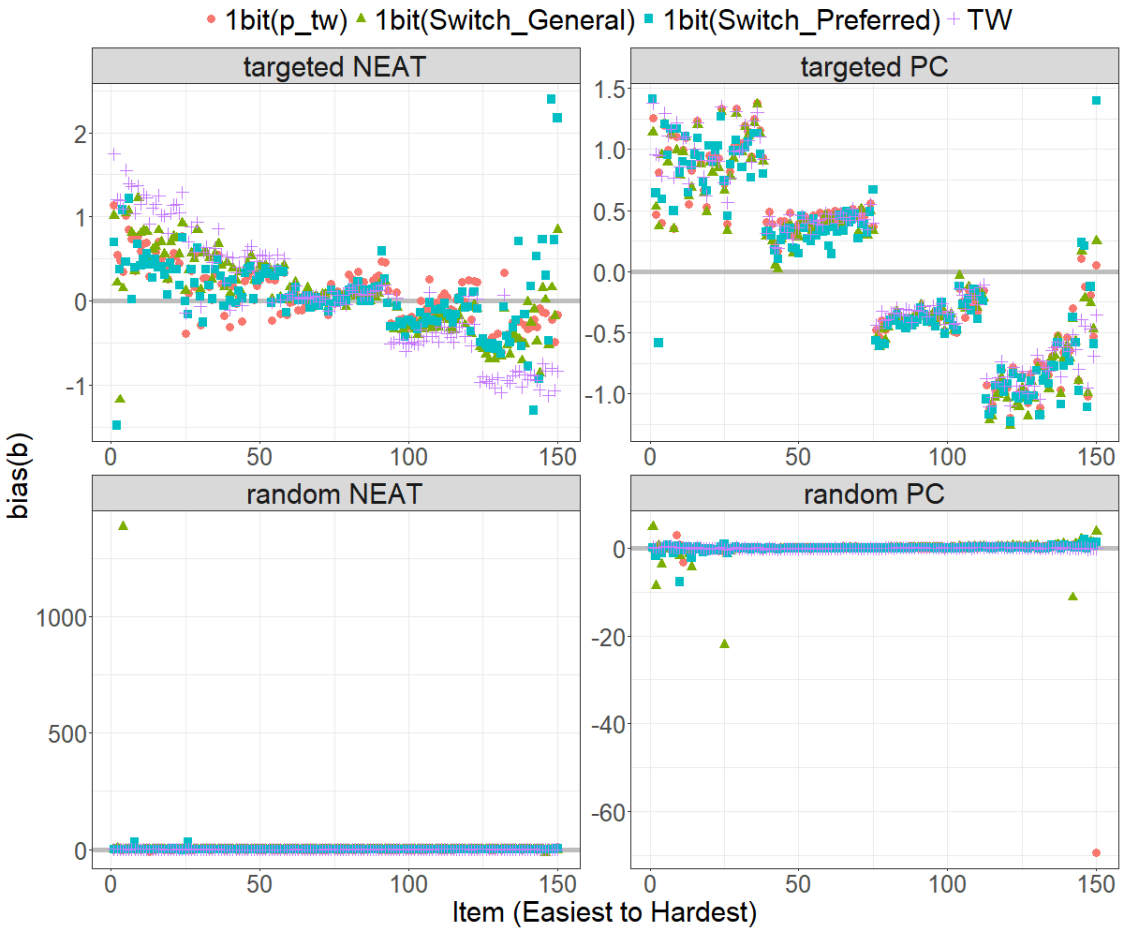


Figure 4.10: Bias(b) per Item for Switching Method with *1bit(p_tw)* and *TW* in Practical Item Bank and $N = 125$ (Free Y-Axis)

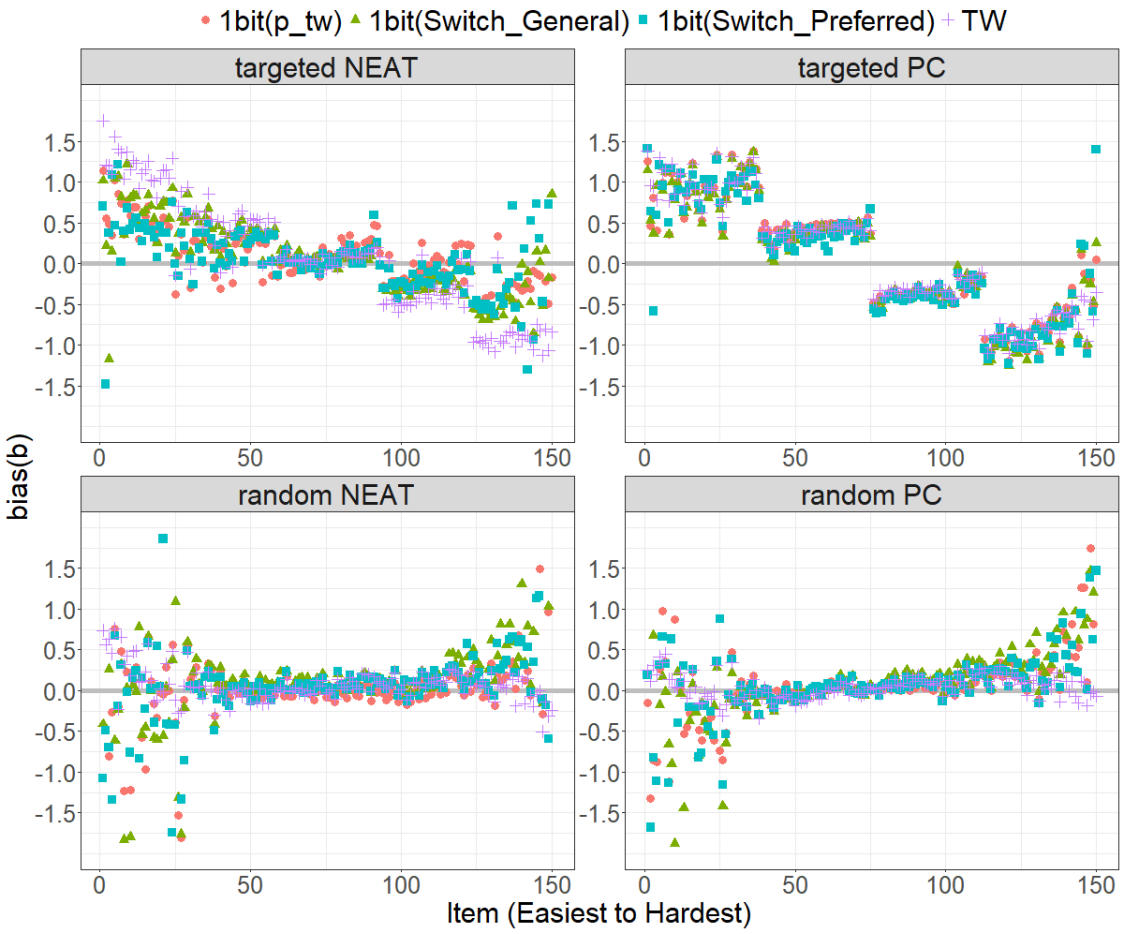


Figure 4.11: Bias(b) per Item for Switching Method with *1bit(p_tw)* and *TW* in Practical Item Bank and $N = 125$ (Zoomed-In with Fixed Y-Axis)

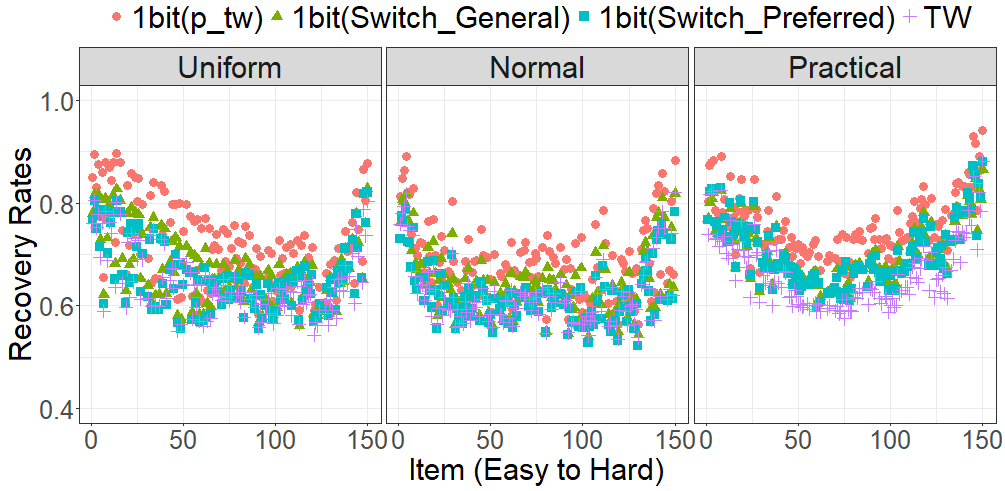


Figure 4.12: Recovery Rates per Item for Switching Method with *1bit(p_tw)* and *TW* Across Item Banks with $N = 125$ In Targeted NEAT Designs

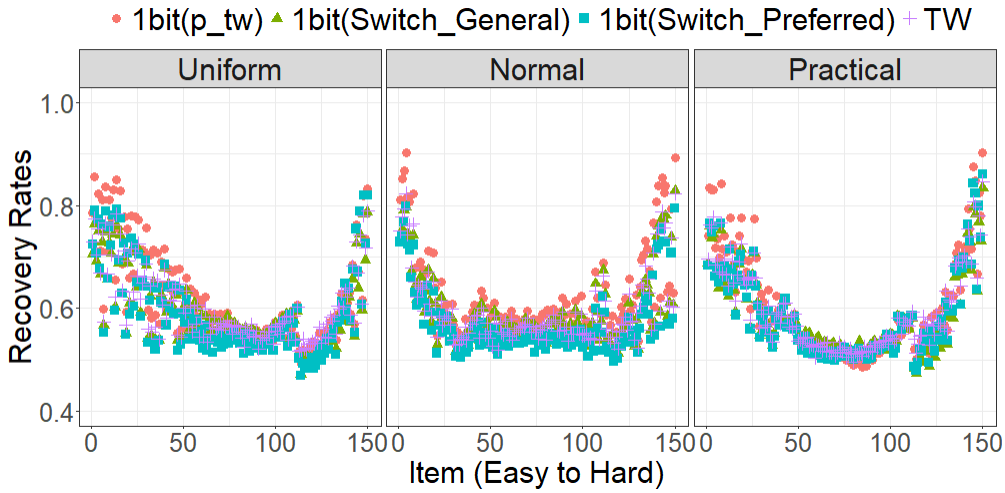


Figure 4.13: Recovery Rates per Item for Switching Method with *1bit(p_tw)* and *TW* Across Item Banks with $N = 125$ in Targeted PC Designs

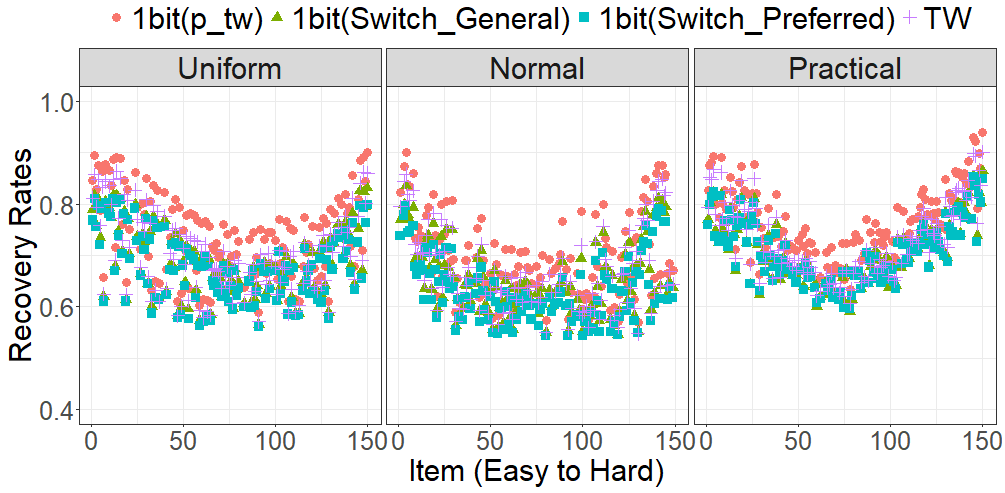


Figure 4.14: Recovery Rates per Item for Switching Method with *1bit(p_tw)* and *TW* Across Item Banks with $N = 125$ in Random NEAT Designs

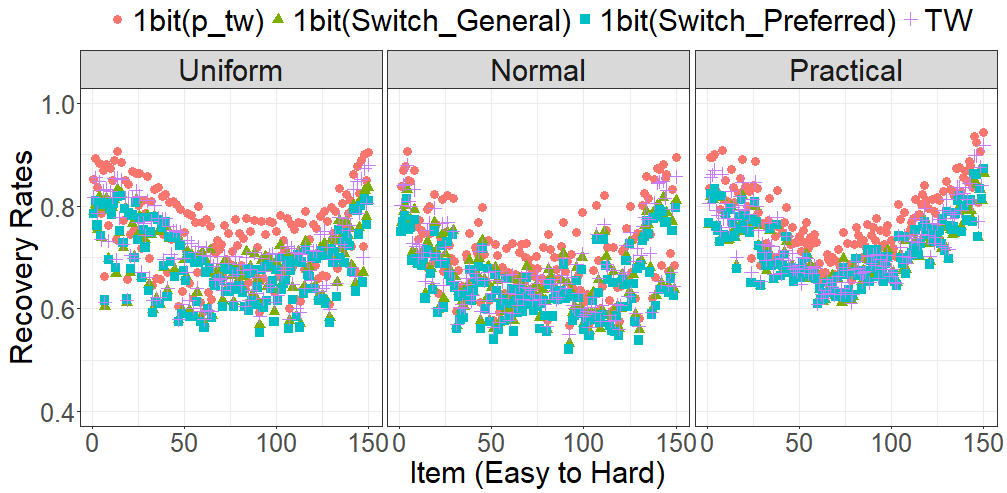


Figure 4.15: Recovery Rates per Item for Switching Method with *1bit(p_tw)* and TW Across Item Banks with $N = 125$ in Random PC Designs

Secondly, the results of both *1bit(Switch_General)* and *1bit(Switch_Preferred)* for item discrimination parameters were similar across item banks. For example, Figure 4.16 presents the RMSE(a) across item banks when $N = 125$. Both *1bit(Switch_General)* and *1bit(Switch_Preferred)* had much smaller RMSE(a) than *1bit(p_tw)* in all three item banks, especially in the targeted NEAT, random NEAT, and random PC designs. Their improvement to RMSE(a) was less obvious in the targeted PC design because *1bit(p_tw)*'s RMSE(a) was not too large to begin with. In addition, Figure 4.17 presents the bias(a) per item across three item banks under four incomplete calibration designs. The bias(a) showed similar patterns across three item banks, though bias(a) showed different patterns in different incomplete calibration designs as discussed before in practical item bank.

Thirdly, the results of RMSE(b) and bias(b) from *1bit(Switch_General)* and *1bit(Switch_Preferred)* were also similar across item banks (Figures 4.18 and 4.19). Generally, both *1bit(Switch_General)* and *1bit(Switch_Preferred)* had slightly larger RMSE(b) as *1bit(p_tw)* with a few more extreme estimations. Bias(b) from the four methods were generally similar across item banks.

Figure 4.16: RMSE(a) for Switching Method with *1bit(p_tw)* and *TW* Across Item Banks with $N = 125$

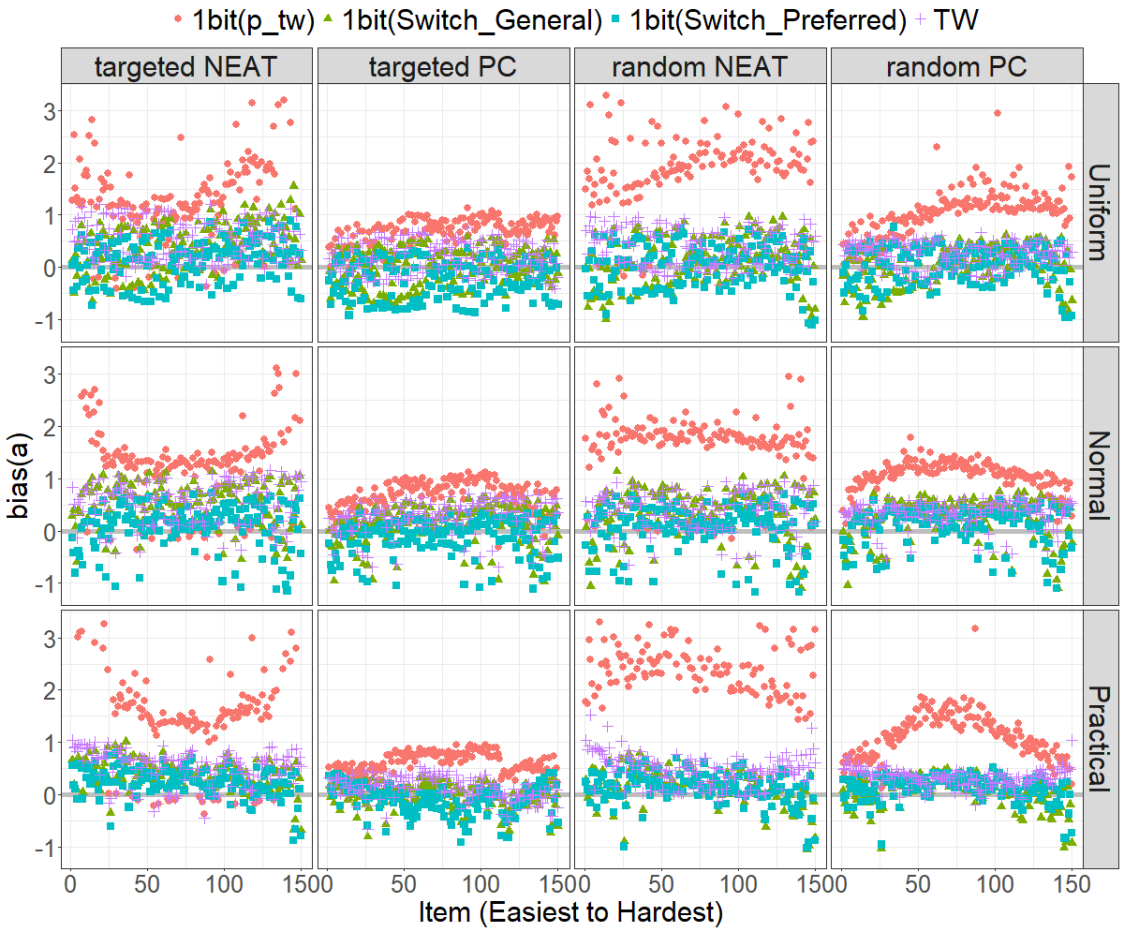


Figure 4.17: Bias(a) per item for Switching Method with *1bit(p_tw)* and *TW* Across Item Banks with $N = 125$

Figure 4.18: RMSE(b) for Switching Method with *1bit(p_tw)* and *TW* Across Item Banks with $N = 125$

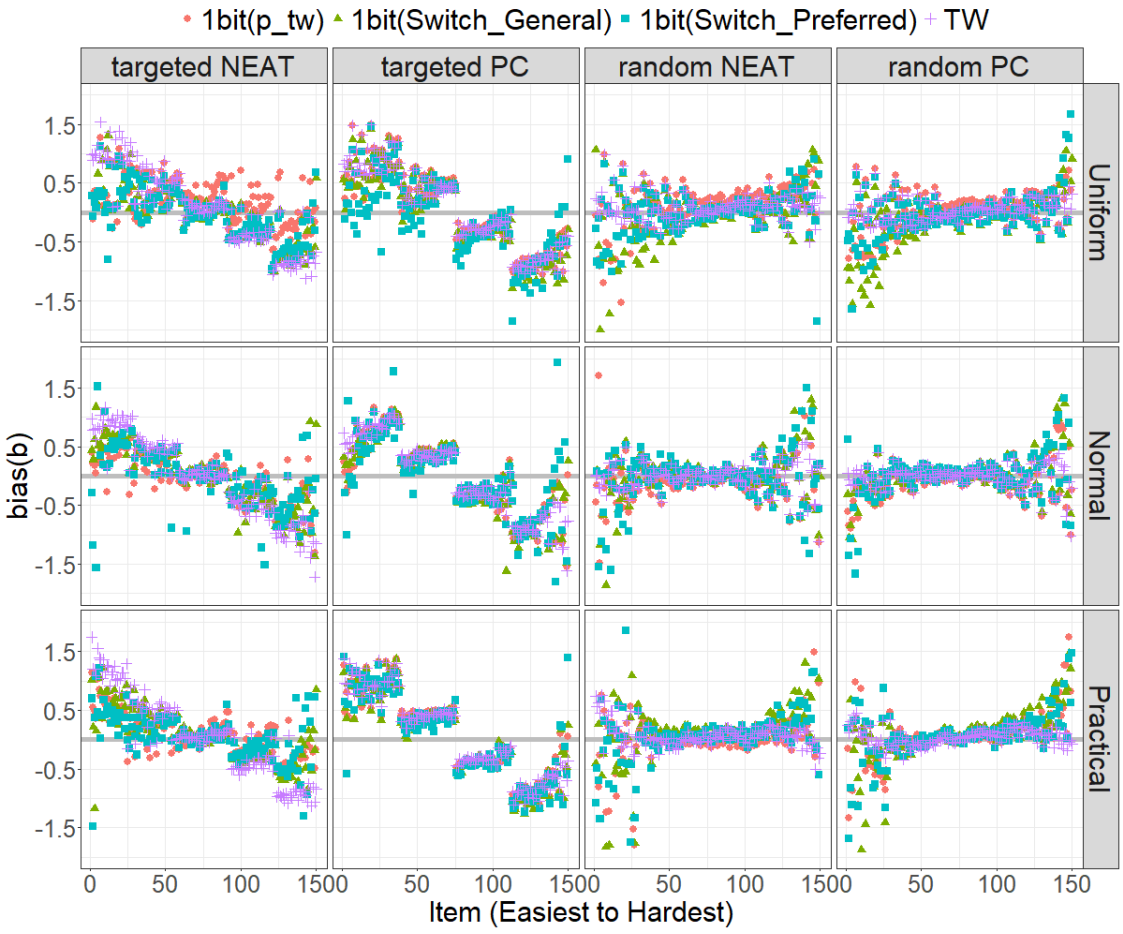


Figure 4.19: Bias(b) per item for Switching Method with *1bit(p_tw)* and *TW* Across Item Banks with $N = 125$

Influence of sample sizes on *1bit(Switch_General)* and *1bit(Switch_Preferred)*.

The influence of sample sizes on the *1bit(Switch_General)* and *1bit(Switch_Preferred)* was very little. For example, Table 4.3 presents the recovery rates across three sample sizes in the practical item bank. The recovery rates from *1bit(Switch_General)* and *1bit(Switch_Preferred)* were almost the same across three sample sizes. Similar observations were also presented in uniform and normal item banks (Tables D.20 and D.21 in Appendix). In addition, Table 4.4 presents the results for estimations of item discrimination parameters across sample sizes. While bias(a) did not always decrease with increasing sample sizes, RMSE(a) decreased with increasing sample sizes. In addition, the bias(a) and RMSE(a) of *1bit(Switch_General)* and *1bit(Switch_Preferred)* were also smaller than TW when N was larger in random NEAT and random PC designs. Lastly, Table 4.5 presents the average bias(b) and RMSE(b) across three sample sizes in practical item bank. Similar to the findings in estimation for discrimination parameters, RMSE(b) decreased with increasing sample sizes, though bias(b) did not always decrease. In addition, *1bit(Switch_General)* and *1bit(Switch_Preferred)* did not have extreme estimation in random NEAT designs when $N = 250$ and $N = 500$. Their bias(b) and RMSE(b) were larger than *1bit(p_tw)* and TW when N was larger than 125.

Summary. To summarize, switching method improved item discrimination parameters from *1bit(p_tw)* in all four incomplete calibration designs especially in the targeted NEAT, random NEAT, and random PC designs, where the overall bias(a) were smaller than the best baseline TW. On the other hand, they slightly sacrificed the accuracy of item difficulty parameter estimations except in random PC designs. Generally, their bias(b) and RMSE(b) were comparable to *1bit(p_tw)* and TW. In addition, the performance of *1bit(Switch_General)* and *1bit(Switch_Preferred)* were very similar, which indicated that the switching method with the set of general switching parameters is applicable in real applications where no prior simulation is required.

Results from *SMC(1+Switch_General)* and *SMC(1+Switch_Preferred)* compared to *SMC(1+p_tw)* and *TW*.

Recovery rates in the practical item bank with $N = 125$.

Table 4.3: Average Recovery Rates Across Sample Sizes in Practical Item Banks

Condition	Method	N = 125	N = 250	N = 500
Targeted NEAT	1bit(p_tw)	0.759	0.763	0.767
	1bit(Switch_General)	0.707	0.712	0.714
	1bit(Switch_Preferred)	0.706	0.711	0.714
	TW	0.666	0.668	0.670
Targeted PC	1bit(p_tw)	0.602	0.602	0.601
	1bit(Switch_General)	0.584	0.580	0.580
	1bit(Switch_Preferred)	0.583	0.579	0.578
	TW	0.592	0.592	0.592
Random NEAT	1bit(p_tw)	0.767	0.773	0.775
	1bit(Switch_General)	0.710	0.715	0.716
	1bit(Switch_Preferred)	0.708	0.710	0.712
	TW	0.731	0.735	0.736
Random PC	1bit(p_tw)	0.775	0.780	0.781
	1bit(Switch_General)	0.712	0.716	0.717
	1bit(Switch_Preferred)	0.714	0.719	0.718
	TW	0.730	0.732	0.733

Table 4.4: Item Discrimination Parameter Estimations Across Sample Sizes in Practical Item Banks

		N = 125		N = 250		N = 500	
Condition	Method	Bias(a)	RMSE(a)	Bias(a)	RMSE(a)	Bias(a)	RMSE(a)
Targeted NEAT	1bit(p_tw)	2.024	3.087	1.752	2.13	1.758	1.922
	1bit(Switch_General)	0.378	0.753	0.397	0.653	0.386	0.617
	1bit(Switch_Preferred)	0.238	0.653	0.265	0.576	0.257	0.511
	TW	0.515	0.723	0.560	0.664	0.502	0.585
Targeted PC	1bit(p_tw)	0.606	0.818	0.587	0.696	0.521	0.596
	1bit(Switch_General)	-0.027	0.455	-0.058	0.381	-0.078	0.356
	1bit(Switch_Preferred)	-0.110	0.544	-0.132	0.459	-0.164	0.421
	TW	0.111	0.413	0.135	0.333	0.110	0.288
Random NEAT	1bit(p_tw)	2.33	4.018	1.799	2.114	1.679	1.771
	1bit(Switch_General)	0.137	0.611	0.133	0.498	0.117	0.439
	1bit(Switch_Preferred)	0.121	0.582	0.120	0.460	0.123	0.416
	TW	0.477	0.741	0.450	0.554	0.414	0.484
Random PC	1bit(p_tw)	1.121	1.549	1.038	1.194	0.996	1.099
	1bit(Switch_General)	0.064	0.573	-0.012	0.428	-0.003	0.399
	1bit(Switch_Preferred)	0.115	0.564	0.094	0.437	0.091	0.400
	TW	0.318	0.581	0.293	0.419	0.279	0.360

Table 4.5: Item Difficulty Parameter Estimations Across Sample Sizes in Practical Item Banks

		N = 125		N = 250		N = 500	
Condition	Method	Bias(b)	RMSE(b)	Bias(b)	RMSE(b)	Bias(b)	RMSE(b)
Targeted NEAT	1bit(p_tw)	0.077	0.667	0.115	0.517	0.11	0.461
	1bit(Switch_General)	0.080	0.873	0.010	0.655	0.007	0.694
	1bit(Switch_Preferred)	0.074	1.364	0.012	0.809	-0.052	0.528
	TW	0.072	0.783	0.085	0.726	0.079	0.665
Targeted PC	1bit(p_tw)	0.059	0.824	0.051	0.758	0.027	0.720
	1bit(Switch_General)	0.020	0.891	0.023	0.831	0.018	0.742
	1bit(Switch_Preferred)	0.020	1.032	0.030	0.872	0.009	0.762
	TW	0.077	0.792	0.083	0.756	0.064	0.720
Random NEAT	1bit(p_tw)	0.091	2.221	0.030	0.508	0.035	0.319
	1bit(Switch_General)	9.313	68.228	0.018	0.807	-0.009	0.494
	1bit(Switch_Preferred)	0.512	4.593	0.037	0.920	-0.034	0.488
	TW	0.072	0.585	0.062	0.399	0.050	0.308
Random PC	1bit(p_tw)	-0.401	4.944	-0.031	0.470	-0.035	0.358
	1bit(Switch_General)	-0.186	3.661	-0.185	1.820	0.005	0.608
	1bit(Switch_Preferred)	0.009	1.447	-0.020	0.621	-0.019	0.454
	TW	0.044	0.544	0.056	0.352	0.039	0.269

Recovery rates per item of *SMC(1+Switch_General)* and *SMC(1+Switch_Preferred)* were slightly worse than *SMC(1+p_tw)* in targeted NEAT, random NEAT, and random PC designs, and their recovery rates were similar to *SMC(1+p_tw)* in targeted PC design (Figure 4.20). Both of *SMC(1+Switch_General)* and *SMC(1+Switch_Preferred)* had slightly higher recovery rates than *TW* in targeted NEAT designs, and similar recovery rates with *TW* in the other three incomplete calibration designs. Generally, both *SMC(1+Switch_General)* and *SMC(1+Switch_Preferred)* had similar recovery rates across the four incomplete calibration designs. This finding was consistent with *1bit(Switch_General)* and *1bit(Switch_Preferred)*.

Item discrimination parameter estimations in the practical item bank with $N = 125$.

The average bias(a) and RMSE(a) for item discrimination parameters were summarized in Table 4.6, and the bold numbers indicate that the switching method produced better results than *SMC(1+p_tw)*. Both *SMC(1+Switch_General)* and *SMC(1+Switch_Preferred)* produced better bias(a) and RMSE(a) than

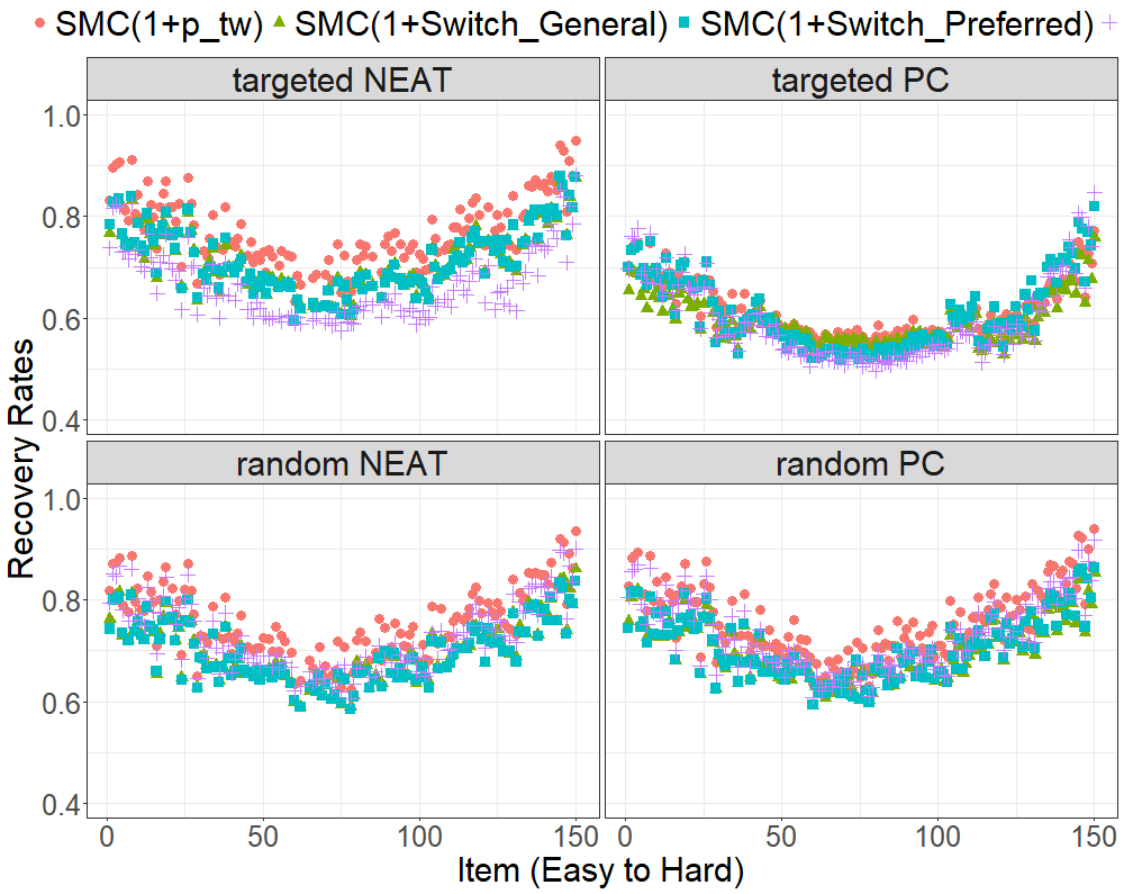


Figure 4.20: Recovery Rates for Switching Method with *SMC(1+p_tw)* and *TW* in Practical Item Bank and $N = 125$

SMC(1+p_tw) in all four incomplete calibration designs. In addition, their bias(a) and RMSE(a) were better than *TW* except in targeted PC designs.

To further evaluate their performance, their bias(a) per item were plotted in Figure 4.21. The y-axis was not fixed across the four incomplete calibration designs in this set of figures. The bias(a) from both *SMC(1+Switch_General)* and *SMC(1+Switch_Preferred)* were smaller than *SMC(1+p_tw)* and were closer to zero. Thus, both of them improved item discrimination parameter estimations. In addition, their bias(a) were slightly closer to zero than *TW* except in the targeted PC designs. They underestimated a more than *TW* in targeted PC designs. Across the four incomplete calibration designs, *SMC(1+Switch_Preferred)* did not show advantages over *SMC(1+Switch_General)*.

Table 4.6: Summary Statistics for Item Parameter Estimations for Switching Method with *SMC(1+p_tw)* and *TW* in Practical Item Bank and $N = 125$

Condition	Method	Bias(a)	RMSE(a)	Bias(b)	RMSE(b)
Targeted NEAT	1bit(p_tw)	1.857	2.917	0.062	1.155
	1bit(Switch_General)	**0.156**	**0.631**	-0.281	5.550
	1bit(Switch_Preferred)	**0.124**	**0.584**	**0.059**	1.922
	TW	0.515	0.723	0.072	0.783
Targeted PC	1bit(p_tw)	0.736	1.051	0.074	1.277
	1bit(Switch_General)	**0.003**	**0.481**	**0.037**	**0.991**
	1bit(Switch_Preferred)	**-0.174**	**0.488**	**-0.048**	**2.192**
	TW	0.111	0.413	0.077	0.792
Random NEAT	1bit(p_tw)	2.208	4.09	0.122	2.422
	1bit(Switch_General)	**0.094**	**0.592**	0.578	7.023
	1bit(Switch_Preferred)	**0.047**	**0.520**	**0.009**	**1.842**
	TW	0.477	0.741	0.072	0.585
Random PC	1bit(p_tw)	0.931	1.462	-0.065	2.457
	1bit(Switch_General)	**0.011**	**0.539**	0.120	3.137
	1bit(Switch_Preferred)	**0.037**	**0.512**	**0.018**	**1.909**
	TW	0.318	0.581	0.044	0.544

Note. The bold numbers indicated the better item parameter estimations from the switching method than *SMC(1+p_tw)*.

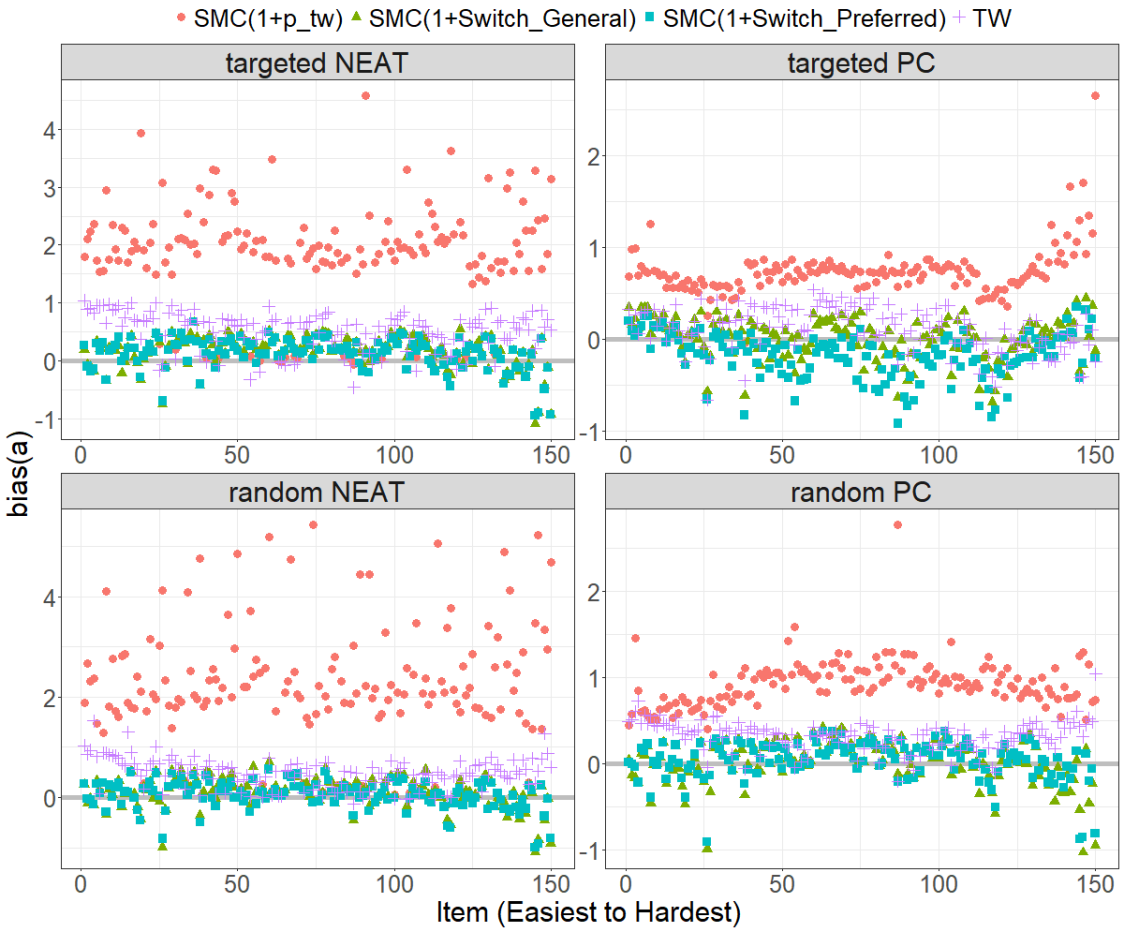


Figure 4.21: Bias(a) per Item for Switching Method with *SMC(1+p_tw)* and *TW* in Practical Item Bank and $N = 125$

Item difficulty parameter estimations in practical item bank with $N = 125$.

From Table 4.6, *SMC(1+Switch_General)* and *SMC(1+Switch_Preferred)* did not show much improvements than *SMC(1+p_tw)* in terms of bias(b) and RMSE(b). Their bias(b) and RMSE(b) were smaller than *SMC(1+p_tw)* in the targeted PC designs, though their RMSE(b) were still slightly larger than *TW*. In addition, *SMC(1+Switch_Preferred)* had better bias(b) and RMSE(b) than *SMC(1+p_tw)* in random NEAT and random PC designs. The reason that *SMC(1+Switch_General)* did not have better performance in random NEAT and random PC designs and had a larger RMSE(b) in targeted NEAT designs was due to a few extreme estimations. Figure 4.22 presents bias(b) per item without fixing the range of y-axis. From this set of figures, *SMC(1+Switch_Genera)* had a few extreme estimations for easy or difficult items in targeted NEAT, random NEAT, and random PC designs. In addition, *SMC(1+Switch_Preferred)* also had an extreme estimation for a very easy item in the targeted PC designs. Furthermore, Figure 4.23 presents the bias(b) per item of these four methods after fixing the y-axis in the range of -2 and 2. If ignoring the extreme estimations, both *SMC(1+Switch_General)* and *SMC(1+Switch_Preferred)* had bias(b) very close to zero except for easy and difficult items.

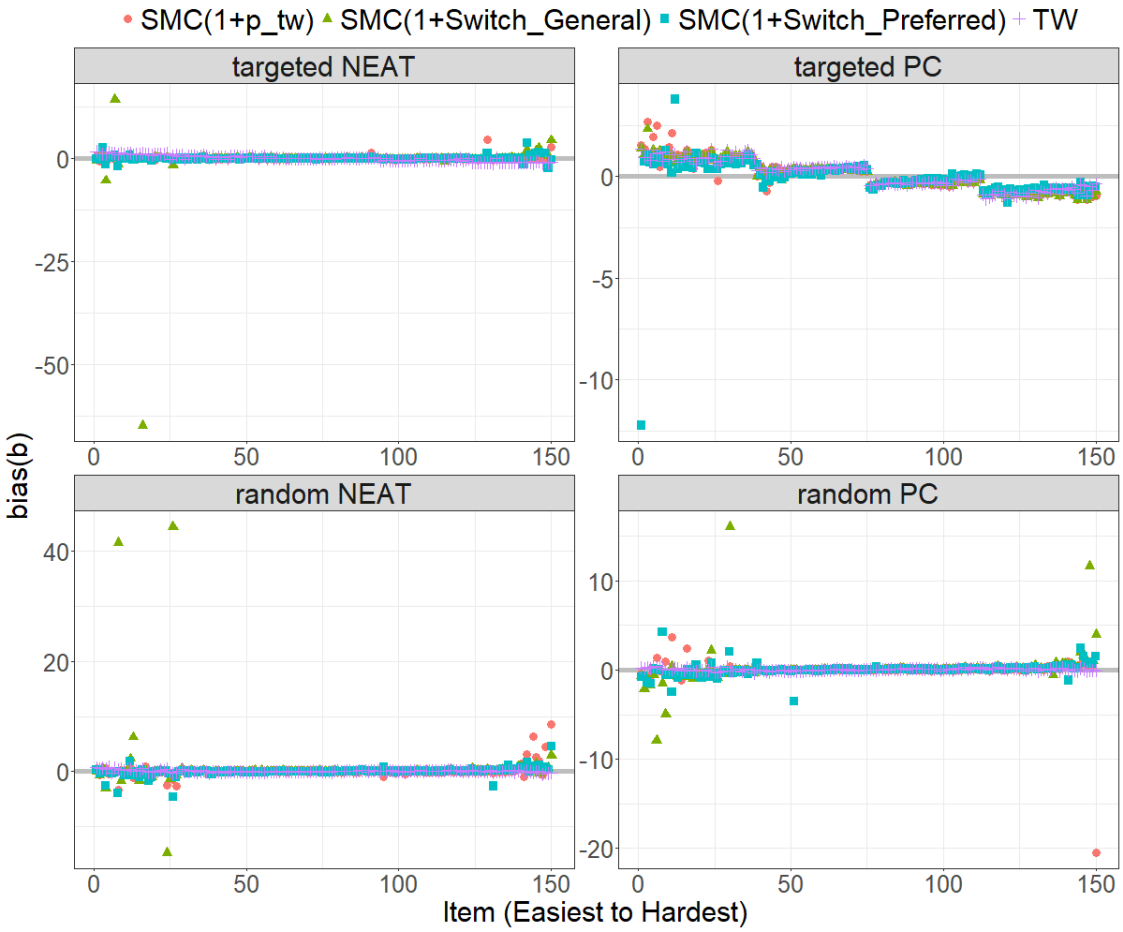


Figure 4.22: Bias(b) per Item for Switching Method with *SMC(1+p_tw)* and *TW* in Practical Item Bank and $N = 125$ (Free Y-Axis)

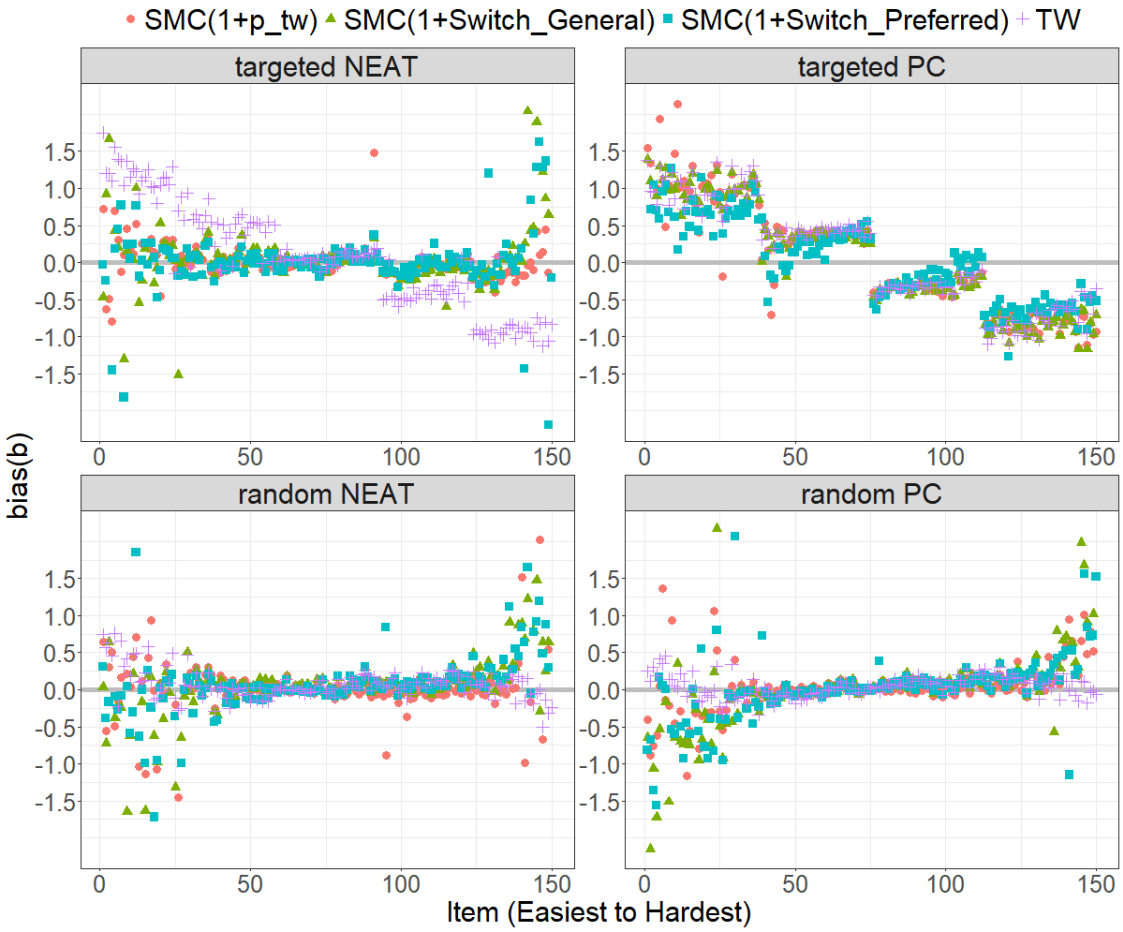


Figure 4.23: Bias(b) per Item for Switching Method with *SMC(1+p_tw)* and *TW* in Practical Item Bank and $N = 125$ (Zoomed in)

Influence of item banks on *SMC(1+Switch_General)* and *SMC(1+Switch_Preferred)*.

Item banks did not have much influence on *SMC(1+Switch_General)* and *SMC(1+Switch_Preferred)*. Firstly, their recovery rates were similar across item banks. Figures 4.24, 4.25, 4.26, and 4.27 present their recovery rates in four incomplete calibration designs respectively. From this set of figures, the recovery rates from *SMC(1+Switch_General)* and *SMC(1+Switch_Preferred)* were similar across item banks and their relations to *SMC(1+p_tw)* and *TW* were similar. Across three item banks, recovery rates of *SMC(1+Switch_General)* and *SMC(1+Switch_Preferred)* were similar to *TW* in targeted PC, random NEAT, and random PC designs, and were slightly better than *TW* in targeted NEAT design. In addi-

tion, their recovery rates were slightly lower than *SMC(1+p_tw)* in targeted NEAT, random NEAT, and random PC designs.

Secondly, the RMSE(a) and bias(a) of *SMC(1+Switch_General)* and *SMC(1+Switch_Preferred)* in the uniform and normal item banks were also similar to those in the practical item bank. From Figure 4.28, both *SMC(1+Switch_General)* and *SMC(1+Switch_Preferred)* had smaller RMSE(a) than *SMC(1+p_tw)* across three item banks. They also had smaller RMSE(a) than *TW* across three item banks in the targeted NEAT and random NEAT designs. In addition, the pattern of bias(a) per item of *SMC(1+Switch_General)* and *SMC(1+Switch_Preferred)* were similar across item banks (Figure 4.29).

Lastly, the RMSE(b) and bias(b) from both *SMC(1+Switch_General)* and *SMC(1+Switch_Preferred)* were similar across item banks as shown in Figures 4.30 and 4.31. Generally, *SMC(1+Switch_General)* and *SMC(1+Switch_Preferred)* had similar RMSE(b) across item banks, and they were similar or slightly larger than *SMC(1+p_tw)*.

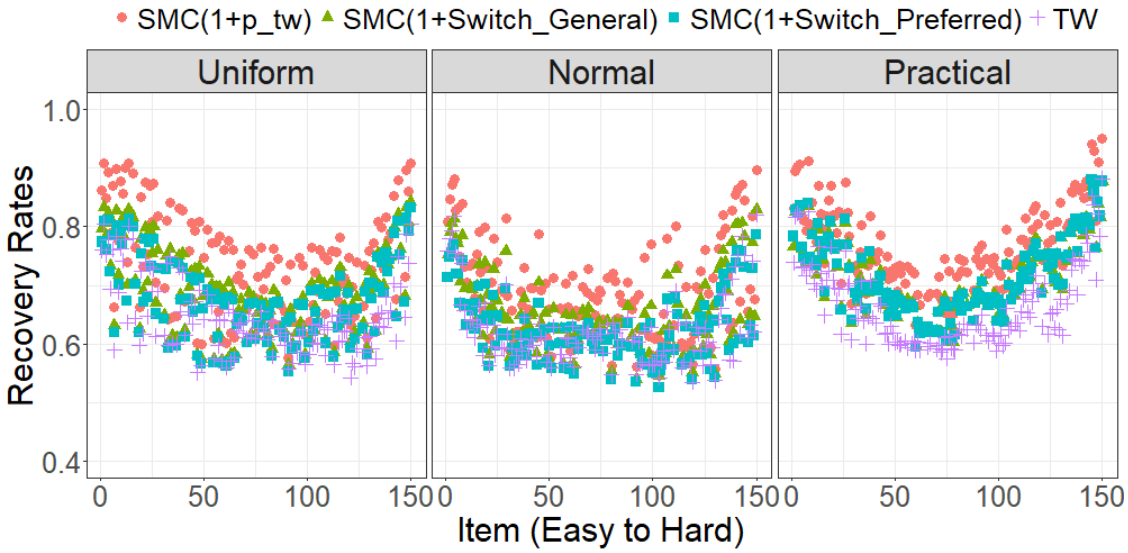


Figure 4.24: Recovery Rates per Item for Switching Method with *SMC(1+p_tw)* and *TW* Across Item Banks with $N = 125$ in Targeted NEAT Designs

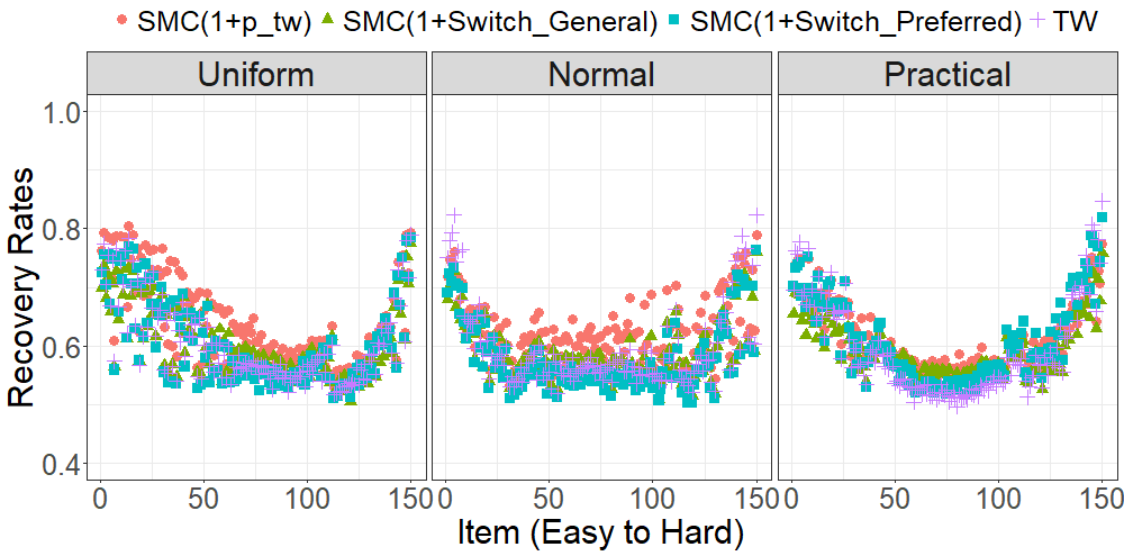


Figure 4.25: Recovery Rates per Item for Switching Method with *SMC(1+p_tw)* and *TW* Across Item Banks with $N = 125$ in Targeted PC Designs

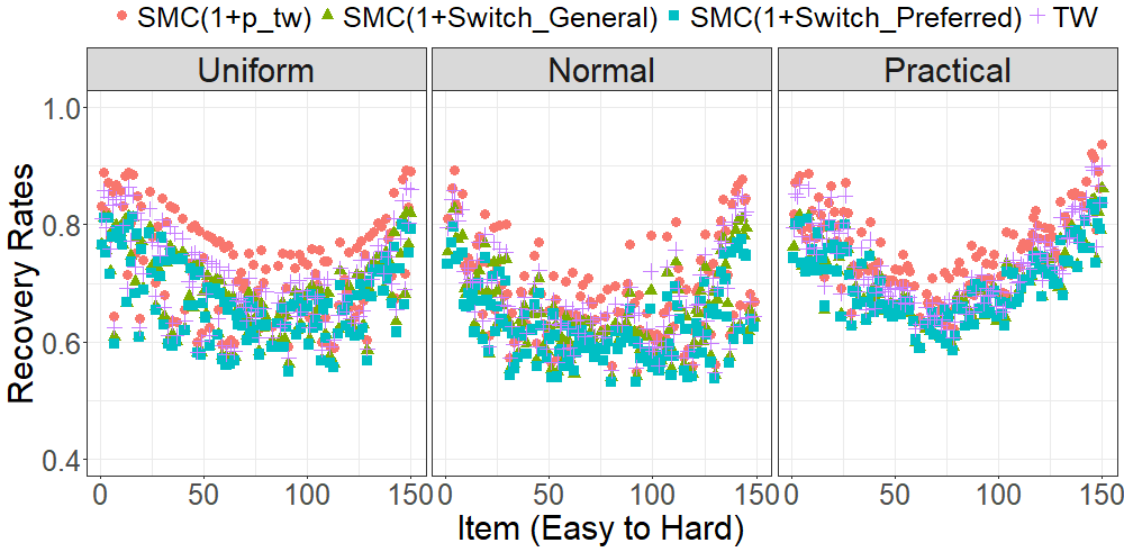


Figure 4.26: Recovery Rates per Item for Switching Method with *SMC(1+p_tw)* and *TW* Across Item Banks with $N = 125$ in Random NEAT Designs

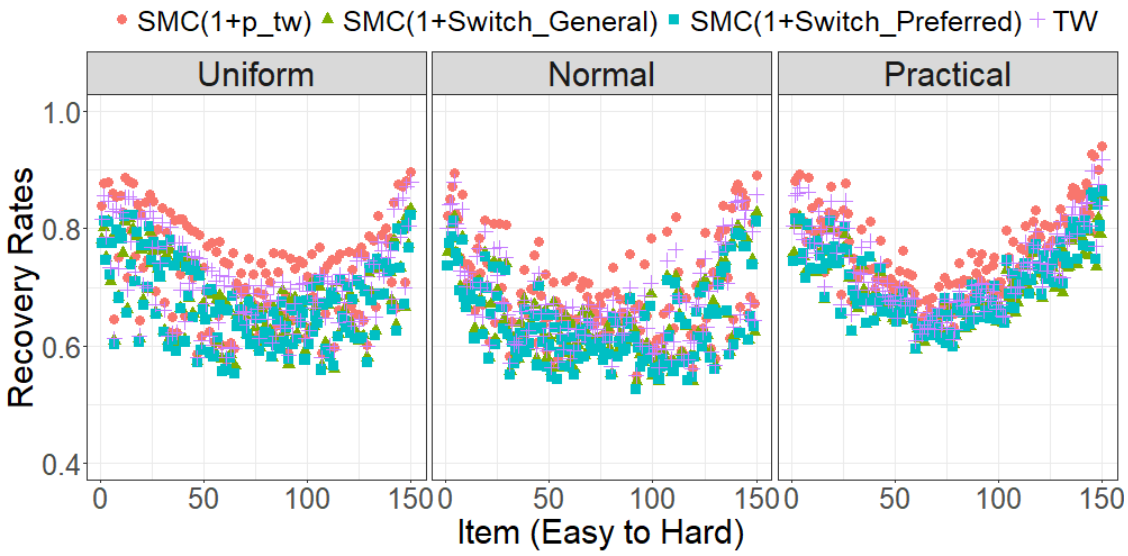


Figure 4.27: Recovery Rates per Item for Switching Method with *SMC(1+p_tw)* and *TW* Across Item Banks with $N = 125$ in Random PC Designs

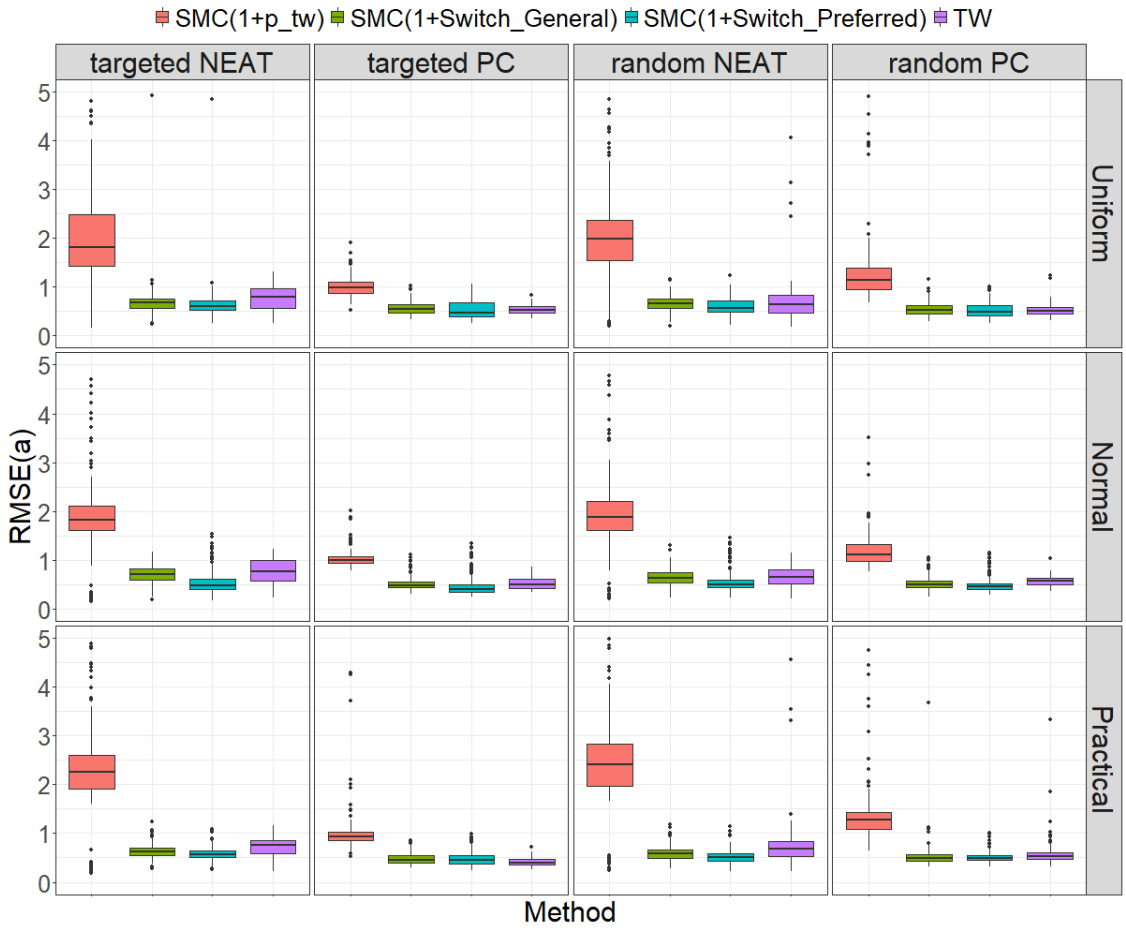


Figure 4.28: RMSE(a) for Switching Method with *SMC(1+p_tw)* and *TW* Across Item Banks with $N = 125$

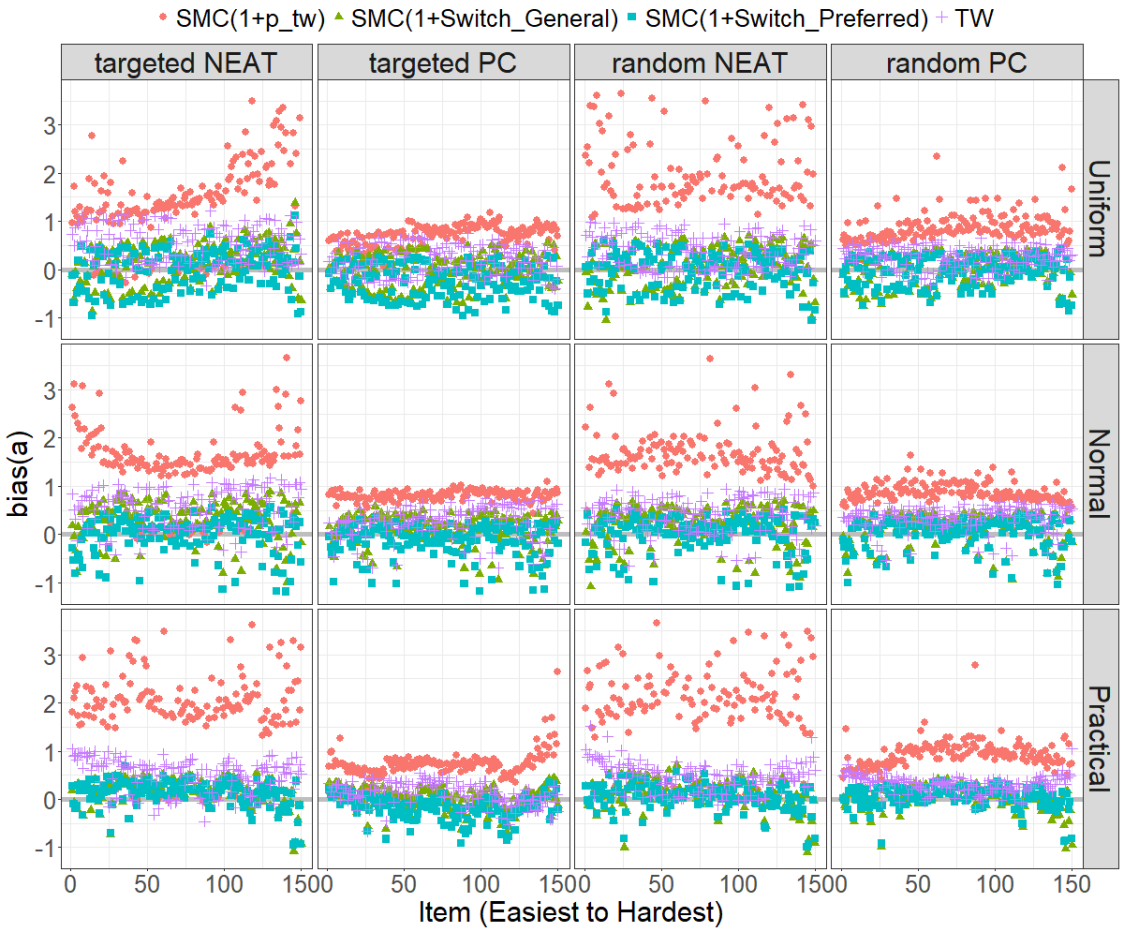


Figure 4.29: Bias(a) per item for Switching Method with *SMC(1+p_tw)* and *TW* Across Item Banks with $N = 125$

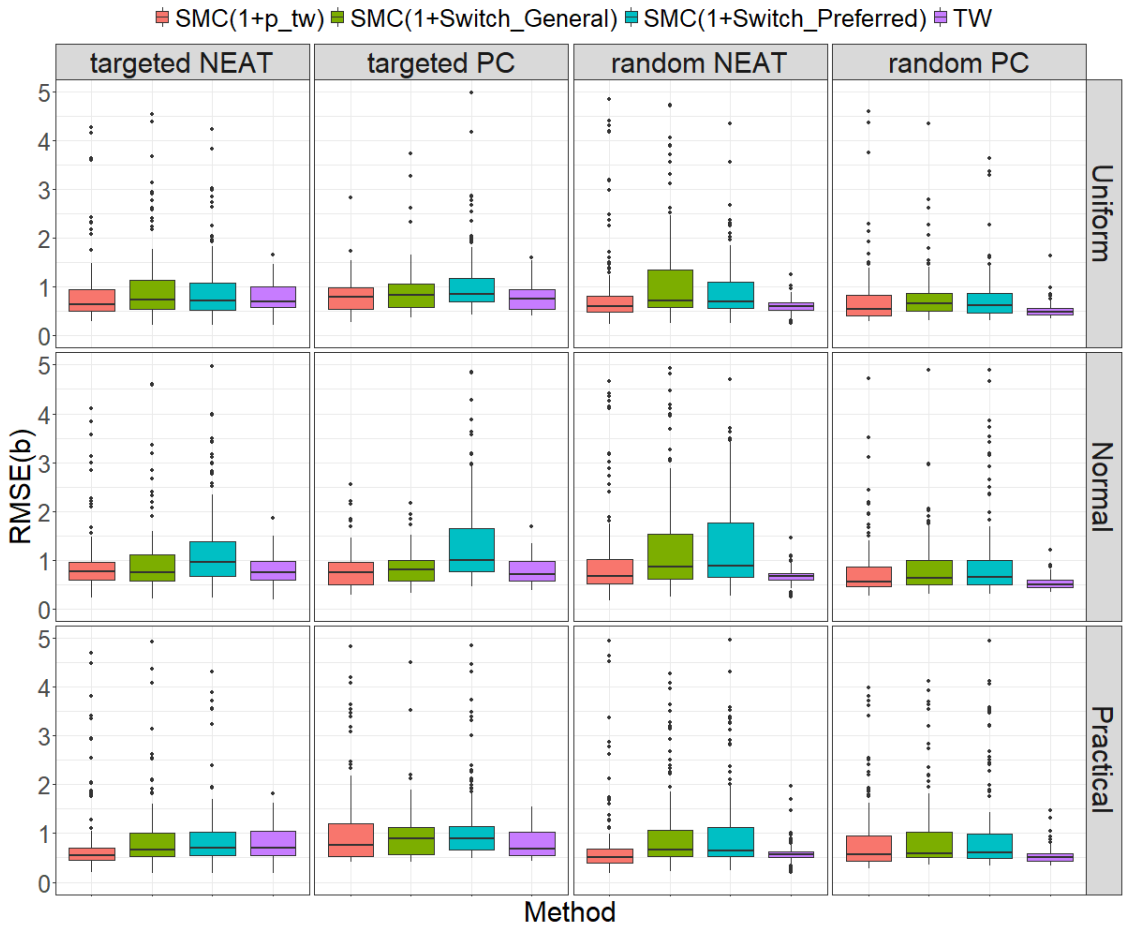


Figure 4.30: RMSE(b) for Switching Method with *SMC(1+p_tw)* and *TW* Across Item Banks with $N = 125$

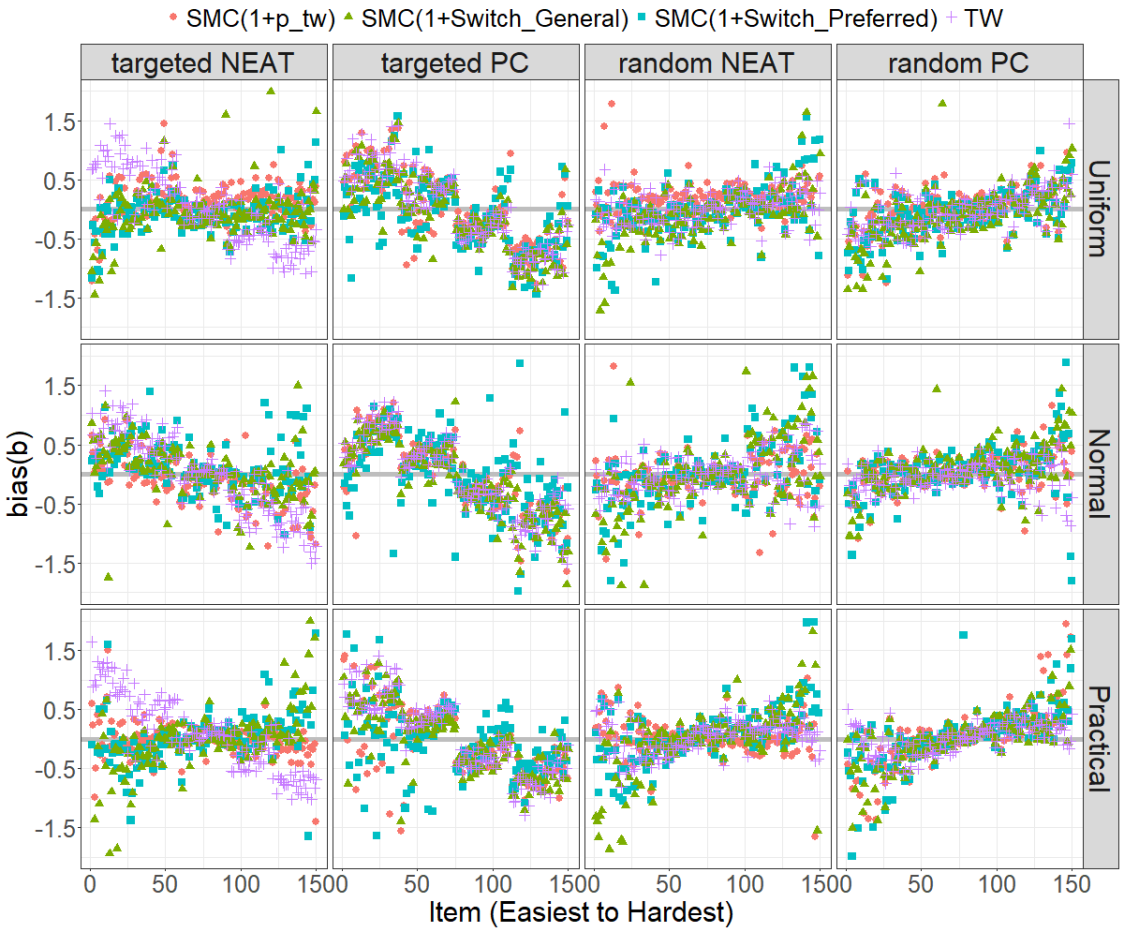


Figure 4.31: Bias(b) per item for Switching Method with *SMC(1+p_tw)* and *TW* Across Item Banks with $N = 125$

Influence of sample sizes on *SMC(1+Switch_General)* and *SMC(1+Switch_Preferred)*.

The sample size had a very small effect on *SMC(1+Switch_General)* and *SMC(1+Switch_Preferred)*. For example, the average recovery rates from *SMC(1+Switch_General)* and *SMC(1+Switch_Preferred)* were very similar across sample sizes in practical item bank (Table 4.7). Their recovery rates slightly increased as sample sizes increased except in the targeted PC designs. This pattern was also shown in the uniform and normal item banks (Tables D.22 and D.23 in the Appendix). In addition, their estimations for item discrimination parameters and item difficulty parameters were similar across sample sizes. Table 4.8 shows the bias(a) and RMSE(a) in the practical item bank across three sample sizes. While bias(a) did not decrease with the increasing sample sizes, RMSE(a) decreased with increasing sample sizes. In

addition, their RMSE(a) were smaller than *TW* across sample sizes except in the targeted PC designs. Finally, the estimations for item difficulty parameters were similar across sample sizes (Table 4.9). The RMSE(b) of *SMC(1+Switch_Preferred)* in the random NEAT design and $N = 250$ was large (6.185), which was due to an extreme estimation for an hard item.

Table 4.7: Average Recovery Rates Across Sample Sizes in Practical Item Bank

Condition	Method	N = 125	N = 250	N = 500
Targeted NEAT	SMC(1+p_tw)	0.773	0.782	0.785
	SMC(1+Switch_General)	0.717	0.725	0.727
	SMC(1+Switch_Preferred)	0.717	0.724	0.726
	TW	0.666	0.668	0.670
Targeted PC	SMC(1+p_tw)	0.615	0.610	0.605
	SMC(1+Switch_General)	0.592	0.582	0.579
	SMC(1+Switch_Preferred)	0.606	0.597	0.594
	TW	0.592	0.592	0.592
Random NEAT	SMC(1+p_tw)	0.759	0.765	0.767
	SMC(1+Switch_General)	0.703	0.709	0.711
	SMC(1+Switch_Preferred)	0.700	0.706	0.707
	TW	0.731	0.735	0.736
Random PC	SMC(1+p_tw)	0.758	0.766	0.771
	SMC(1+Switch_General)	0.701	0.705	0.707
	SMC(1+Switch_Preferred)	0.705	0.709	0.711
	TW	0.730	0.732	0.733

Summary. In summary, the switching method that applied to *SMC(1+p_tw)* showed similar results to that applied to *1bit(1+p_tw)*. This was expected as *SMC(1+p_tw)* and *1bit(p_tw)* had similar imputed response patterns to begin with. Specifically, switching method, both *Switch_General* and *Switch_Preferred*, applied to *SMC(1+p_tw)* improved item discrimination parameters in all four incomplete calibration designs especially in targeted and random NEAT designs. Their bias(a) and RMSE(a) were better than the best baseline of *TW*. On the other hand, they sacrificed the estimation for item difficulty parameter a little bit except for the random PC designs, but their estimations for item difficulty parameters were still comparable to *SMC(1+p_tw)* and *TW*. Same to the findings of *1bit(Switch_General)* and *1bit(Switch_Preferred)*, *SMC(1+Switch_Preferred)* did not show advantage over *SMC(1+Switch_General)*.

Table 4.8: Summary of Item Discrimination Parameter Estimation Across Sample Sizes in Practical Item Bank

		N = 125		N = 250		N = 500	
Condition	Method	Bias(a)	RMSE(a)	Bias(a)	RMSE(a)	Bias(a)	RMSE(a)
Targeted NEAT	SMC(1+p_tw)	1.857	2.917	1.645	1.915	1.608	1.741
	SMC(1+Switch_General)	0.156	0.631	0.199	0.517	0.178	0.470
	SMC(1+Switch_Preferred)	0.124	0.584	0.148	0.482	0.147	0.444
	TW	0.515	0.723	0.560	0.664	0.502	0.585
Targeted PC	SMC(1+p_tw)	0.736	1.051	0.759	0.887	0.706	0.770
	SMC(1+Switch_General)	0.003	0.481	0.022	0.421	-0.010	0.372
	SMC(1+Switch_Preferred)	-0.174	0.488	-0.142	0.418	-0.147	0.392
	TW	0.111	0.413	0.135	0.333	0.110	0.288
Random NEAT	SMC(1+p_tw)	2.208	4.090	1.549	1.876	1.420	1.577
	SMC(1+Switch_General)	0.094	0.592	0.046	0.464	0.043	0.413
	SMC(1+Switch_Preferred)	0.047	0.520	0.084	0.437	0.064	0.385
	TW	0.477	0.741	0.450	0.554	0.414	0.484
Random PC	SMC(1+p_tw)	0.931	1.462	0.865	1.057	0.822	0.928
	SMC(1+Switch_General)	0.011	0.539	-0.032	0.392	-0.033	0.342
	SMC(1+Switch_Preferred)	0.037	0.512	0.031	0.386	0.035	0.335
	TW	0.318	0.581	0.293	0.419	0.279	0.360

Table 4.9: Summary of Item Difficulty Parameter Estimation Across Sample Sizes in Practical Item Bank

		N = 125		N = 250		N = 500	
Condition	Method	Bias(b)	RMSE(b)	Bias(b)	RMSE(b)	Bias(b)	RMSE(b)
Targeted NEAT	SMC(1+p_tw)	0.062	1.155	0.044	0.405	0.063	0.327
	SMC(1+Switch_General)	-0.281	5.55	0.122	1.429	0.199	2.066
	SMC(1+Switch_Preferred)	0.059	1.922	0.024	0.766	-0.128	1.149
	TW	0.072	0.783	0.085	0.726	0.079	0.665
Targeted PC	SMC(1+p_tw)	0.074	1.277	0.101	0.855	0.087	0.793
	SMC(1+Switch_General)	0.037	0.991	0.046	0.855	0.053	0.771
	SMC(1+Switch_Preferred)	-0.048	2.192	0.026	0.785	0.034	0.694
	TW	0.077	0.792	0.083	0.756	0.064	0.720
Random NEAT	SMC(1+p_tw)	0.122	2.422	0.018	0.552	0.022	0.327
	SMC(1+Switch_General)	0.578	7.023	-0.016	1.415	-0.059	1.388
	SMC(1+Switch_Preferred)	0.009	1.842	-0.772	6.185	-0.003	0.508
	TW	0.072	0.585	0.062	0.399	0.050	0.308
Random PC	SMC(1+p_tw)	-0.065	2.457	-0.011	0.499	-0.025	0.369
	SMC(1+Switch_General)	0.120	3.137	0.046	0.835	-0.004	0.507
	SMC(1+Switch_Preferred)	0.018	1.909	-0.023	0.686	-0.005	0.416
	TW	0.044	0.544	0.056	0.352	0.039	0.269

4.3.2 Results from the softened threshold method.

In this section, the results from *SMC(1+Soft)* and *SMC(p1+Soft)* were compared to *SMC(1+p_tw)* and *SMC(p1+p_tw)*, which were their corresponding methods in study 1, and *TW*, the best baseline in study 1. Their recovery rates, estimations for item discrimination parameters and item difficulty parameters in practical item bank with $N = 125$ were first presented, and the effects from item banks and sample sizes were then presented.

Recovery rates in the practical item bank with $N = 125$.

The recovery rate per item is presented in Figure 4.32. The recovery rates from *SMC(1+Soft)* and *SMC(p1+Soft)* were relatively similar to *SMC(1+p_tw)* and *SMC(p1+p_tw)* and varied a little across incomplete calibration designs. In the targeted NEAT design, both *SMC(1+Soft)* and *SMC(p1+Soft)* had recovery rates slightly lower than their corresponding methods in study 1 but slightly higher than *TW*. In targeted PC designs, *SMC(p1+p_tw)* and *SMC(p1+Soft)* had similar recovery rates and slightly higher than the other methods, especially for the easy and hard items. In random NEAT and random PC designs, all five methods had similar recovery rates.

Estimation for item discrimination parameters in the practical item bank with $N = 125$.

The average bias(a) and RMSE(a) were summarized in Table 4.10. The bold number indicated that the softened threshold method had better results than its corresponding method in study 1. From this table, *SMC(p1+Soft)* had smaller bias(a) and RMSE(a) than *SMC(p1+p_tw)* in all four incomplete calibration designs. *SMC(1+Soft)* had slightly smaller bias(a) and similar RMSE(a) than *SMC(1+p_tw)* in targeted PC, random NEAT, and random PC designs. However, neither *SMC(1+Soft)* nor *SMC(p1+Soft)* had better bias(a) and RMSE(a) than *TW*. In addition, Figure 4.33 presents the bias(a) per item in practical item bank with $N = 125$. From this set of figures, *SMC(1+Soft)* and *SMC(p1+Soft)* had similar bias(a) to their corresponding methods, and all four of them were larger than *TW* especially in targeted NEAT and random NEAT designs.

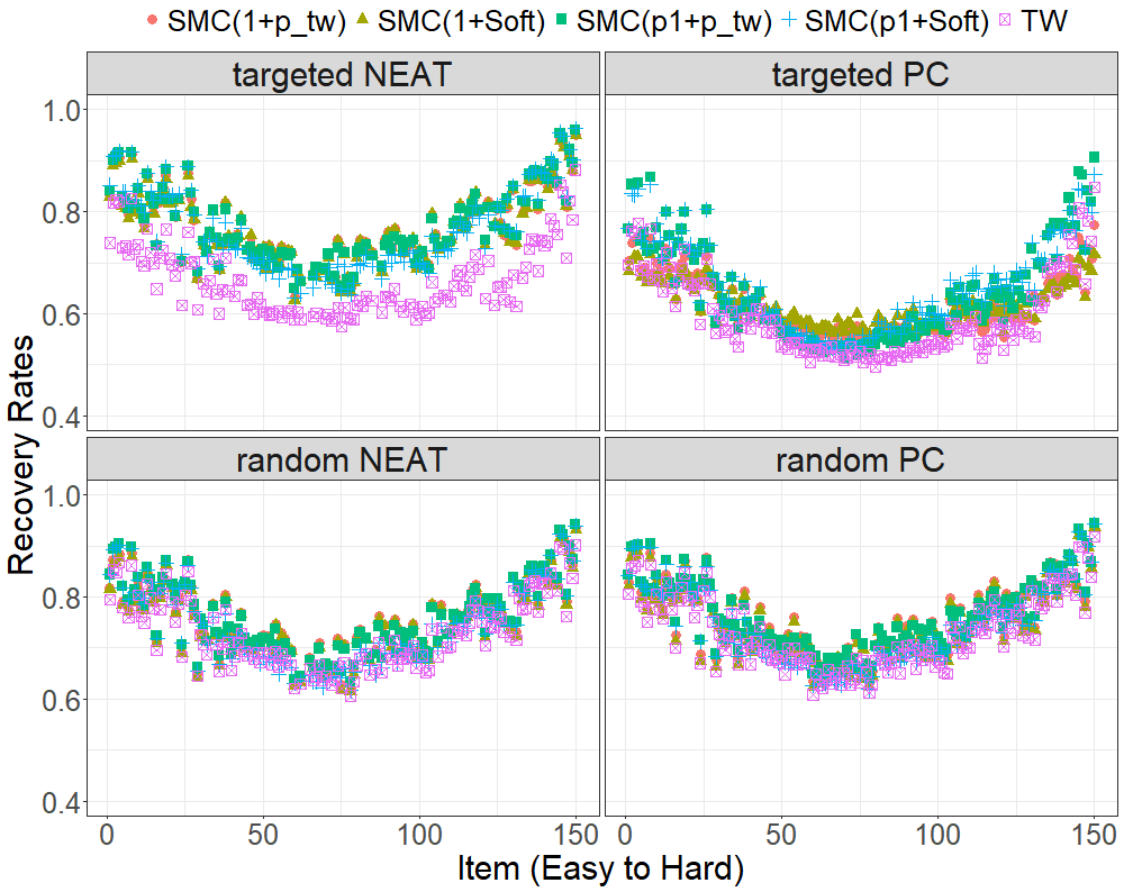


Figure 4.32: Recovery Rates of the Softened Threshold Methods in Practical Item Bank with $N = 125$.

Table 4.10: Summary Statistics for Item Parameter Estimations for Softened Threshold Method with *SMC(1+p_tw)*, *SMC(p1+p_tw)* and *TW* in Practical Item Bank with $N = 125$.

Condition	Method	Bias(a)	RMSE(a)	Bias(b)	RMSE(b)
Targeted NEAT	SMC(1+p_tw)	1.857	2.917	0.062	1.155
	SMC(1+Soft)	2.286	5.962	0.073	1.677
	SMC(p1+p_tw)	1.828	2.65	0.122	1.047
	SMC(p1+Soft)	**1.603**	**2.429**	**-0.002**	1.624
	TW	0.515	0.723	0.072	0.783
Targeted PC	SMC(1+p_tw)	0.736	1.051	0.074	1.277
	SMC(1+Soft)	**0.633**	1.372	0.174	**1.149**
	SMC(p1+p_tw)	0.331	0.651	0.951	7.391
	SMC(p1+Soft)	**0.213**	**0.609**	4.082	30.030
	TW	0.111	0.413	0.077	0.792
Random NEAT	SMC(1+p_tw)	2.208	4.090	0.122	2.422
	SMC(1+Soft)	**2.204**	4.370	**-0.025**	3.566
	SMC(p1+p_tw)	1.718	2.740	0.534	5.686
	SMC(p1+Soft)	**1.541**	**2.505**	6.223	51.762
	TW	0.477	0.741	0.072	0.585
Random PC	SMC(1+p_tw)	0.931	1.462	-0.065	2.457
	SMC(1+Soft)	**0.839**	**1.425**	-0.079	3.336
	SMC(p1+p_tw)	0.608	1.082	-0.673	9.178
	SMC(p1+Soft)	**0.448**	**1.004**	-2.587	29.141
	TW	0.318	0.581	0.044	0.544

Note. The bold numbers indicated better item parameter estimations from switching method than its corresponding method in Study 1.

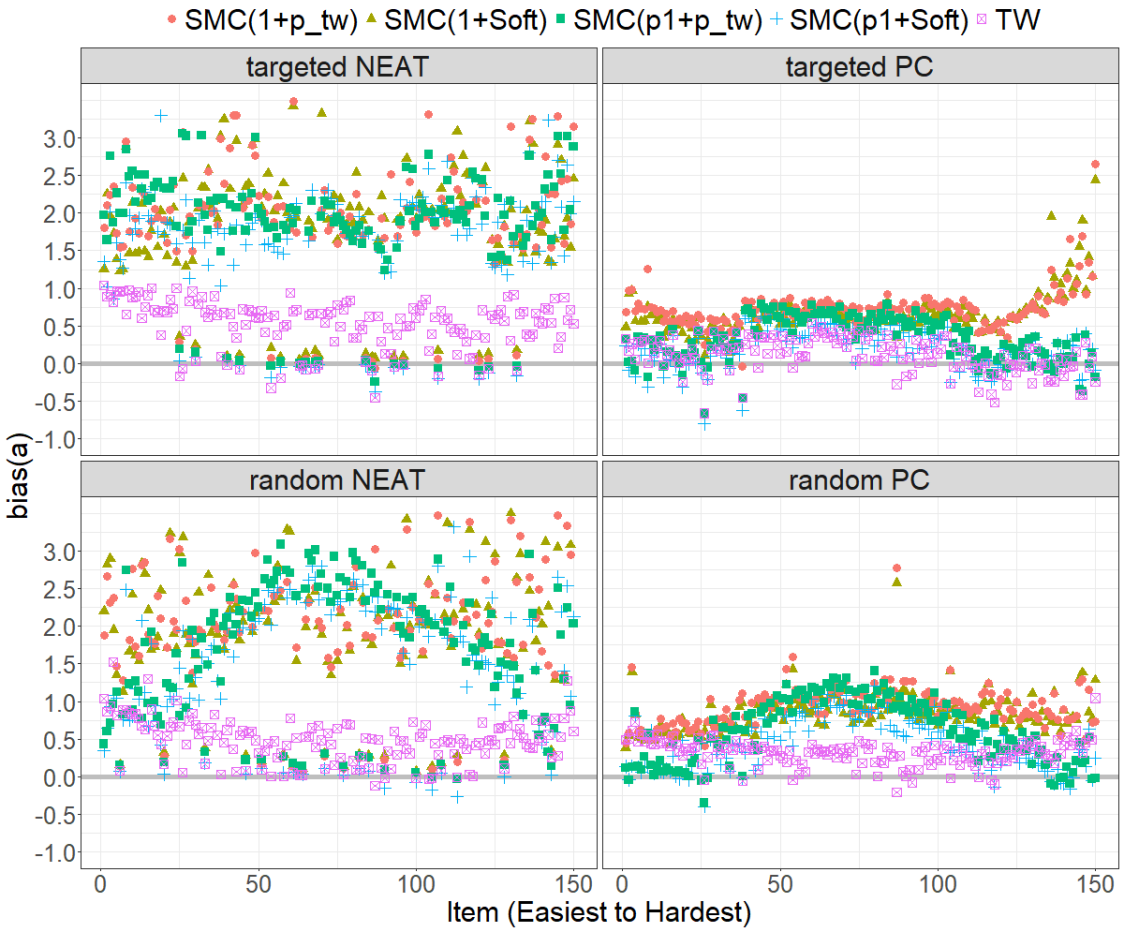


Figure 4.33: Bias(a) per Item of the Softened Threshold Method in Practical Item Bank with $N = 125$.

Estimation for item difficulty parameters in the practical item bank with $N = 125$.

From Table 4.10, *SMC(1+Soft)* and *SMC(p1+Soft)* barely had better bias(b) nor RMSE(b) than their corresponding methods in study 1 for item difficulty parameters. As shown in Figure 4.34 where the y-axis was not fixed, *SMC(p1+Soft)* had a few very extreme estimations for the easy items in targeted PC, random NEAT, and random PC designs. This explained that *SMC(p1+Soft)* had very large bias(b) and RMSE(b) in targeted PC, random NEAT, and random PC designs in Table 4.10. In addition, Figure 4.35 presents bias(b) per item when fixed the y-axis in the range of -2 to 2. After excluding extreme estimations, *SMC(1+Soft)* and *SMC(p1+Soft)* were better than TW in the targeted NEAT design especially for easy and hard items. *SMC(1+Soft)* and its corresponding method, *SMC(1+p_tw)*, had bias(b) closer to zero than TW in the random PC design. In addition, *SMC(p1+Soft)* and its corresponding method, *SMC(p1+p_tw)*, had bias(b) closer to zero than TW in the targeted PC designs but with larger variations for the easy items. Neither *SMC(1+Soft)* or *SMC(p1+Soft)* had better bias(b) than TW in the random NEAT designs. To summarize, the softened threshold method kept the features of *p_tw* and produced a few more extreme estimations.

Influence from item banks.

The results of *SMC(1+Soft)* and *SMC(p1+Soft)* from the uniform and normal item banks were similar to the findings from the practical item bank. Firstly, the recovery rates were similar across item banks. Figures 4.36, 4.37, 4.38, and 4.39 present the recovery rates per item across three item banks in four incomplete calibration designs respectively. Generally, the recovery rates of all methods across item banks were similar, while recovery rates of the practical item banks were slightly higher than the other two item banks in targeted NEAT, random NEAT, and random PC designs.

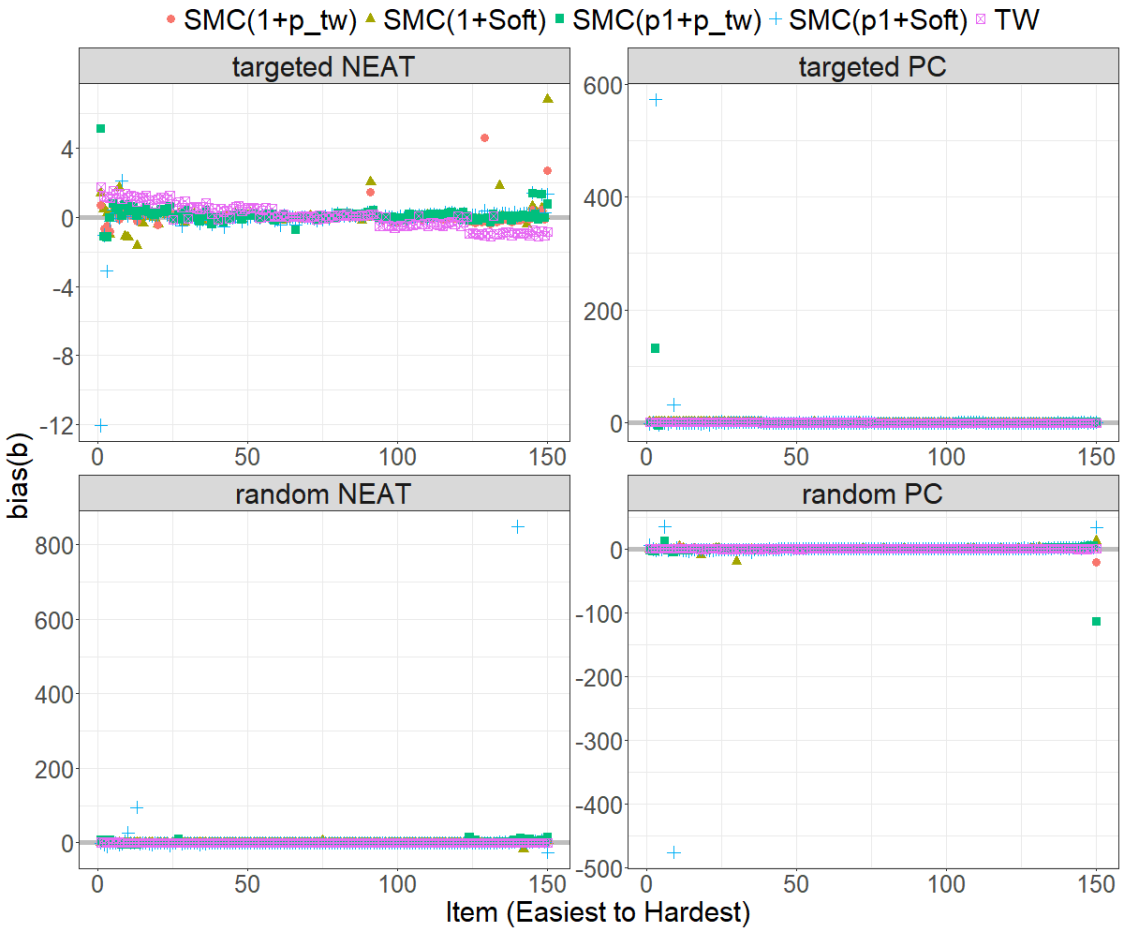


Figure 4.34: Bias(b) per Item for Softened Threshold Methods in Practical Item Bank with $N = 125$ (Free Y-Axis).

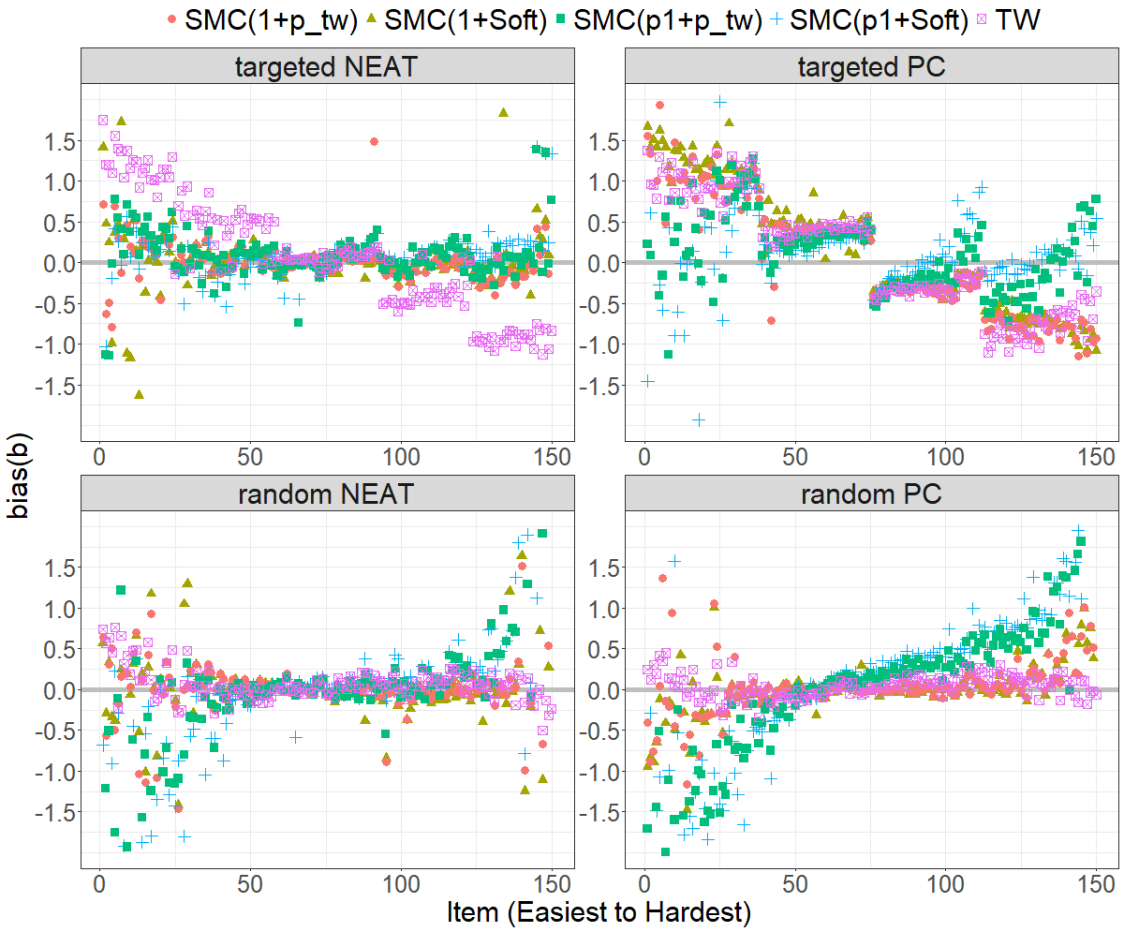


Figure 4.35: Bias(b) per Item for Softened Threshold Methods in Practical Item Bank with $N = 125$ (Zoomed In).

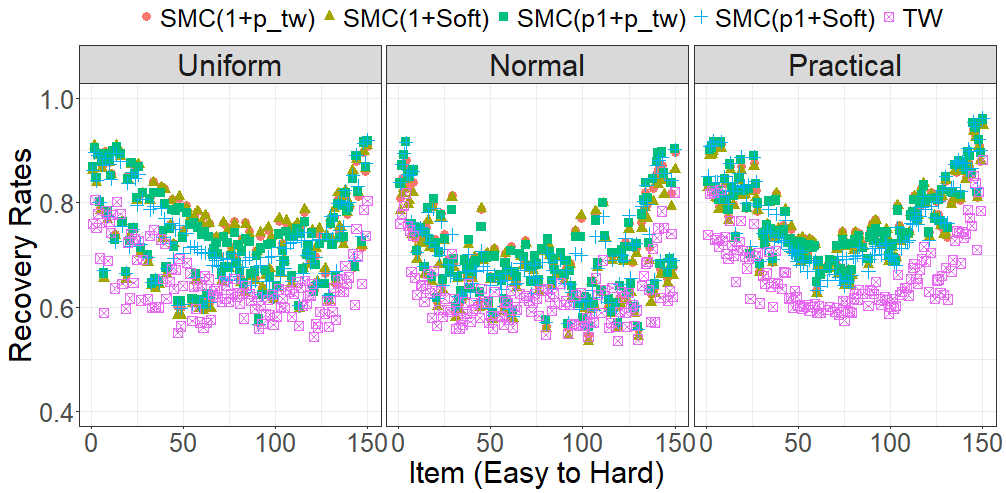


Figure 4.36: Recovery Rates per Item for the Softened Threshold Methods and *TW* Across Item Banks with $N = 125$ In Targeted NEAT Designs

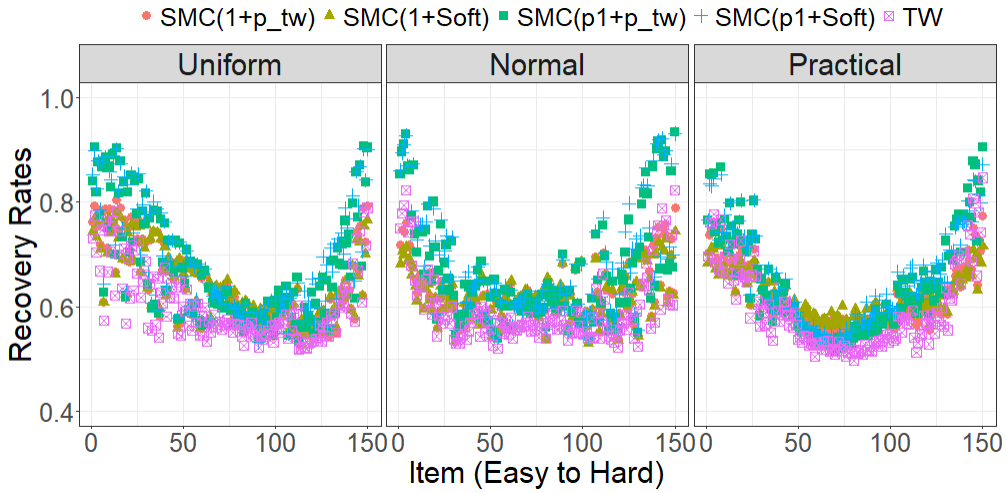


Figure 4.37: Recovery Rates per Item for Switching Method with *1bit(p_tw)* and *TW* Across Item Banks with $N = 125$ in Targeted PC Designs

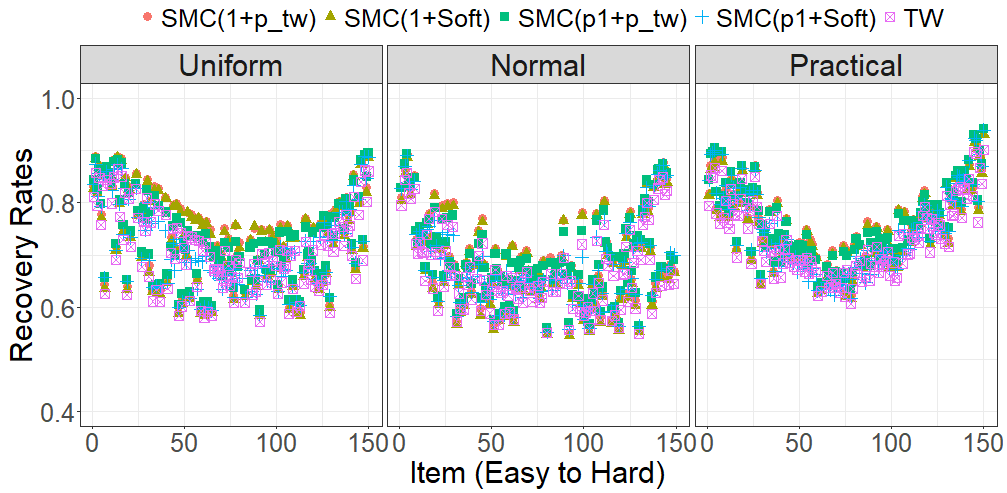


Figure 4.38: Recovery Rates per Item for Switching Method with *1bit(p_tw)* and TW Across Item Banks with $N = 125$ in Random NEAT Designs

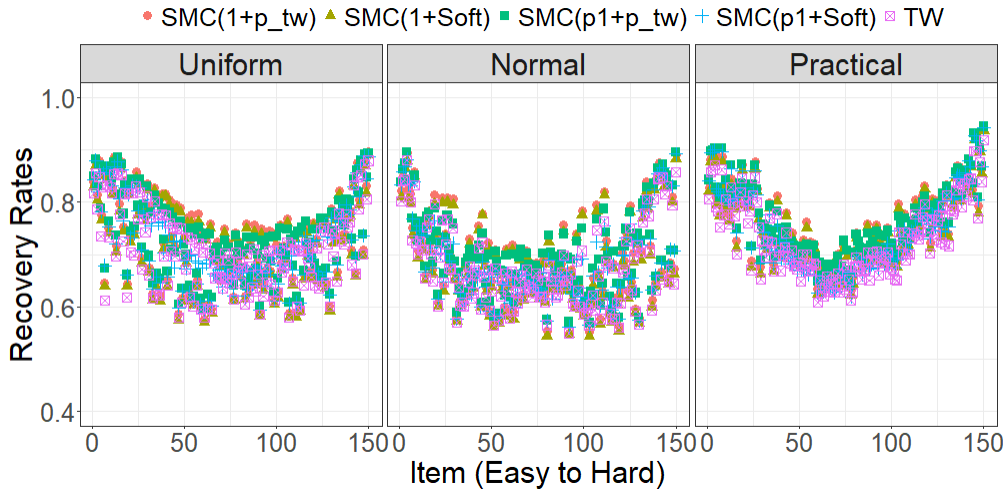


Figure 4.39: Recovery Rates per Item for Switching Method with *1bit(p_tw)* and TW Across Item Banks with $N = 125$ in Random PC Designs

Secondly, the results of both *SMC(1+Soft)* and *SMC(p1+Soft)* for item discrimination parameters were similar across item banks. For example, Figure 4.40 presents the RMSE(a) across item banks when $N = 125$. Both *SMC(1+Soft)* and *SMC(p1+Soft)* had similar RMSE(a) with *SMC(1+p_tw)* and *SMC(p1+p_tw)* across item banks and incomplete calibration designs, and thus their RMSE(a) were larger than TW. Their bias(a) were also larger than TW across item banks and incomplete calibration designs (Figure 4.41).

Thirdly, the results of RMSE(b) and bias(b) from *SMC(1+Soft)* and *SMC(p1+Soft)* were also similar across item banks (Figures 4.42 and 4.43). Generally, both *SMC(1+Soft)* and *SMC(p1+Soft)* had a few more extreme estimation for item difficulty parameters than *TW*, similar to their corresponding methods in study 1.

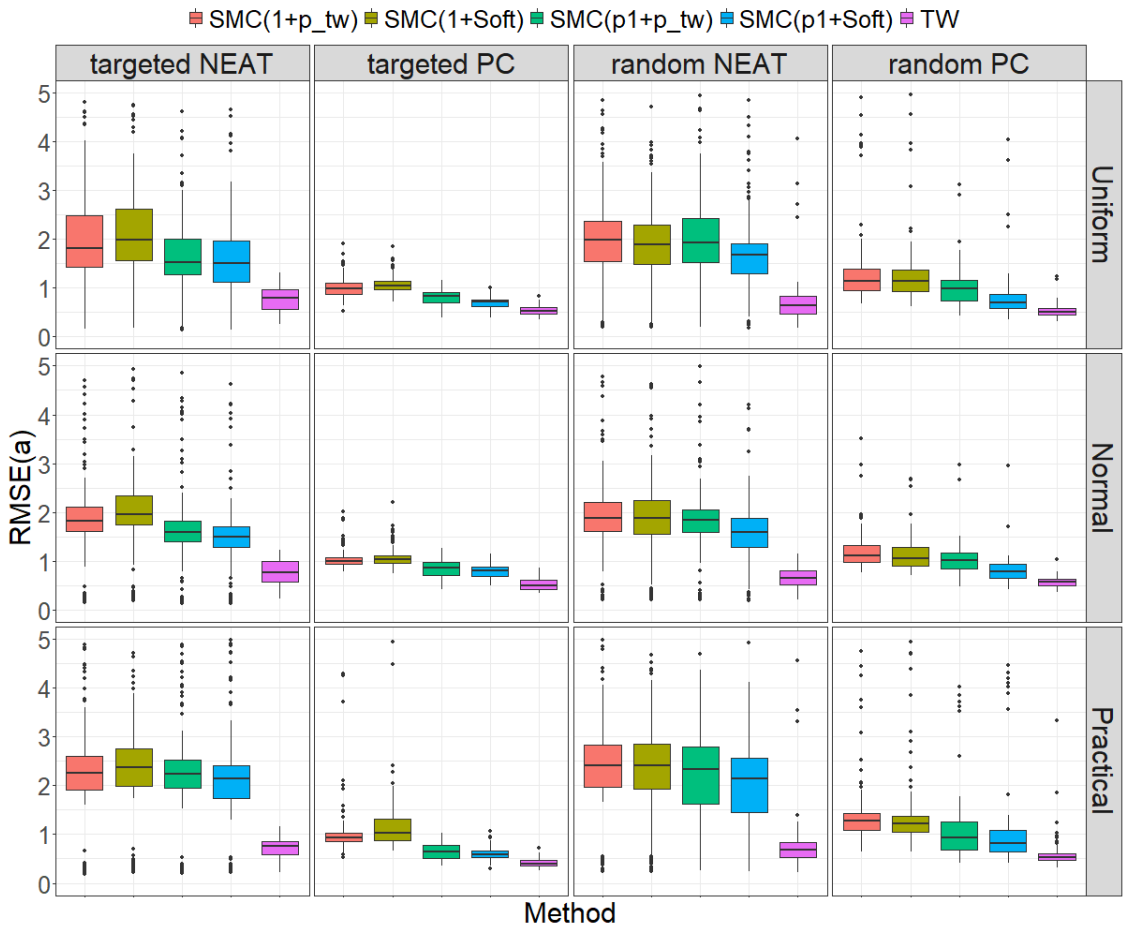


Figure 4.40: RMSE(a) for the Softened Threshold Methods and *TW* Across Item Banks with $N = 125$

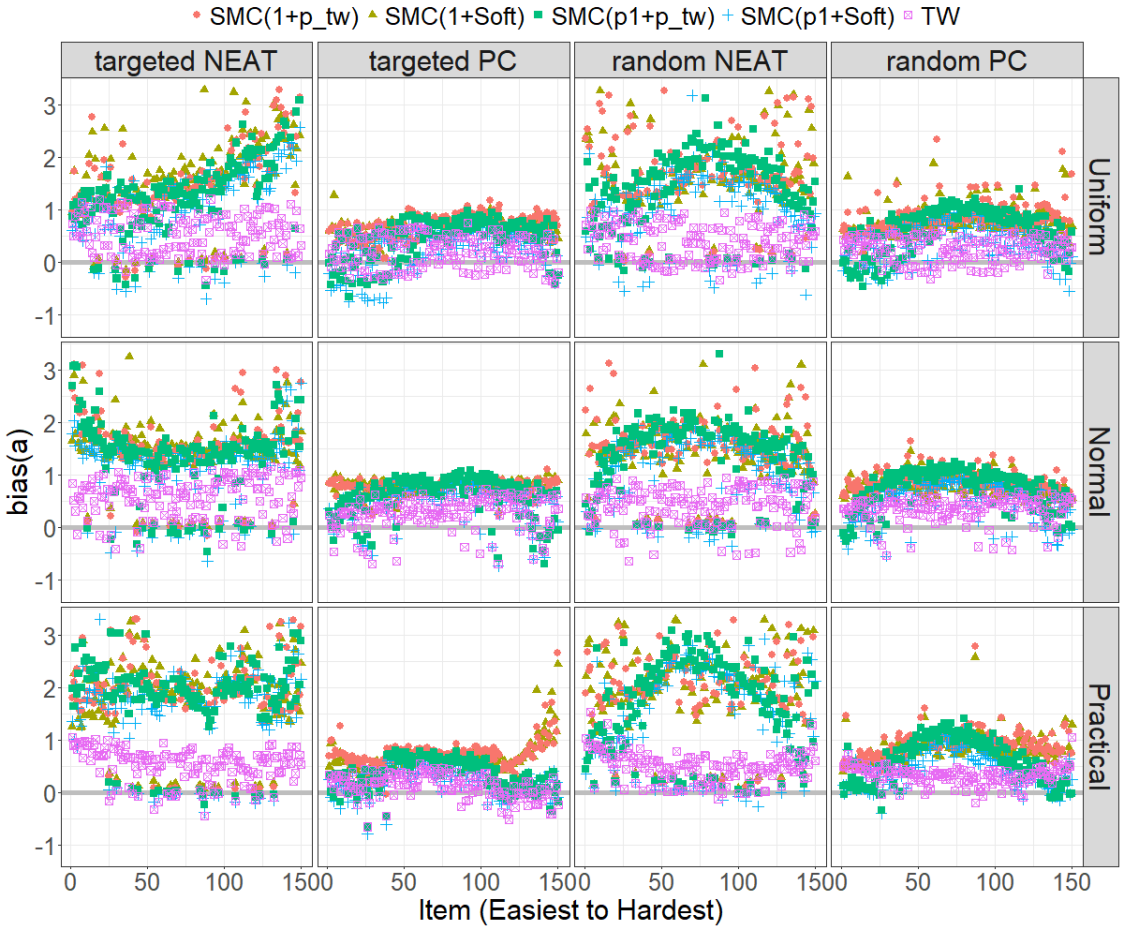


Figure 4.41: Bias(a) per item for the Softened Threshold Methods and *TW* Across Item Banks with $N = 125$

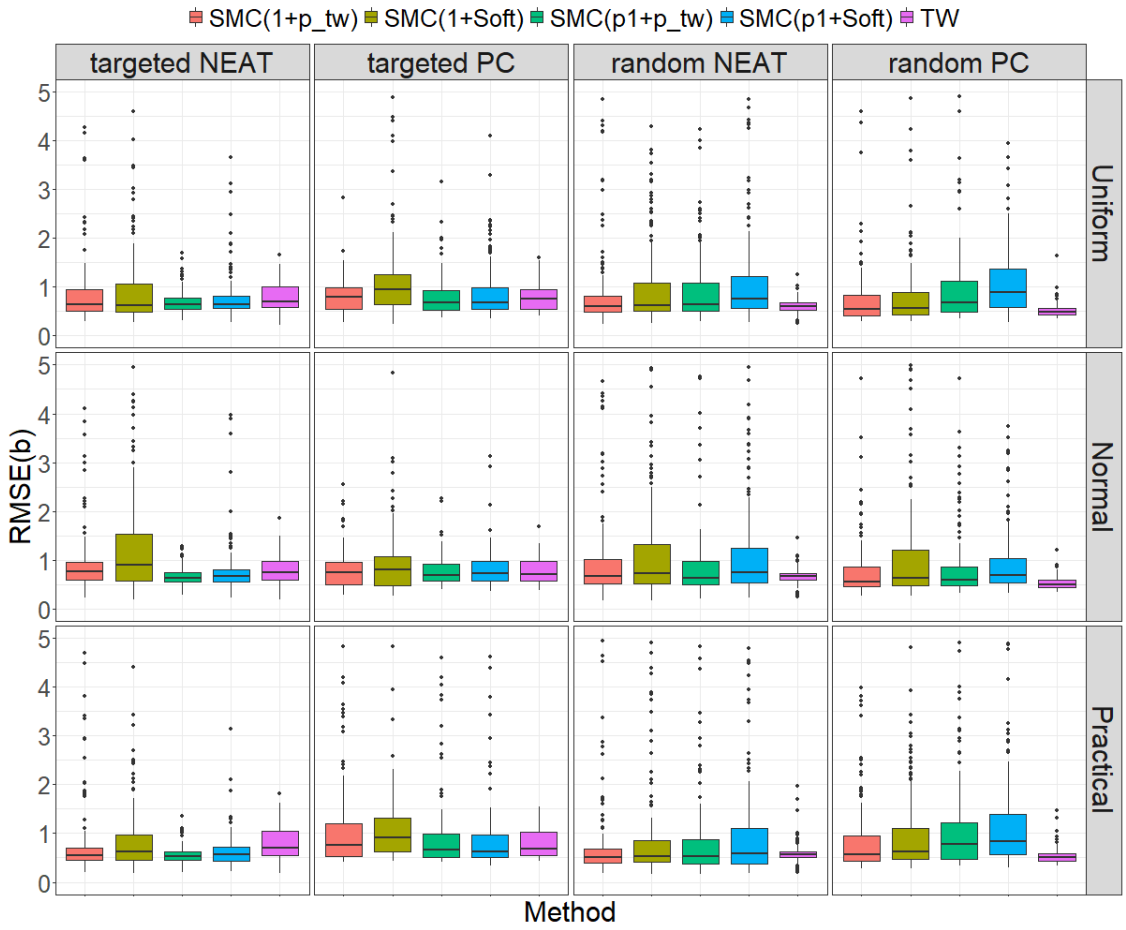


Figure 4.42: RMSE(b) for the Softened Threshold Methods and *TW* Across Item Banks with $N = 125$

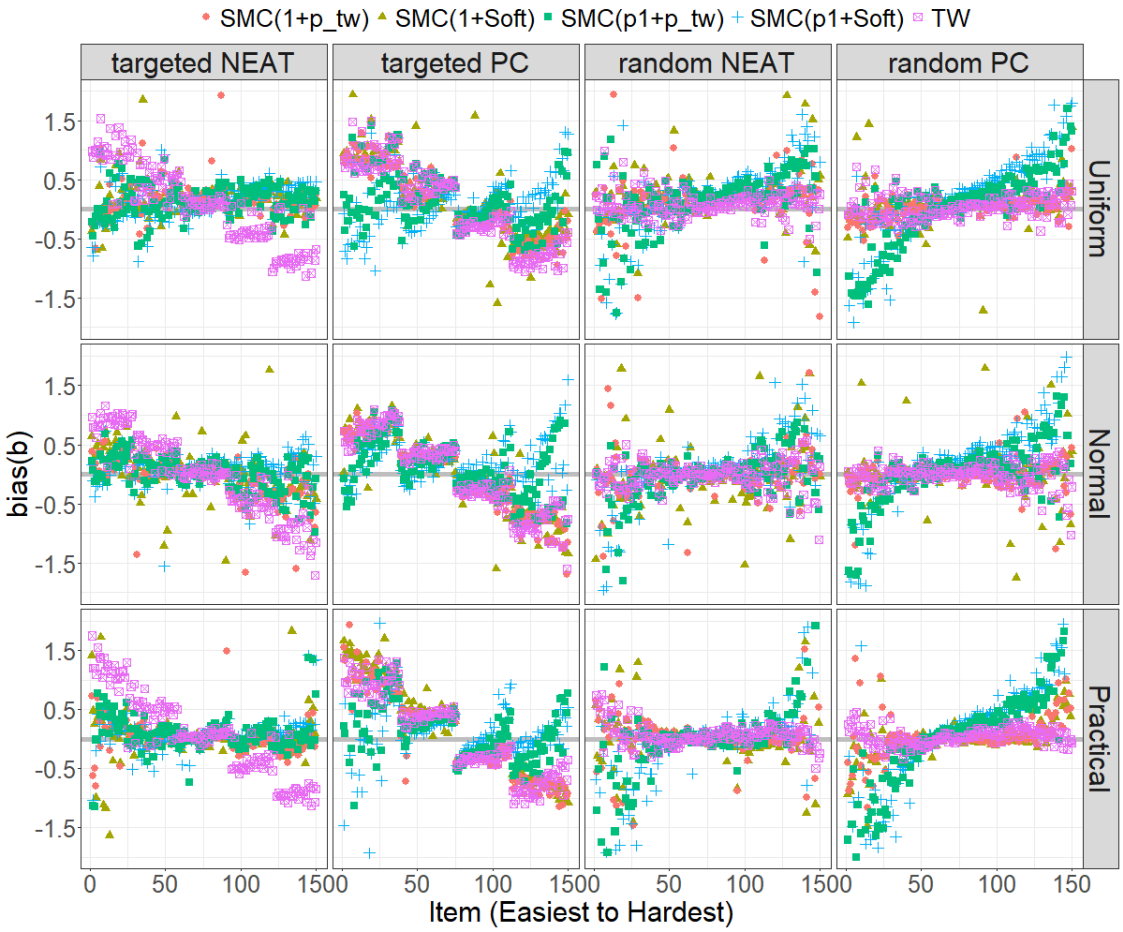


Figure 4.43: Bias(b) per item for the Softened Threshold Methods and *TW* Across Item Banks with $N = 125$

Influence from sample sizes.

The influence of sample sizes on the *SMC(1+Soft)* and *SMC(p1+Soft)* was very little. For example, Table 4.11 presents the recovery rates across three sample sizes in the practical item bank. The recovery rates from *SMC(1+Soft)* and *SMC(p1+Soft)* were very similar across three sample sizes. Similar observations were also presented in uniform and normal item banks (Tables D.24 and D.25 in Appendix). In addition, Table 4.12 presents the results for estimations of item discrimination parameters across sample sizes. Generally, bias(a) and RMSE(a) of *SMC(1+Soft)* and *SMC(p1+Soft)* decreased with increasing sample sizes. Lastly, Table 4.13 presents the average bias(b) and RMSE(b) across three sample sizes in practical item bank. Bias(b) and RMSE(b) of *SMC(1+Soft)* and *SMC(p1+Soft)* did not strictly decrease as sample sizes in-

creased. In addition, there were no extreme estimations of b using *SMC(1+Soft)* and *SMC(p1+Soft)*when $N = 250$ or $N = 500$ in the targeted NEAT and targeted PC designs as shown in Figures 4.44a and 4.44b, and there were less extreme estimations of b when $N = 250$ and $N = 500$ than $N = 125$ in the random NEAT and random PC designs (Figures 4.44c and 4.44d).

Table 4.11: Average Recovery Rates of Softened Threshold Methods Across Sample Sizes in the Practical Item Bank

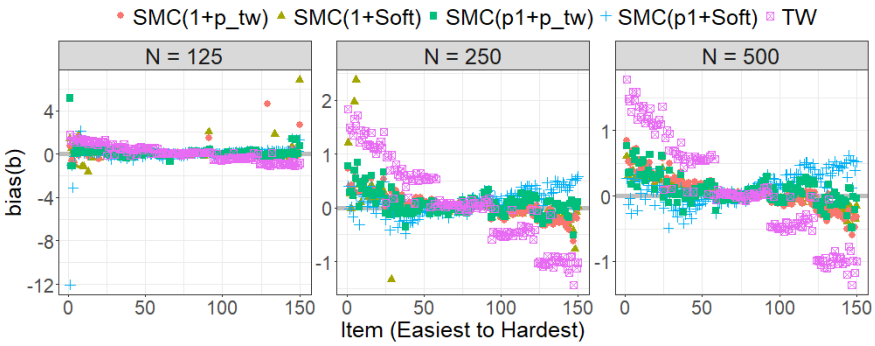
Condition	Method	N = 125	N = 250	N = 500
Targeted NEAT	SMC(1+p_tw)	0.773	0.782	0.785
	SMC(1+Soft)	0.770	0.781	0.784
	SMC(p1+p_tw)	0.775	0.781	0.785
	SMC(p1+Soft)	0.766	0.772	0.774
	TW	0.666	0.668	0.670
Targeted PC	SMC(1+p_tw)	0.615	0.610	0.605
	SMC(1+Soft)	0.615	0.610	0.605
	SMC(p1+p_tw)	0.641	0.616	0.595
	SMC(p1+Soft)	0.651	0.615	0.582
	TW	0.592	0.592	0.592
Random NEAT	SMC(1+p_tw)	0.759	0.765	0.767
	SMC(1+Soft)	0.753	0.759	0.761
	SMC(p1+p_tw)	0.762	0.766	0.768
	SMC(p1+Soft)	0.746	0.748	0.750
	TW	0.731	0.735	0.736
Random PC	SMC(1+p_tw)	0.758	0.766	0.771
	SMC(1+Soft)	0.749	0.758	0.766
	SMC(p1+p_tw)	0.762	0.765	0.766
	SMC(p1+Soft)	0.743	0.745	0.746
	TW	0.730	0.732	0.733

Table 4.12: Item Discrimination Parameters Estimations Across Sample Sizes in the Practical Item Bank

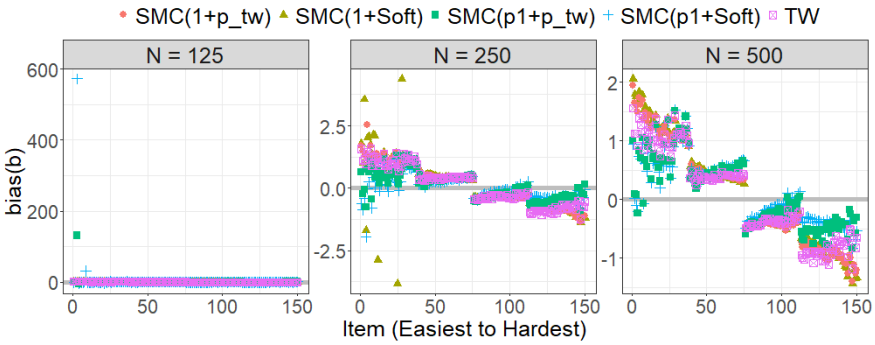
		N = 125		N = 250		N = 500	
Condition	Method	Bias(a)	RMSE(a)	Bias(a)	RMSE(a)	Bias(a)	RMSE(a)
Targeted NEAT	SMC(1+p_tw)	1.857	2.917	1.645	1.915	1.608	1.741
	SMC(1+Soft)	2.286	5.962	1.582	1.874	1.503	1.669
	SMC(p1+p_tw)	1.828	2.65	1.511	1.724	1.505	1.626
	SMC(p1+Soft)	1.603	2.429	1.146	1.413	1.063	1.247
	TW	0.515	0.723	0.560	0.664	0.502	0.585
Targeted PC	SMC(1+p_tw)	0.736	1.051	0.759	0.887	0.706	0.770
	SMC(1+Soft)	0.633	1.372	0.759	0.924	0.725	0.795
	SMC(p1+p_tw)	0.331	0.651	0.34	0.551	0.231	0.408
	SMC(p1+Soft)	0.213	0.609	0.241	0.532	0.135	0.398
	TW	0.111	0.413	0.135	0.333	0.110	0.288
Random NEAT	SMC(1+p_tw)	2.208	4.090	1.549	1.876	1.420	1.577
	SMC(1+Soft)	2.204	4.370	1.484	1.887	1.316	1.517
	SMC(p1+p_tw)	1.718	2.740	1.212	1.511	1.079	1.261
	SMC(p1+Soft)	1.541	2.505	0.862	1.239	0.739	0.996
	TW	0.477	0.741	0.450	0.554	0.414	0.484
Random PC	SMC(1+p_tw)	0.931	1.462	0.865	1.057	0.822	0.928
	SMC(1+Soft)	0.839	1.425	0.782	1.000	0.787	0.912
	SMC(p1+p_tw)	0.608	1.082	0.448	0.687	0.418	0.612
	SMC(p1+Soft)	0.448	1.004	0.239	0.518	0.229	0.461
	TW	0.318	0.581	0.293	0.419	0.279	0.360

Table 4.13: Item Difficulty Parameters Estimations Across Sample Sizes in Practical Item Banks

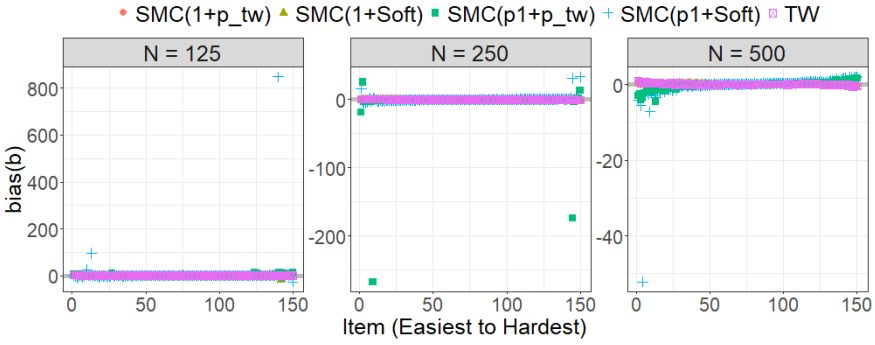
		N = 125		N = 250		N = 500	
Condition	Method	Bias(b)	RMSE(b)	Bias(b)	RMSE(b)	Bias(b)	RMSE(b)
Targeted NEAT	SMC(1+p_tw)	0.062	1.155	0.044	0.405	0.063	0.327
	SMC(1+Soft)	0.073	1.677	0.038	0.716	0.043	0.317
	SMC(p1+p_tw)	0.122	1.047	0.077	0.401	0.085	0.319
	SMC(p1+Soft)	-0.002	1.624	0.109	0.483	0.129	0.428
	TW	0.072	0.783	0.085	0.726	0.079	0.665
Targeted PC	SMC(1+p_tw)	0.074	1.277	0.101	0.855	0.087	0.793
	SMC(1+Soft)	0.174	1.149	0.089	1.685	0.106	0.803
	SMC(p1+p_tw)	0.951	7.391	0.064	0.668	0.119	0.610
	SMC(p1+Soft)	4.082	30.030	0.027	0.742	0.156	0.607
	TW	0.077	0.792	0.083	0.756	0.064	0.720
Random NEAT	SMC(1+p_tw)	0.122	2.422	0.018	0.552	0.022	0.327
	SMC(1+Soft)	-0.025	3.566	0.028	0.497	0.015	0.355
	SMC(p1+p_tw)	0.534	5.686	-2.936	24.471	-0.137	0.893
	SMC(p1+Soft)	6.223	51.762	0.533	5.057	-0.421	3.656
	TW	0.072	0.585	0.062	0.399	0.050	0.308
Random PC	SMC(1+p_tw)	-0.065	2.457	-0.011	0.499	-0.025	0.369
	SMC(1+Soft)	-0.079	3.336	-0.021	0.541	-0.014	0.387
	SMC(p1+p_tw)	-0.673	9.178	-0.073	1.042	-0.042	0.910
	SMC(p1+Soft)	-2.587	29.141	0.008	1.769	0.247	2.276
	TW	0.044	0.544	0.056	0.352	0.039	0.269



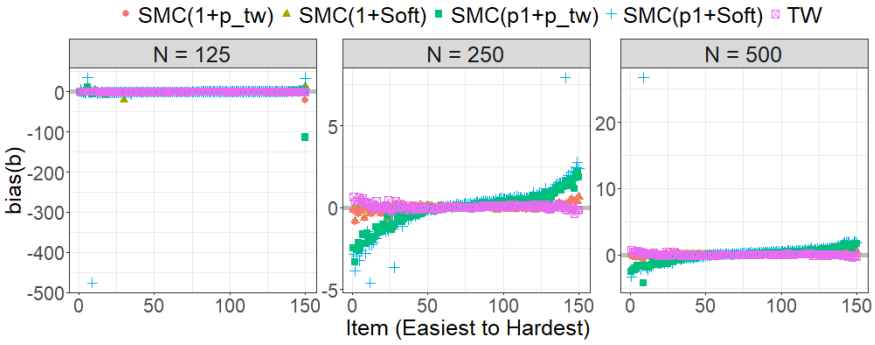
(a) Targeted NEAT Design



(b) Targeted PC Design



(c) Random NEAT Design



(d) Random PC Design

Figure 4.44: Bias(b) per Item of the Softened Methods in Practical Item Bank Across Sample Sizes

Summary. General speaking, the softened method was not able to improve the item bank calibration results. They achieved slightly better item discrimination parameter estimations than their corresponding methods in study 1, but their item difficulty parameter estimations were worse. Therefore, this method was not recommended.

4.3.3 Results from pooling converting methods

In this section, the results from *SMC(pool+p_tw)* were summarized and compared to *SMC(1+p_tw)*, *SMC(p1+p_tw)*, and *SMC(p2+p_tw)*, which were their corresponding methods in study 1, and *TW*, which was the best baseline in study 1. Similar to the previous sections, their recovery rates, estimations for item discrimination parameters and item difficulty parameters were firstly presented in practical item bank with $N = 125$, and the effects from item banks and sample sizes were examined.

Recovery Rates in the practical item bank with $N = 125$.

From Figure 4.45, the recovery rates of *SMC(pool+p_tw)* were similar to its corresponding methods and similar to *TW* in all four incomplete calibration designs. In addition, *SMC(pool+p_tw)* had slightly higher recovery rates than *TW* in the targeted NEAT designs.

Estimation for item discrimination parameters in the practical item bank with $N = 125$.

The average bias(a) and RMSE(a) were summarized in Table 4.14. The bold number indicated that the pooling converting methods produced better item parameter estimations than its corresponding methods in study 1. From this table, the pooling method resulted in better bias(a) and RMSE(a) in all four incomplete calibration designs. Its bias(a) and RMSE(a) were close but slightly worse than *TW*. In addition, Figure 4.46 presents the bias(a) per item. From this set of figures, *SMC(pool+p_tw)* had bias(a) closer to zero than *SMC(1+p_tw)*, *SMC(p1+p_tw)*, and *SMC(p2+p_tw)* in all four incomplete calibration designs. In addition, it had better bias(a) than *TW* for the easy and difficulty items but not for the medium-difficulty items in targeted NEAT, random NEAT and random PC designs.

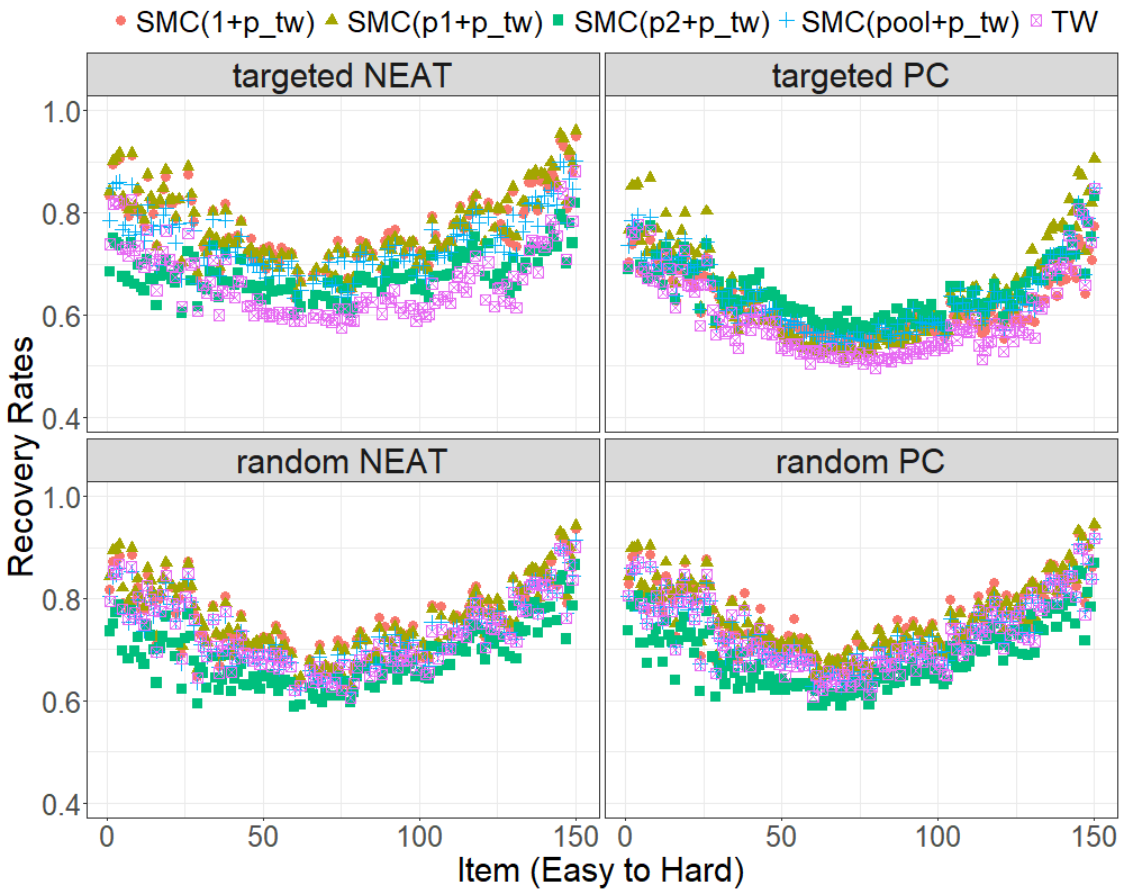


Figure 4.45: Recovery Rates per Item for Pooling Converting Methods in Practical Item Bank with $N = 125$.

Table 4.14: Summary Statistics for Item Parameter Estimations for the Pooling Method

Condition	Method	Bias(a)	RMSE(a)	Bias(b)	RMSE(b)
Targeted NEAT	SMC(1+p_tw)	1.857	2.917	0.062	1.155
	SMC(p1+p_tw)	1.828	2.650	0.122	1.047
	SMC(p2+p_tw)	1.833	3.276	0.172	1.045
	SMC(pool+p_tw)	**0.750**	**1.003**	0.165	**0.659**
	TW	0.515	0.723	0.072	0.783
Targeted PC	SMC(1+p_tw)	0.736	1.051	0.074	1.277
	SMC(p1+p_tw)	0.331	0.651	0.951	7.391
	SMC(p2+p_tw)	0.445	0.743	0.237	0.947
	SMC(pool+p_tw)	**0.235**	**0.540**	0.167	**0.726**
	TW	0.111	0.413	0.077	0.792
Random NEAT	SMC(1+p_tw)	2.208	4.090	0.122	2.422
	SMC(p1+p_tw)	1.718	2.740	0.534	5.686
	SMC(p2+p_tw)	1.364	2.277	0.189	1.189
	SMC(pool+p_tw)	**0.725**	**1.105**	0.193	**0.788**
	TW	0.477	0.741	0.072	0.585
Random PC	SMC(1+p_tw)	0.931	1.462	-0.065	2.457
	SMC(p1+p_tw)	0.608	1.082	-0.673	9.178
	SMC(p2+p_tw)	0.685	0.960	0.418	0.865
	SMC(pool+p_tw)	**0.350**	**0.701**	0.202	**0.860**
	TW	0.318	0.581	0.044	0.544

Note. The bold number indicates better item parameter estimations from the pooling converting method than the three structured matrix completion methods.

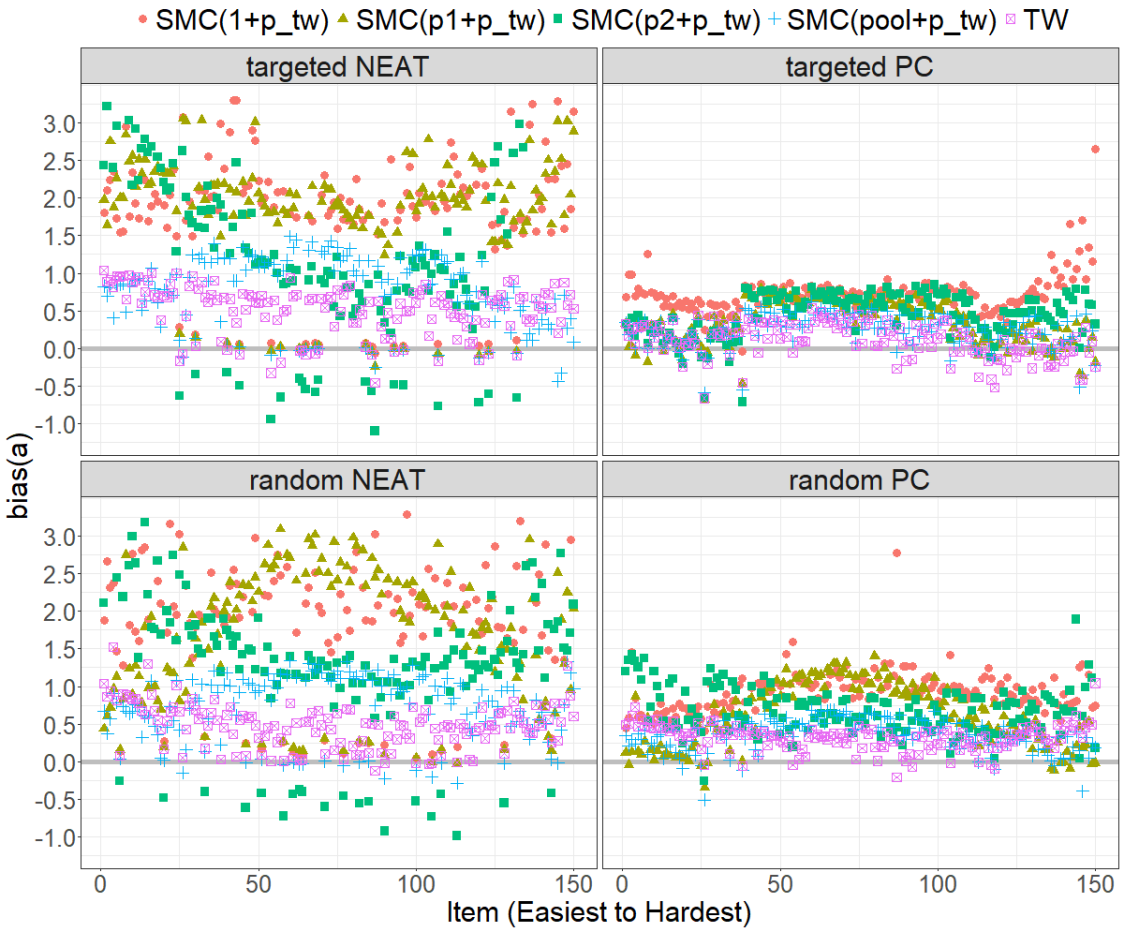


Figure 4.46: Bias(a) per Item for the Pooling Method in Practical Item Bank with $N = 125$.

Estimation for item difficulty parameters in the practical item bank with $N = 125$.

From Table 4.14, the pooling method had the smaller RMSE(b) than *SMC(1+p_tw)*, *SMC(p1+p_tw)*, and *SMC(p2+p_tw)*, and it had bias(b) close to the best case among the three structured matrix completion based imputation methods. In addition, *SMC(pool+p_tw)* had smaller RMSE(b) than *TW* in targeted NEAT and targeted PC designs but larger than *TW* in random NEAT and random PC designs. In addition, Figure 4.47 presents the bias(b) per item. *SMC(pool+p_tw)* had better bias(b) for easy items than *SMC(p2+p_tw)* in targeted NEAT, random NEAT, and random PC designs. However, it was not clear the advantage of *SMC(pool+p_tw)* in terms of estimating item difficulty parameters. Figure 4.48 presents bias(b) per item with y-axis was not fixed. From this set of figures, *SMC(pool+p_tw)* had less extreme estimations in all four incomplete calibration designs.

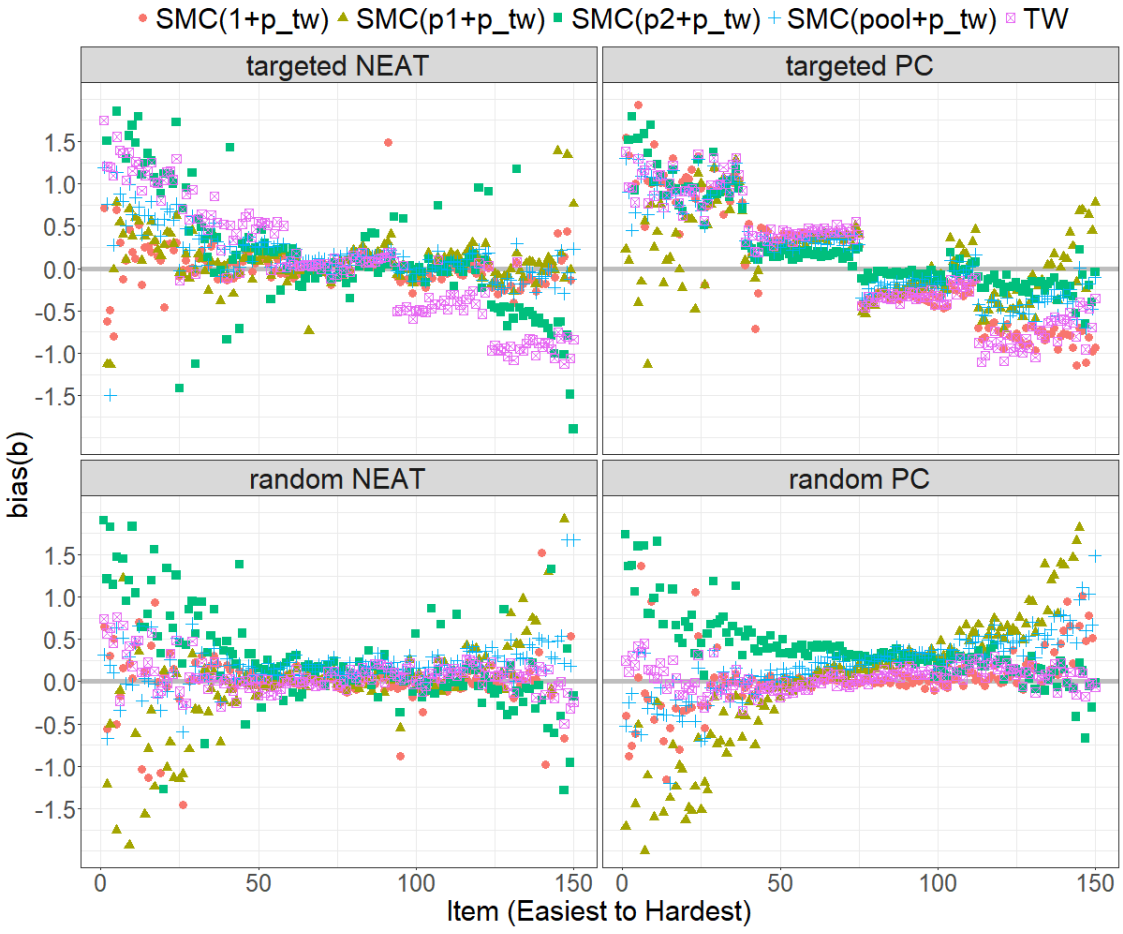


Figure 4.47: Bias(b) per Item for Pooling Converting Methods in Practical Item Bank with $N = 125$.

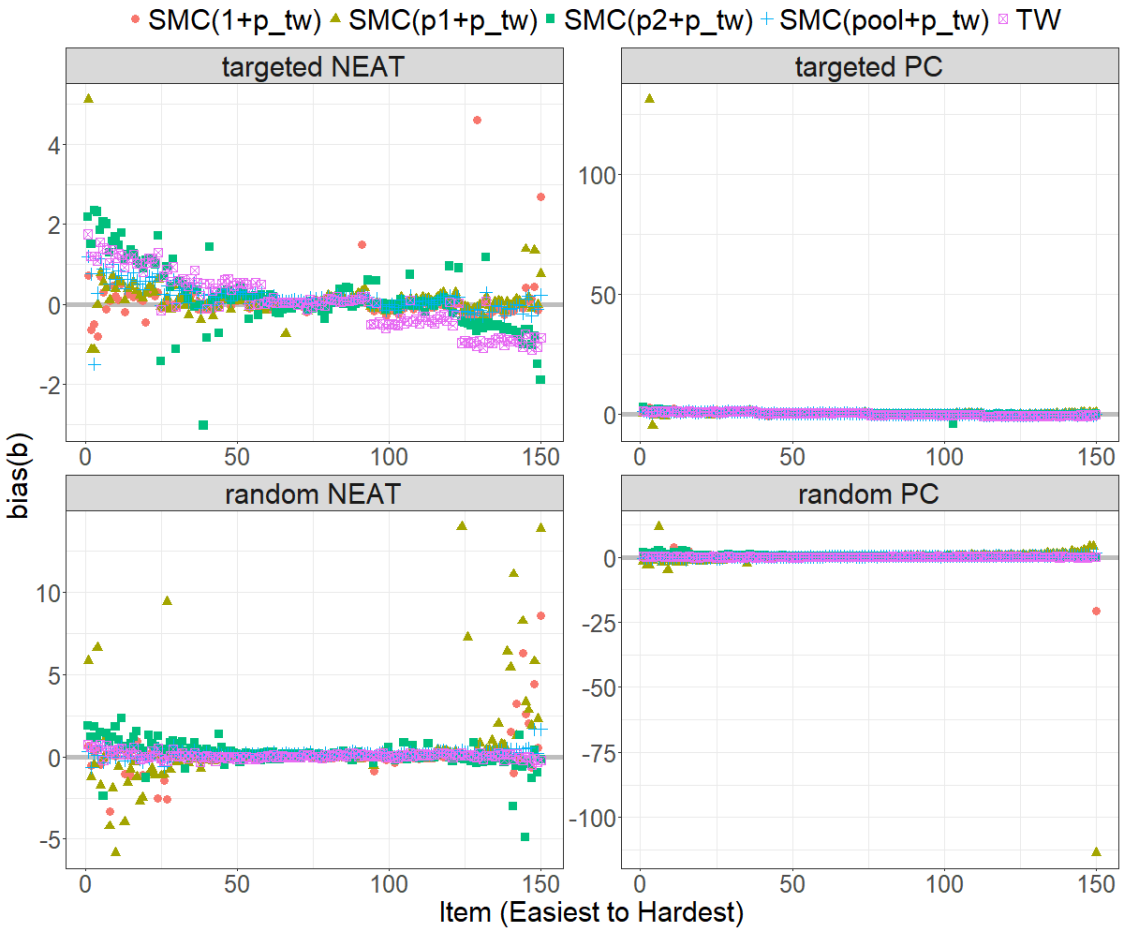


Figure 4.48: Bias(b) per Item for Pooling Converting Methods in Practical Item Bank with $N = 125$.

Influence from item banks.

The results of *SMC(pool+p_tw)* from the uniform and normal item banks were similar to the findings from the practical item bank. Firstly, the recovery rates were similar across item banks. Figures 4.49, 4.50, 4.51, and 4.52 present the recovery rates per item across three item banks in four incomplete calibration designs respectively. Generally speaking, the recovery rates of *SMC(pool+p_tw)* were in the middle of *SMC(1+p_tw)*, *SMC(p1+p_tw)*, and *SMC(p2+p_tw)* across four incomplete calibration designs, and the recovery rates of them were slightly higher in the practical item bank than the other two item banks in targeted NEAT, random NEAT, and random PC designs.

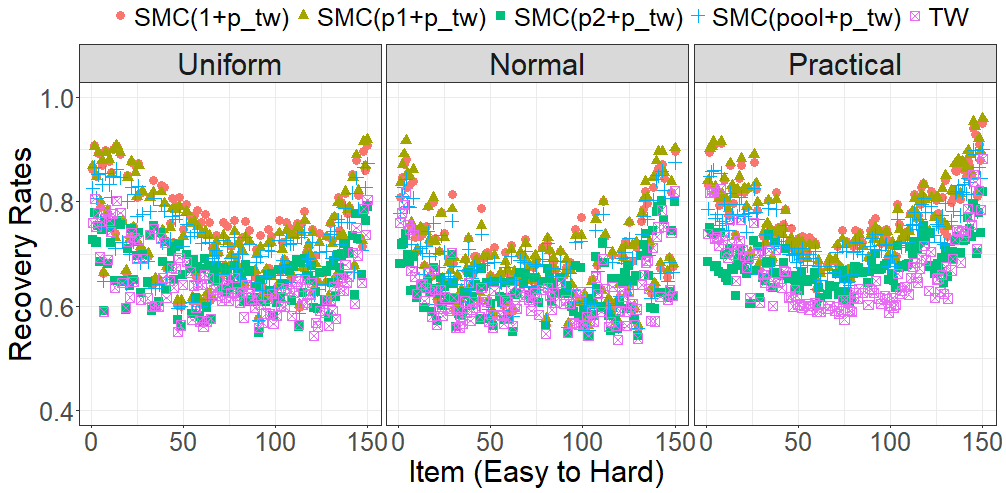


Figure 4.49: Recovery Rates per Item for the Pooling Method Across Item Banks with $N = 125$ In Targeted NEAT Designs

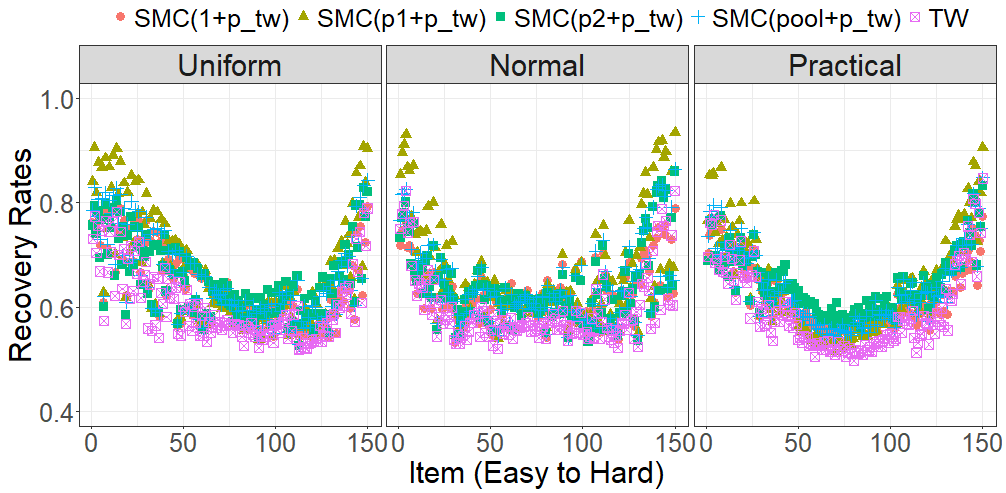


Figure 4.50: Recovery Rates per Item the Pooling Method Across Item Banks with $N = 125$ in Targeted PC Designs

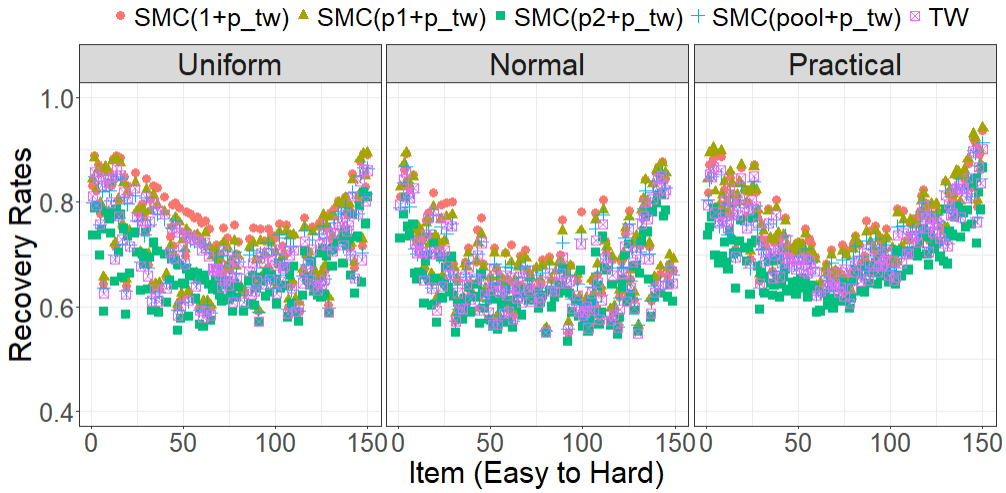


Figure 4.51: Recovery Rates per Item for the Pooling Method Across Item Banks with $N = 125$ in Random NEAT Designs

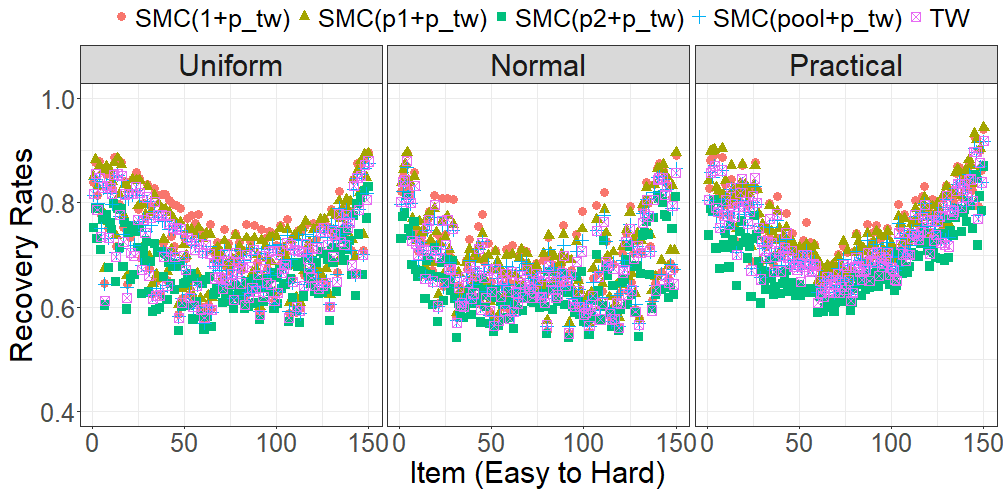


Figure 4.52: Recovery Rates per Item for the Pooling Method Across Item Banks with $N = 125$ in Random PC Designs

Secondly, the estimations of item discrimination parameters of *SMC(pool+p_tw)* were similar across item banks. For example, Figure 4.53 presents the RMSE(a) across item banks when $N = 125$. *SMC(pool+p_tw)* had the smaller RMSE(a) than its corresponding methods in study 1 but had larger RMSE(a) than *TW* in all three item banks. In addition, Figure 4.54 presents bias(a) for *SMC(pool+p_tw)*, *SMC(p2+p_tw)*, and *TW*. *SMC(1+p_tw)* and *SMC(p1+p_tw)* were not presented because their RMSE(a) were larger than *SMC(p2+p_tw)*. When using targeted NEAT and random NEAT designs, *SMC(pool+p_tw)* had bias(a) much closer to zero than *SMC(p2+p_tw)* especially for easy and difficult items in all three item banks, and its bias(a) were only slightly larger than *TW* for median difficulty items. In the other cases, bias(a) of *SMC(pool+p_tw* were very close to *TW*.

Thirdly, the results of RMSE(b) and bias(b) from *SMC(Pool+p_tw)* were also similar across item banks. From Figure 4.55, RMSE(b) of *SMC(Pool+p_tw)* were similar to *SMC(1+p_tw)* and *SMC(p1+p_tw)* but with less extreme estimations. Figure 4.56 presents the bias(b) per item. *SMC(Pool+p_tw)* had better bias(b) than its corresponding methods at either easy or difficult items across four incomplete calibration designs. In addition, it had smaller bias(b) than *TW* in targeted NEAT designs across three item banks.

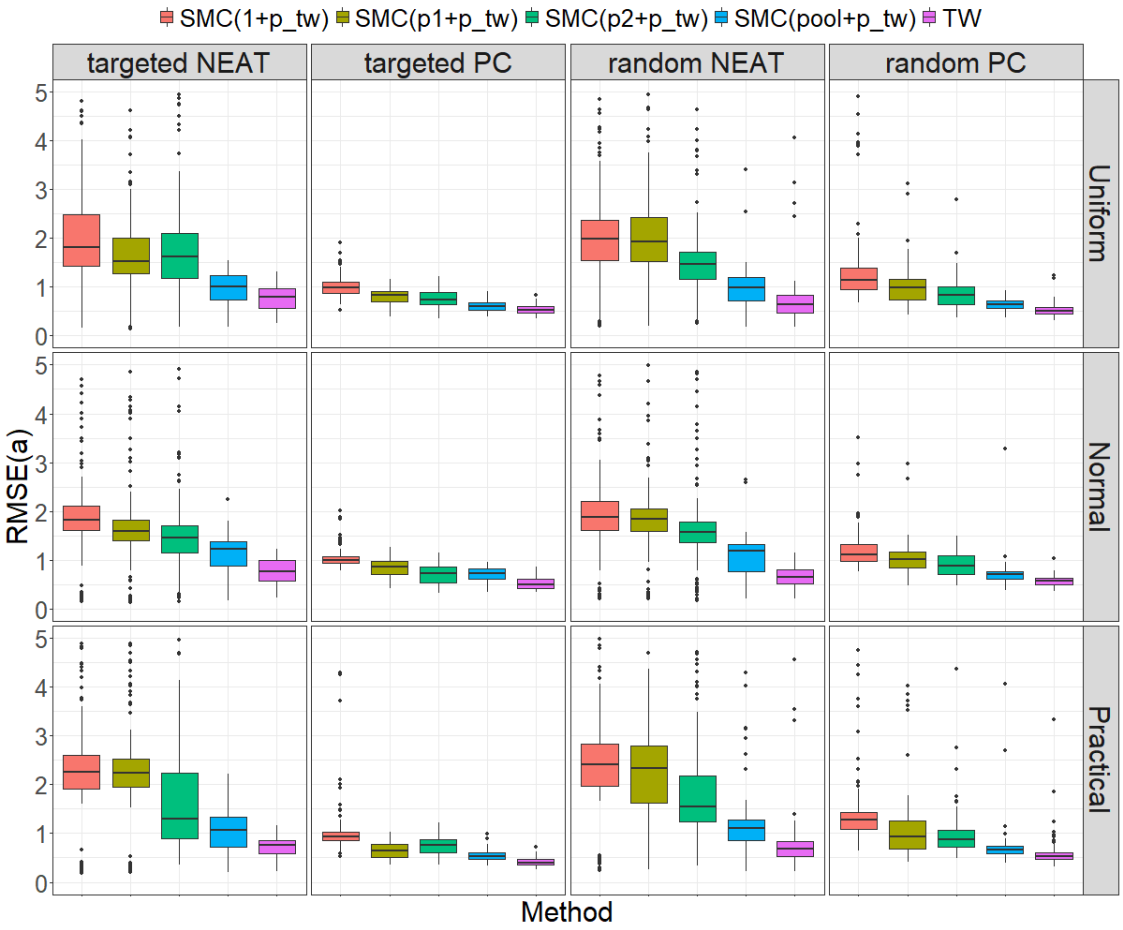


Figure 4.53: RMSE(a) for the Pooling Method and TW Across Item Banks with $N = 125$

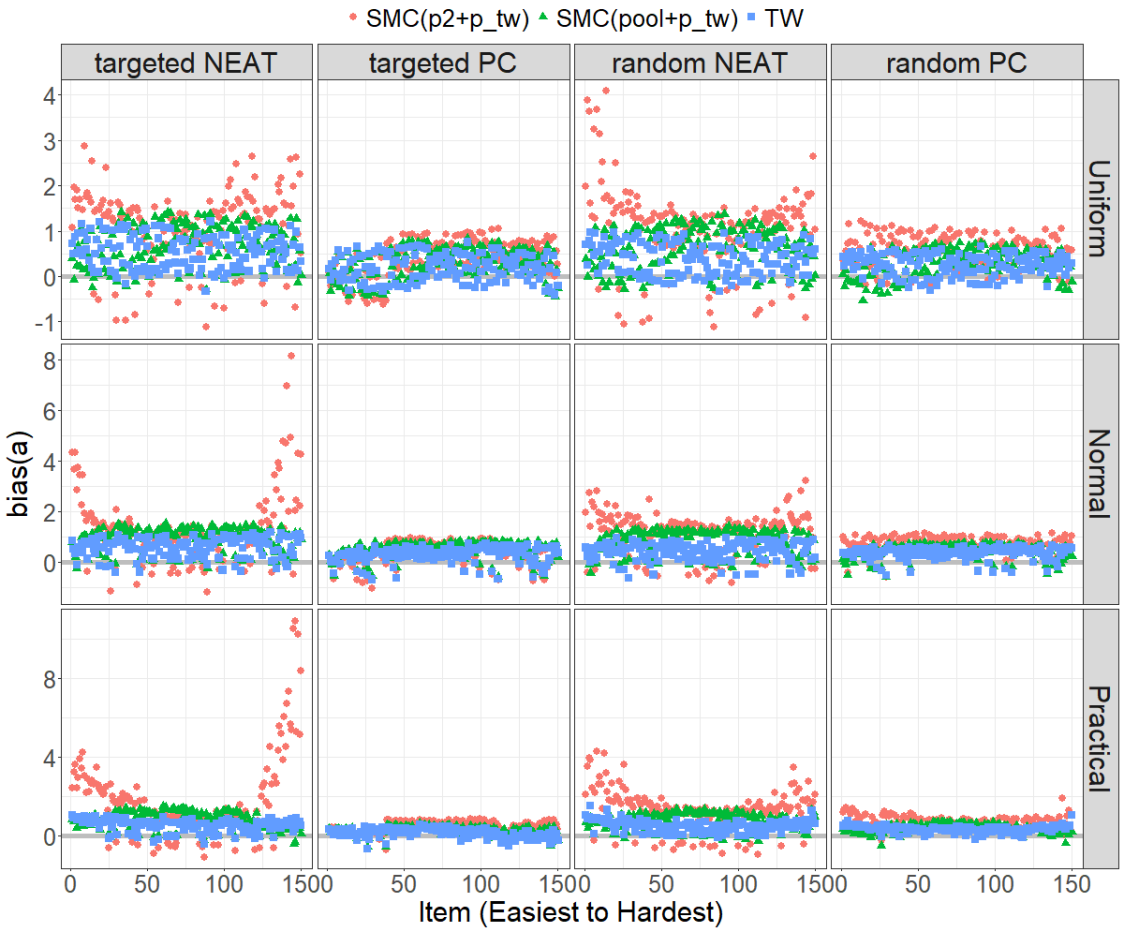


Figure 4.54: Bias(a) per item for the Pooling Method, *SMC(p2+p_tw)*, and *TW* Across Item Banks with $N = 125$

Figure 4.55: RMSE(b) for the Pooling Method Across Item Banks with $N = 125$

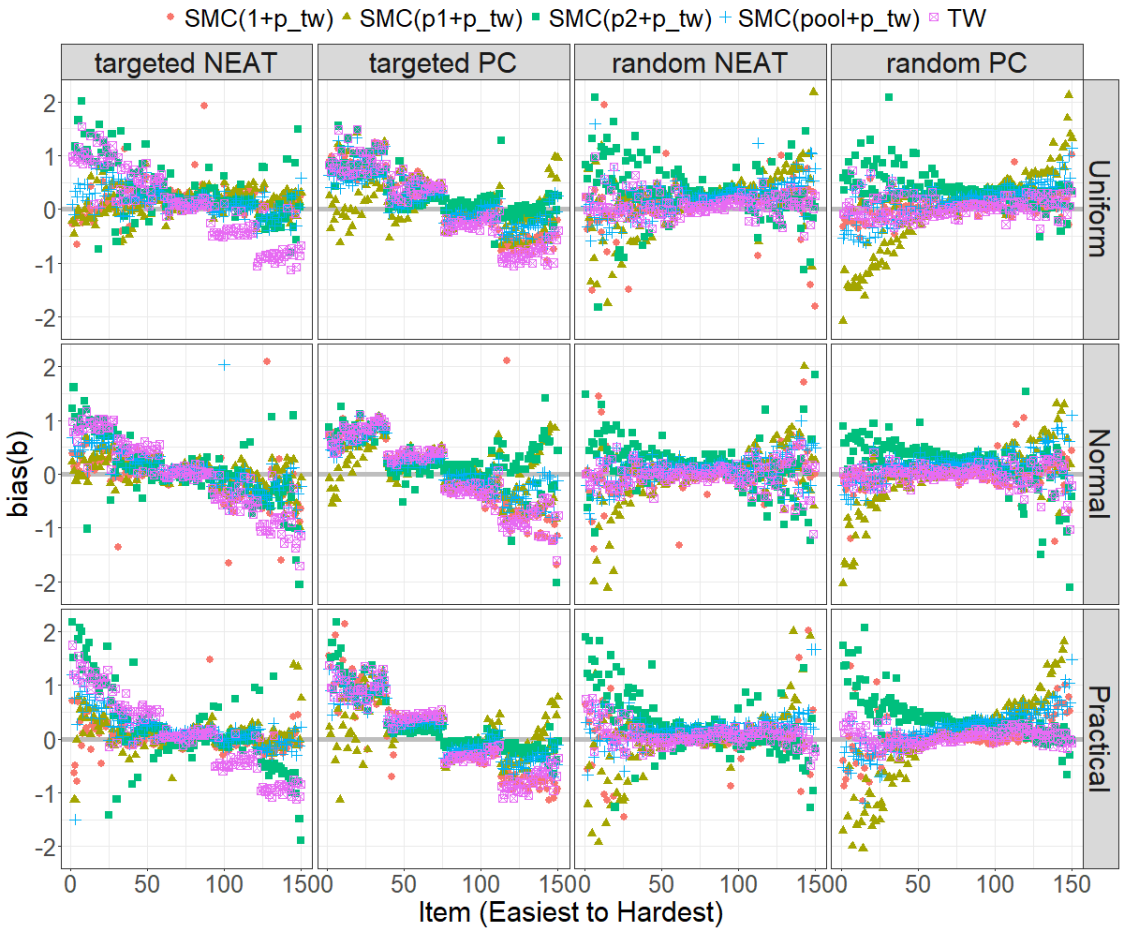


Figure 4.56: Bias(b) per item for the Pooling Method and *TW* Across Item Banks with $N = 125$

Influence from sample sizes.

The influence of sample sizes on the *SMC(pool+p_tw))* was very little. For example, Table 4.15 presents the recovery rates across three sample sizes in the practical item bank. The recovery rates of *SMC(pool+p_tw)* were very similar across three sample sizes, and they slightly increased as sample sizes increased except in the targeted PC designs. Similar observations are also presented in uniform and normal item banks (Tables D.26 and D.27 in Appendix). In addition, Table 4.16 presents the results for estimations of item discrimination parameters across sample sizes. Generally, RMSE(a) of *SMC(pool+p_tw)* decreased with increasing sample sizes, though bias(a) did not strictly decreased. In addition, across three sample sizes, *SMC(pool+p_tw)* had better Bias(a) and RMSE(a) than its corresponding methods in study 1, though it was still not as good as *TW*. Lastly, Table 4.17 presents the average bias(b) and RMSE(b) across three

sample sizes in practical item bank. Bias(b) of *SMC(pool+p_tw)* did not strictly decrease as sample sizes increased, but RMSE(b) of *SMC(pool+p_tw)* decreased slightly. *SMC(pool+p_tw)* had RMSE(b) close the best case of *SMC(1+p_tw)*, *SMC(p1+p_tw)*, and *SMC(p2+p_tw)*, and their RMSE(b) were smaller than *TW* in the targeted NEAT and targeted PC designs.

Table 4.15: Average Recovery Rates of the Pooling Method Across Sample Sizes in the Practical Item Bank

Condition	Method	N = 125	N = 250	N = 500
Targeted NEAT	SMC(1+p_tw)	0.773	0.782	0.785
	SMC(p1+p_tw)	0.775	0.781	0.785
	SMC(p2+p_tw)	0.681	0.690	0.686
	SMC(pool+p_tw)	0.744	0.751	0.752
	TW	0.666	0.668	0.670
Targeted PC	SMC(1+p_tw)	0.615	0.610	0.605
	SMC(p1+p_tw)	0.641	0.616	0.595
	SMC(p2+p_tw)	0.644	0.646	0.631
	SMC(pool+p_tw)	0.633	0.624	0.610
	TW	0.592	0.592	0.592
Random NEAT	SMC(1+p_tw)	0.759	0.765	0.767
	SMC(p1+p_tw)	0.762	0.766	0.768
	SMC(p2+p_tw)	0.687	0.688	0.686
	SMC(pool+p_tw)	0.736	0.740	0.740
	TW	0.731	0.735	0.736
Random PC	SMC(1+p_tw)	0.758	0.766	0.771
	SMC(p1+p_tw)	0.762	0.765	0.766
	SMC(p2+p_tw)	0.685	0.686	0.683
	SMC(pool+p_tw)	0.735	0.739	0.740
	TW	0.730	0.732	0.733

Summary. *SMC(pool+p_tw)* improved item discrimination parameter estimations from *SMC(1+p_tw)*, *SMC(p1+p_tw)*, and *SMC(p2+p_tw)*, thought it was still not as good as *TW*. In addition, it had relatively good estimations for item difficulty parameters, as its bias(b) and RMSE(b) were close to the best case among *SMC(1+p_tw)*, *SMC(p1+p_tw)*, and *SMC(p2+p_tw)*. In addition, their RMSE(b) were smaller than *TW* in the targeted NEAT and targeted PC designs.

Table 4.16: Item Discrimination Parameters Estimations of Pooling Method Across Sample Sizes in the Practical Item Bank

		N = 125		N = 250		N = 500	
Condition	Method	Bias(a)	RMSE(a)	Bias(a)	RMSE(a)	Bias(a)	RMSE(a)
Targeted NEAT	SMC(1+p_tw)	1.857	2.917	1.645	1.915	1.608	1.741
	SMC(p1+p_tw)	1.828	2.650	1.511	1.724	1.505	1.626
	SMC(p2+p_tw)	1.833	3.276	1.550	2.075	1.546	1.847
	SMC(pool+p_tw)	0.750	1.003	0.771	0.911	0.719	0.833
	TW	0.515	0.723	0.560	0.664	0.502	0.585
Targeted PC	SMC(1+p_tw)	0.736	1.051	0.759	0.887	0.706	0.770
	SMC(p1+p_tw)	0.331	0.651	0.340	0.551	0.231	0.408
	SMC(p2+p_tw)	0.445	0.743	0.442	0.607	0.348	0.482
	SMC(pool+p_tw)	0.235	0.540	0.273	0.429	0.229	0.359
	TW	0.111	0.413	0.135	0.333	0.110	0.288
Random NEAT	SMC(1+p_tw)	2.208	4.090	1.549	1.876	1.420	1.577
	SMC(p1+p_tw)	1.718	2.740	1.212	1.511	1.079	1.261
	SMC(p2+p_tw)	1.364	2.277	1.218	1.506	1.232	1.426
	SMC(pool+p_tw)	0.725	1.105	0.599	0.779	0.545	0.677
	TW	0.477	0.741	0.45	0.554	0.414	0.484
Random PC	SMC(1+p_tw)	0.931	1.462	0.865	1.057	0.822	0.928
	SMC(p1+p_tw)	0.608	1.082	0.448	0.687	0.418	0.612
	SMC(p2+p_tw)	0.685	0.960	0.663	0.773	0.635	0.700
	SMC(pool+p_tw)	0.350	0.701	0.291	0.470	0.288	0.422
	TW	0.318	0.581	0.293	0.419	0.279	0.360

Table 4.17: Item Difficulty Parameters Estimations of the Pooling Method Across Sample Sizes in the Practical Item Bank

		N = 125		N = 250		N = 500	
Condition	Method	Bias(b)	RMSE(b)	Bias(b)	RMSE(b)	Bias(b)	RMSE(b)
Targeted NEAT	SMC(1+p_tw)	0.062	1.155	0.044	0.405	0.063	0.327
	SMC(p1+p_tw)	0.122	1.047	0.077	0.401	0.085	0.319
	SMC(p2+p_tw)	0.172	1.045	0.310	0.657	0.393	0.659
	SMC(pool+p_tw)	0.165	0.659	0.147	0.456	0.147	0.378
	TW	0.072	0.783	0.085	0.726	0.079	0.665
Targeted PC	SMC(1+p_tw)	0.074	1.277	0.101	0.855	0.087	0.793
	SMC(p1+p_tw)	0.951	7.391	0.064	0.668	0.119	0.610
	SMC(p2+p_tw)	0.237	0.947	0.203	0.559	0.257	0.511
	SMC(pool+p_tw)	0.167	0.726	0.147	0.667	0.171	0.664
	TW	0.077	0.792	0.083	0.756	0.064	0.72
Random NEAT	SMC(1+p_tw)	0.122	2.422	0.018	0.552	0.022	0.327
	SMC(p1+p_tw)	0.534	5.686	-2.936	24.471	-0.137	0.893
	SMC(p2+p_tw)	0.189	1.189	0.240	0.558	0.308	0.511
	SMC(pool+p_tw)	0.193	0.788	0.146	0.455	0.169	0.361
	TW	0.072	0.585	0.062	0.399	0.050	0.308
Random PC	SMC(1+p_tw)	-0.065	2.457	-0.011	0.499	-0.025	0.369
	SMC(p1+p_tw)	-0.673	9.178	-0.073	1.042	-0.042	0.910
	SMC(p2+p_tw)	0.418	0.865	0.307	0.490	0.346	0.455
	SMC(pool+p_tw)	0.202	0.860	0.134	0.494	0.156	0.396
	TW	0.044	0.544	0.056	0.352	0.039	0.269

4.3.4 Comparing results from all the proposed methods.

The performance of all 7 methods proposed in study 2, their corresponding methods in study 1, and the best baseline *TW*, were investigated together in this section. In particular, recovery rates, bias and RMSE for both item discrimination and difficulty parameters were calculated. The results from practical item bank with $N = 125$ were first presented for each evaluation criterion, and the effects from item banks and sample sizes were discussed next.

Recovery Rates of all methods.

Figure 4.57 presents the box plot of the recovery rates from all methods in the practical item bank with the sample size of 125. Among the proposed methods in study 2, *SMC(1+soft)*, *SMC(p1+Soft)*, and *SMC(pool+p_tw)* had higher recovery rates than the proposed switching methods in the targeted NEAT, random NEAT, and random PC designs. The proposed methods in study 2, though had recovery rates close to or slightly worse than *1bit(p_tw)* and *SMC(1+p_tw)*, had higher recovery rates than *TW* in targeted NEAT designs. In addition, *SMC(1+soft)*, *SMC(p1+Soft)*, and *SMC(pool+p_tw)* had higher recovery rates than *TW* in the random NEAT and random PC designs, while the switching method had similar recovery rates with *TW*. In targeted PC designs, the recovery rates were generally lower of all methods. In addition, *SMC(1+Soft)* and *SMC(p1+Soft)* had large variations in recovery rates, which were similar to *SMC(1+p_tw)* and *SMC(p1+p_tw)*. *1bit(Switch_Genera)* and *1bit(Switch_Preferred)* had small variations, but their recovery rates were slightly lower than *TW*.

As discussed before, the item banks and sample sizes had small effects on the recovery rates of the proposed methods. For example, Figure 4.58 presents the recovery rates in the practical item bank across sample sizes and incomplete calibration designs. The recovery rates were similar across sample sizes with smaller variations as sample sizes decreased. Figures C.33 and C.34 in the Appendix presented the corresponding figures in the uniform and normal item banks respectively. Generally speaking, the switching method, either using *Switch_General* or *Switch_Preferred*, produced lower recovery rates than the other two proposed methods in study 2.

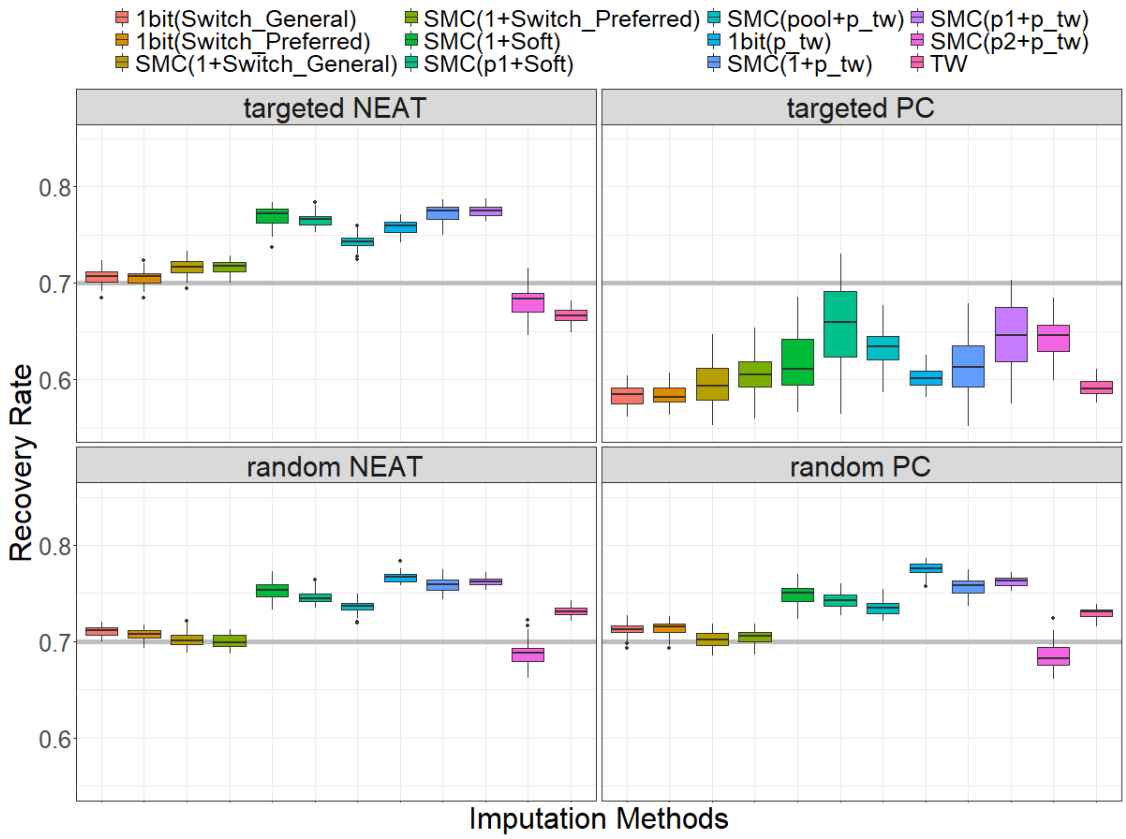


Figure 4.57: Recovery Rates of All Methods in Practical Item Bank with $N = 125$.

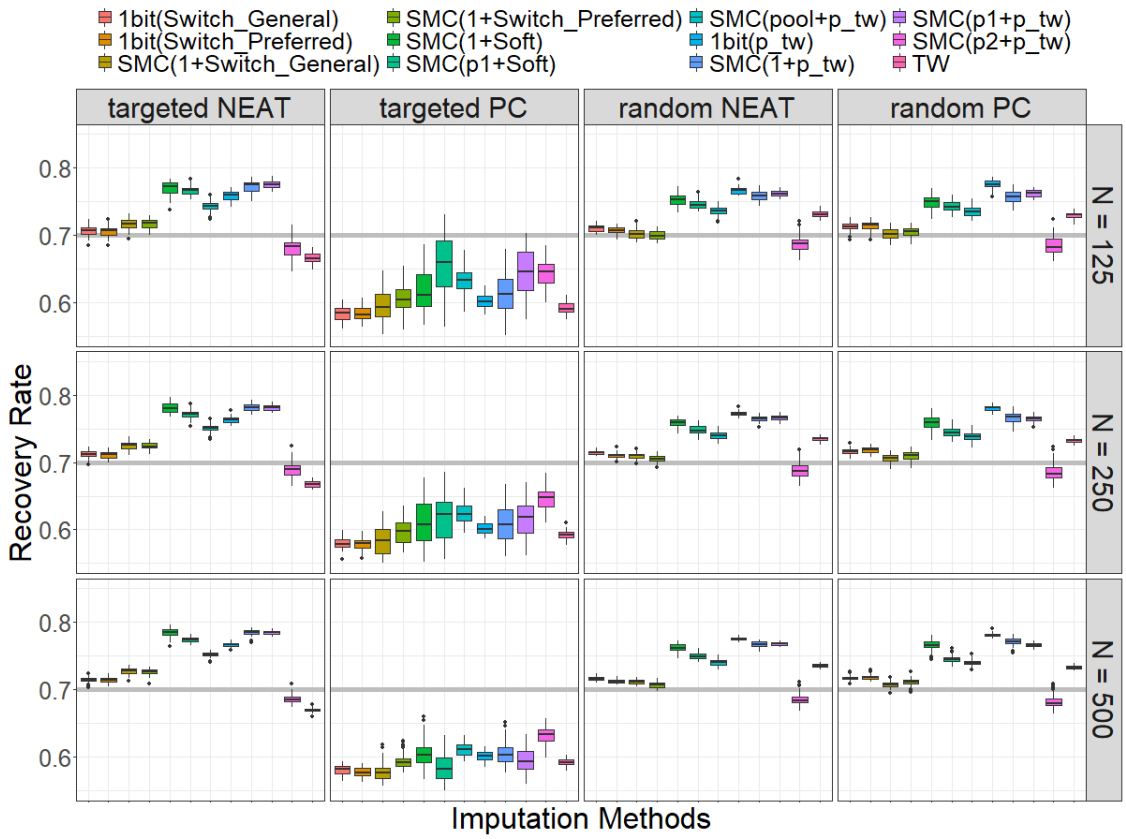
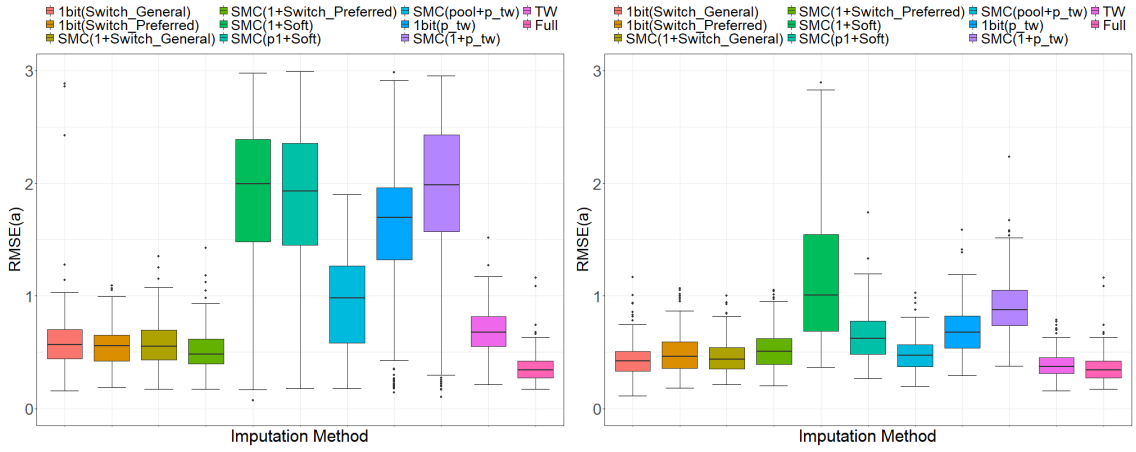


Figure 4.58: Recovery Rates of All Methods in Practical Item Bank with $N = 125$.

Comparing bias(a) and RMSE(a) among all methods.

Figure 4.59 presents the box plot for RMSE(a) of the 7 proposed methods in study 2 and *1bit(p_tw)*, *SMC(1+p_tw)*, *TW*, and *Full*. *Full* is the item calibration results using the original full response matrix, which served as the best case scenario. The results of *SMC(p1+p_tw)* and *SMC(p2+p_tw)* were not presented because of their similar or worse performance than *SMC(1+p_tw)* as shown in study 1. In targeted NEAT and random NEAT designs, switching methods and pooling converting methods produced smaller RMSE(a) than their corresponding methods in study 1, and switching methods achieved RMSE(a) that were close to or even better than *TW*. In targeted PC designs, in addition to the switching methods and pooling converting methods, softened threshold method paired with *SMC(p1+Soft)* slightly improved RMSE(a), but the improvement was not as much as the switching methods and the pooling method. In addition, the switching and pooling methods had RMSE(a) slightly larger than *TW*. In random PC designs, all proposed methods in study 2 had smaller RMSE(a). In particular, the switching methods, either using the switching parameters of *Switch_General* or *Switch_Preferred*, had similar RMSE(a) with *TW* with fewer extreme estimations.



(a) Targeted NEAT Design

(b) Targeted PC Design

(c) Random NEAT Design

(d) Random PC Design

Figure 4.59: RMSE(a) of Selected Methods in Practical Item Bank with Sample Size of 125

From the results from RMSE(a), the switching method improved estimations for item discrimination parameters the most. To further evaluate their performance, their bias(a) per item were plotted in Figure 4.60 along with *TW*. From this set of figures, the switching method had bias(a) closer to zero than *TW* using either set of switching parameters from *Switch_General* or *Switch_Preferred* and paired with either *1bit(p_tw)* or *SMC(1+p_tw)*. As discussed before, the item banks and sample sizes had little impact on the item discrimination parameter estimations. Figure 4.61 presents the bias(a) from the switching methods across sample sizes in the practical item banks, and bias(a) were very similar when sample sizes were different. The corresponding figures for uniform and normal item banks were presented in Figure C.35 and C.36 in the Appendix.

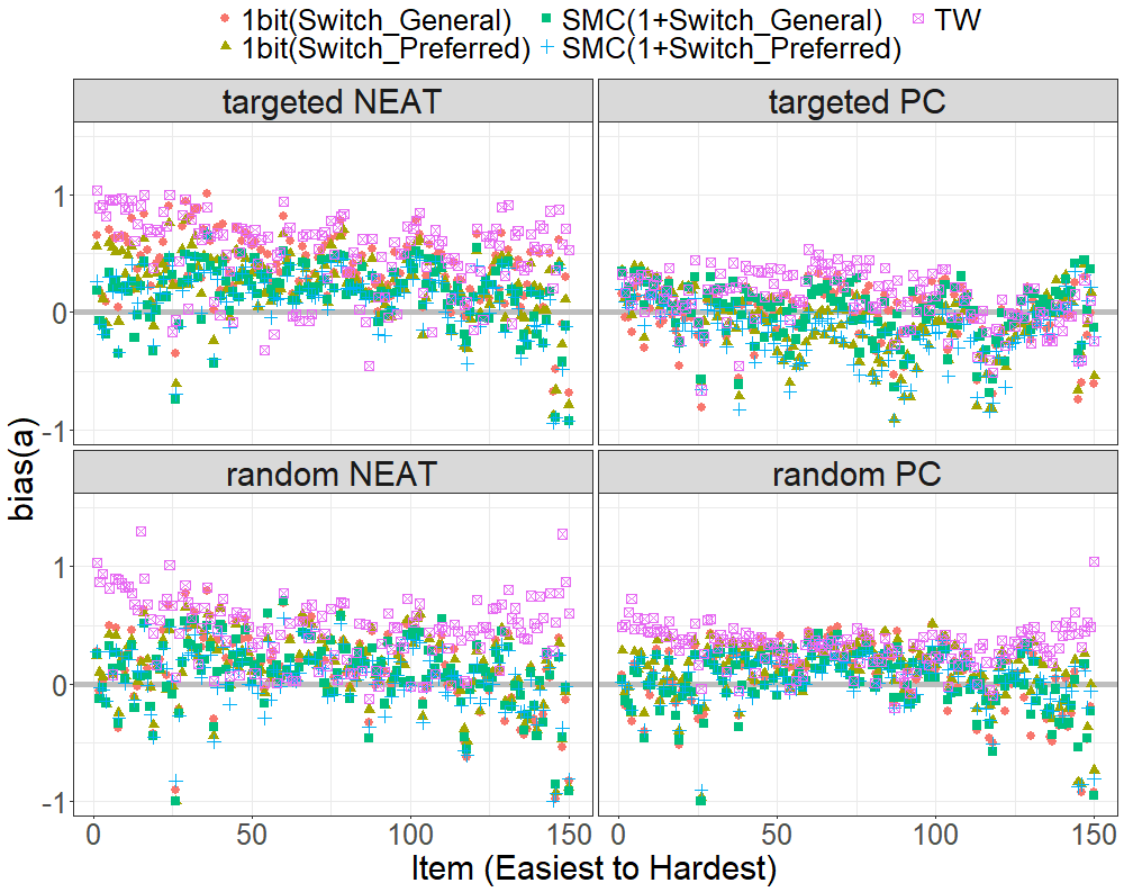


Figure 4.60: Bias(a) Per Item of Switching Methods in Practical Item Bank Across Incomplete Calibration Designs with $N = 125$.

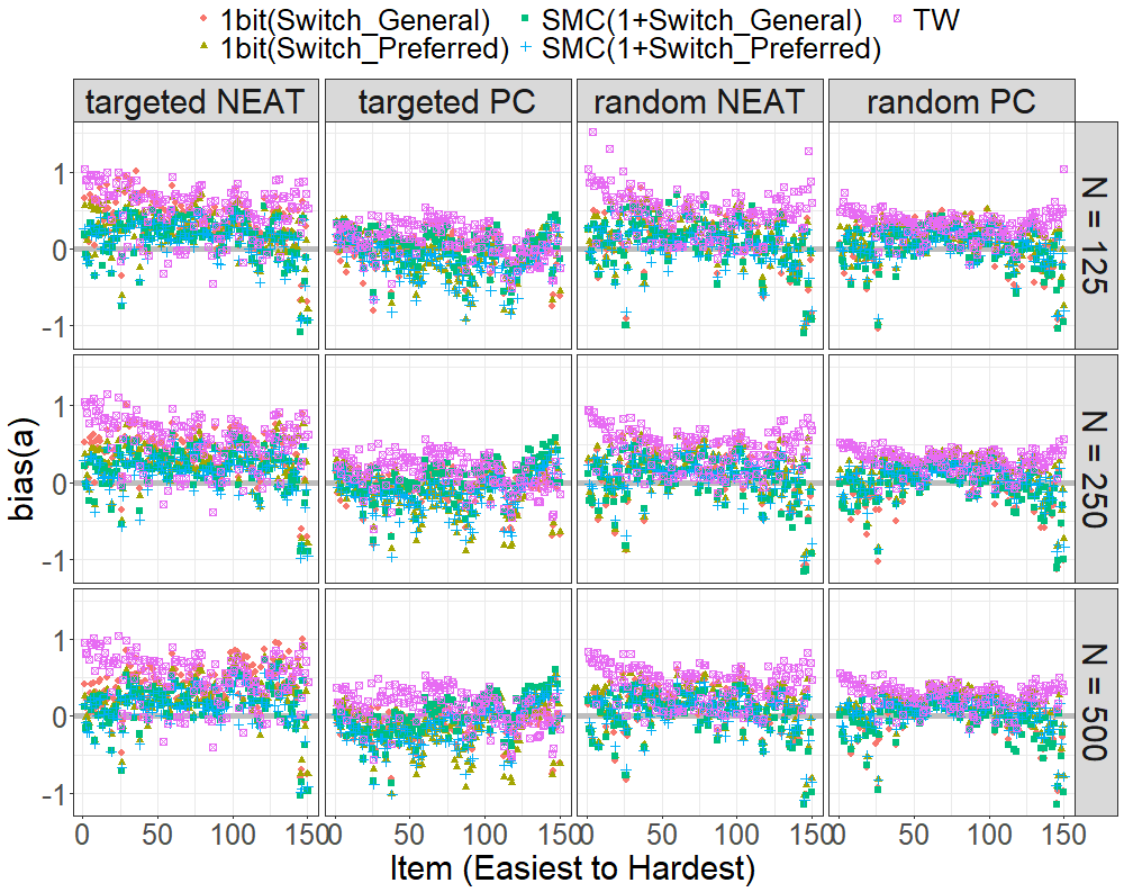


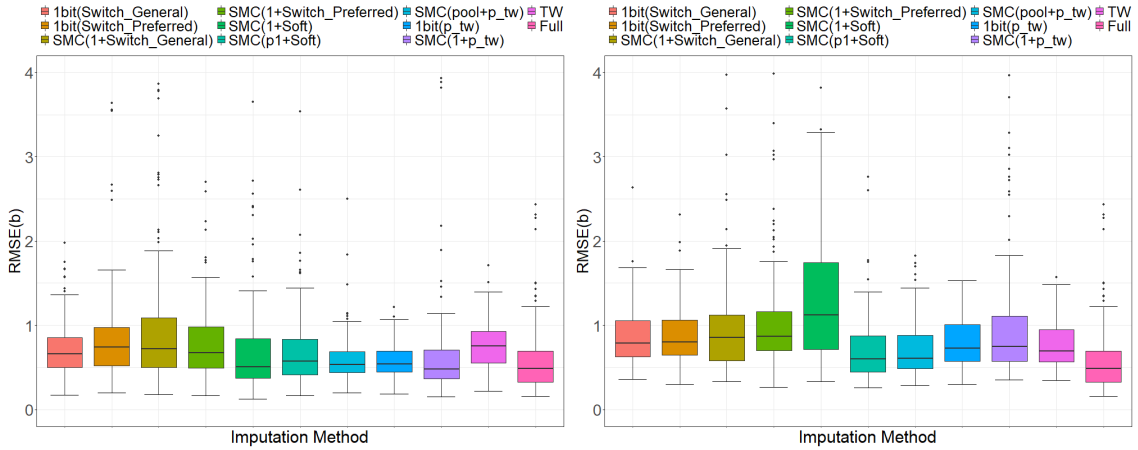
Figure 4.61: Bias(a) Per Item of Switching Methods in Practical Item Bank Across Incomplete Calibration Designs and Sample Sizes.

Comparing bias(b) and RMSE(b) among all methods.

Similar to the last section, Figure 4.62 presents the box plot for RMSE(b) of the proposed methods in study 2 and *1bit(p_tw)*, *SMC(1+p_tw)*, *TW*, and *Full*. All the methods had similar RMSE(b) in general. In targeted NEAT and random NEAT designs, all methods had smaller medium RMSE(b) than *TW*, and *SMC(p1+Soft)* and *SMC(pool+p_tw)* had the smallest variations. In targeted PC designs, *SMC(1+Soft)* was the worst, while *SMC(p1+Soft)* and *SMC(pool+p_tw)* were slightly better than the others. In addition, switching methods using both *Switch_General* and *Switch_Preferred* had similar RMSE(b) for both *1bit(p_tw)* and *SMC(1+p_tw)*. This finding was consistent with the result of recovery rates. In random PC designs, *SMC(p1+Soft)* has the worst RMSE(b), and the other proposed methods have RMSE(b) slightly larger than *TW*.

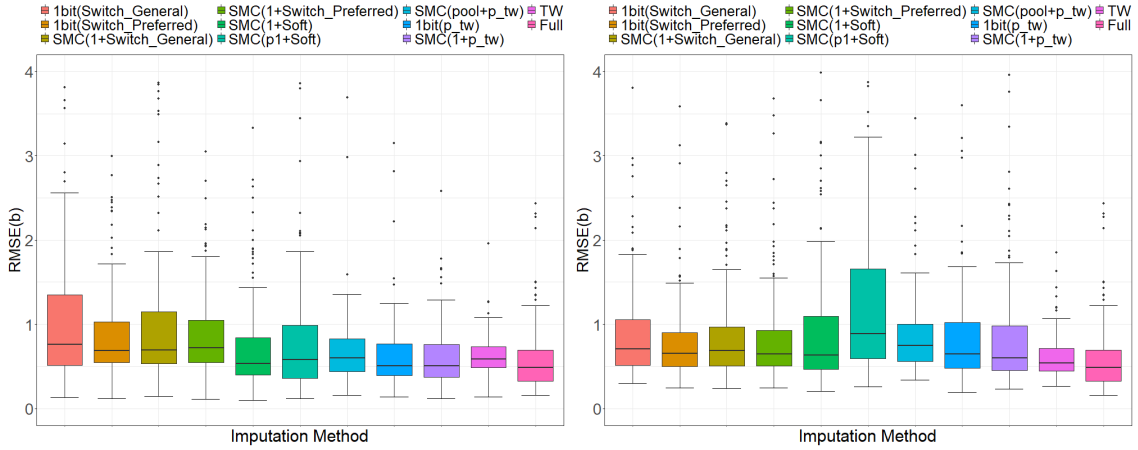
Given the results from RMSE(b), *SMC(pool+p_tw)* was the best among the proposed methods in study 2, though many of them have relatively similar RMSE(b) in different incomplete calibration designs. This finding was not consistent with the finding of RMSE(a) and bias(a), where switching methods outperformed the other proposed methods. To further evaluate the performance of *SMC(pool+p_tw)* along with the switching methods, their bias(b) per items were plotted. In addition, only *1bit(Switch_General)* and *SMC(1+Switch_General)*, were plotted (Figure 4.63). The *Switch_General* was selected, because it was more practical than *Switch_Preferred* in reality. From Figure 4.63, *SMC(pool+p_tw)* mainly had advantage over switching methods for easy and difficult items in targeted NEAT, targeted PC, and random NEAT designs. This observation was also shown in other item banks and sample sizes. Figures 4.64, C.37, and C.38 presents bias(b) across incomplete calibration designs and sample sizes in the practical, uniform, and normal item banks.

Summary. In this simulation, the proposed switching method, softened threshold method, and pooling converting methods were investigated in four incomplete calibration designs, three item banks, and three sample sizes. Among the three methods, the switching method, either using the set of general switching parameters (*Switch_General*) or using the sets of specific switching parameters determined from study 1 (*Switch_Preferred*), was able to improve item discrimination parameters, and provided better item discrim-



(a) Targeted NEAT Design

(b) Targeted PC Design



(c) Random NEAT Design

(d) Random PC Design

Figure 4.62: RMSE(b) of Selected Methods in Practical Item Bank with Sample Size of 125

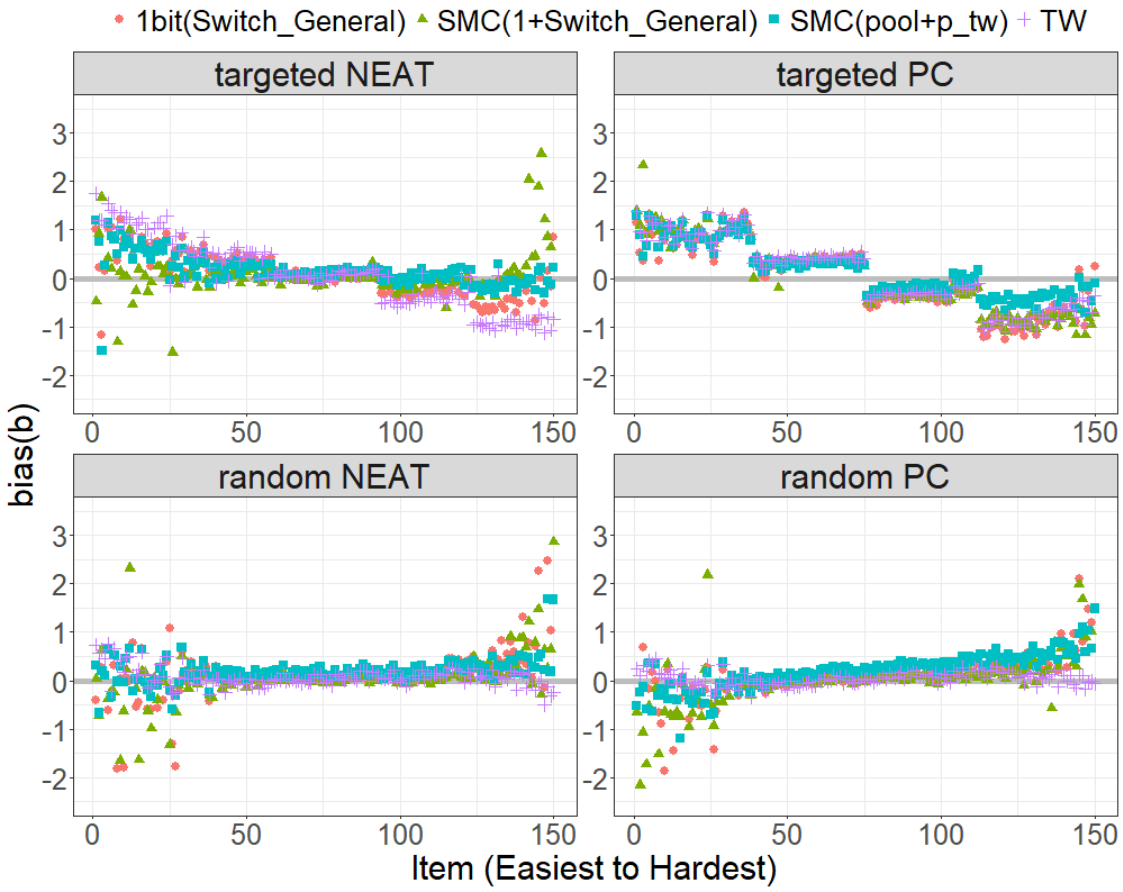


Figure 4.63: Bias(b) Per Item of Selected Methods in Practical Item Bank with $N = 125$.

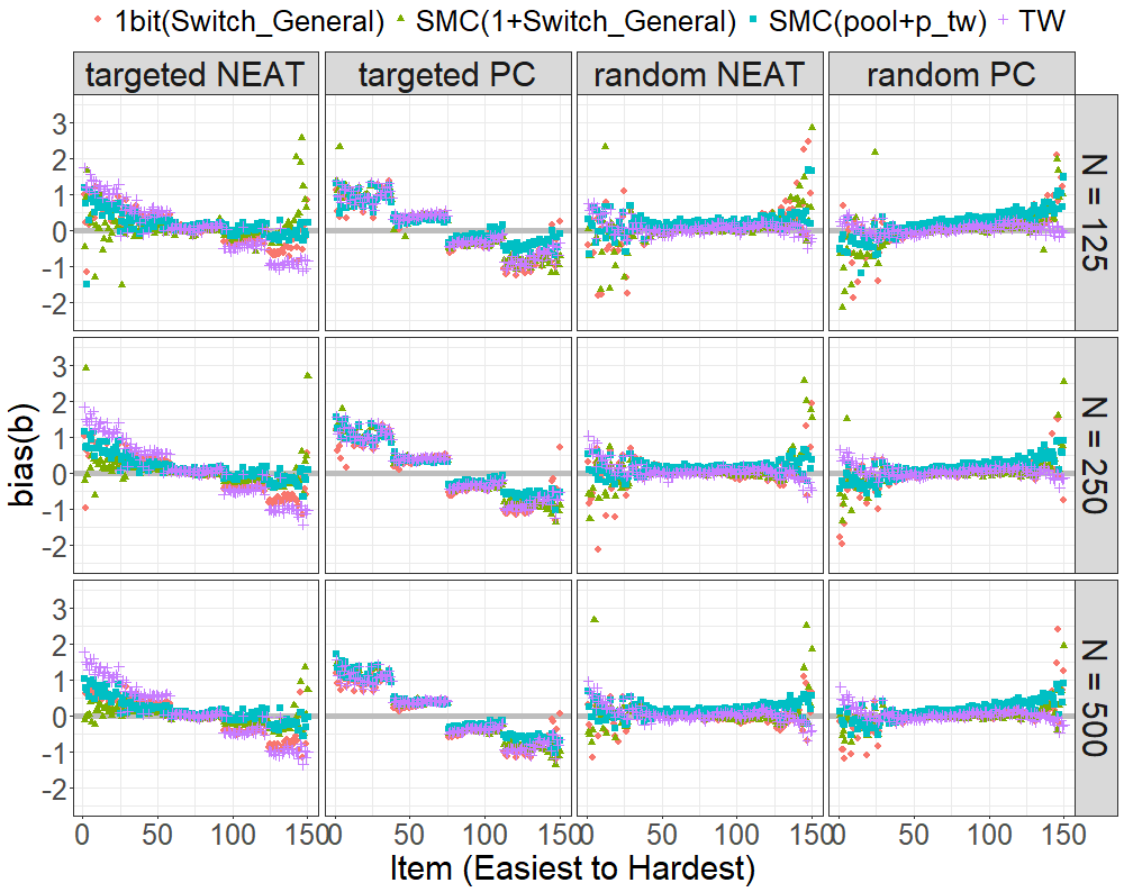


Figure 4.64: Bias(b) Per Item of Selected Methods in Practical Item Bank with $N = 125$.

ination parameter estimations than *TW* in targeted NEAT, random NEAT, and random PC designs. In addition, it also obtained relatively accurate item difficulty parameter estimations, which were comparable to their corresponding methods in study 1 and *TW*. In addition, the pooling method also improved item discrimination parameter estimations, though the amount of improvement was less than the switching methods. In addition, the pooling method had the relatively accurate item difficulty parameter estimations. On the contrary, the softened threshold method was not able to improve item discrimination parameter estimations, and it had worse item difficulty parameters than its corresponding method in study 1. Thus, the switching method and pooling the converting methods were recommended in general.

Chapter 5

Study 3: Applications of the Proposed Item Bank Calibration Strategies

Simulation studies 1 and 2 evaluated the proposed item bank calibration strategies in terms of the recovery rate and the accuracy of item parameter estimation. In this study, we examined the proposed strategies from two additional perspectives. Firstly, the potential benefits of improving the accuracy of examinees' ability estimations in a CAT were investigated. The focus was on whether the CAT using the item bank calibrated with the proposed strategies provided more precise estimations for examinees' abilities than using baseline calibration methods, such as *FIML* and *TW*. Secondly, the proposed strategies were applied to an empirical data set, which was collected from a recent study to develop an item bank for CAT. The purpose was to demonstrate and investigate the proposed strategies in an empirical data set.

5.1 The CAT Simulation

To investigate the potential benefits of the proposed item bank calibration strategies, a CAT simulation was conducted using the calibrated item banks in Study 2, and the estimated examinees' abilities were

evaluated. In particular, *SMC(pool+p_tw)*, *SMC(1+Switch)*, and *1bit(Switch)* were selected in this CAT simulation study, because they demonstrated the best performance in Study 2. Note that *SMC(1+Switch)* and *1bit(Switch)* used the general sets of switching parameters. In addition, the estimated item parameters using *FIML* was served as the lower baseline. True item parameters were also used to present the best scenario. The estimated item parameters from *TW* was also investigated as they were shown to be relatively accurate in study 1 especially in the random calibration designs. The details of the CAT simulation study design were introduced next in this section and the results were followed.

5.1.1 CAT Simulation Study Design

Simulation Factors

As introduced in Section 2.2, six components are required for a CAT. Table 5.1 presents the simulated components for CAT and other simulation factors in this CAT simulation study. In particular, the measurement model was the 2PL model as our item bank was built on the 2PL model. In addition, only practical item bank was used in this study. Uniform and Normal item bank were not included because their results were similar to the practical item bank in Study 1 and 2. Furthermore, the calibrated item parameters using the targeted NEAT and random NEAT designs were included because the proposed calibration strategies showed the most improvements in these two incomplete calibration designs. In this simulated CAT, the first 3 items were randomly selected, because no prior information about examinees were available. Examinees' interim $\hat{\theta}$ were estimated using EAP with the prior of $N(0, 1)$ for the first 3 items. After the first 3 items, items were selected using maximum Fisher information, and examinees' interim $\hat{\theta}$ were estimated using MLE. The stopping criterion was the fixed test length, which meant the CAT would stop when examinees answered a predetermined number of items. The test lengths were fixed at 20 and 40 items, representing short and long tests respectively. Lastly, the examinees were simulated using two types of distributions. The first one was similar to that in Study 1 and 2, which was a mixture of normal distributions, $\theta \sim N(\boldsymbol{\mu}, 1)$, where $\boldsymbol{\mu} = (-1, -0.5, 0, 0.5, 1)$. The second one was different from that in Study 1 or 2, and it simulated examinees from a standard normal distribution,

$\theta \sim N(0, 1)$. The first procedure was a common practice that the item bank should be calibrated on the same population as the targeted population of the CAT application. The second procedure was added to evaluate the scenario where the calibration population did not exactly match the targeted population for CAT. The sample size was fixed at 1000 for both procedures. In particular, 200 examinees were simulated per group in the first procedure. The simulation was repeated for 50 replications where examinees' abilities were regenerated in every replication and item parameters were fixed across replications.

Table 5.1: Simulation Factors in Study 3

Factors	Design
Measurement Model	2PL
Item Bank	Practical item bank with estimated item parameters using: (1) True item parameters (2) FIML (3) TW (4) 1bit(Switch) (5) SMC(1+Switch) (6) SMC(Pool+p_tw) in four incomplete calibration designs
Starting Criterion	First 3 items were randomly selected
Ability Estimation	EAP with N(0,1) as prior for the first 3 items MLE since the 4th item
Item Selection Algorithm	Maximum Fisher information
Stopping Criterion	Fixed test length
Test Length	20 and 40
Examinees	$\theta \sim N(\mu, 1), \mu = c(-1, -0.5, 0, 0.5, 1)$ $\theta \sim N(\mu, 1)$
Sample Size	N = 1000

Simulation Procedures

Figure 5.1 presents the flow chart for the CAT simulation. There were two simulating processes for each examinee. Firstly, using the administered item's true parameter a and b and the examinee's true ability θ, his or her response of that item was simulated using the 2PL model. That is, the probability of a correct response was $P(X = 1|\theta, a, b)$. In the second simulating process, the simulated response, X, either correct or incorrect, and the estimated item parameters $\hat{a}$ and $\hat{b}$ were used for the CAT to obtain

the MLE or EAP for examinee's interim ability estimation $\hat{\theta}^k$, where k indicated that k items have been administered. Then, Fisher information for the remaining items in the item bank were computed using their estimated parameters $\hat{a}$ and $\hat{b}$ with $\hat{\theta}^k$. The item with the maximum Fisher information would be selected as the next item. These two simulating processes were repeated until this examinee answered the predetermined number of items and then the test stopped. This simulation was conducted in R (R Core Team, 2021) using $mirtCAT$ package (Chalmers, 2016).

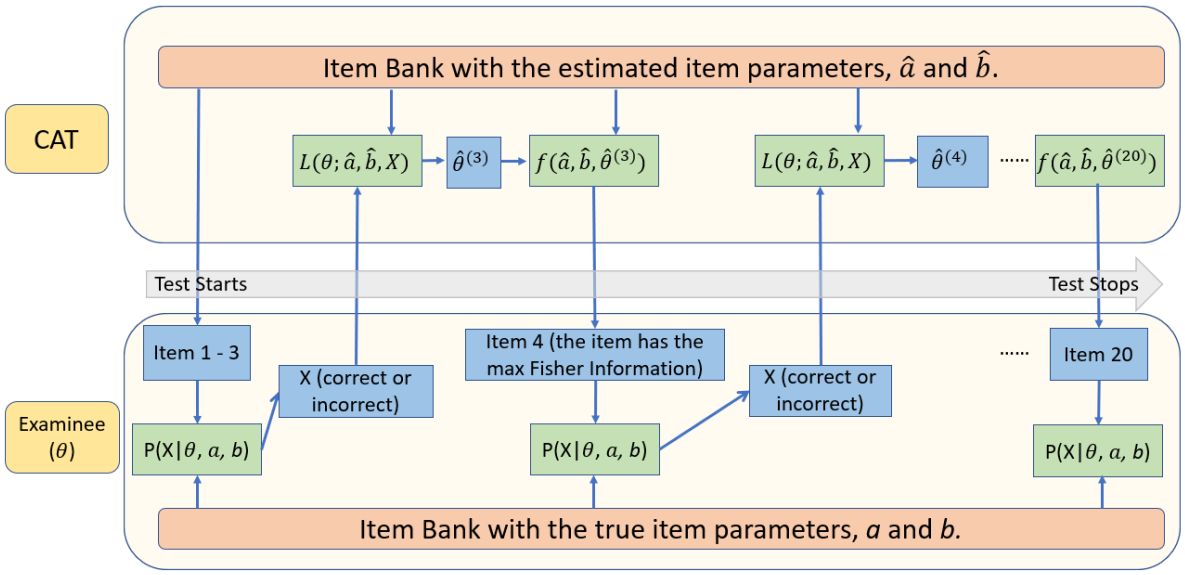


Figure 5.1: The Flow Chart for the CAT Simulation Process

Evaluation criteria

In this simulation, with item banks calibrated using 6 different methods in 2 incomplete calibration designs, 2 test lengths, and 2 populations, there were 48 conditions in total. To evaluate the accuracy of the estimations of examinees' abilities, mean deviation, mean squared of error (MSE), standard deviation (SD), relative absolute deviation, and correlation were calculated across replications as the evaluation criteria. Relative absolute deviation (RAD; Wang et al., 2020) is defined as the ratio of the absolute difference between the true and estimated value to the absolute true value. Smaller RAD, more accurate parameter estimations.

5.1.2 CAT Simulation Study Results

In this section, the simulation results of the ability estimations of CAT in 6 different calibrated item banks and 2 incomplete calibration designs were presented sequentially. The mean deviation, SD, MSE, RAD, and correlation of estimation of θ from 6 item banks calibrated in the targeted NEAT designs were presented first to identify the best item bank calibration strategies. The influence of test lengths and examinees' population was then investigated. The results of different item bank calibration strategies in random NEAT designs were presented next.

Results of using calibrated item banks in the targeted NEAT designs.

Figures 5.2 and 5.3 show the correlation plots between θ and $\hat{\theta}$ of the six calibration methods when test lengths were 40 and 20 respectively. In these two Figures, the y-axis was fixed in the range of -5 to 5 to better visualize the difference across methods. Generally, the variation of the estimated θ was larger when test length was 20 than 40. It was expected because more responses in a test would provide more information about the examinees' abilities. Thus, the results from 40-item tests were considered with heavier weights.

When test length was 40 (Figure 5.2), *TW* overestimated θ when true θ were less than -1 and slightly underestimated θ when true θ was larger than 1. The three proposed calibration strategies, *1bit(Switch)*, *SMC(1+Switch)*, and *SMC(pool+p_tw)* overestimated the small θ like *TW* but to a much lesser extent. In addition, *1bit(Switch)* and *SMC(pool+p_tw)* had $\hat{\theta}$ close to θ when θ were larger than 1, while *SMC(1+Switch)* overestimated θ when θ were large. Thus, *1bit(Switch)* and *SMC(pool+p_tw)* provided better estimations of θ than *TW* and *SMC(1+Switch)*, because their $\hat{\theta}$ had higher correlation for small and large θ. When test length was 20 (Figure 5.3), the correlation plot of each method was similar to that in 40-item tests, but the patterns were less obvious because of larger variations of $\hat{\theta}$.

In addition, because the y-axis of these two figures was fixed in the range of -5 to 5, the extreme estimations were not shown. Table 5.2 summarized the number of $\hat{\theta}$ with absolute deviation greater than

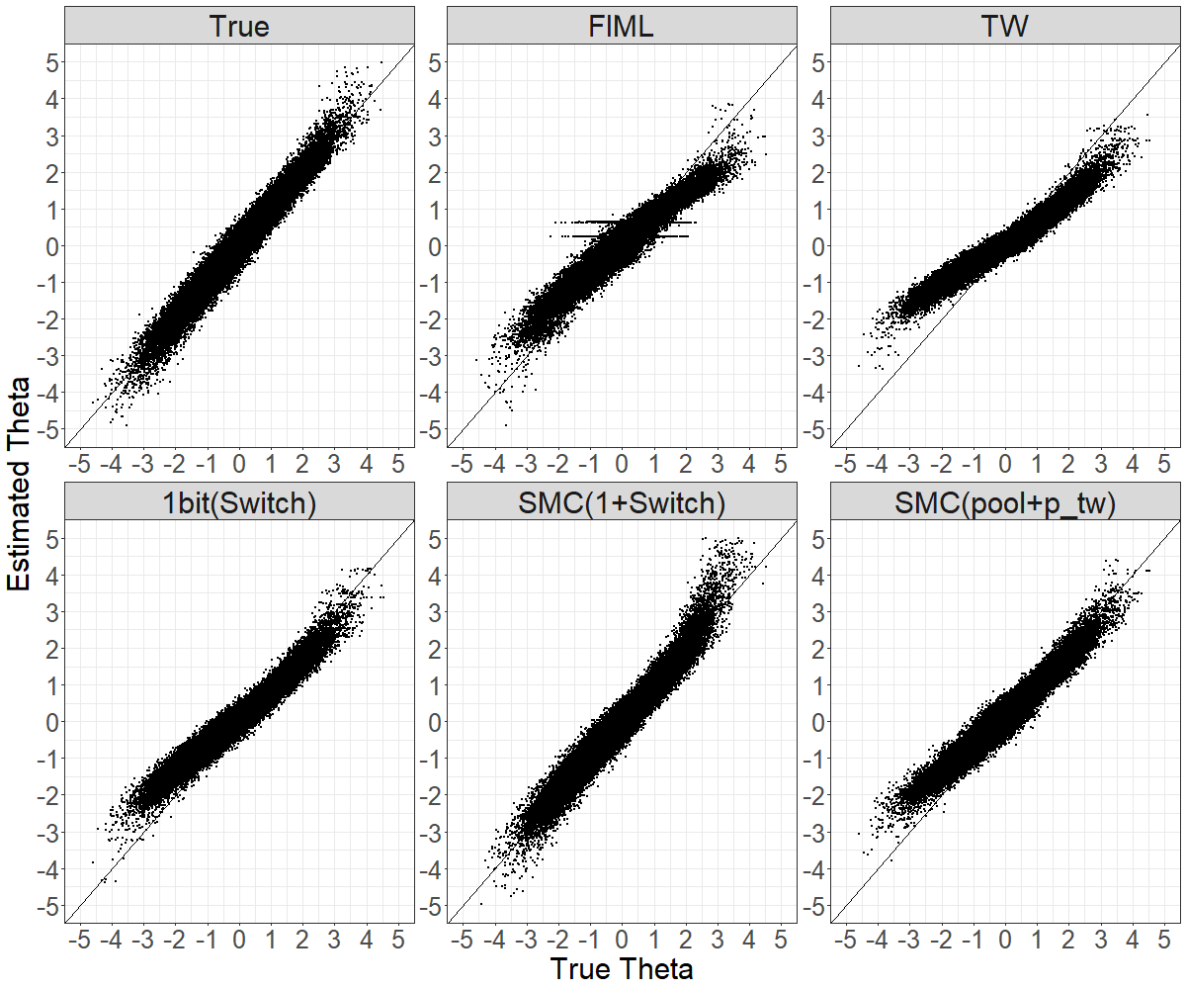


Figure 5.2: Correlation Plots in 40-Item Tests when $\theta \sim N(\mu, 1)$ Using Item Bank Calibrated From Targeted NEAT Designs.

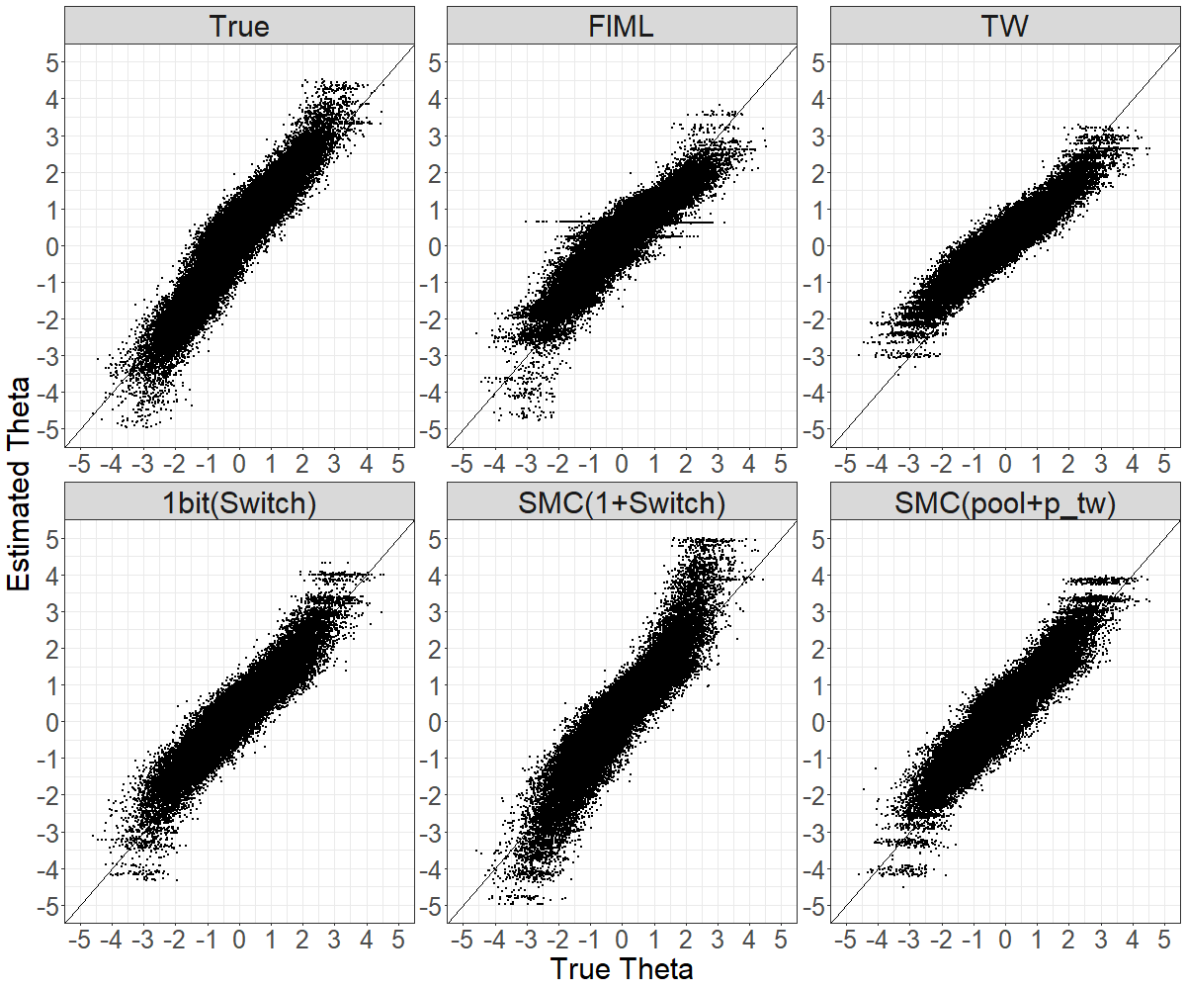


Figure 5.3: Correlation Plots in 20-Item Tests when $\theta \sim N(\mu, 1)$ Using Item Bank Calibrated From Targeted NEAT Designs.

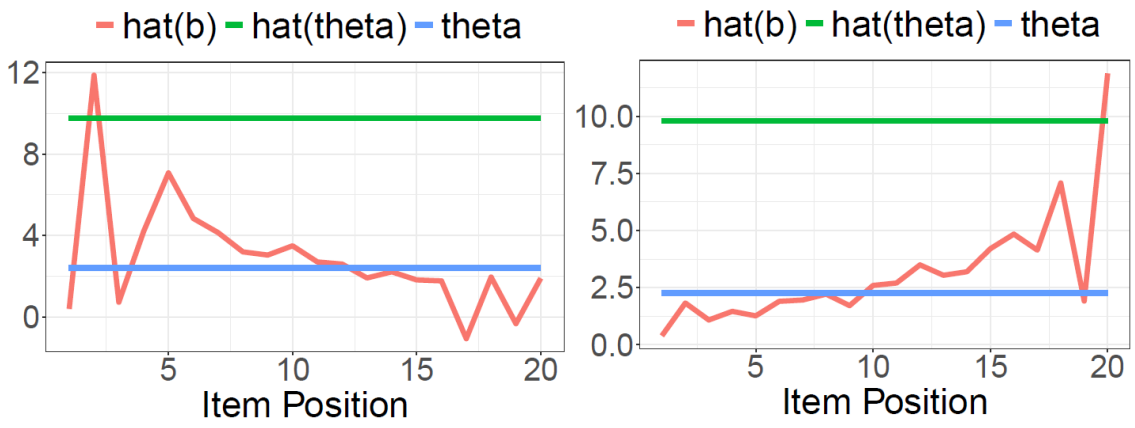
5 using the 6 methods. From this table, *TW*, *1bit(Switch)*, and *SMC(pool+p_tw)* did not have extremely estimated θ.

A worthy noting observation in Table 5.2 was that *SMC(1+Switch)* had a few extremely estimated θ in both 20-item and 40-item tests. To further investigate this observation, those extremely estimated θ were identified. Figure 5.4 presents the administered item difficulties versus item positions of two extreme cases when test length was 20. In the case 1, the 2^{nd} item was very difficult, though the item difficulties were approaching to the true θ at the end of the test. Remember that the first three items were randomly selected, thus this was unavoidable when there were items with extreme item parameters in the item bank. On the contrary, in the case 2, the last item was very difficult. Note that the true θ was about 2.5, which was relatively high. This examinee may answer most of the items correctly and had higher interim θ estimation than their true θ. This indicated that 20 items were not enough to provide an accurate θ estimations for some examinees. In addition, Figure 5.5 presents the administered item difficulties versus item positions when test length was 40. In both cases, one or two items with large b values were selected in the middle of the test, and b were approaching to θ towards the end of the tests. The final estimated θ were still much larger than the true θ because of the extremely estimated item difficulty parameters from *SMC(1+Switch)*. Note that examinees' ability may be less extreme if using EAP with an appropriate prior distribution. However, the ability estimation methods of CAT were not the focus of this study, and thus they were not further investigated.

Table 5.2: Number of $\hat{\theta}$ with Absolute Deviation Larger Than 5 Using Item Bank Calibrated from Targeted NEAT Designs

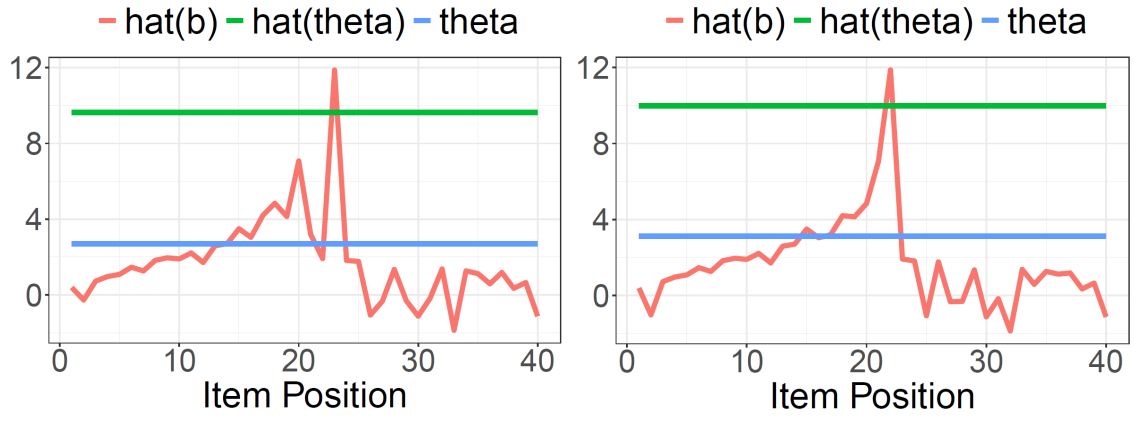
	True	FIML	TW	1bit(Switch)	SMC(1+Switch)	SMC(pool+p_tw)
20-Item Test	0	12	0	0	8	0
40-item Test	3	1	0	0	11	0

In addition to the visualizations of the correlation of θ and $\hat{\theta}$ across different calibration strategies, Tables 5.3 and 5.4 present the summary statistics of the estimation of θ in 40-item and 20-item tests given three θ ranges when θ were from $N(\mu, 1)$. The bold numbers indicated better results than *FIML* method.



(a) Case 1 (b) Case 2

Figure 5.4: Two Examples of Extreme Estimations of θ Using *SMC(1+Switch)* in Targeted NEAT Design with Test Length of 20



(a) Case 1 (b) Case 2

Figure 5.5: Two Examples of Extreme Estimations of θ Using *SMC(1+Switch)* in Targeted NEAT Design with Test Length of 40

As observed in the Figures 5.2 and 5.3, estimations of θ had larger SD in the 20-item tests than 40-item tests. Thus, more weights were put on results of 40-item tests.

In 40-item tests (Table 5.3), when θ was between -1 and 1, all imputation methods provided smaller MSE, RAD, and higher correlation than *FIML*. In particular, *SMC(1+Switch)* had the smallest MSE and highest correlation. When θ were less than -1, *SMC(1+Switch)* had the smallest deviation, MSE and RAD among all the imputation methods, and it had relatively high correlation. When θ were larger than 1, *1bit(Swtich)*, *SMC(1+Switch)*, and *SMC(pool+p_tw)* had smaller deviation, MSE, RAD, and higher correlation than *FIML*. Though *TW* had relatively high correlation, it had larger deviation, SD, MSE, and RAD than *FIML*. Given the above-mentioned evaluation criterion, *1bit(Switch)*, *SMC(1+Switch)*, and *SMC(pool+p_tw)* had the better estimations for θ than *TW* and *FIML*. In particular, *SMC(1+Switch)* had the most accurate $\hat{\theta}$ when θ were less than 1, and *SMC(pool+p_tw)* had the most accurate $\hat{\theta}$ when θ were larger than 1. When compared to the best scenario when using the true item parameters, their estimations were still worse, but they improved estimations for θ from the baselines.

The summary statistics across methods were slightly different in 20-item tests (Table 5.4) than 40-item tests. In particular, *1bit(Switch)* was the only method that had smaller MSE, RAD and larger correlation then *FIML* among four imputation methods in all three θ ranges. The other three imputation methods provided better θ estimations than *FIML* only in certain ranges of θ. In particular, *SMC(1+Switch)* had better results when θ were smaller than 1, and *SMC(pool+p_tw)* had better results than *FIML* when θ were less than -1 and larger than 1. In addition, *TW* only had better results than *FIML* when θ were in the range of -1 to 1. When compared to the best scenario when using true item parameters, the imputation method even had better results in some conditions. For example, when θ were in the range of -1 and 1, *1bit(Switch)* had smaller deviation, SD, MSE, and RAD, and the same correlation with results of using true item parameters. In addition, when θ were larger than 1, *TW* also had smaller slightly smaller SD, RAD, and higher correlation than using true item parameters.

Table 5.3: Summary Statistics of $\hat{\theta}$ when $\theta \sim N(\mu, 1)$ using Item Parameters Calibrated in Targeted NEAT Design with Test Length of 40

θ Ranges	Method	Deviation	SD	MSE	RAD	Correlation
$\theta \leq -1$	True	-0.013	0.638	0.089	0.149	0.883
	FIML	0.456	0.529	0.301	0.287	0.840
	TW	0.808	0.370	0.739	0.486	0.871
	1bit(Switch)	**0.564**	0.465	**0.393**	**0.343**	0.868
	SMC(1+Switch)	**0.104**	0.641	**0.111**	**0.164**	0.870
	SMC(pool+p_tw)	**0.535**	0.440	**0.369**	**0.321**	0.857
$-1 < \theta \leq 1$	True	-0.005	0.609	0.063	2.163	0.911
	FIML	0.050	0.492	0.098	3.332	0.831
	TW	**0.014**	**0.323**	**0.094**	**1.971**	**0.887**
	1bit(Switch)	0.065	**0.436**	**0.066**	**1.838**	**0.900**
	SMC(1+Switch)	**-0.004**	0.553	**0.058**	**2.465**	**0.905**
	SMC(pool+p_tw)	0.129	0.511	**0.079**	**2.747**	**0.893**
$\theta > 1$	True	0.011	0.626	0.067	0.127	0.910
	FIML	-0.494	0.442	0.370	0.292	0.828
	TW	-0.618	0.457	0.441	0.377	0.902
	1bit(Switch)	**-0.319**	0.546	**0.159**	**0.210**	**0.905**
	SMC(1+Switch)	**-0.022**	0.827	**0.195**	**0.162**	0.878
	SMC(pool+p_tw)	**-0.176**	0.545	**0.091**	**0.145**	**0.901**

Note. The bold numbers indicated better θ estimation from the imputation methods than *TW* method.

Table 5.4: Summary Statistics of $\hat{\theta}$ when $\theta \sim N(\mu, 1)$ using Item Parameters Calibrated in Targeted NEAT Design with Test Length of 20

θ Ranges	Method	Deviation	SD	MSE	RAD	Correlation
$\theta \leq -1$	True	-0.072	0.828	0.294	0.269	0.768
	FIML	0.464	0.757	0.502	0.364	0.708
	TW	0.665	**0.559**	0.565	0.422	**0.801**
	1bit(Switch)	0.464	**0.701**	**0.419**	**0.343**	**0.767**
	SMC(1+Switch)	**0.051**	0.974	**0.454**	**0.329**	**0.746**
	SMC(pool+p_tw)	0.466	**0.707**	**0.435**	**0.345**	**0.750**
$-1 \leq \theta < 1$	True	0.328	0.735	0.303	5.634	0.800
	FIML	0.267	0.575	0.254	5.362	0.723
	TW	**0.183**	**0.441**	**0.147**	**3.029**	**0.794**
	1bit(Switch)	**0.283**	**0.562**	**0.205**	**4.779**	**0.800**
	SMC(1+Switch)	0.279	0.668	**0.247**	**5.212**	**0.789**
	SMC(pool+p_tw)	0.338	0.620	0.260	5.382	**0.794**
$\theta > 1$	True	0.272	0.698	0.252	0.254	0.796
	FIML	-0.294	0.622	0.472	0.235	0.657
	TW	-0.378	**0.516**	**0.255**	0.248	**0.808**
	1bit(Switch)	**-0.014**	0.693	**0.186**	**0.208**	**0.782**
	SMC(1+Switch)	0.349	1.104	0.731	0.323	**0.753**
	SMC(pool+p_tw)	**0.093**	0.686	**0.203**	**0.216**	**0.768**

Note. The bold numbers indicated better θ estimation from the imputation methods than *FIML* method.

Influence from the examinees' population

When examinees were from the standard normal distribution, the estimations of θ were similar to those when examinees were from the mixture normal distribution. Figures 5.6 and 5.7 show the correlation between θ and $\hat{\theta}$ when test lengths were 40 and 20 respectively. Though examinees' abilities were in a smaller range when simulated from $N(0,1)$ than from the mixture normal distribution, *TW* still overestimated θ when they were less than -1 and underestimated θ when they were larger than 1, similar to the patterns shown when $\theta \sim N(\mu, 1)$. When compared to *TW*, *1bit(Switch)* and *SMC(pool+p_tw)* had less overestimations when θ were less than -1, and they did not show underestimations when θ were larger than 1. *SMC(1+Switch)* also showed better correlation than *TW* except that it had a few overestimation when θ were larger than 1. These observations were also shown when $\theta \sim N(\mu, 1)$. In addition, their summary statistics for 40-item tests and 20-item tests were presented in Tables D.28 and D.29 in the Appendix. To summarize, the examinees' population did not have much influence on the estimations of ability parameters.

Results of using calibrated item bank in the random NEAT designs.

Figures 5.8 and 5.9 present the correlation between θ and $\hat{\theta}$ of the six calibration methods in random NEAT designs when test lengths were 20 and 40 respectively. Similar to the results of the item bank calibrated using targeted NEAT designs, the variation of the estimated θ was larger in 20-item tests than 40-item tests. When test length was 40 (Figure 5.8), *TW* slightly overestimated θ that were less than -2, and it slightly underestimated θ that were larger than 2. *1bit(Switch)* and *SMC(1+Switch)* slightly overestimated the large θ, and they slightly underestimated the small θ. In addition, *SMC(pool+p_tw)* slightly overestimated θ across all θ as the the correlation dots were slightly above the diagonal line. The estimated θ showed similar patterns in the 20-item tests with those in the 40-item tests. In addition, because the y-axis of Figures 5.9 and 5.8 was fixed in the range of -5 to 5, the extreme estimations were not presented. Table 5.5 summarized the number of $\hat{\theta}$ with absolute values greater than 5 for the six methods of the two test lengths. From this table, *FMIL* had extremely estimated θ in both test lengths, and *SMC(1+Switch)* had three extreme estimations in 20-item tests.

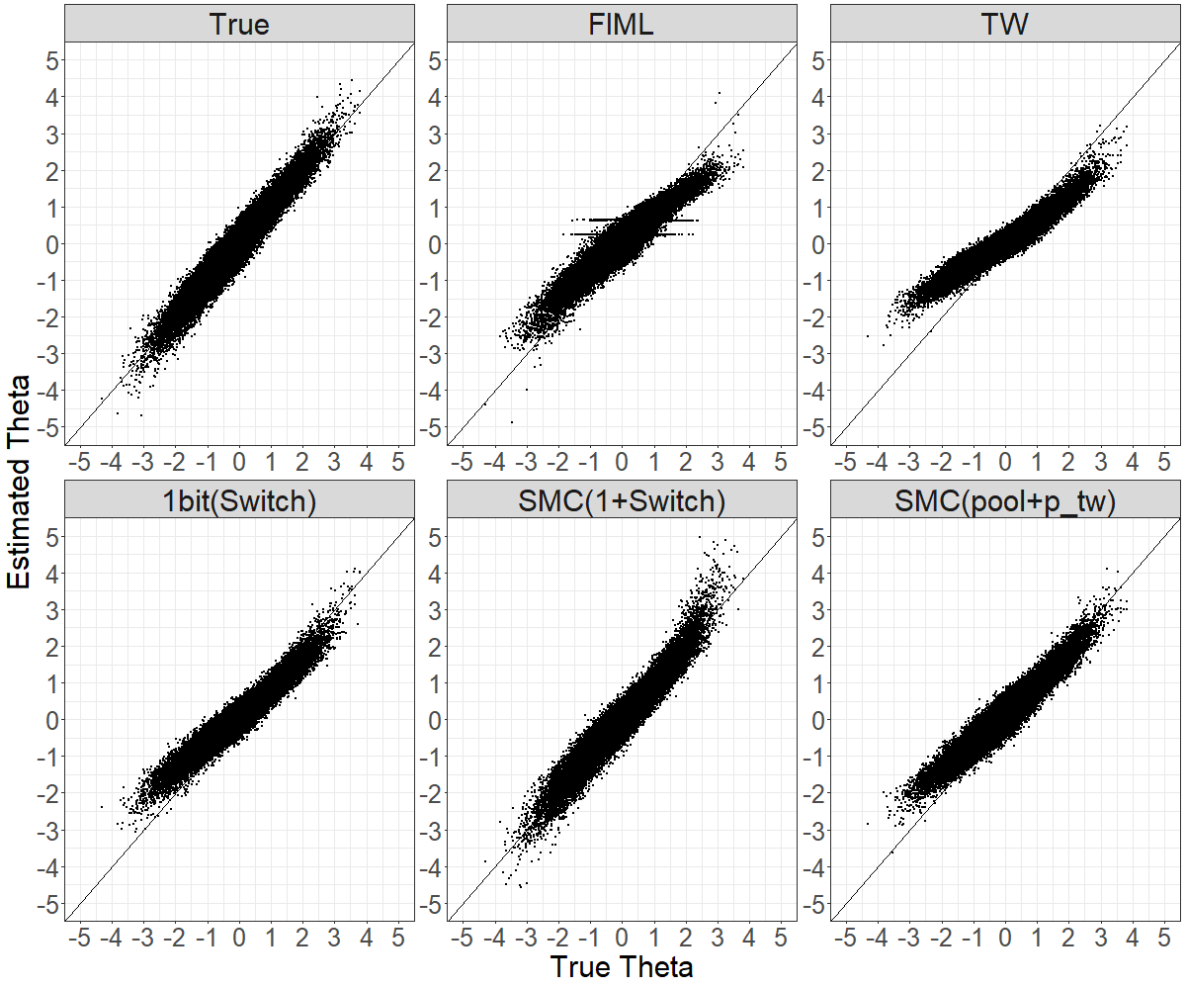


Figure 5.6: Correlation Plots in 40-Item Tests when $\theta \sim N(0,1)$ Using Item Bank Calibrated From Targeted NEAT Designs.

Table 5.5: Number of $\hat{\theta}$ with Deviation larger than 5 Using Item Bank Calibrated from Random NEAT Designs

	True	FIML	TW	1bit(Switch)	SMC(1+Switch)	SMC(pool+p_tw)
20-Item Test	0	122	0	0	3	0
40-item Test	0	44	0	0	0	0

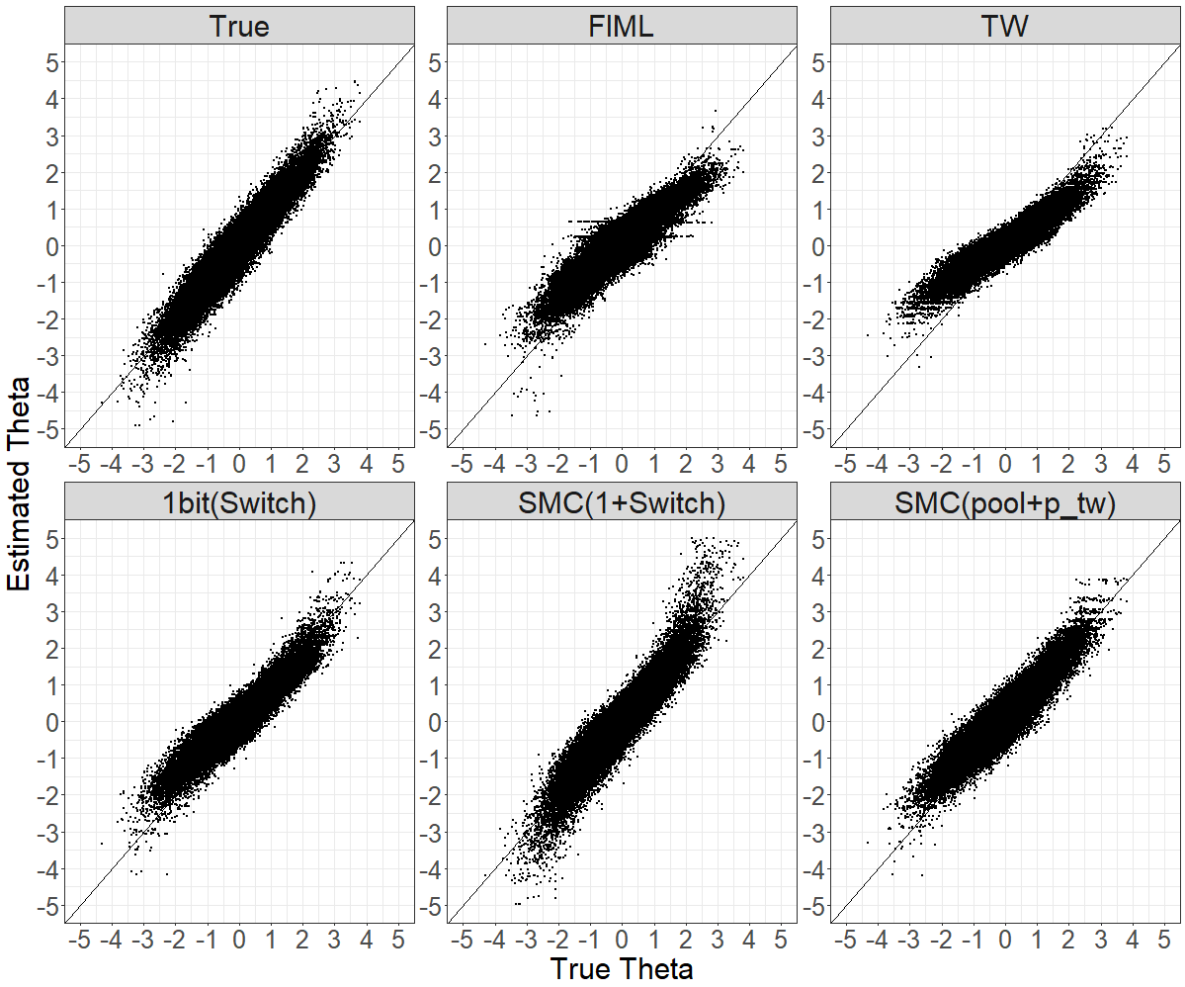


Figure 5.7: Correlation Plots in 20-Item Tests when $\theta \sim N(0,1)$ Using Item Bank Calibrated From Targeted NEAT Designs.

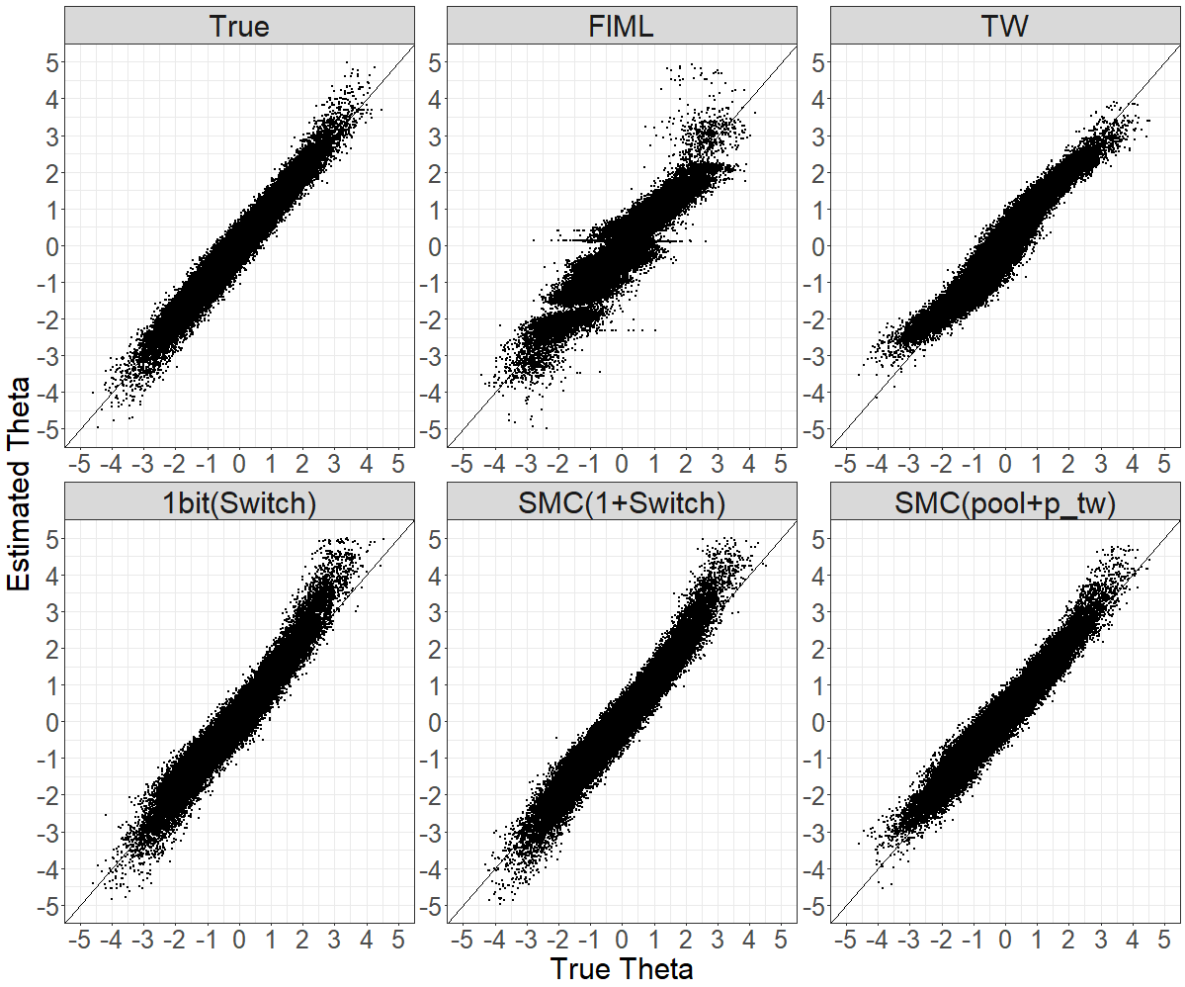


Figure 5.8: Correlation Plots in 40-Item Tests when $\theta \sim N(\mu, 1)$ Using Item Bank Calibrated From Random NEAT Designs.

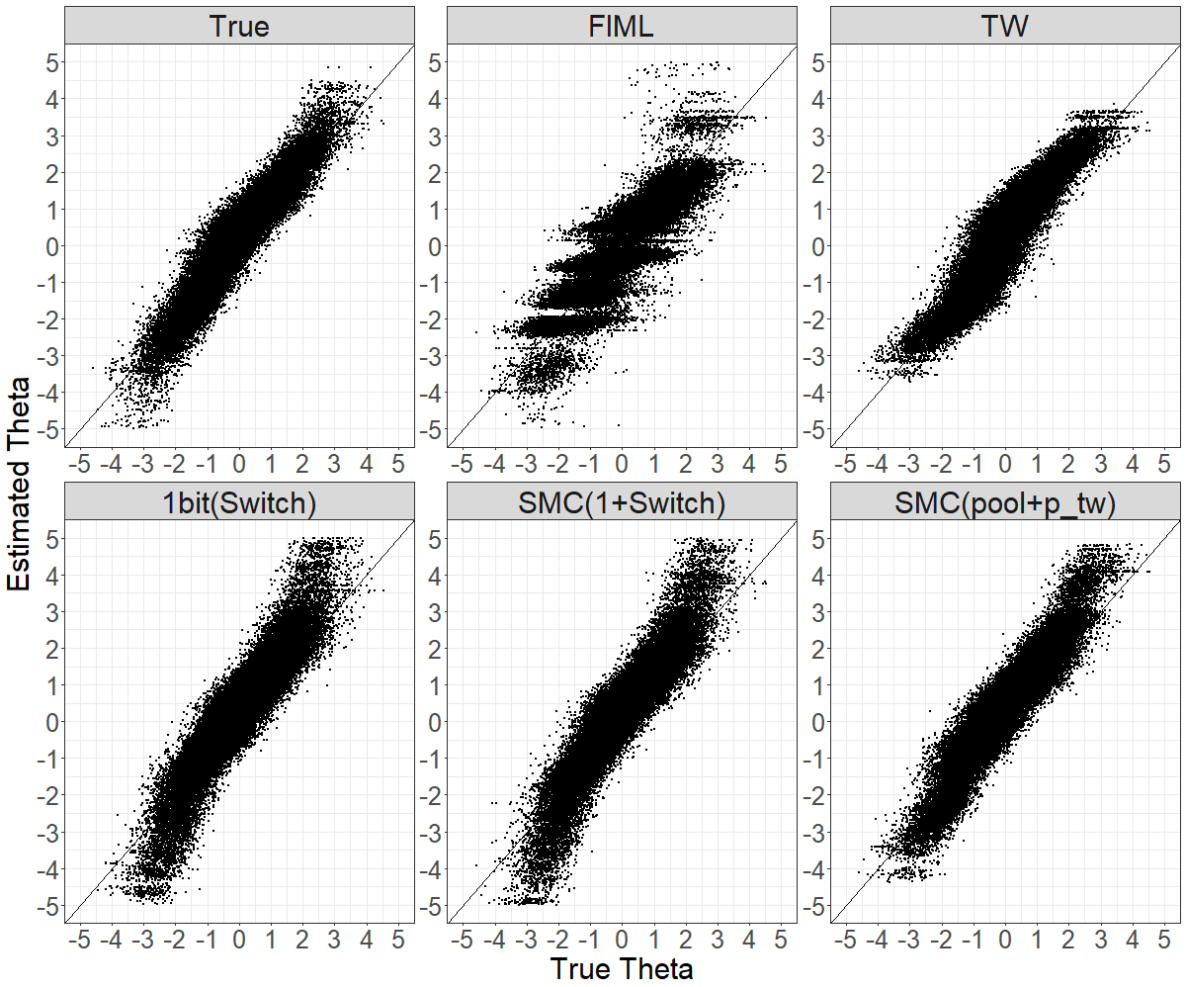
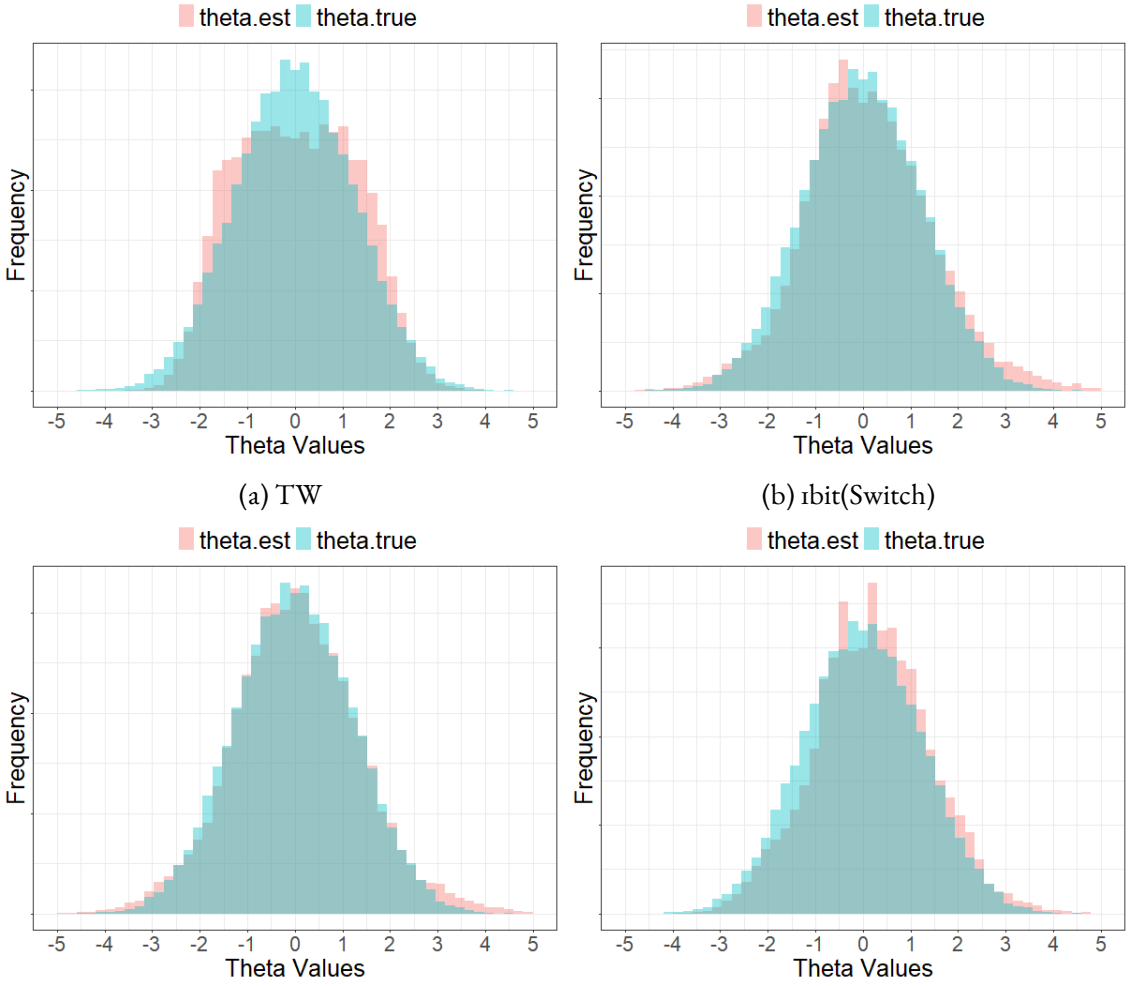


Figure 5.9: Correlation Plots in 20-Item Tests when $\theta \sim N(\mu, 1)$ Using Item Bank Calibrated From Random NEAT Designs.

In addition, Figure 5.10 shows the histogram of the estimated θ versus the true θ in 40-item tests using *TW* and three proposed item bank calibration strategies. The histogram of $\hat{\theta}$ from *TW* was closer to a uniform distribution rather than a normal distribution. On the contrary, the histograms of *1bit(Switch)* and *SMC(1+Switch)* had $\hat{\theta}$ mostly aligned with true θ. In addition, *SMC(pool+p_tw)* slightly overestimated θ across all θ, which was also observed in the correlation plots. These observations were also shown in the 20-item tests (see Figure C.39 in the Appendix).



(a) TW

(b) 1bit(Switch)

(c) SMC(1+Switch)

(d) SMC(pool+p_tw)

Figure 5.10: Histogram of Estimated θ Versus True θ with Test Length of 40 When $\theta \sim N(\mu, 1)$ Using Item Bank Calibrated in Random NEAT Designs

In addition to the visualizations, Table 5.6 presents the summary statistics of the estimation of θ given θ ranges in 40-item tests when θ were generated from $N(\mu, 1)$. The bold numbers indicated better results

Table 5.6: Summary Statistics of $\hat{\theta}$ when $\theta \sim N(\mu, 1)$ using Item Parameters Calibrated in Random NEAT Design with Test Length of 40

θ Ranges	Method	Deviation	SD	MSE	RAD	Correlation
$\theta \leq -1$	True	-0.010	0.638	0.091	0.150	0.881
	FIML	-0.026	0.746	0.244	0.219	0.760
	TW	0.077	**0.439**	**0.096**	**0.148**	**0.840**
	1bit(Switch)	0.117	**0.699**	**0.144**	**0.189**	**0.859**
	SMC(1+Switch)	**-0.003**	**0.715**	**0.134**	**0.176**	**0.864**
	SMC(pool+p_tw)	0.294	**0.619**	**0.189**	0.232	**0.855**
$-1 \leq \theta < 1$	True	-0.005	0.610	0.062	2.085	0.912
	FIML	-0.103	0.635	0.163	4.199	0.793
	TW	**0.025**	0.708	**0.106**	**3.211**	**0.896**
	1bit(Switch)	**0.081**	**0.619**	**0.080**	**2.651**	**0.899**
	SMC(1+Switch)	**0.023**	**0.619**	**0.071**	**2.657**	**0.903**
	SMC(pool+p_tw)	0.189	**0.578**	**0.105**	**3.397**	**0.893**
$\theta > 1$	True	0.012	0.628	0.070	0.128	0.908
	FIML	-0.228	0.978	0.607	0.261	0.680
	TW	**0.034**	**0.473**	**0.066**	**0.130**	**0.891**
	1bit(Switch)	0.295	**0.817**	**0.248**	**0.215**	**0.898**
	SMC(1+Switch)	**0.177**	**0.798**	**0.178**	**0.175**	**0.901**
	SMC(pool+p_tw)	**0.152**	**0.651**	**0.106**	**0.159**	**0.898**

Note. The bold numbers indicated better θ estimation from the imputation methods than *FIML* method.

than *FIML* method. In 40-item tests, *1bit(Switch)*, *SMC(1+Switch)*, and *SMC(pool+p_tw)* had smaller SD, MSE, RAD, and higher correlation than *FIML* across three ranges of θ, which was the same as their results in the targeted NEAT designs. In addition, *TW* had smaller MSE, RAD, and higher correlation than *FIML* when θ were either small or large, which was different from its results in the targeted NEAT designs. This finding was consistent from the item parameter estimations shown in Study 1, where *TW* had higher recovery rates and more accurate item parameter estimations in the random NEAT designs than in the targeted NEAT designs. When comparing the proposed item bank calibration strategies to *TW*, *1bit(Switch)* and *SMC(1+Switch)* had smaller deviation, SD, MSE, RAD, and higher correlation than *TW* when θ were in the range of -1 and 1. When θ were less than -1 or larger than 1, *TW* had smaller MSE and RAD than three proposed strategies but had smaller correlation. These observations were also shown in 20-item tests (see Table D.30 in the Appendix).

Influence from examinees' population

When examinees were from the standard normal distribution, the results of θ estimations were similar to those when examinees were from the mixture normal distribution. Firstly, the correlation plots when $\theta \sim N(0,1)$ (Figures 5.11 and 5.12) were similar to the correlation plots when $\theta \sim N(\mu,1)$ (Figures 5.8 and 5.9) in both 20- and 40-item tests. *FIML* had 17 extreme θ estimations in both 20- and 40-item tests, which were not shown in these two figures because the y-axis was fixed in the ranges of -5 to 5. There were not extreme estimations of θ using all four imputation methods. Secondly, similar to the results with $\theta \sim N(\mu,1)$, *TW* produced $\hat{\theta}$ that showed a pattern of uniform distribution rather than normal distribution with $\theta \sim N(0,1)$. Figure 5.13 presents the histograms of $\hat{\theta}$ versus θ from item banks calibrated using 6 methods. This pattern was also shown in 20-item tests (Figure C.40). Lastly, the summary statistics of the estimated θ also showed similar patterns when $\theta \sim N(0,1)$ and $\theta \sim N(\mu,1)$ in both 20-item and 40-item tests (Tables D.31 and D.32 in the Appendix).

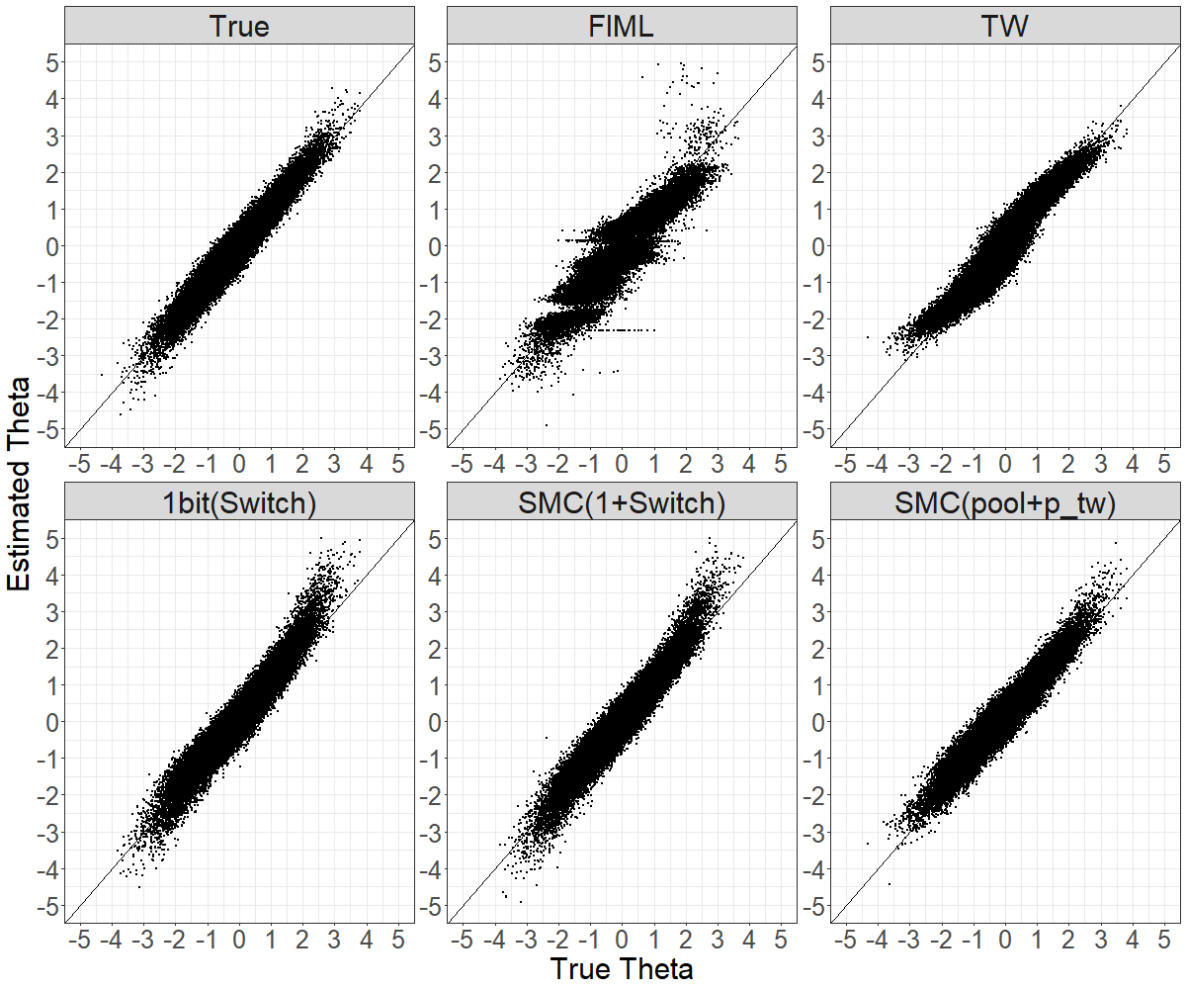


Figure 5.11: Correlation Plots in 40-Item Tests when $\theta \sim N(0, 1)$ Using Item Bank Calibrated From Random NEAT Designs.

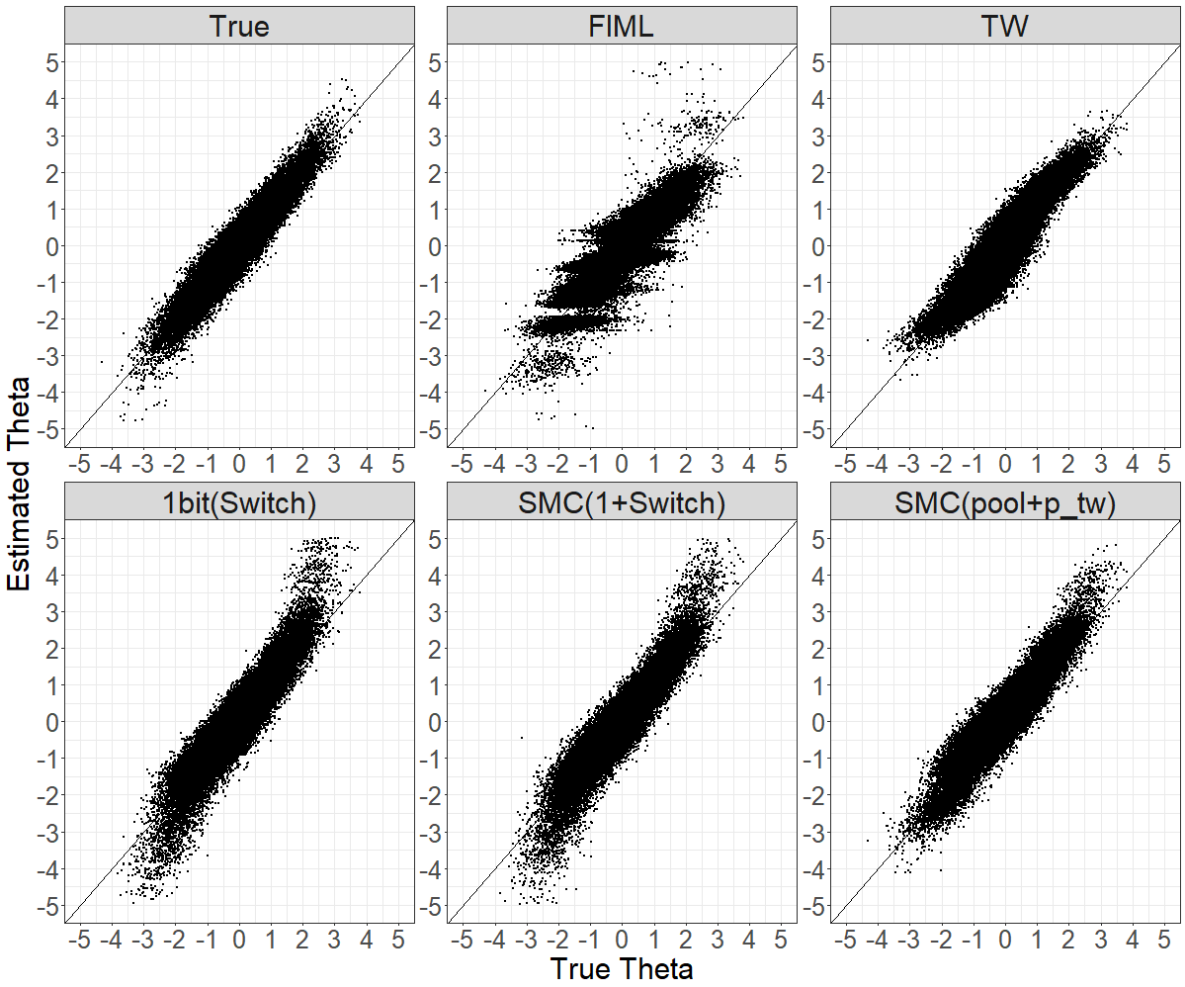
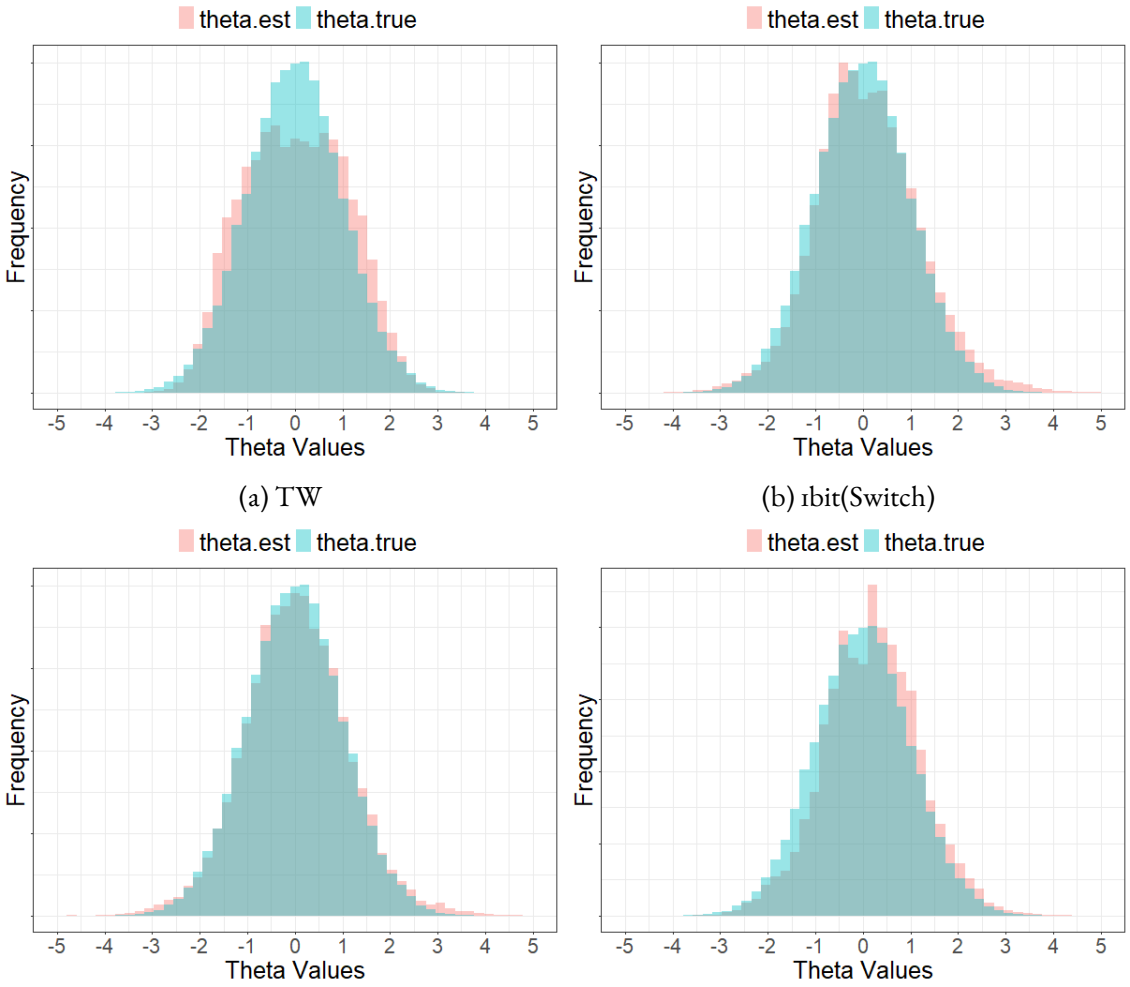


Figure 5.12: Correlation Plots in 20-Item Tests when $\theta \sim N(0,1)$ Using Item Bank Calibrated From Random NEAT Designs.



(a) TW

(b) 1bit(Switch)

(c) SMC(1+Switch)

(d) SMC(pool+p_tw)

Figure 5.13: Histogram of Estimated θ Versus True θ with Test Length of 40 When $\theta \sim N(0, 1)$ Using Item Bank Calibration From Random NEAT Designs

5.1.3 Summary of CAT simulation study.

From the CAT simulation, we noticed that 20-item test was too short to provide a stable estimation for θ. Thus, the main conclusion was drawn from the results of 40-item tests. The estimated examinees' abilities were more accurate if using the item bank that was calibrated with imputation methods than *FIML*. In addition, the proposed item bank calibration strategies, *1bit(Switch)*, *SMC(1+Switch)*, and *SMC(pool+p_tw)* led to better estimations of θ than *TW*, especially when calibrated in the targeted NEAT designs. In particular, *SMC(1+Switch)* provided best ability estimations when θ were less than 1 but slightly overestimated θ when θ were larger than 1. *SMC(pool+p_tw)* and *1bit(Switch)* had better ability estimations when θ were larger than 1 but slightly underestimated θ when θ were less than 1. Lastly, the ability estimations when $\theta \sim N(0, 1)$ were similar to $\theta \sim N(\mu, 1)$, which indicated that the imputation methods were relatively robust to the calibration population.

5.2 Real Data Demonstration

The previous studies were all based on simulations. In this study, the developed item calibration strategies from previous sections were applied to a real data set from small-scale assessments. This data set was collected from a recent study to develop an item bank for CAT for a graduate statistic course. A total of 148 multiple choice questions were developed and needed to be calibrated before using them to deliver a CAT. Using the 148 items, six test forms were created with a set of anchor items within each of them. These forms were distributed to six graduate statistics courses based on instructors' prior knowledge about the students' background. Thus, the calibration design was similar to the 5-form targeted NEAT designs in the previous simulation studies but had 6 forms. However, as requested by each instructor, each test form was adjusted to fit the goal and teaching status of the classroom. Thus, some items were removed, while some items were added to each test form. In this way, the calibration design did not follow the targeted NEAT designs exactly. The response matrix given responded or missed was plotted in Figure 5.14. In addition, the test length of each form and the sample size of each form were summarized in Table

5.7. The total sample size across the six statistics courses was 166, which was more than 125, the smallest sample size investigated in the previous simulation studies. The overall average missing percentage was about 55.4%, which was slightly lower than that in the previous simulation studies. The distribution of the missing percentages over all the items was documented in Figure 5.15. Note that a few items had missing rate close to 1.

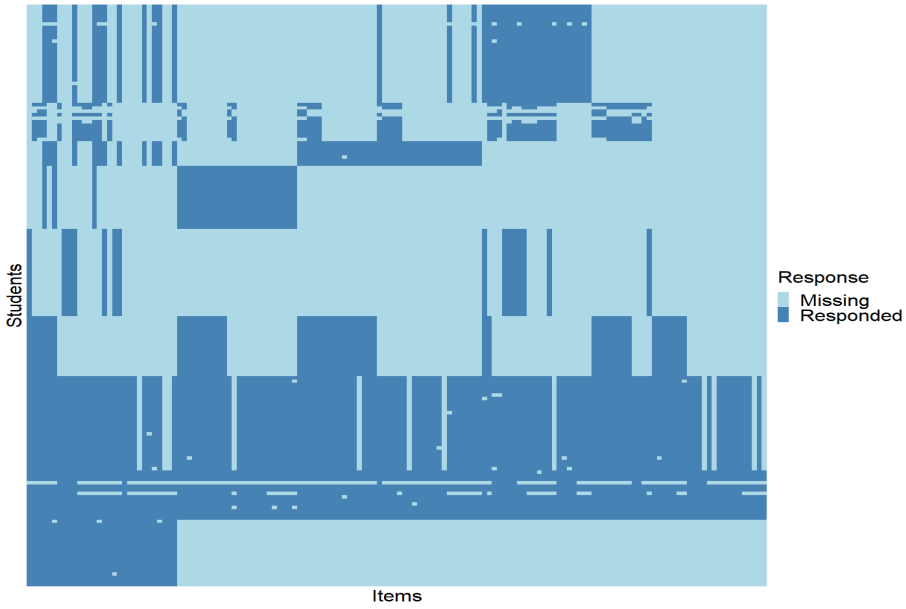


Figure 5.14: Response Patterns

Table 5.7: Test Length and Sample Size for Each Test Form

Forms	ERSH6300	ERSH8310	ERSH8320	ERSH8350	STAT8200	Dr. Fan
Test length	30	148*	27	49	37	50
Sample Size	19	83	18	7	28	11

Note. The data for ERSH8310 was collected from multiple semesters, and the test lengths varied a little across semesters.

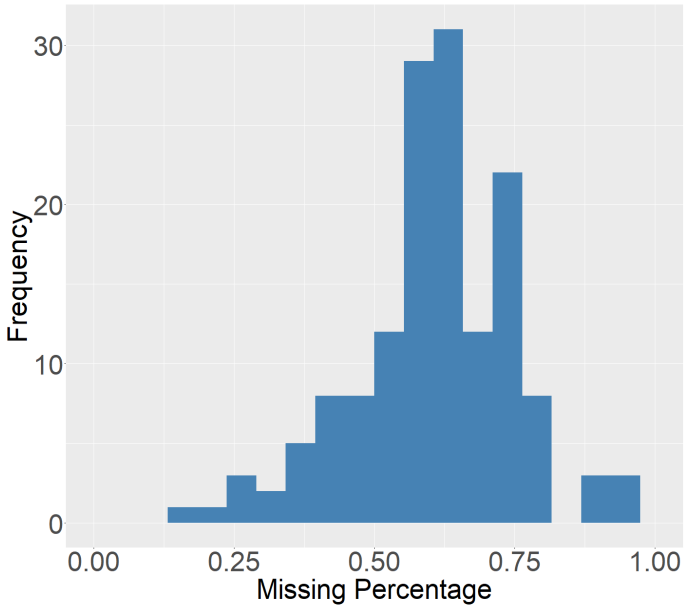


Figure 5.15: Distribution of Missing Percentages

5.2.1 Method

Among the proposed methods in the previous section, *1bit(p_tw)* and *1bit(Switch)* with the general set of switching parameters was applied to this data set. The structure matrix completion was not used in this study. As introduced above, the collected data set did not have an exactly structured missing pattern because of the request from each instructor and the skipped responses from students. Thus, it was difficult to manipulate this real data set to fit in the required missing pattern of structured matrix completion. In addition, *FIML* and *TW* was also used in this study. *FIML* was selected because it did not require imputation and was a common practice of calibrating item parameters with missing responses. *TW* was selected because it recovered the incomplete response matrix with relatively high recovery rates and relatively accurate item parameter estimations.

Because the true item parameters were unknown in this real data set, it was hard to directly evaluate the results of *1bit(p_tw)*, *1bit(Switch)*, *FIML*, and *TW*. On the contrary, the similarities of the results across these four methods were examined. In particular, two evaluation criteria were calculated in this study. The first was the proportion of the same imputed responses across three methods, and the second was the

correlation between the estimated item parameters. Since *FIML* did not impute the missing responses, it was not evaluated by the proportion of the same imputed responses.

5.2.2 Results

Proportion of the same imputed responses across three methods.

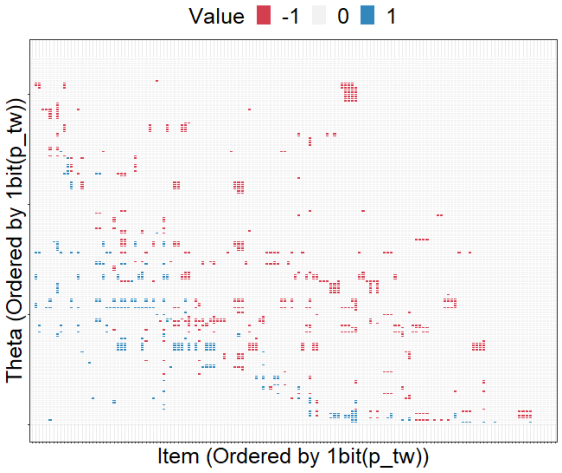
Table 5.8 presented the proportion of the same imputed responses across three imputation methods. The imputed response matrices between *1bit(p_tw)* and *1bit(Switch)* were very similar with 96.4% of the same imputed responses. It was expected because *1bit(Switch)* was based on *1bit(p_tw)* and changed a few responses in the response matrix of *1bit(p_tw)*. In this case, only 3.6% of responses were changed. Their difference was plotted in Figure 5.16a. Thus, *1bit(Switch)* changed a few correct responses of large item scores and person scores given the imputed response matrix of *1bit(p_tw)* to incorrect, and it changed a few incorrect responses of small item scores and small person scores to correct. The proportions of the same imputed responses between *1bit(p_tw)* and *TW* and between *1bit(Switch)* and *TW* were lower but still larger than 0.850. In addition, the difference of the heat maps were plotted in Figure 5.16b and 5.16c. From both heat maps, *1bit(p_tw)* and *1bit(Switch)* imputed more correct responses in the diagonal where item difficulties and examinees' abilities were similar than *TW*.

Table 5.8: Proportion of the Same Imputed Responses

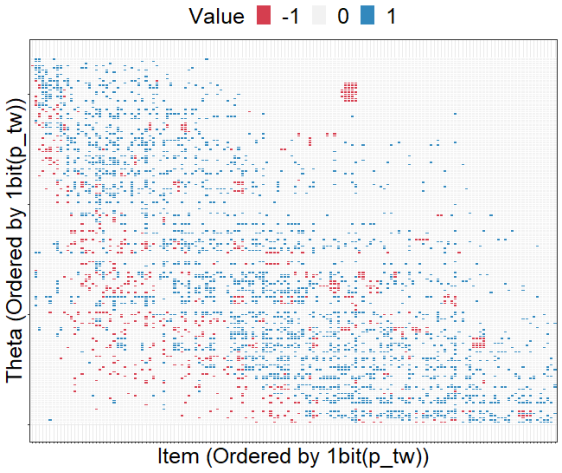
	TW	1bit(p_tw)	1bit(Switch)
TW	1.000	0.885	0.872
1bit(p_tw)		1.000	0.964
1bit(Switch)			1.000

Agreement of the item discrimination parameter estimations.

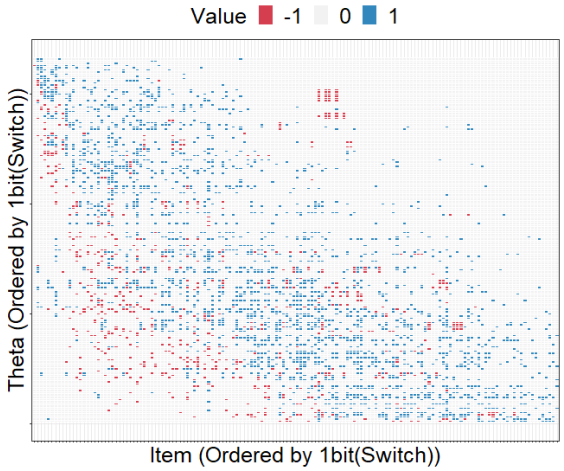
The correlation of estimated item discrimination parameters among *FIML*, *1bit(p_tw)*, *1bit(Switch)*, and *TW* were calculated (Table 5.9). Firstly, the correlations of estimated item discrimination parameters between *1bit(p_tw)* and *1bit(Switch)* were much greater than other correlations. Figure 5.17a presents the correlation of $\hat{a}$ between *1bit(p_tw)* and *1bit(Switch)*. Generally, $\hat{a}$ of *1bit(p_tw)* and *1bit(Switch)* were quite



(a) *1bit(p_tw)* Versus *1bit(Switch)*



(b) *1bit(p_tw)* Versus *TW*



(c) *1bit(Switch)* Versus *TW*

Figure 5.16: The Difference Between the Imputed Response Matrices

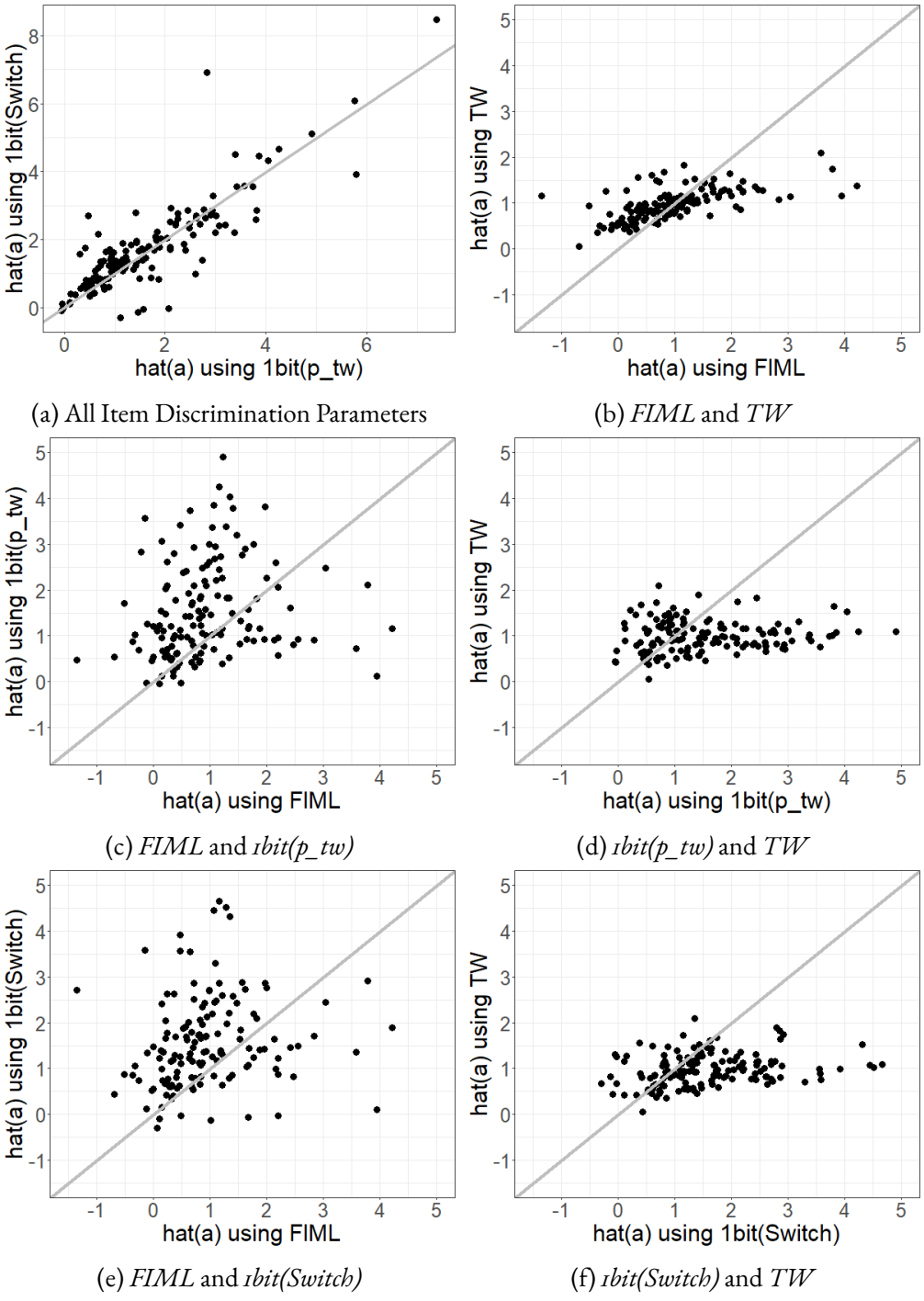
similar, and they had large $\hat{a}$ for same items ($\hat{(a)}$ larger than 5). However, *1bit(Switch)* had four items with $\hat{a}$ much closer to zero than *1bit(p_tw)*. It was expected because *1bit(Switch)* purposely switched a few selected responses to mitigate the extreme response patterns and to release the overestimations of a from *1bit(p_tw)*.

Table 5.9: Correlation of Estimated Item Discrimination Parameters

	All Estimated Item Parameters			
Cor(a)	FIML	TW	1bit(p_tw)	1bit(Switch)
FIML	1.000	0.188	-0.065	-0.034
TW		1.000	0.126	0.239
1bit(p_tw)			1.000	0.851
1bit(Switch)				1.000
	Estimated Item Parameter ≤ 5			
Cor(a)	FIML	TW	1bit(p_tw)	1bit(Switch)
FIML	1.000	0.639	0.161	0.170
TW		1.000	0.158	0.260
1bit(p_tw)			1.000	0.833
1bit(Switch)				1.000

Secondly, from Table 5.9, the correlation between *FIML* and *TW* increased from 0.188 to 0.639 after excluding $\hat{a}$ that were larger than 5. Figure 5.17b presents their correlation after excluding extremely estimated item discrimination parameters. *TW* had $\hat{a}$ in much smaller range than *FIML*. In previous simulation studies, *TW* also showed this characteristic. Note that *FIML* had three $\hat{a}$ that were larger than 10 and thus not shown in this figure.

Lastly, after excluding the extremely estimated a, the correlation of *1bit(p_tw)* with *TW* and *FIML* were still smaller than 0.3. Though $\hat{a}$ from both *FIML* and *1bit(p_tw)* had wide ranges, the items with large $\hat{a}$ using *FIML* were not the items with large a using *1bit(p_tw)* (Figure 5.17c). It explained the low correlation between $\hat{a}$ from these two methods. In addition, $\hat{a}$ from *TW* were close to 1 for most of items, while $\hat{a}$ from *1bit(p_tw)* were in the range of 0 to 5 (Figure 5.17d). These findings of *1bit(p_tw)* were also shown if using *1bit(Switch)* (Figures 5.17e and 5.17f).



(a) All Item Discrimination Parameters

(b) *FIML* and *TW*

(c) *FIML* and *1bit(p_tw)*

(d) *1bit(p_tw)* and *TW*

(e) *FIML* and *1bit(Switch)*

(f) *1bit(Switch)* and *TW*

Figure 5.17: Correlation Plots of Estimated Item Discrimination Parameters

To further investigate the difference of $\hat{a}$ from different methods, 7 items with the largest difference were selected. Table 5.10 presents their missing percentages, point biserial correlation and $\hat{a}$. Point biserial correlation was computed between the binary responses and the raw total scores. Larger the point biserial correlation, more examinees with higher raw total scores get this item correctly. It is another statistics that capture item's power of discrimination. Thus, it was used as a reference to capture the characteristics of the items with different item discrimination parameter estimations. In this table, the items were ordered by proportion of missing responses. Among the 7 items, there were four types of items. The first 2 items had missing rates that were larger than 0.920. In addition, all the observed responses for item 47 were correct, and the imputed responses were correct using *1bit(p_tw)* and *1bit(Switch)*. For the third item with missing rages of 0.880, the point biserial correlations were widely different from four methods. In particular, the point biserial correlations from *1bit(p_tw)* and *1bit(Switch)* were much larger than *FIML* and *TW*, and their $\hat{a}$ were larger. For the items 106, 138, and 77, $\hat{a}$ of $FIML$ were very large and they were not converged. On the contrary, $\hat{a}$ from the other three methods were converged, $\hat{a}$ of TW and *1bit(Switch)* were more close to each other. For the last item, *FIML* had very small point biserial correlation while it had large $\hat{a}$.

Table 5.10: Selected Items with Different Estimated Item Discrimination Parameters

		Point Biserial Correlation				$\hat{a}$			
Item ID	P(Missing)	FIML	TW	1bit(p_tw)	1bit(Switch)	FIML	TW	1bit(p_tw)	1bit(Switch)
Item 117	0.928	0.552	0.555	0.761	0.789	2.078	0.925	7.367	8.465
Item 47	0.922	na^*	0.304	na^*	na^*	-0.216	1.256	2.826	6.910
Item 55	0.880	0.035	0.567	0.750	0.713	0.475	0.986	5.779	3.925
Item 106	0.723	0.314	0.350	0.175	0.168	152.946	1.211	0.718	1.239
Item 138	0.651	0.155	0.262	0.159	0.156	2941.590	1.734	0.680	1.157
Item 77	0.602	0.288	0.337	0.296	0.288	46.304	1.893	1.416	2.798
Item 137	0.651	0.024	0.306	0.222	0.361	5.775	1.359	0.678	2.160

Note. The point biserial correlations were not applicable because all the responses were correct.

Agreement of the item difficulty parameter estimations.

The correlation of estimated item difficulty parameters among *FIML*, *TW*, *1bit(p_tw)*, and *1bit(Switch)* were computed (Table 5.11). Firstly, the correlation of b between *1bit(p_tw)* and *1bit(Switch)* were higher than the other correlations, especially if only including $\hat{b}$ with absolute values less than 5. The correlation

plot of *1bit(p_tw)* and *1bit(Switch)* without fixing x-axis and y-axis was shown in Figure 5.18b. From this figure, *1bit(p_tw)* had a few more extremely estimated $\hat{b}$. To better visualize their correlation, Figure 5.18a presents their correlation where both x-axis and y-axis were fixed in the range of -5 to 5. *1bit(Switch)* and *1bit(p_tw)* had similar $\hat{b}$ when $\hat{b}$ were in the range of -1 to 2. *1bit(Switch)* had larger $\hat{b}$ than *1bit(p_tw)* when $\hat{b}$ of *1bit(p_tw)* were less than -1 or larger than 2. It was because the switching method targeted the extreme response patterns, which were items of almost all correct or incorrect responses. Thus, the difference of $\hat{b}$ between these two methods were shown when $\hat{b}$ were either small or large.

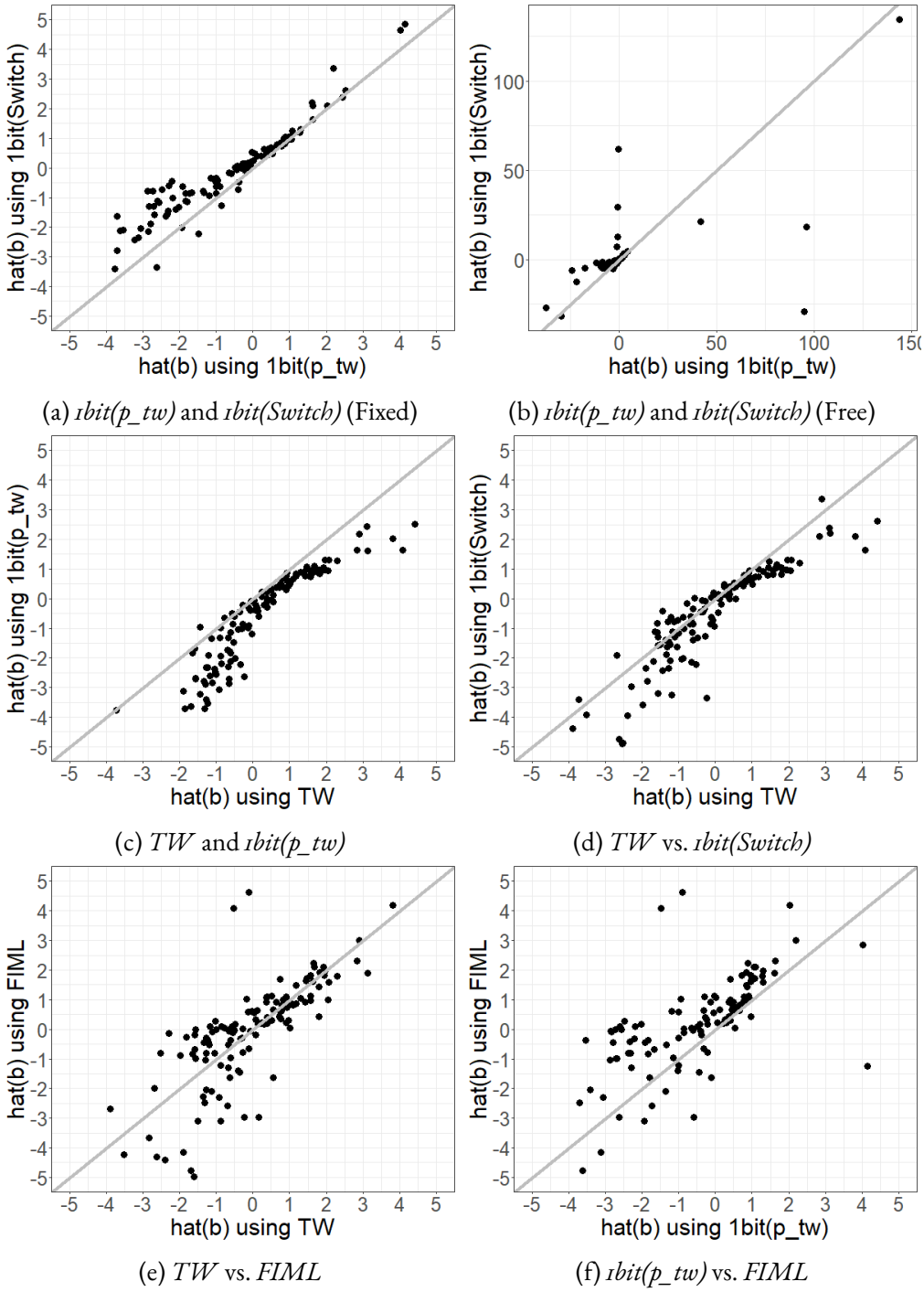
Table 5.11: Correlation of Estimated Item Difficulty Parameters

	All Estimated Item Parameters			
Cor(b)	FIML	TW	1bit(p_tw)	1bit(Switch)
FIML	1.000	-0.014	-0.029	-0.019
TW		1.000	0.059	0.065
1bit(p_tw)			1.000	0.633
1bit(Switch)				1.000
	Estimated Item Parameter ≤ 5			
Cor(b)	FIML	TW	1bit(p_tw)	1bit(Switch)
FIML	1.000	0.709	0.682	0.671
TW		1.000	0.918	0.891
1bit(p_tw)			1.000	0.933
1bit(Switch)				1.000

Secondly, after excluding $\hat{b}$ with absolute values larger than 5, the correlation of $\hat{b}$ between *TW* with *1bit(p_tw)* and *1bit(Switch)* were high, about 0.9. Figure 5.18c shows the correlation between *TW* and *1bit(p_tw)*. *1bit(p_tw)* had smaller $\hat{b}$ than *TW* for easy items, which were more extreme $\hat{b}$. In addition, *1bit(p_tw)* had smaller $\hat{b}$ for difficulty items, which were less extreme $\hat{b}$. Similar patterns were also shown in *1bit(Switch)* and *TW* but to a less extent (Figure 5.18d).

Lastly, the correlation of $\hat{b}$ of *FIML* were relatively high with *TW*, *1bit(p_tw)*, and *1bit(Switch)* if only using $\hat{a}$ with absolute values less than 5. Their correlations were shown in the Figures 5.18e and 5.18f.

To further investigate the difference of $\hat{b}$ from different methods, 11 items with the largest difference were selected. Table 5.12 presents their missing percentages, proportion of correct responses, and $\hat{b}$. These



(a) *1bit(p_tw)* and *1bit(Switch)* (Fixed)

(b) *1bit(p_tw)* and *1bit(Switch)* (Free)

(c) *TW* and *1bit(p_tw)*

(d) *TW* vs. *1bit(Switch)*

(e) *TW* vs. *FIML*

(f) *1bit(p_tw)* vs. *FIML*

Figure 5.18: Correlation Plots of Estimated Item Difficulty Parameters

statistics were used to summarize the characteristics of these items with different $\hat{b}$. Generally, they were categorized into six types. The first item, item 47, had very high missing rate of 0.922 with all correct responses observed. *FIML* and *1bit(p_tw)* did not have converged estimations for b. This item also had very different $\hat{a}$ as shown in Table 5.10. Secondly, item 114 had relatively high proportion of correctness from the observed response matrix, and its $\hat{b}$ from the *TW*, *1bit(p_tw)* and *1bit(Switch)* were all negative indicating this item was relatively easy. However, its $\hat{b}$ from *FIML* was large (4.073), which was contrast to the high proportion of correctness (0.750). Thirdly, item 136 had relatively high missing rate and large proportion of correctness from the observed response matrix, *TW*, *1bit(p_tw)*, and *1bit(Switch)* provided relatively small $\hat{b}$ that were in the range of -0.620 to -2.113, while *FIML* provided very small $\hat{b}$ of -6.836. Fourthly, the next three items, 10, 132, and 137, *1bit(p_tw)* had the smallest $\hat{b}$ and had the largest proportion of correctness among all the methods. This pattern was consistent with the previous finding that *1bit(p_tw)* had the tendency to impute more extreme response matrix than the other methods. Fifthly, the next four items, 4, 57, 106, and 138, *1bit(p_tw)* and *1bit(Switch)* had the same proportions of correctness that were close to 1, but *1bit(p_tw)* had smaller $\hat{b}$. Thus, though *1bit(Switch)* did not directly change the extreme responses patterns for these four items, it still mitigate the extreme estimations for $\hat{b}$. It was expected because other factors, such as examinees' θ estimations, also contributed to the estimation for b. Lastly, item 81 had smaller proportions of correctness using the observed response matrix and *TW* than *1bit(p_tw)* and *1bit(Switch)*, and thus its $\hat{b}$ were larger than *1bit(p_tw)* and *1bit(Switch)*. From this observation, *1bit(p_tw)* and *1bit(Switch)* did not always lead to extreme response patterns.

5.2.3 Summary of the real data application.

In this real data application, we applied two proposed item bank calibration strategies, *1bit(p_tw)* and *1bit(Switch)*, and two baseline methods, *FIML* and *TW* to an empirical data collected from small assessments. This empirical data set did not had the exactly same structure as the data set simulated in the previous simulation studies. Firstly, this data set did not follow the planned targeted NEAT designs due to practical issues. For example, instructors deleted a few items in each form that did not match their

Table 5.12: Selected Items with Different Estimated Item Difficulty Parameters

		P(Correct Response)				$\hat{b}$			
item_id	P(Missing)	FIML	TW	1bit(p_tw)	1bit(Switch)	FIML	TW	1bit(p_tw)	1bit(Switch)
Item 47	0.922	1.000	0.970	1.000	1.000	184.413	-2.538	-17.554	-4.895
Item 114	0.928	0.750	0.831	0.952	0.934	4.073	-0.519	-1.483	-2.210
Item 136	0.759	0.775	0.807	0.928	0.898	-6.836	-0.620	-2.113	-1.401
Item 10	0.657	0.930	0.904	0.976	0.976	-0.808	-2.523	-8.668	-4.868
Item 132	0.717	0.936	0.952	0.982	0.964	-2.281	-1.355	-8.849	-1.394
Item 137	0.651	0.948	0.940	0.982	0.964	-0.182	-1.587	-5.306	-1.134
Item 4	0.608	0.923	0.934	0.970	0.970	-0.885	-1.981	-6.015	-3.592
Item 57	0.759	0.975	0.940	0.994	0.994	-0.512	-1.181	-6.529	-3.246
Item 106	0.723	0.957	0.940	0.988	0.988	-0.139	-2.293	-5.599	-2.971
Item 138	0.651	0.966	0.970	0.988	0.988	-0.001	-1.564	-5.905	-3.188
Item 81	0.596	0.299	0.247	0.349	0.373	6.888	4.084	1.633	1.631

teaching plans and goals, and students skipped a few items in the tests. In addition, the missing rates were not uniformly distributed where some items had missing rates close to 1. Furthermore, the sample size was slightly larger than the minimum sample size investigated in the simulation studies.

After comparing the imputed response matrices, *1bit(p_tw)*, *1bit(Switch)*, and *TW* had over 85% same imputed responses. In addition, their estimated item difficulty parameters had high correlation, about 0.9, after excluding extreme estimations. Among these three imputation methods, *1bit(p_tw)* had the most $\hat{b}$ with absolute values larger than 10. However, the estimated item discrimination parameters of *TW* had small correlations with *1bit(p_tw)* and *1bit(Switch)* even after excluding $\hat{a}$ with absolute values larger than 5. With regard to item discrimination parameters, both *1bit(p_tw)* and *1bit(Switch)* had wider ranges of $\hat{a}$ than *TW*. The inconsistency between estimations for item discrimination parameters and item difficulty parameters were also shown in the results of simulation study 1. Furthermore, the item parameters were also estimated using *FIML* where no imputation were required. It produced more extreme $\hat{a}$ and $\hat{b}$ than the other three imputation methods.

By identifying the items with different item parameter estimations across methods, we found these items had certain characteristics, such as very high missing rates and all correct responses observed. These

findings can help to guide item analyses. In other words, extra attention should be paid to items with very high missing rate, either all correct or incorrect responses, and different estimated item parameters and summary statistics across different methods, especially when the difference was not expected. For example, an item had very small point biserial correlation but large estimated item discrimination parameter using one method. In addition, some items had similar proportion of correctness across methods, but their $\hat{b}$ were different or even with different signs.

Chapter 6

Conclusion and Discussion

Though the great benefits of CAT, its applications in small-assessments are limited. One main reason is the gap between the large pretesting sample sizes required by IRT models and the small sample sizes avaliable in the small-assessments. With the goal of promoting the use of CAT in small-scale assessments, we developed item bank calibration strategies in pretesting scenarios of small sample sizes to close this gap. In particular, through three interdependent studies, we proposed 15 item bank calibration strategies based on two matrix completion algorithms, 1-bit matrix completion and structured matrix completions, in four incomplete calibration designs, targeted NEAT, targeted PC, random NEAT, and random PC designs. These four incomplete calibration designs were combinations of two different missing mechanisms, MAR and MNAR, and two different calibration designs, NEAT and PC designs. In this chapter, we first summarized and discussed the key findings from these three studies. We then concluded the implications of this dissertation and ended this chapter with limitations and future directions.

6.1 Summary

In **Study 1**, we explored and proposed adaptations to two selected matrix completion algorithms in the context of four incomplete calibration designs with binary observed responses. Specifically, *1bit(p_tw)* was proposed to deal with the violation of the uniformly missing assumption of the 1-bit matrix completion

when using the four incomplete calibration designs. In addition, the original structured matrix completion algorithm was developed for continuous matrices and thus was modified for the binary response matrices with four different converting methods and two dichotomizing methods, such as *SMC(1+p_tw)* and *SMC(p1+p_tw)*. Furthermore, structured matrix completion method was adapted to handle the mismatch between its required missing structure and the specific missing patterns in the four incomplete calibration designs. These methods were evaluated in the simulation study 1 with three varying factors: incomplete calibration designs, item banks, and sample sizes. In particular, their performance were compared to four baseline methods investigated in the previous studies: multiple imputation with two-way imputation (*TW*), multivariate imputation by chained equations with classification trees and random forest, denoted as *MICE(CART)* and *MICE(RF)*, and full information maximum likelihood estimation (*FIML*). The key findings were summarized and discussed next.

The proposed dichotomzing method *p_tw*, when paired with 1-bit matrix completion and structured matrix completion, provided higher recovery rates than *Sign* in the targeted NEAT, random NEAT, and random PC designs. For example, *1bit(p_tw)*, *SMC(1+p_tw)*, and *SMC(p1+p_tw)* had higher recovery rates than *1bit(Sign)*, *SMC(1+Sign)*, and *SMC(p1+Sign)*. The main advantage of *p_tw* is that it used both person effect and item effect to determine the cut-off point for each response rather than using zero for all the responses. In this way, it resolved the violation of the uniformly missing assumption of 1-bit matrix completion algorithm. It also provided more reasonable cut-off point to dichotomize continuous imputed values from structured matrix completion than zero by reducing the impact of the specific characteristics of the targeted NEAT, random NEAT, and random PC designs. However, *p_tw* did not work well in the targeted PC designs. In the targeted PC designs, all the items were answered by the examinees with matched abilities. Thus, the person means, which were computed as the average score of the answered responses for examinees, were close to 0.5 for most examinees even their abilities were quite different. That is to say, the person mean in the targeted PC designs was not a good indicator for examinees' abilities. In this case, *p_tw* could not provide informative cut-off points.

Higher recovery rates were not associated with better item discrimination parameter estimations. Compared to *TW*, *1bit(p_tw)*, *SMC(1+p_tw)*, and *SMC(p1+p_tw)* had higher recovery rates and better item difficulty parameter estimations; however, they had were item discrimination parameter estimations than *TW*. The reason was that the adapted matrix completion methods imputed the response matrix with more extreme responses than *TW*. That is, in the imputed response matrix from the adapted matrix completion algorithms using *p_tw*, examinees of high ability had more correct responses for the difficult items, and examinees of low ability had more incorrect responses for the easy items. The item discrimination parameters were thus overestimated. On the contrary, the correct and incorrect responses were more evenly distributed using *TW* across examinees and items. Though it was less accurate than the imputed response matrix based on adapted matrix completion methods with *p_tw* and had less accurate item difficulty parameter estimations, it provided better item discrimination parameter estimations.

Item bank and sample sizes did not have much influence on the performance of the modified matrix completion methods. The results of the adapted matrix completion were similar across three investigated item banks and sample sizes. Though the results from the practical item bank were slightly better than the other two item banks, the relative performance across investigated methods were similar. That is to say, no matter which item bank was used, *1bit(p_tw)* had better results than *1bit(Sign)*, and *SMC(1+p_tw)* had better results than *SMC(1+Sign)*. In addition, the recovery rates of the imputed responses were similar across three sample sizes. The bias of the estimated item parameters were similar when using 125 examinees and 500 examinees, though there were less extremely estimated item parameters when using 500 examinees than 125 examinees. This finding suggests the potential to develop item banks with smaller sample sizes as recommended by previous studies.

The multiple imputation with two-way imputation and full imputation maximum likelihood method provided relatively accurate imputations in random NEAT and random PC designs. Though *TW* provided slightly less accurate imputed responses than the adapted matrix completion methods in the targeted NEAT designs, both *TW* and *FIML* estimated item parameters relatively accurate in the random NEAT

and random PC designs. The other two baselines, *MICE(CART)* and *MICE(RF)*, neither provided the accurate imputations nor accurate item parameter estimations.

In **Study 2**, we developed three item bank calibration strategies to further improve the item parameter estimations from the proposed methods in Study 1. Inspired by the findings in Study 1, the switching method, softened threshold method, and pooling the converting methods, were proposed to mitigate the extreme response patterns from *p_tw*. In particular, the switching method selected a few responses from items and examinees with extreme patterns, such as having mostly correct or mostly incorrect responses, and then switched those responses to the opposite. The amount of responses to be switched was dependent on recovery rates, items' difficulties and examinees' abilities. Two sets of switching parameters which captured the proportion of responses to switch, the proportion of selected correct responses to switch to incorrect, and the proportion of selected incorrect responses to switch in correct were proposed and evaluated in the simulation study 2. In addition, softened threshold method made the cut point that dichotomized the continuous imputed values closer to 0.5 to reduce extreme response patterns from using *p_tw*. Lastly, pooling the converting methods randomly select one response from the three imputed response matrix after excluding the responses with negative point biserial correlation. These three methods were also evaluated in the simulation study 2 with the same design as the simulation in Study 1.

The switching method was relatively robust to the switching parameters and provided more accurate item discrimination parameter estimations than *TW*. Switching method was a direct approach to mitigate the extreme responses and dependent on a set of switching parameters to regulate the amount of responses to switch and which direct to switch. The results of simulation study 2 showed that the two sets of proposed switching parameters had similar recovery rates and item parameter estimations, which indicated that the performance of the switching method was robust to the switching parameters. This finding provided evidence to apply switching method to empirical data sets without previous simulation studies to locate the extreme responses and the degree of the extreme responses. In addition, the switching method improved item discrimination parameter estimations from *1bit(p_tw)* and *SMC(1+p_tw)*, and it maintained the accuracy of item difficulty parameter estimations of *1bit(p_tw)* and *SMC(1+p_tw)*.

The pooling converting methods improved item discrimination parameter estimations but to a less degree than the switching method. Compared to *1* and *p1*, the two converting methods proposed in Study 1, *p_2* led to less extreme response matrices when paired with *p_tw* as the dichotomzing method. Thus, it was expected that by pooling these three converting methods, the extreme response pattern in Study 1 would be mitigated from the imputed response matrices using each converting method with *p_tw*. In addition, during the pooling process, the responses with negative point biserial correlation were removed. From the simulation study 2, the pooling converting methods improved item discrimination parameter estimations but not as good as the switching method. On the other hand, pooling converting methods provided slightly better item difficulty parameter estimations than the switching method. Compared to the switching method, which directly switched the extreme responses, the pooling method was a less direct approach by pooling the imputed responses for all the responses. Thus, it was expected that the pooling method had less impact on reducing the extreme response patterns.

The softened threshold method slightly improved item discrimination parameter estimations and had a few extremely estimated item difficulty parameters. Different from the switching method which targeted on the extreme responses, the softened threshold method modified the cut-off points for all the responses. The advantage of the softened threshold method was that it did not require a set of parameters to control where to switch and which direction to switch. However, using this approach, the extreme response pattern was not eliminated, and a few very easy and very difficult items had extremely estimated item difficulty parameters. If excluding these extreme cases, the softened threshold method had similar results as their corresponding adapted structured matrix completion methods. Because of the small amount of the improvement on the item discrimination parameter estimations and the loss of the accuracy on the item difficulty parameters, the softened threshold method was not recommended.

In **study 3**, to further evaluate the proposed item bank calibration strategies, a CAT simulation and a real data application were conducted. In the CAT simulation, the benefits of the proposed item bank calibration strategies for improving examinees' ability estimations were measured. In the real data application, the feasibility and the performance of the proposed strategies were investigated. In particular, the

empirical data was collected from a set of small-assessments, and its structure was not exactly the same as the simulated response matrix. Thus, this real data application presented insights for researchers and practitioners when conducting item bank calibrations using the proposed strategies of this study in practical scenarios.

CAT using the item bank calibrated by the proposed strategies had more accurate ability estimations than using the item bank calibrated by *FIML* and *TW*. Item banks that were calibrated using three proposed strategies, *1bit(Switch)*, *SMC(1+Switch)*, and *SMC(pool+p_tw)*, along with three baselines, *TW*, *FIML* and the true item parameters were used in this CAT simulation. The estimated examinees' abilities were more accurate if using the estimated item parameters from three proposed strategies and *TW* than *FIML*. In addition, the examined examinees' abilities were more accurate if using the estimated item parameters from three proposed strategies than *TW*, especially for examinees with high and low abilities. From results of simulation study 1, *TW* had estimated item parameters with relatively small bias and RMSE, but it narrowed the ranges of the item parameters. Thus, the estimated abilities from *TW* clustered close to zero and showing an uniform pattern than a normal distribution. This pattern was more obvious if using the random NEAT design than using targeted NEAT designs. On the other hand, from the results of simulation study 1 and 2, *1bit(Switch)*, *SMC(1+Switch)*, and *SMC(pool+p_tw)* had wider ranges of estimated item parameters than *TW*, and their corresponding ability estimations for examinees with high and low abilities were more accurate.

The estimated item parameters were robust to the targeted calibration population. Given the nature of small-assessments and the small sample sizes, the examinees who participate in the calibration may not from the exactly same population of examinees who take the tests. Thus, in this CAT simulation, two populations were simulated. One was a mixture of normal distributions, which was the same as the calibration population in the simulation study 1 and 2, and the other one was a standard normal distribution. The results of the estimated abilities were similar between these two populations. This finding eliminated the concern of the mismatch between the calibration population and the targeted population in applications.

Estimated item parameters from proposed methods showed similar patterns in the real data application as the simulation studies. The true item parameters were unknown in the real data application, and it was very hard to directly evaluate the performance of the proposed item bank calibration strategies. However, by comparing the estimated item parameters across *1bit(p_tw)*, *1bit(Switch)*, *TW*, and *FIML*, their patterns were similar with the previous findings in the simulation study 1 and 2. For example, *1bit(p_tw))* and *1bit(Switch)* had wider ranges of estimated item parameters than *TW*, and *FIML* had more extreme estimations than the imputation methods. In addition, the proportions of the same imputed responses across *1bit(p_tw)*, *1bit(Switch)*, and *TW* were large, and correlations of the estimated item difficulty parameters were also high. On the contrary, the correlations of the estimated item discrimination parameters were low. The inconsistency between estimated item discrimination parameters and item difficulty parameters were also shown in the simulation study 1. Furthermore, the structure of the empirical data, including the missing pattern and the missing rate, was different from the simulated response matrices in the simulation study 1 and 2. The comparable results between the empirical data application and the previous simulation studies indicated that the proposed strategies were not limited to a few specific calibration designs and data structures.

6.2 Implications

For the 2PL model that was used in this dissertation, many research studies have recommended to use large sample sizes, such as more than 500 examinees per item, to estimate item parameters. However, as demonstrated in this dissertation, well-planned incomplete calibration designs and imputation methods can provide more accurate item bank calibration results with small sample sizes than commonly used *FIML* and other baselines such as *TW* and *MICE*. In addition, the minimum sample size that has been investigated in previous studies was 300 with an existing well-developed item bank using an optimal design (Zhu, 2006). In this dissertation, the minimal sample size was 125 without an existing item bank. In the CAT simulation, the estimations of the examinees' abilities using the item bank calibrated by *1bit(p_tw)* and

SMC(p1+Switch) were relatively accurate and only slightly worse than the estimated examinees' abilities when using the true item parameters.

With the currently proposed item bank calibration strategies, targeted NEAT, random NEAT, and random PC designs are recommended over random PC designs. In addition, *1bit(Switch)*, *SMC(1+Switch)*, and *SMC(Switch)* are recommended to handle the missing responses from these three incomplete calibration designs. In particular, both *SMC(1+Switch)* and *SMC(Switch)* require specific missing patterns, while *1bit(Switch)* does not. Thus, if unexpected missing pattern shown, such as students skipping some items, *1bit(p_tw)* is still applicable.

Lastly, it is recommended to apply multiple imputation methods in practice. As demonstrated in the real data application, the different estimated item parameters among different methods may provide useful insights for item analyses. Since the true item parameters are unknown and the sample sizes are small, the inconsistency of the results from different methods can flag certain items from the statistic perspective. Then content experts and instructors can review these items and either revise or remove them.

6.3 Limitations and Future Directions

There are a few limitations of this dissertation. Firstly, the dichotomizing method, *p_tw*, is effective in the targeted NEAT, random NEAT, and random PC designs, but it cannot recover the missing response patterns in the targeted PC designs. It is because the person means in the targeted PC designs were not good indicators for person effects. Thus, one future direction is to modify or to find a different indicator for the person effect in the targeted PC designs. Secondly, all the proposed item bank calibration strategies in this dissertation have two steps: imputation and estimation, which were relatively lengthy. A future direction is to develop item bank calibration strategies that only have one step by integrating the imputation and estimation together. Thirdly, this dissertation only focused on 2PL model and found that high recovery rates were not associated with accurate item discrimination parameter estimations. Thus, a future direction is to investigate the performance of the proposed item bank calibration strategy on other IRT models and psychometric models to investigate the relation of recovery rates and the item

parameter estimations. Fourthly, the proposed item bank calibration strategies, such as *SMC(1+Switch)* and *SMC(pool+p_tw)*, used certain ways of permutation and rearrangement to organize the incomplete response matrix, and thus loosened the strong requirement about the missing structure of the structured matrix completion. However, they cannot handle more complex missing structures such as students skipping a few items either randomly or on purpose. Lastly, this dissertation investigated three different sample sizes, but the missing percentage was fixed in the three sample sizes. A future direction is to investigate influence of the missing percentages while fixing the sample sizes. In addition, the missing percentages investigated in this study were 69.3% in the NEAT designs and 60.0% in the PC designs. Though these two missing percentages were already higher than most of the previous studies, another future direction will be investigate the influence of even larger missing percentages.

Bibliography

Ali, U. S. and Chang, H.-H. (2014). An item-driven adaptive design for calibrating pretest items. *ETS Research Report Series*, 2014(2):1–12.

Almond, R. G. (1995). *Graphical belief modeling*. CRC Press.

Anderson, T. W. (1957). Maximum likelihood estimates for a multivariate normal distribution when some observations are missing. *Journal of the american Statistical Association*, 52(278):200–203.

Attali, Y., Saldivia, L., Jackson, C., Schuppan, F., and Wanamaker, W. (2014). Estimating item difficulty with comparative judgments. *ETS Research Report Series*, 2014(2):1–8.

Avouac, R. and Bjaï, B. (2020). 1-bit matrix completion. `https://github.com/avouacr/1-Bit-Matrix-Completion`.

Azur, M. J., Stuart, E. A., Frangakis, C., and Leaf, P. J. (2011). Multiple imputation by chained equations: what is it and how does it work? *International journal of methods in psychiatric research*, 20(1):40–49.

Baker, F. B. and Kim, S.-H. (2004). *Item response theory: Parameter estimation techniques*. CRC Press.

Ban, J.-C., Hanson, B. A., Wang, T., Yi, Q., and Harris, D. J. (2001). A comparative study of on-line pretest item—calibration/scaling methods in computerized adaptive testing. *Journal of Educational Measurement*, 38(3):191–212.

Ban, J.-C., Hanson, B. A., Yi, Q., and Harris, D. J. (2002). Data sparseness and on-line pretest item calibration-scaling methods in cat. *Journal of Educational Measurement*, 39(3):207–218.

Beale, E. M. and Little, R. J. (1975). Missing values in multivariate analysis. *Journal of the Royal Statistical Society: Series B (Methodological)*, 37(1):129–145.

Berger, M. P. (1992). Sequential sampling designs for the two-parameter item response theory model. *Psychometrika*, 57(4):521–538.

Bernaards, C. A. and Sijtsma, K. (1999). Factor analysis of multidimensional polytomous item response data suffering from ignorable item nonresponse. *Multivariate Behavioral Research*, 34(3):277–313.

Bernaards, C. A. and Sijtsma, K. (2000). Influence of imputation and em methods on factor analysis when item nonresponse in questionnaire data is nonignorable. *Multivariate Behavioral Research*, 35(3):321–364.

Bjorner, J. B., Kosinski, M., and Ware Jr, J. E. (2003). Calibration of an item pool for assessing the burden of headaches: An application of item response theory to the headache impact test (hit™). *Quality of Life Research*, 12(8):913–933.

Bock, R. D. and Aitkin, M. (1981). Marginal maximum likelihood estimation of item parameters: Application of an em algorithm. *Psychometrika*, 46(4):443–459.

Brown, J. M. and Weiss, D. J. (1977). An adaptive testing strategy for achievement test batteries. Technical report, MINNESOTA UNIV MINNEAPOLIS DEPT OF PSYCHOLOGY.

Buyske, S. G. et al. (1998). Optimal design for item calibration in computerized adaptive testing: The 2pl case. *New developments and applications in experimental design*, 34:115–125.

Cai, T., Cai, T. T., and Zhang, A. (2016). Structured matrix completion with applications to genomic data integration. *Journal of the American Statistical Association*, 111(514):621–633.

Cai, T. and Zhou, W.-X. (2013). A max-norm constrained minimization approach to 1-bit matrix completion. *J. Mach. Learn. Res.*, 14(1):3619–3647.

Candes, E. J. and Plan, Y. (2011). Tight oracle inequalities for low-rank matrix recovery from a minimal number of noisy random measurements. *IEEE Transactions on Information Theory*, 57(4):2342–2359.

Candès, E. J. and Recht, B. (2009). Exact matrix completion via convex optimization. *Foundations of Computational mathematics*, 9(6):717–772.

Candès, E. J. and Tao, T. (2010). The power of convex relaxation: Near-optimal matrix completion. *IEEE Transactions on Information Theory*, 56(5):2053–2080.

Chalmers, R. P. (2016). Generating adaptive and non-adaptive test interfaces for multidimensional item response theory applications. *Journal of Statistical Software*, 71(5):1–39.

Chang, Y.-c. I. and Lu, H.-Y. (2010). Online calibration via variable length computerized adaptive testing. *Psychometrika*, 75(1):140–157.

Chang, Y.-P., Chiu, C.-Y., and Tsai, R.-C. (2019). Nonparametric cat for cd in educational settings with small samples. *Applied psychological measurement*, 43(7):543–561.

Davenport, M. A., Plan, Y., Van Den Berg, E., and Wootters, M. (2014). 1-bit matrix completion. *Information and Inference: A Journal of the IMA*, 3(3):189–223.

De Ayala, R., Plake, B. S., and Impara, J. C. (2001). The impact of omitted responses on the accuracy of ability estimation in item response theory. *Journal of educational measurement*, 38(3):213–234.

De Ayala, R. J. (2013). *The theory and practice of item response theory*. Guilford Publications.

DeMars, C. (2002). Incomplete data and item parameter estimates under jmle and mml estimation. *Applied Measurement in Education*, 15(1):15–31.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22.

Dodd, B. G., De Ayala, R., and Koch, W. R. (1995). Computerized adaptive testing with polytomous items. *Applied psychological measurement*, 19(1):5–22.

Donahue, P. (1999). *NAEP 1998 reading report card for the nation and the states*. Department of Education Office of Educational.

Edwards, J. M. and Finch, W. H. (2018). Recursive partitioning methods for data imputation in the context of item response theory: A monte carlo simulation. *Psicológica Journal*, 39(1):88–117.

Eggen, T. J. and Verhelst, N. D. (2011). Item calibration in incomplete testing designs. *Psicológica*, 32(1):107–132.

Enders, C. K. (2001). A primer on maximum likelihood algorithms available for use with missing data. *Structural Equation Modeling*, 8(1):128–141.

Enders, C. K. and Bandalos, D. L. (2001). The relative performance of full information maximum likelihood estimation for missing data in structural equation models. *Structural equation modeling*, 8(3):430–457.

ETS (2021). Interpreting your gre® scores: 2021–22.

Finch, H. (2008). Estimation of item response theory parameters in the presence of missing data. *Journal of Educational Measurement*, 45(3):225–245.

Fink, A., Born, S., Spoden, C., and Frey, A. (2018). A continuous calibration strategy for computerized adaptive testing. *Psychological Test and Assessment Modeling*, 60(3):327–346.

Flaugher, B. and Meier, K. (1990). A compilation of jet finding algorithms. Technical report, Fermi National Accelerator Lab., Batavia, IL (USA).

Fliege, H., Rose, M., Arck, P., Walter, O. B., Kocalevent, R.-D., Weber, C., and Klapp, B. F. (2005). The perceived stress questionnaire (psq) reconsidered: validation and reference values from different clinical and healthy adult samples. *Psychosomatic medicine*, 67(1):78–88.

Graham, J. W., Olchowski, A. E., and Gilreath, T. D. (2007). How many imputations are really needed? some practical clarifications of multiple imputation theory. *Prevention science*, 8(3):206–213.

Graham, J. W., Taylor, B. J., and Cumsille, P. E. (2001). Planned missing-data designs in analysis of change.

Graham, J. W., Taylor, B. J., Olchowski, A. E., and Cumsille, P. E. (2006). Planned missing data designs in psychological research. *Psychological methods*, 11(4):323.

Gross, D. (2011). Recovering low-rank matrices from few coefficients in any basis. *IEEE Transactions on Information Theory*, 57(3):1548–1566.

Hambleton, R. K. and Cook, L. L. (1977). Latent trait models and their use in the analysis of educational test data. *Journal of educational measurement*, pages 75–96.

Hendrickson, A. (2007). An ncme instructional module on multistage testing. *Educational Measurement: Issues and Practice*, 26(2):44–52.

Huisman, M. (1999). *Item nonresponse: Occurrence, causes, and imputation of missing answers to test items*. DSWO Press Leiden.

Jiang, B., Ma, S., Causey, J., Qiao, L., Hardin, M. P., Bitts, I., Johnson, D., Zhang, S., and Huang, X. (2016). Sparrec: An effective matrix completion framework of missing data imputation for gwas. *Scientific reports*, 6(1):1–15.

Jones, D. H. and Jin, Z. (1994). Optimal sequential designs for on-line item estimation. *Psychometrika*, 59(1):59–75.

Kalish, L. and Rosenberger, J. (1978). Optimal designs for the estimation of the logistic function. In *Technical Report 33*. The Pennsylvania State University, Dept. of Statistics, State College PA.

Kang, H. (2013). The prevention and handling of the missing data. *Korean journal of anesthesiology*, 64(5):402.

Kaya, E. and Kalender, İ. (2018). Use of computerized adaptive testing in class-level language assessment. *Bridging Teaching, Learning and Assessment in the English Language Classroom*, page 73.

Kingsbury, G. (2009). Adaptive item calibration: A process for estimating item parameters within a computerized adaptive test. In *Proceedings of the 2009 GMAC conference on computerized adaptive testing*.

Kingsbury, G. G. and Zara, A. R. (1989). Procedures for selecting items for computerized adaptive tests. *Applied measurement in education*, 2(4):359–375.

Kirisci, L., Hsu, T.-c., and Yu, L. (2001). Robustness of item parameter estimation programs to assumptions of unidimensionality and normality. *Applied psychological measurement*, 25(2):146–162.

Kohli, N., Koran, J., and Henn, L. (2015). Relationships among classical test theory and item response theory frameworks via factor analytic models. *Educational and Psychological Measurement*, 75(3):389–405.

Krass, I. A. and Williams, B. (2003). Calibrating cat pools and online pretest items using nonparametric and adjusted marginal maximum likelihood methods. In *Annual Meeting of the National Council on Measurement in Education, Chicago, IL, USA*.

Lee, J. E. (2011). Full-metric concurrent calibration for the development of cat item banks.

Little, T. D. and Rhemtulla, M. (2013). Planned missing data designs for developmental researchers. *Child Development Perspectives*, 7(4):199–204.

Liu, Y., Zhang, A., Cai, T., T., and Cai, T. (2015). *StructureMC: Structured Matrix Completion*. R package version 1.0.

Lord, F. (1980). Application of item response theory to practical testing problems. hillsdale, nj, lawrence erlbaum ass.

Lord, F. M. (1955). Estimation of parameters from incomplete data. *Journal of the American Statistical Association*, 50(271):870–876.

Lord, F. M. (1968). An analysis of the verbal scholastic aptitude test using birnbaum's three-parameter logistic model. *Educational and Psychological Measurement*, 28(4):989–1020.

Lord, F. M. (1971). A theoretical study of two-stage testing. *Psychometrika*, 36(3):227–242.

Lord, F. M. (1974). Estimation of latent ability and item parameters when there are omitted responses. *Psychometrika*, 39(2):247–264.

Lord, F. M. (1977). A broad-range tailored test of verbal ability. *Applied Psychological Measurement*, 1(1):95–100.

Lord, F. M. (1982). Maximum likelihood estimation of item response parameters when some responses are omitted. *ETS Research Report Series*, 1982(1):i–13.

Lord, F. M. (1986). Maximum likelihood and bayesian parameter estimation in item response theory. *Journal of Educational Measurement*, pages 157–162.

Lu, H.-Y. (2014). Application of optimal designs to item calibration. *PloS one*, 9(9):e106747.

Makransky, G. and Glas, C. A. (2014). An automatic online calibration design in adaptive testing. *Journal of Applied Testing Technology*, 11(1):1–20.

Martin, A. J. and Lazendic, G. (2018). Computer-adaptive testing: Implications for students' achievement, motivation, engagement, and subjective test experience. *Journal of educational psychology*, 110(1):27.

Mathew, T. and Sinha, B. K. (2001). Optimal designs for binary data under logistic regression. *Journal of Statistical Planning and Inference*, 93(1-2):295–307.

McBride, J. R. and Martin, J. T. (1983). Reliability and validity of adaptive ability tests in a military setting. In *New horizons in testing*, pages 223–236. Elsevier.

McKnight, P. E., McKnight, K. M., Sidani, S., and Figueredo, A. J. (2007). *Missing data: A gentle introduction*. Guilford Press.

Mellenbergh, G. J. (1989). Item bias and item response theory. *International journal of educational research*, 13(2):127–143.

Meredith, W. (1993). Measurement invariance, factor analysis and factorial invariance. *Psychometrika*, 58(4):525–543.

Meredith, W. and Millsap, R. E. (1992). On the misuse of manifest variables in the detection of measurement bias. *Psychometrika*, 57(2):289–311.

Mislevy, R. J. (1986). Bayes modal estimation in item response models. *Psychometrika*, 51(2):177–195.

Mislevy, R. J. and Sheehan, K. M. (1989). The role of collateral information about examinees in item parameter estimation. *Psychometrika*, 54(4):661–679.

Mislevy, R. J. and Wu, P.-K. (1996). Missing responses and irt ability estimation: Omits, choice, time limits, and adaptive testing. *ETS Research Report Series*, 1996(2):i–36.

Moreno, K. E., Wetzel, C. D., McBride, J. R., and Weiss, D. J. (1984). Relationship between corresponding armed services vocational aptitude battery (asvab) and computerized adaptive testing (cat) subtests. *Applied Psychological Measurement*, 8(2):155–163.

OECD, N. (2003). *The PISA 2003 assessment framework: Mathematics, reading, science and problem solving knowledge and skills*. OECD Paris.

Orlando, M. and Thissen, D. (2000). Likelihood-based item-fit indices for dichotomous item response theory models. *Applied psychological measurement*, 24(1):50–64.

Petersen, M. A., Aaronson, N. K., Chie, W.-C., Conroy, T., Costantini, A., Hammerlid, E., Hjermstad, M. J., Kaasa, S., Loge, J. H., Velikova, G., et al. (2016). Development of an item bank for computerized adaptive test (cat) measurement of pain. *Quality of Life Research*, 25(1):1–11.

Pokropek, A. et al. (2011). Missing by design: Planned missing-data designs in social science. *ASK. Research & Methods*, (20):81–105.

Prestwood, J. S., Vale, C., Massey, R., and Welsh, J. (1985). Armed services vocational aptitude battery: Development of an adaptive item pool. Technical report, ASSESSMENT SYSTEMS CORP ST PAUL MN.

R Core Team (2021). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Raghunathan, T. E., Lepkowski, J. M., Van Hoewyk, J., Solenberger, P., et al. (2001). A multivariate technique for multiply imputing missing values using a sequence of regression models. *Survey methodology*, 27(1):85–96.

Rasch, G. (1960). Studies in mathematical psychology: I. probabilistic models for some intelligence and attainment tests.

Recht, B., Fazel, M., and Parrilo, P. A. (2010). Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM review*, 52(3):471–501.

Reckase, M. D. (2009). Multidimensional item response theory models. In *Multidimensional item response theory*, pages 79–112. Springer.

Rhemtulla, M. and Hancock, G. R. (2016). Planned missing data designs in educational psychology research. *Educational Psychologist*, 51(3-4):305–316.

Rhemtulla, M. and Little, T. D. (2012). Planned missing data designs for research in cognitive development. *Journal of Cognition and Development*, 13(4):425–438.

Rubin, D. B. (1976). Inference and missing data. *Biometrika*, 63(3):581–592.

Rupp, A., Templin, J., and Henson, R. (2010). Diagnostic assessment: Theory, methods, and applications. *New York: Guilford*.

Şahin, A. and Weiss, D. J. (2015). Effects of calibration sample size and item bank size on ability estimation in computerized adaptive testing. *Educational Sciences: Theory & Practice*, 15(6).

Samejima, F. (1983). Some methods and approaches of estimating the operating characteristics of discrete item responses. *Principals of modern psychological measurement: A Festschrift for Frederic M. Lord*, pages 159–182.

Samsudin, M. A., Chut, T. S., Ismail, M. E., and Ahmad, N. J. (2020). A calibrated item bank for computerized adaptive testing in measuring science timss performance. *EURASIA Journal of Mathematics, Science and Technology Education*, 16(7).

Schafer, J. L. (1997). *Analysis of incomplete multivariate data*. Chapman and Hall/CRC.

Schafer, J. L. and Graham, J. W. (2002). Missing data: our view of the state of the art. *Psychological methods*, 7(2):147.

Schulz, W. H. (2005). Mathematics self-efficacy and student expectations: Results from pisa 2003. *Online Submission*.

Segall, D. O. (2005). Computerized adaptive testing. *Encyclopedia of social measurement*, 1:429–438.

Sijtsma, K. and Van der Ark, L. A. (2003). Investigation and treatment of missing item scores in test and questionnaire data. *Multivariate Behavioral Research*, 38(4):505–528.

Smits, N., Mellenbergh, G. J., and Vorst, H. C. (2002). Alternative missing data techniques to grade point average: Imputing unavailable grades. *Journal of Educational Measurement*, 39(3):187–206.

Stocking, M. L. (1994). Three practical issues for modern adaptive testing item pools 1. *ETS Research Report Series*, 1994(1):i–34.

Suh, Y., Cho, S.-J., and Wollack, J. A. (2012). A comparison of item calibration procedures in the presence of test speededness. *Journal of Educational Measurement*, 49(3):285–311.

Tao, Y.-H., Wu, Y.-L., and Chang, H.-Y. (2008). A practical computer adaptive testing model for small-scale scenarios. *Journal of Educational Technology & Society*, 11(3):259–274.

Thissen, D., Reeve, B. B., Bjorner, J. B., and Chang, C.-H. (2007). Methodological issues for building item banks and computerized adaptive scales. *Quality of Life Research*, 16(1):109–119.

Thompson, N. A. and Weiss, D. A. (2011). A framework for the development of computerized adaptive tests. *Practical Assessment, Research, and Evaluation*, 16(1):1.

U.S. Department of Education, N. C. f. E. S. (2011-2012). Schools and staffing survey (sass): Public school teacher data file. `https://nces.ed.gov/surveys/sass/tables/sass1112_2013314_t1s_007.asp`.

Van Buuren, S. (2012). *Flexible imputation of missing data*. Boca Raton, FL: CRC Press.

Van Buuren, S. and Groothuis-Oudshoorn, K. (2011). mice: Multivariate imputation by chained equations in r. *Journal of statistical software*, 45(1):1–67.

Van Buuren, S. and Oudshoorn, K. (1999). *Flexible multivariate imputation by MICE*. Leiden: TNO.

van der Linden, W. J. and Ren, H. (2015). Optimal bayesian adaptive design for test-item calibration. *Psychometrika*, 80(2):263–288.

van der Linden, W. J., Veldkamp, B. P., and Reese, L. M. (2000). An integer programming approach to item bank design. *Applied Psychological Measurement*, 24(2):139–150.

Van Rijn, P., Eggen, T., Hemker, B., and Sanders, P. (2002). Evaluation of selection procedures for computerized adaptive testing with polytomous items. *Applied Psychological Measurement*, 26(4):393–411.

Veldkamp, B. P. and van der Linden, W. J. (2009). Designing item pools for adaptive testing. In *Elements of adaptive testing*, pages 231–245. Springer.

von Davier, A. A., Holland, P. W., and Thayer, D. T. (2004). The chain and post-stratification methods for observed-score equating: Their relationship to population invariance. *Journal of Educational Measurement*, 41(1):15–32.

Wainer, H. and Mislevy, R. J. (1990). Item response theory, item calibration, and proficiency estimation. *Computerized adaptive testing: A primer*, 4:65–102.

Wang, S., Lin, H., Chang, H.-H., and Douglas, J. (2016). Hybrid computerized adaptive testing: From group sequential design to fully sequential design. *Journal of Educational Measurement*, 53(1):45–62.

Wang, S., Zhang, S., and Shen, Y. (2020). A joint modeling framework of responses and response times to assess learning outcomes. *Multivariate behavioral research*, 55(1):49–68.

Weiss, D. J. (1982). Improving measurement quality and efficiency with adaptive testing. *Applied psychological measurement*, 6(4):473–492.

Weiss, D. J. and Von Minden, S. (2012). A comparison of item parameter estimates from xcalibre 4.1 and bilog-mg. *St. Paul MN: Assessment Systems Corporation*.

Welch, R. E. and Frick, T. W. (1993). Computerized adaptive testing in instructional settings. *Educational Technology Research and Development*, 41(3):47–62.

Wise, S. L. and Kingsbury, G. G. (2000). Practical issues in developing and maintaining a computerized adaptive testing program. *Psicológica*, 21(1):135–155.

Woessmann, L. and West, M. (2006). Class-size effects in school systems around the world: Evidence from between-grade variation in timss. *European Economic Review*, 50(3):695–736.

Wright, B. D. and Stone, M. H. (1979). Best test design.

Xiao, J. and Bulut, O. (2020). Evaluating the performances of missing data handling methods in ability estimation from sparse data. *Educational and Psychological Measurement*, 80(5):932–954.

Zhang, Z. (2020). *Item Parameter Estimation Using Marginal Maximum Likelihood Estimation Under Targeted Testing Design*. PhD thesis, University of Georgia.

Zhu, R. (2006). *Implementation of optimal design for item calibration in computerized adaptive testing (CAT)*. University of Illinois at Urbana-Champaign.

Appendix A

Proximal gradient descent and fast truncated singular value decomposition algorithm for 1-bit matrix completion

With a pre-determined step size parameter t and a penalty parameter λ, the algorithm starts with a $\mathbf{X}^0 = 0 \in \mathbb{R}^{I \times J}$:

1. Find the singular value decomposition for $\mathbf{X}^k$, $\mathbf{X}^k = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$, where $\mathbf{\Sigma} = diag(\sigma_i)_{1 \leq i \leq r}$
2. Compute the gradient value g for the likelihood function $\mathcal{L}_{\Omega,\mathbf{Y}}(\mathbf{X})$:

$$g = \sum_{i,j \in \Omega} \frac{1}{n}(-Y_{ij}) \frac{e^{-Y_{ij}X_{ij}^k}}{1 + e^{-Y_{ij}X_{ij}^k}},$$

   where n is the total number of observations.

3. Update $\mathbf{X}^k$ with $\mathbf{X}^{k+1}$:

$$\mathbf{X}^{k+1} = \mathbf{U}\mathcal{D}_\lambda(\mathbf{X}^k - t \times g)\mathbf{V}^T,$$

   where $\mathcal{D}_\lambda(\mathbf{X}) = \mathbf{U}\mathcal{D}_\lambda(\mathbf{\Sigma})\mathbf{V}^T$ and $\mathcal{D}_\lambda(\mathbf{\Sigma}) = diag(\sigma_i - \lambda)_+$. Note that $x_+ = max(0, x)$, which is the positive part of x.

Repeat the above three steps until convergence.

Appendix B

Determining the switching parameters of *Switch_Preferred*

Following the same pattern as *Switch_General* to determine p, q_1, and q_2 , *Switch_Preferred* identified their specific values for each condition using the simulation results from study 1. That is, the average proportion of correctly imputed responses, proportion of wrongly imputed correct responses, and proportion of wrongly imputed incorrect responses were computed as reference to determine p, q_1, and q_2 for each item bank and each incomplete calibration design. For example, Figure B.1 presents the proportion of wrongly imputed responses per block and the corresponding p of each block in the targeted NEAT design and the uniform item bank. To avoid the overfitting issue, p was not strictly equivalent to the proportion of wrongly imputed responses in each block. The switching parameter p of *Switch_Preferred* did not follow a strictly systematic pattern as p of *Switch_General* in Figure 4.4a but showed a similar pattern.

1 - recovery rates	Item Set 1 (Easy)	Item Set 2	Item Set 3	Item Set 4	Item Set 5 (Hard)
Group 5 (Large theta)	0.04	0.10	0.13	0.17	0.23
Group 4	0.08	0.16	0.22	0.28	0.26
Group 3	0.12	0.25	0.33	0.33	0.18
Group 2	0.19	0.35	0.33	0.19	0.11
Group 1 (Small theta)	0.24	0.22	0.16	0.10	0.06

(a) Proportion of Wrongly Imputed Responses

p% of responses are selected to switch	Item Set 1 (Easy)	Item Set 2	Item Set 3	Item Set 4	Item Set 5 (Hard)
Group 5 (Large theta)	0.05	0.10	0.15	0.15	0.20
Group 4	0.10	0.20	0.25	0.25	0.25
Group 3	0.15	0.25	0.35	0.35	0.20
Group 2	0.20	0.35	0.35	0.20	0.10
Group 1 (Small theta)	0.25	0.20	0.15	0.10	0.05

(b) Switching Parameters p

Figure B.1: Method 2: *Switch_Preferred* for p for Targeted NET Designs and Uniform Item Bank using $1bit(p_tw)$

In addition, Figure B.2 presents the proportion of wrongly imputed correct responses using $1bit(p_tw)$ in the targeted NEAT and the uniform item bank. Larger the proportion of wrongly imputed correct responses (Figure B.2a), more selected correct responses were switched to incorrect (Figure B.2b). The switching parameter $q1$ of *Switch_Preferred* followed a similar pattern as q_1 of *Switch_General* in Figure 4.5a but with different values. Furthermore, the proportion of wrongly imputed correct responses and choices for q_2 are presented in Figure B.3.

Following this procedure, p, q_1, and q_2 of *Switch_Preferred* for the other three incomplete calibration designs using the uniform item banks were determined as shown in the Figures B.4, B.5, and B.6.

P(wrongly imputed correct responses)	Item Set 1 (Easy)	Item Set 2	Item Set 3	Item Set 4	Item Set 5 (Hard)
Group 5 (Large theta)	1.00	1.00	1.00	1.00	0.94
Group 4	1.00	1.00	1.00	0.97	0.68
Group 3	1.00	0.98	0.85	0.65	0.19
Group 2	0.90	0.63	0.36	0.09	0.00
Group 1 (Small theta)	0.34	0.07	0.01	0.00	0.00

(a) Proportion of Wrongly Imputed Correct Responses

q1% of the selected right responses are switched to wrong	Item Set 1 (Easy)	Item Set 2	Item Set 3	Item Set 4	Item Set 5 (Hard)
Group 5 (Large theta)	0.95	0.95	0.95	0.95	0.90
Group 4	0.95	0.95	0.95	0.90	0.50
Group 3	0.95	0.90	0.80	0.50	0.15
Group 2	0.90	0.50	0.35	0.10	0.00
Group 1 (Small theta)	0.35	0.05	0.00	0.00	0.00

(b) Switching Parameters q_1

Figure B.2: Method 2: *Switch_Preferred* for q_1 for Targeted NET Designs and Uniform Item Bank using $1bit(p_tw)$

P(wrongly imputed incorrect responses)	Item Set 1 (Easy)	Item Set 2	Item Set 3	Item Set 4	Item Set 5 (Hard)
Group 5 (Large theta)	0.00	0.00	0.00	0.00	0.06
Group 4	0.00	0.00	0.00	0.03	0.32
Group 3	0.00	0.02	0.15	0.35	0.81
Group 2	0.10	0.37	0.64	0.91	1.00
Group 1 (Small theta)	0.66	0.93	1.00	1.00	1.00

(a) Proportion of Wrongly Imputed Correct Responses

q2% of the selected wrong responses are switched to right	Item Set 1 (Easy)	Item Set 2	Item Set 3	Item Set 4	Item Set 5 (Hard)
Group 5 (Large theta)	0.00	0.00	0.00	0.00	0.05
Group 4	0.00	0.00	0.00	0.05	0.30
Group 3	0.00	0.05	0.15	0.35	0.80
Group 2	0.10	0.35	0.60	0.90	0.95
Group 1 (Small theta)	0.65	0.90	0.95	0.95	0.95

(b) Switching Parameters q_2

Figure B.3: Method 2: *Switch_Preferred* for q_2 for Targeted NET Designs and Uniform Item Bank using $1bit(p_tw)$

p% of responses are selected to switch	Item Set 1 (Easy)	Item Set 2	Item Set 3	Item Set 4 (Hard)
Group 5 (Large theta)	0.05	0.2	0.25	0.35
Group 4	0.1	0.2	0.3	0.3
Group 3	0.1	0.25	0.3	0.2
Group 2	0.2	0.3	0.2	0.1
Group 1 (Small theta)	0.35	0.25	0.2	0.1

(a) Proportion of Wrongly Imputed Correct Responses

q1% of the selected right responses are switched to wrong	Item Set 1 (Easy)	Item Set 2	Item Set 3	Item Set 4 (Hard)
Group 5 (Large theta)	0.95	0.95	0.95	0.95
Group 4	0.95	0.80	0.70	0.60
Group 3	0.70	0.60	0.55	0.30
Group 2	0.55	0.30	0.10	0.00
Group 1 (Small theta)	0.05	0.00	0.00	0.00

(b) Switching Parameters q_1

q2% of the selected wrong responses are switched to right	Item Set 1 (Easy)	Item Set 2	Item Set 3	Item Set 4 (Hard)
Group 5 (Large theta)	0.00	0.00	0.00	0.05
Group 4	0.00	0.00	0.05	0.15
Group 3	0.05	0.10	0.30	0.55
Group 2	0.25	0.55	0.80	0.95
Group 1 (Small theta)	0.90	0.90	0.95	0.95

(c) Switching Parameters q_2

Figure B.4: *Switch_Preferred* for p, q_1, and q_2 for Targeted PC Designs and Uniform Item Bank using $1bit(p_tw)$

p% of responses are selected to switch	Item Set 1 (Easy)	Item Set 2	Item Set 3	Item Set 4	Item Set 5
Group 5 (Large theta)	0.05	0.05	0.1	0.15	0.3
Group 4	0.05	0.15	0.2	0.3	0.25
Group 3	0.1	0.2	0.3	0.25	0.15
Group 2	0.2	0.3	0.25	0.15	0.1
Group 1 (Small theta)	0.3	0.25	0.15	0.05	0.05

(a) Proportion of Wrongly Imputed Correct Responses

q1% of the selected right responses are switched to wrong	Item Set 1 (Easy)	Item Set 2	Item Set 3	Item Set 4	Item Set 5 (Hard)
Group 5 (Large theta)	0.95	0.95	0.95	0.95	0.6
Group 4	0.95	0.95	0.95	0.75	0.1
Group 3	0.95	0.95	0.75	0.15	0
Group 2	0.95	0.7	0.15	0	0
Group 1 (Small theta)	0.6	0.1	0	0	0

(b) Switching Parameters q_1

q2% of the selected wrong responses are switched to right	Item Set 1 (Easy)	Item Set 2	Item Set 3	Item Set 4	Item Set 5 (Hard)
Group 5 (Large theta)	0	0	0	0	0.4
Group 4	0	0	0	0.25	0.85
Group 3	0	0	0.25	0.8	0.9
Group 2	0.05	0.25	0.8	0.95	0.95
Group 1 (Small theta)	0.4	0.9	0.95	0.95	0.95

(c) Switching Parameters q_2

Figure B.5: *Switch_Preferred* for p, q_1, and q_2 for random NEAT Designs and Uniform Item Bank using $1bit(p_tw)$

p% of responses are selected to switch	Item Set 1 (Easy)	Item Set 2	Item Set 3	Item Set 4 (Hard)
Group 5 (Large theta)	0.05	0.05	0.15	0.25
Group 4	0.05	0.15	0.25	0.2
Group 3	0.1	0.2	0.25	0.15
Group 2	0.15	0.25	0.2	0.1
Group 1 (Small theta)	0.25	0.2	0.1	0.05

(a) Proportion of Wrongly Imputed Correct Responses

q1% of the selected right responses are switched to wrong	Item Set 1 (Easy)	Item Set 2	Item Set 3	Item Set 4 (Hard)
Group 5 (Large theta)	0.95	0.95	0.95	0.65
Group 4	0.95	0.95	0.85	0.25
Group 3	0.95	0.9	0.4	0
Group 2	0.95	0.5	0.05	0
Group 1 (Small theta)	0.6	0.05	0	0

(b) Switching Parameters q_1

q2% of the selected wrong responses are switched to right	Item Set 1 (Easy)	Item Set 2	Item Set 3	Item Set 4 (Hard)
Group 5 (Large theta)	0	0	0	0.3
Group 4	0	0	0.1	0.7
Group 3	0	0.05	0.55	0.95
Group 2	0.05	0.45	0.95	0.95
Group 1 (Small theta)	0.35	0.95	0.95	0.95

(c) Switching Parameters q_2

Figure B.6: *Switch_Preferred* for p, q_1, and q_2 for Random PC Designs and Uniform Item Bank using $1bit(p_tw)$

Appendix C

Figures

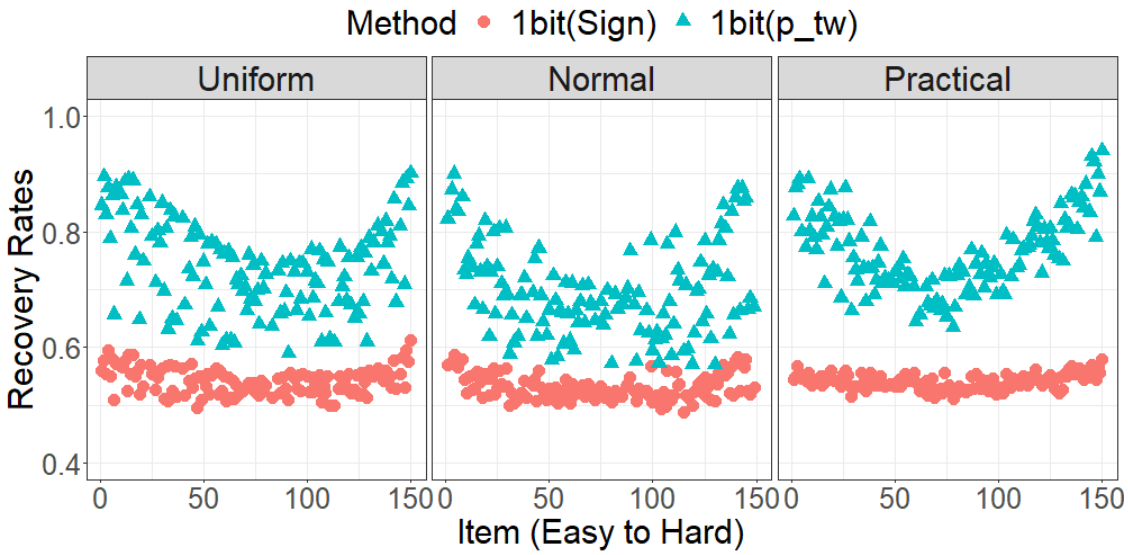


Figure C.1: Recovery Rates of *1bit(Sign)* and *1bit(p_tw)* Across Item Banks in Random NEAT Designs and $N = 125$

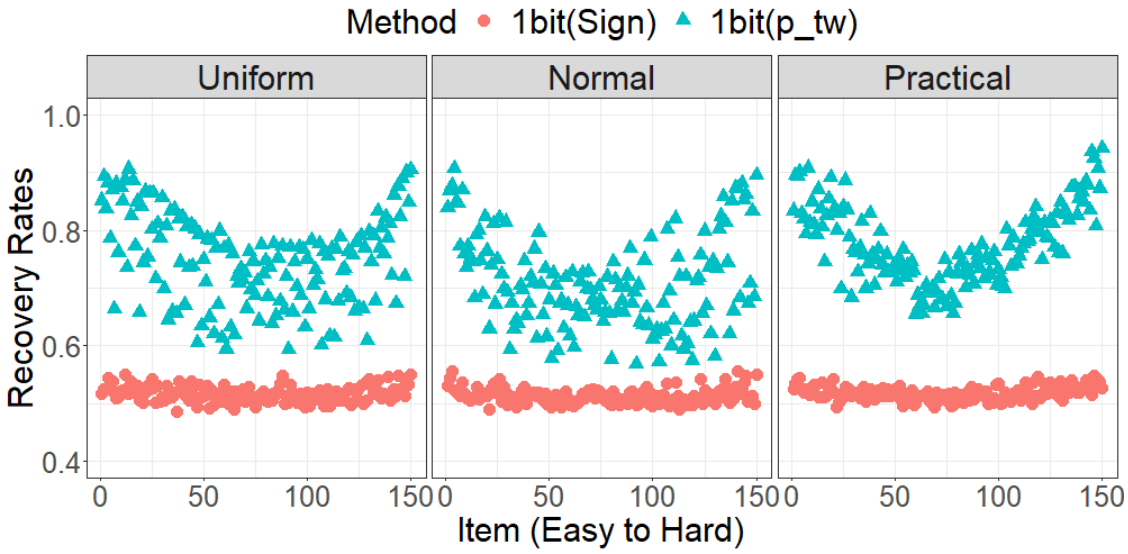


Figure C.2: Recovery Rates of *1bit(Sign)* and *1bit(p_tw)* Across Item Banks in Random PC Designs and $N = 125$

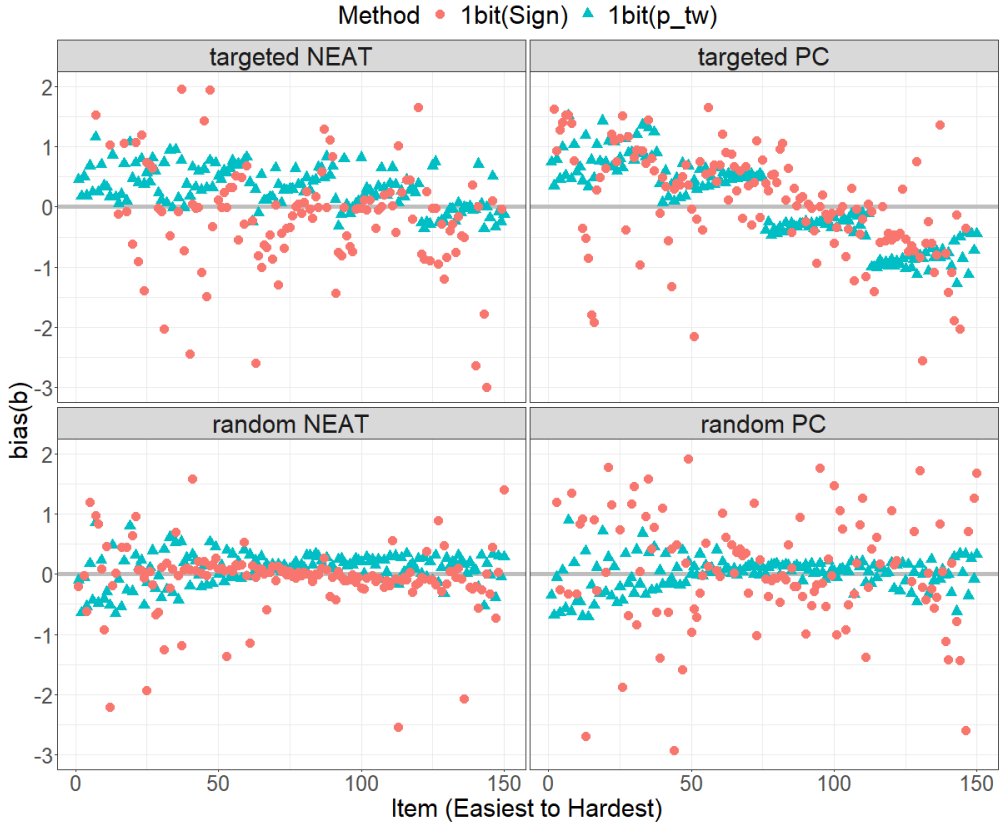


Figure C.3: Bias(b) per Item of *1bit(Sign)* and *1bit(p_tw)* in Uniform Item Bank and $N = 500$

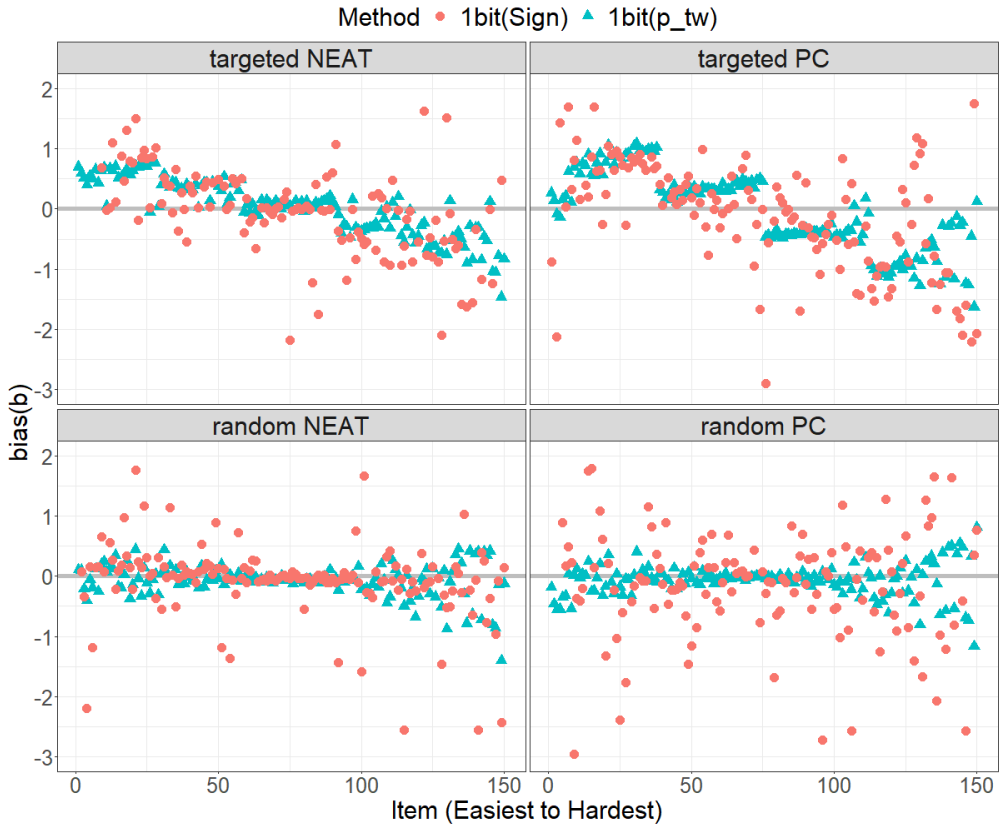


Figure C.4: Bias(b) per Item of *1bit(Sign)* and *1bit(p_tw)* in Normal Item Bank and $N = 500$

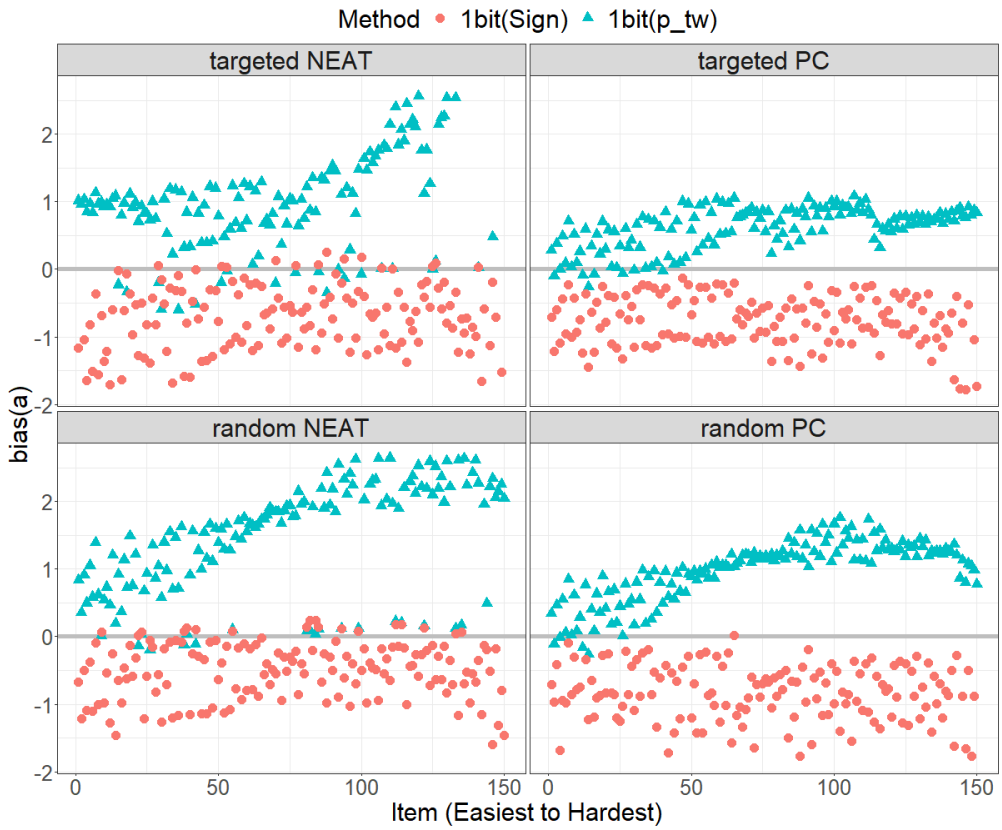


Figure C.5: Bias(a) per Item of *1bit(Sign)* and *1bit(p_tw)* in Practical Item Bank and $N = 500$

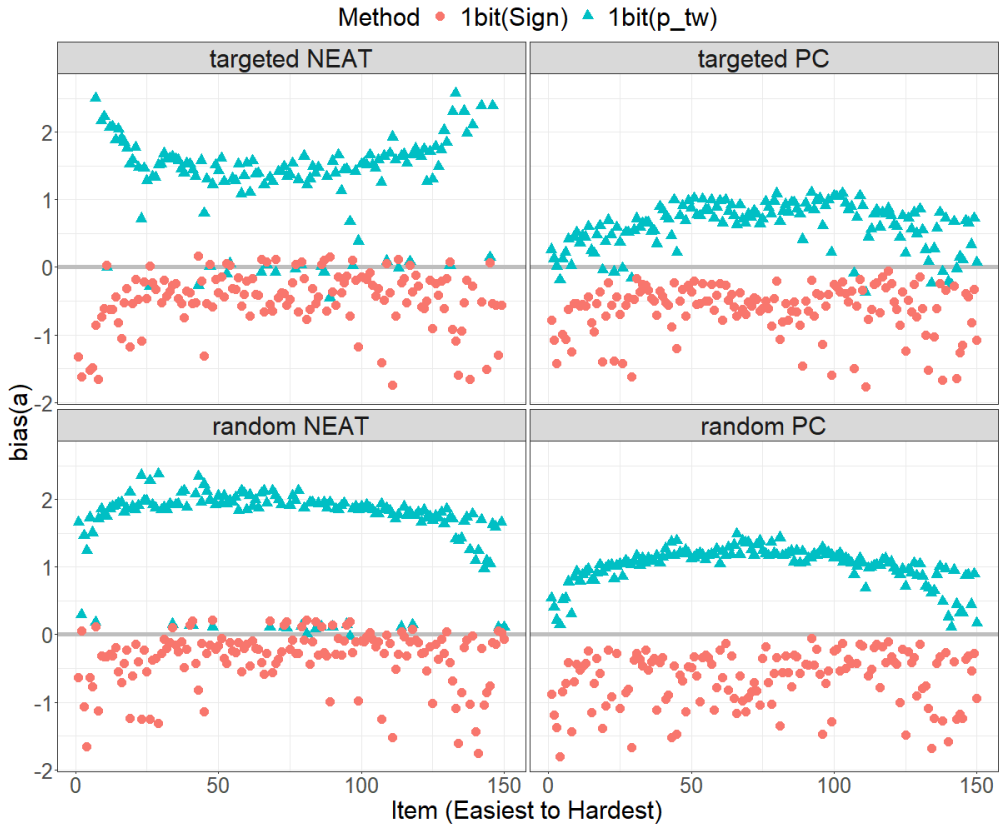
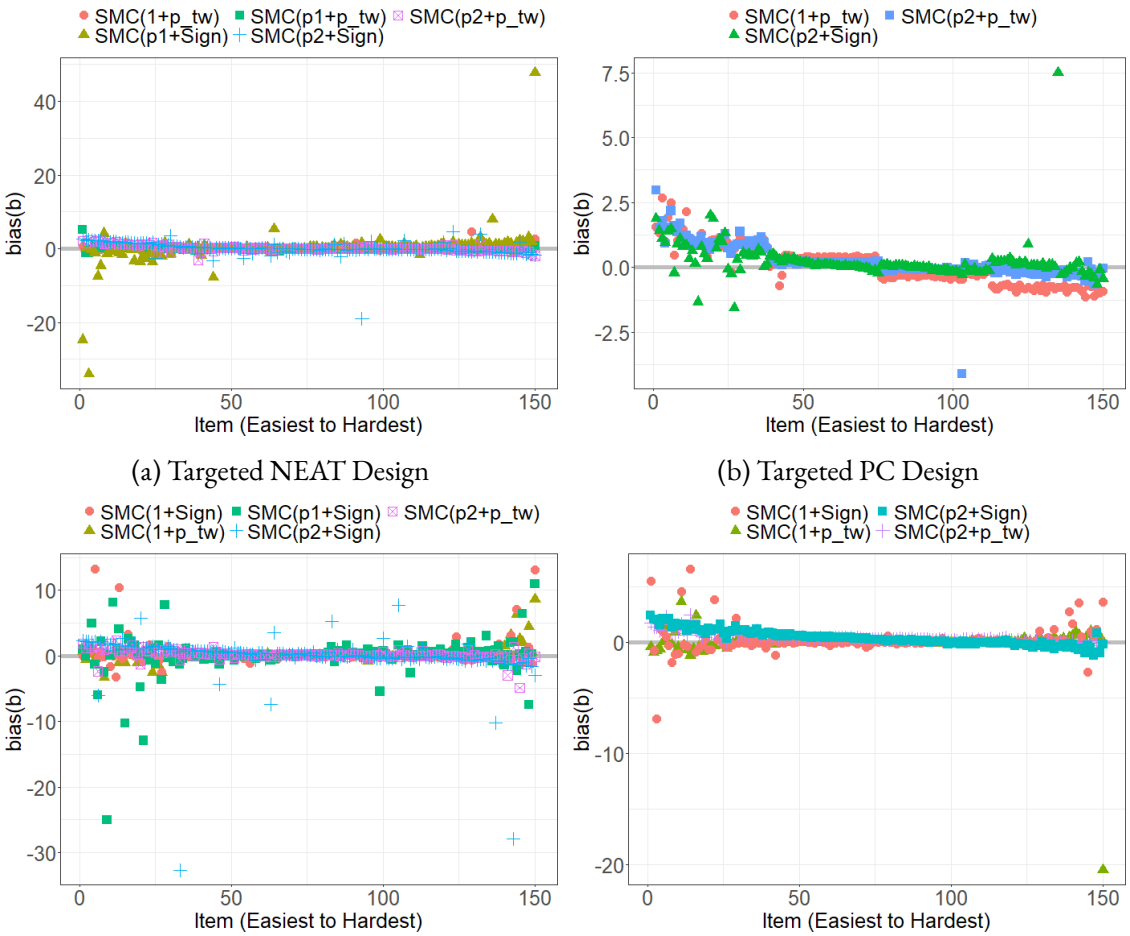


Figure C.6: Bias(a) per Item of *1bit(Sign)* and *1bit(p_tw)* in Normal Item Bank and $N = 500$



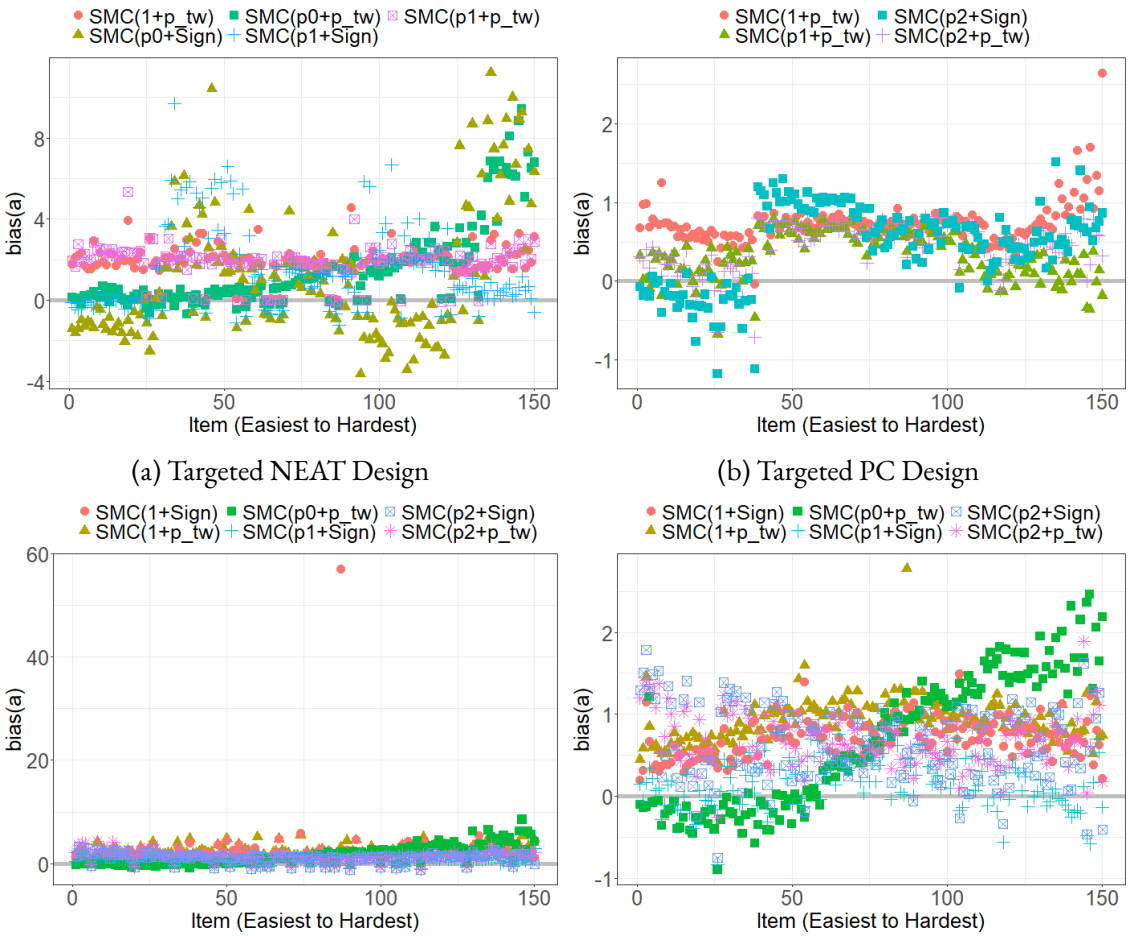
(a) Targeted NEAT Design

(b) Targeted PC Design

(c) Random NEAT Design

(d) Random PC Design

Figure C.7: Bias(b) per Item of Selected Modified Structured Matrix Completion in Practical Item Bank with Sample Size of 125



(a) Targeted NEAT Design

(b) Targeted PC Design

(c) Random NEAT Design

(d) Random PC Design

Figure C.8: Bias(a) per Item of Selected Modified Structured Matrix Completion in Practical Item Bank with Sample Size of 125

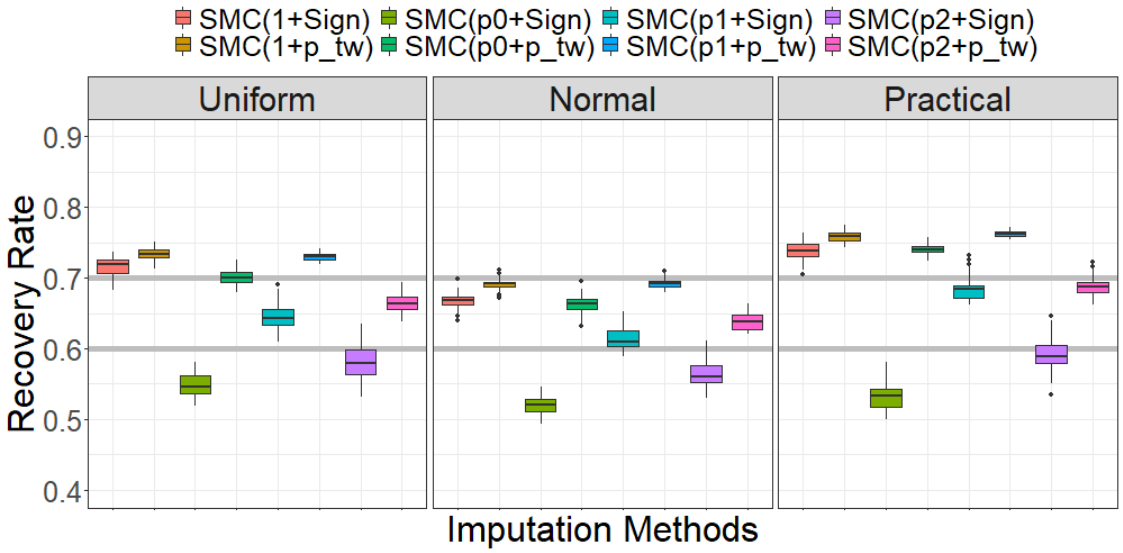


Figure C.9: Recovery Rates of Modified Structured Matrix Completion Across Item Banks in Random NEAT Designs and $N = 125$

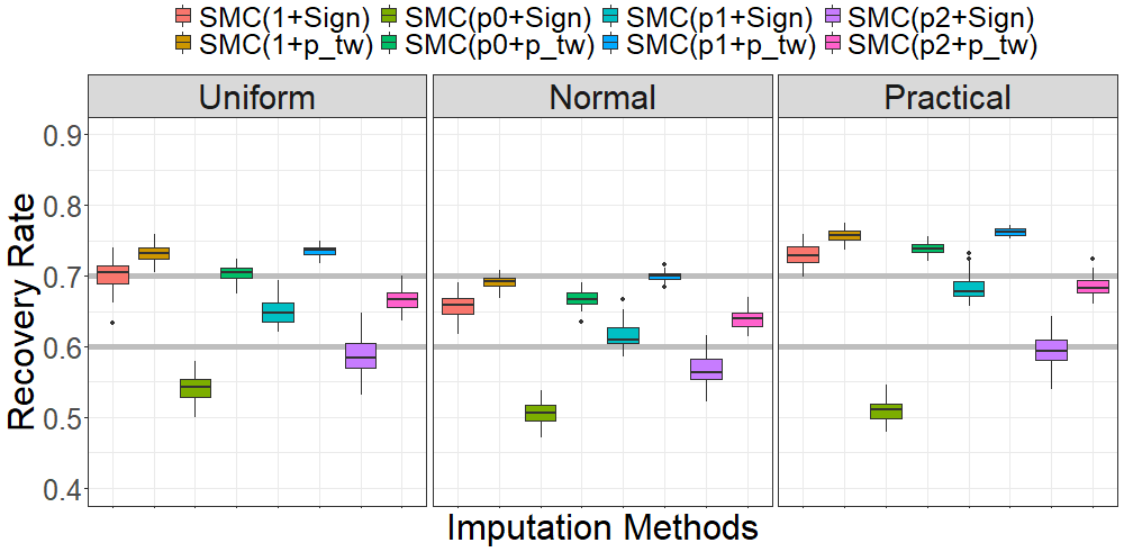


Figure C.10: Recovery Rates of Modified Structured Matrix Completion Across Item Banks in Random PC Designs and $N = 125$

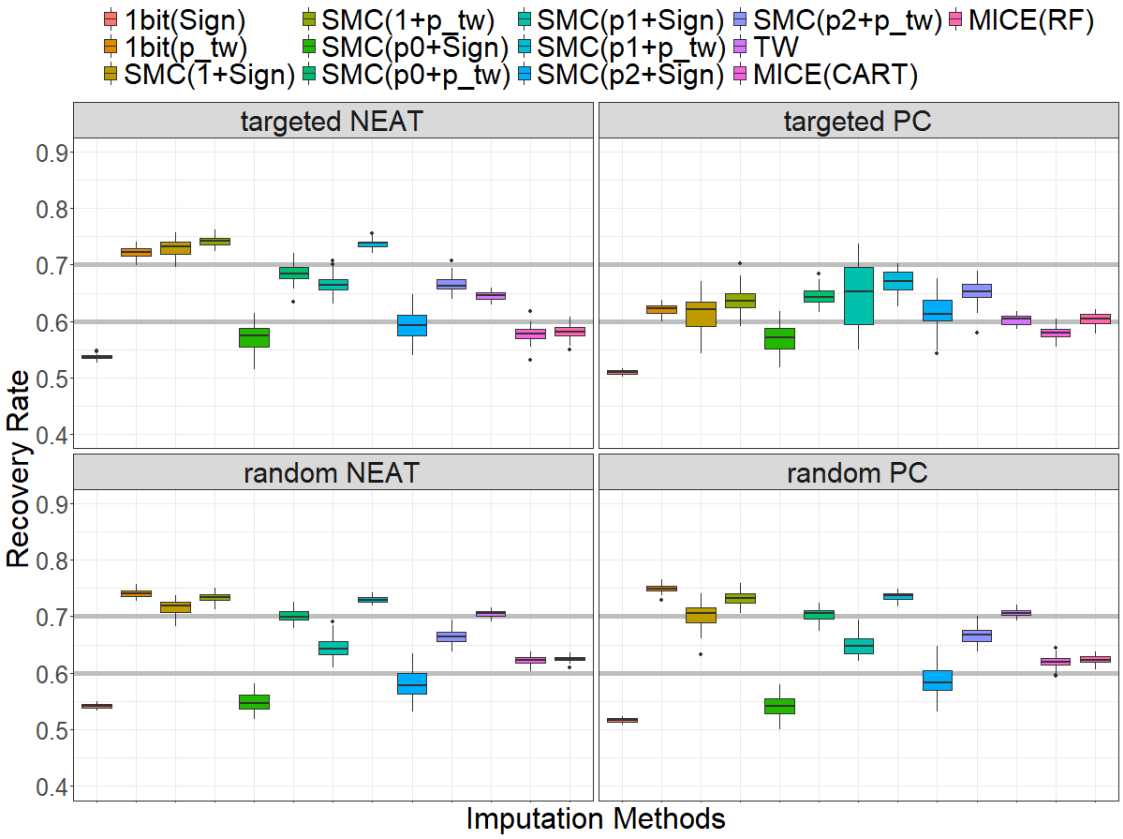


Figure C.11: Recovery Rates of All Methods in Uniform Item Bank and $N = 125$

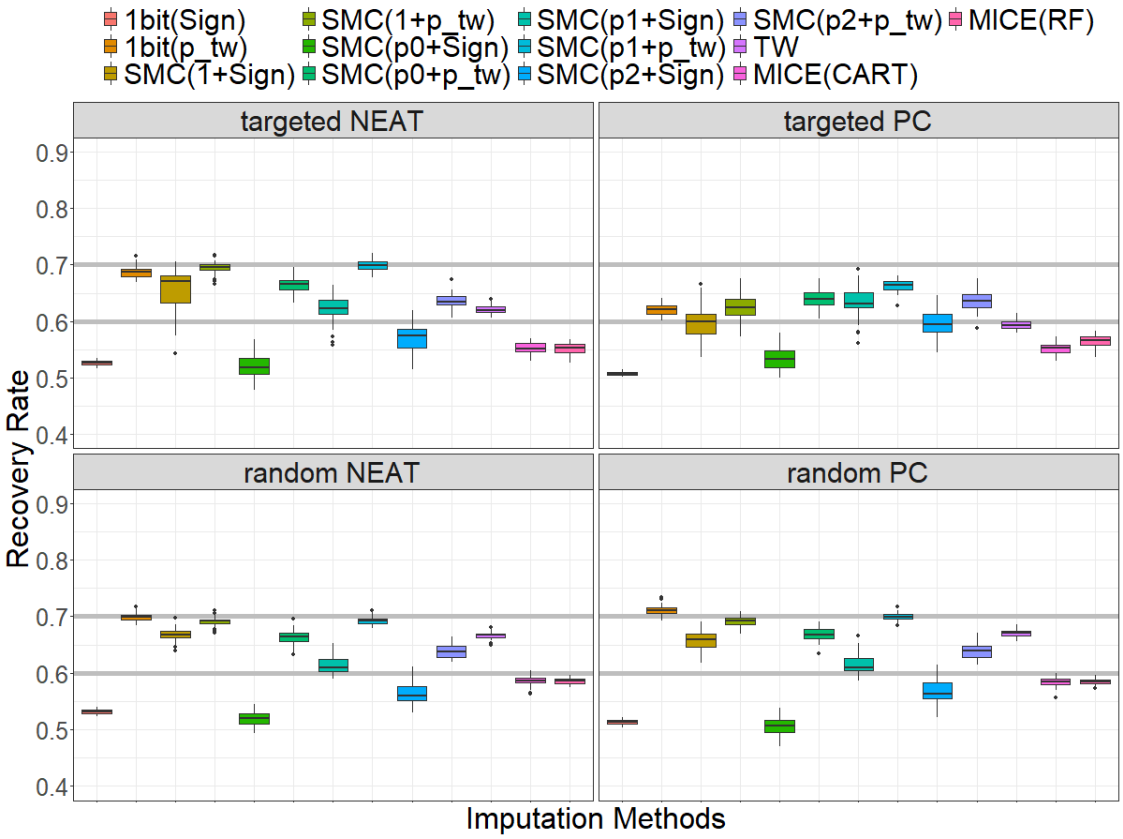


Figure C.12: Recovery Rates of All Methods in Normal Item Bank and $N = 125$

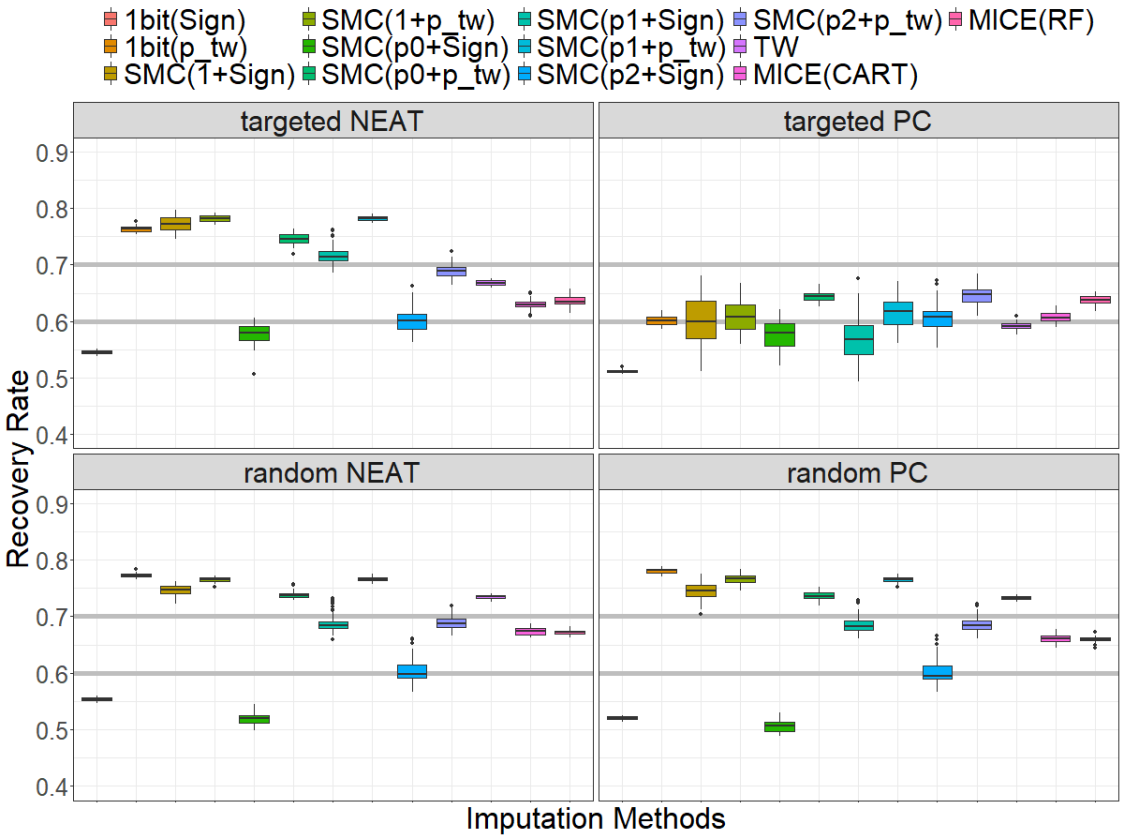


Figure C.13: Recovery Rates of All Methods in Practical Item Bank and $N = 250$

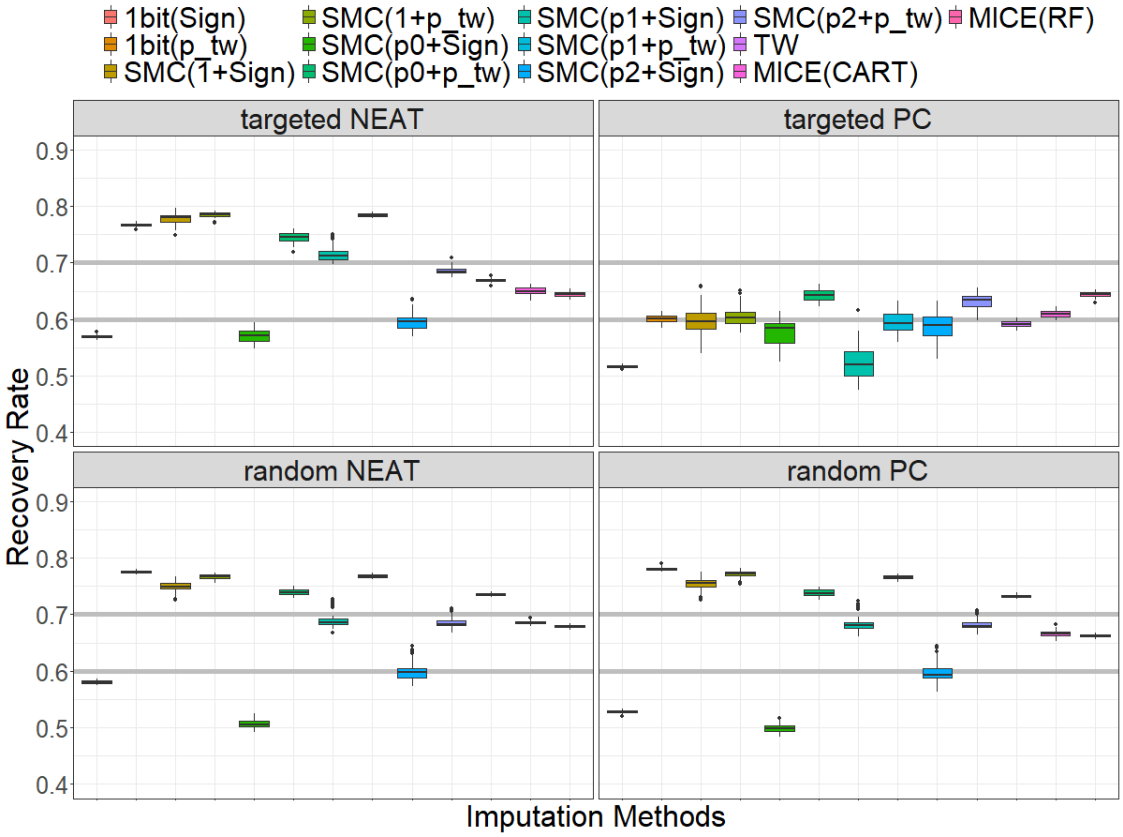
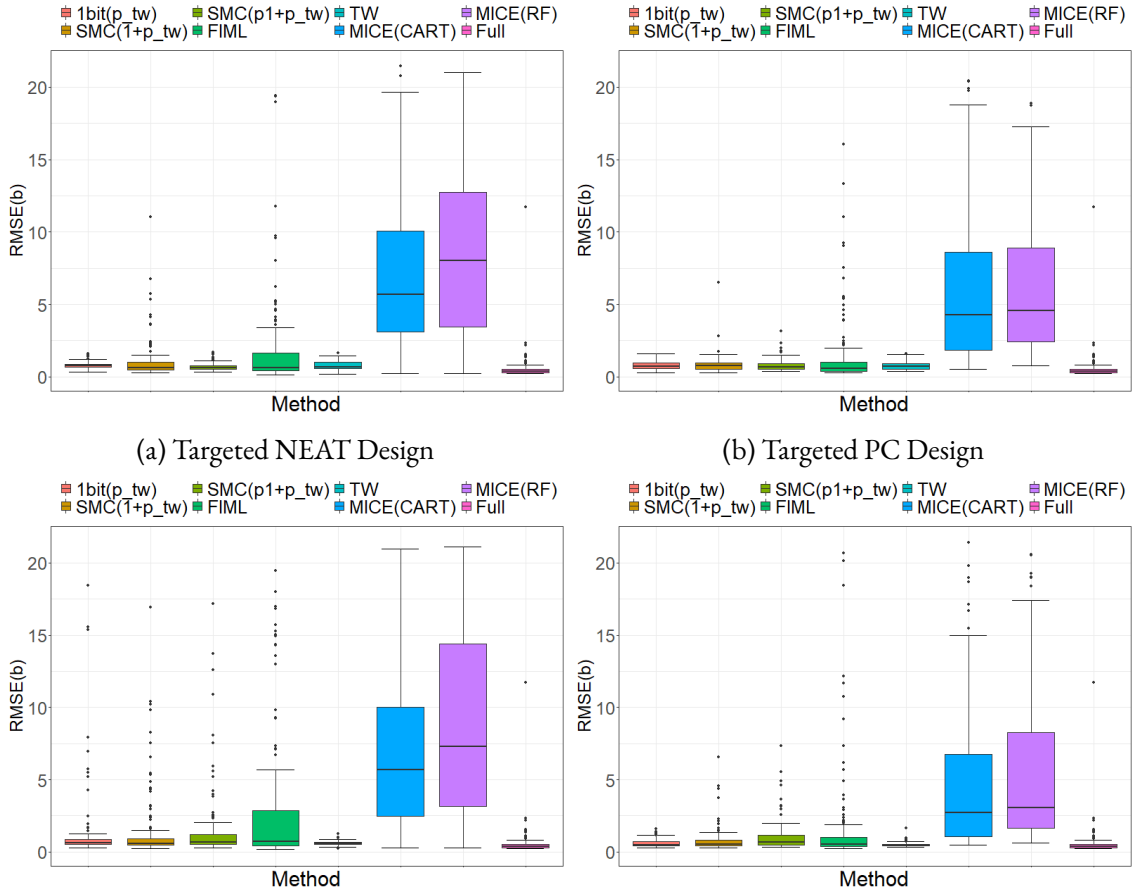


Figure C.14: Recovery Rates of All Methods in Practical Item Bank and $N = 500$



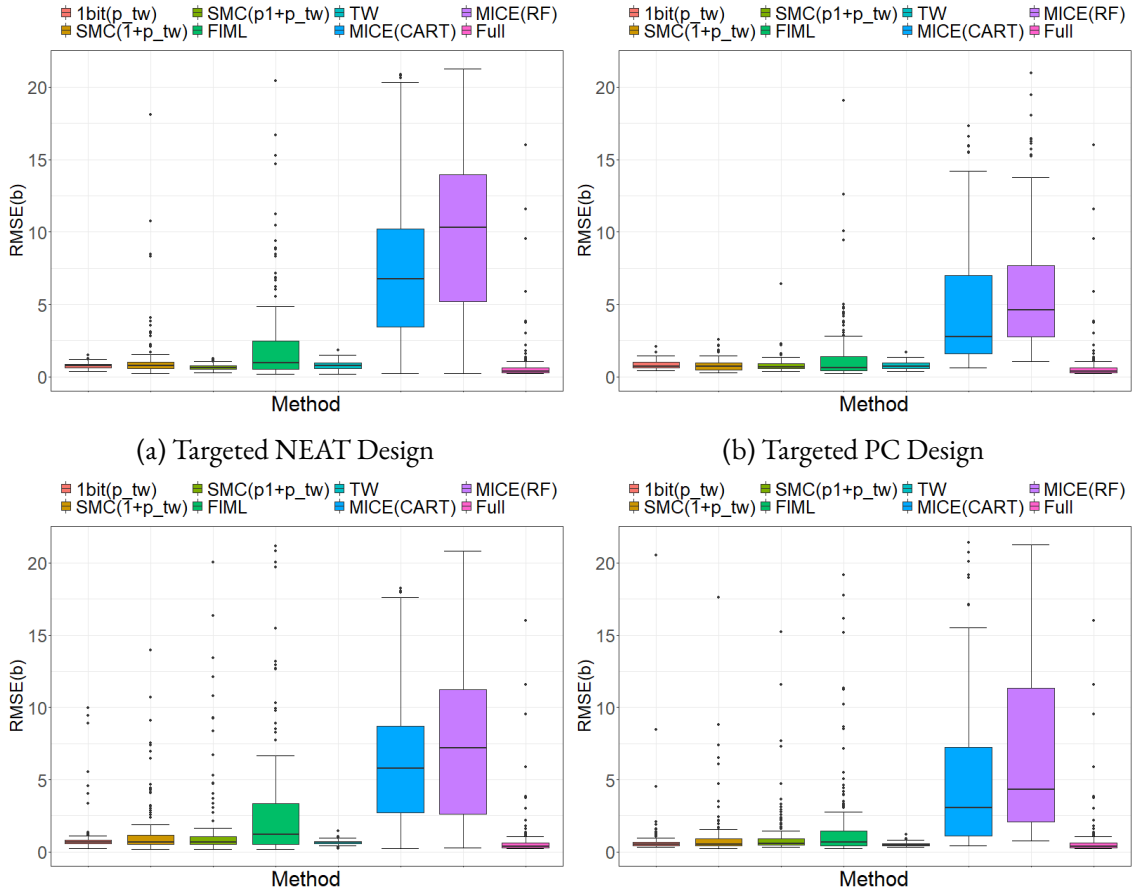
(a) Targeted NEAT Design

(b) Targeted PC Design

(c) Random NEAT Design

(d) Random PC Design

Figure C.15: RMSE(b) of Selected Methods in Uniform Item Bank with $N = 125$



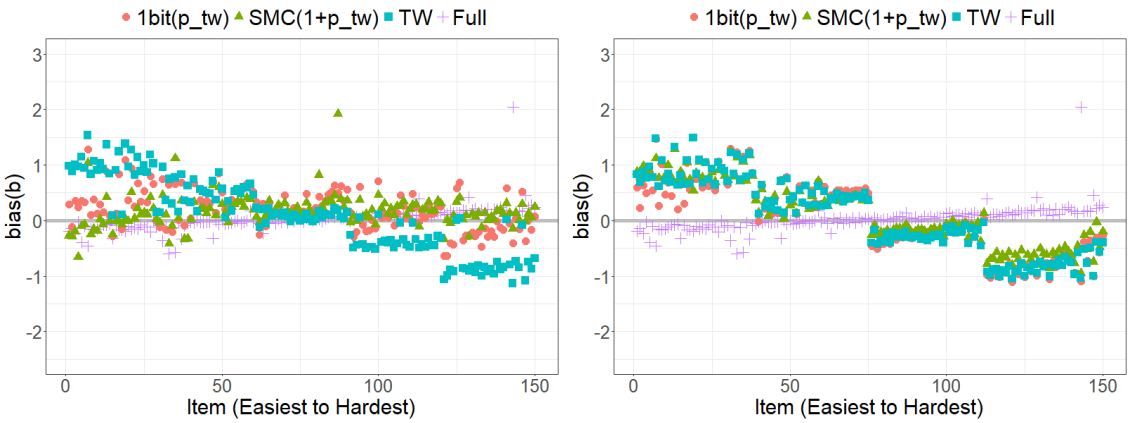
(a) Targeted NEAT Design

(b) Targeted PC Design

(c) Random NEAT Design

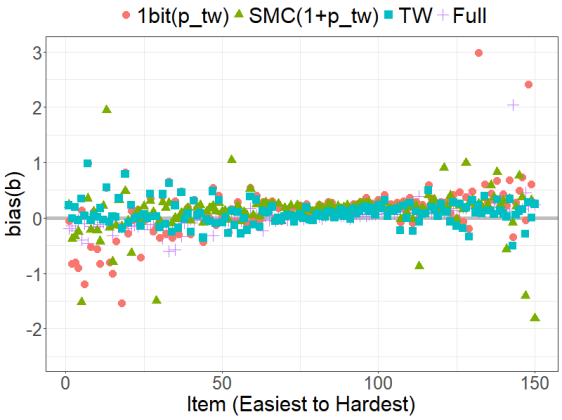
(d) Random PC Design

Figure C.16: RMSE(b) of Selected Methods in Normal Item Bank with $N = 125$

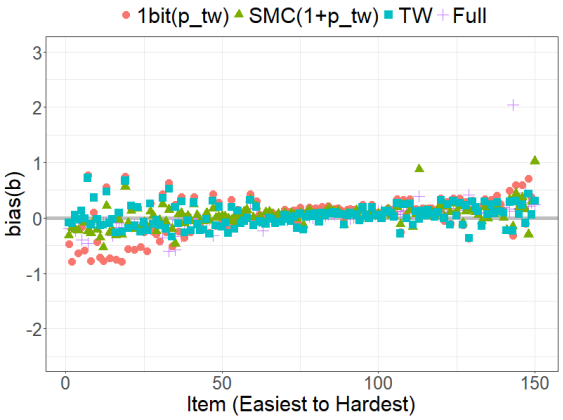


(a) Targeted NEAT Design

(b) Targeted PC Design

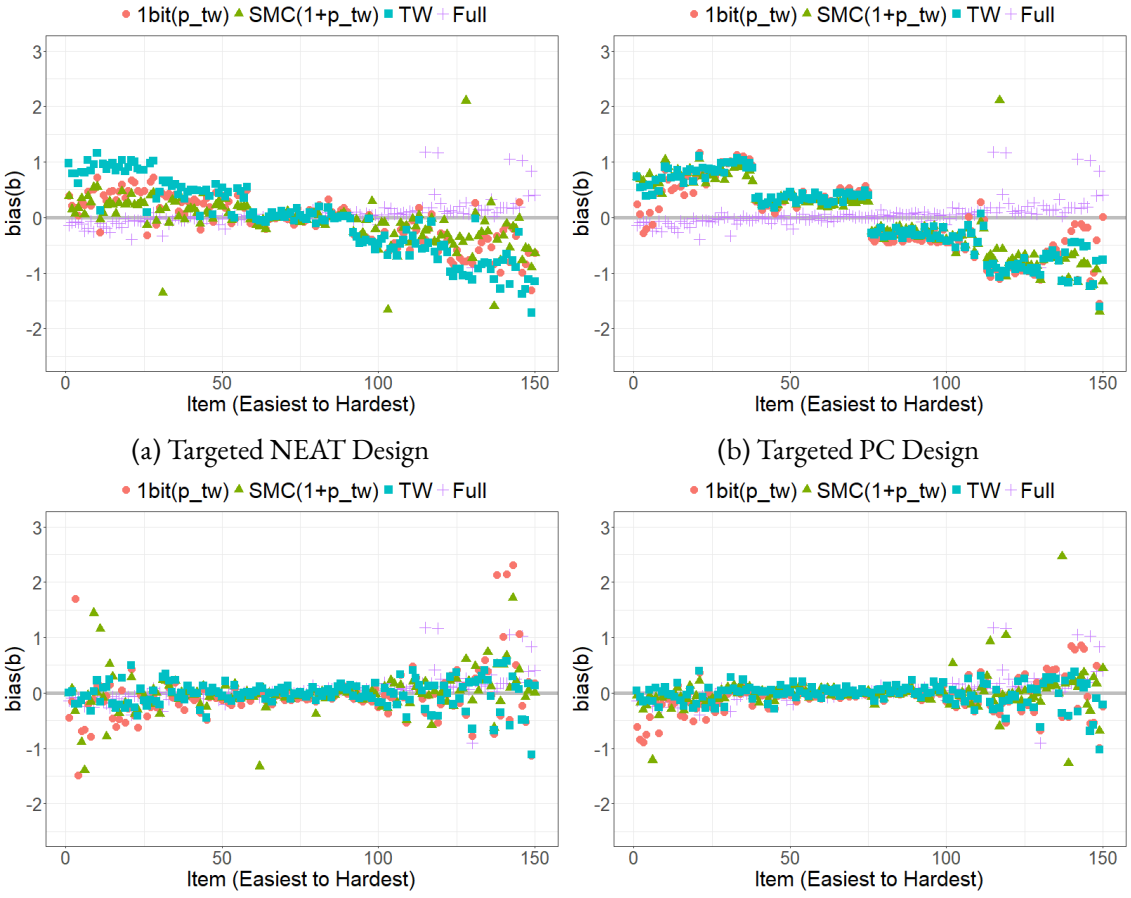


(c) Random NEAT Design



(d) Random PC Design

Figure C.17: Bias(b) of Selected Methods in Uniform Item Bank with $N = 125$



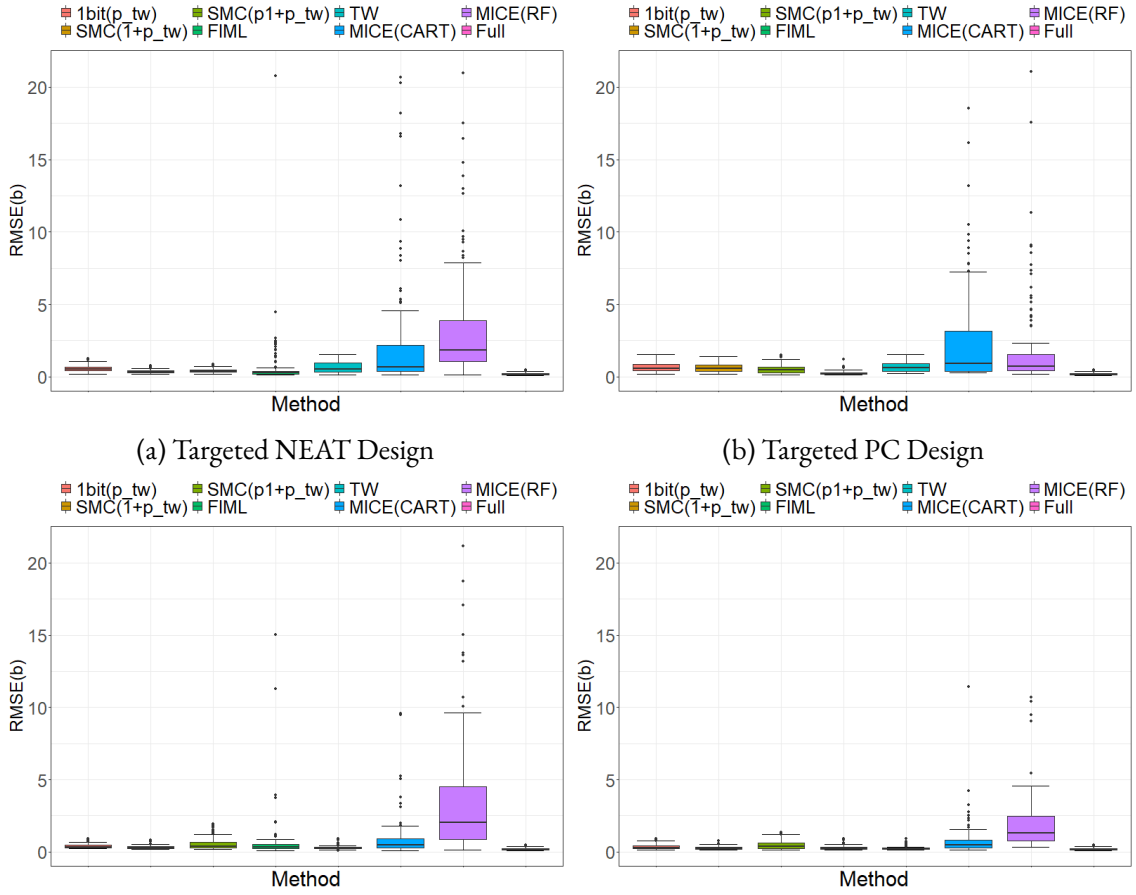
(a) Targeted NEAT Design

(b) Targeted PC Design

(c) Random NEAT Design

(d) Random PC Design

Figure C.18: Bias(b) of Selected Methods in Normal Item Bank with $N = 125$



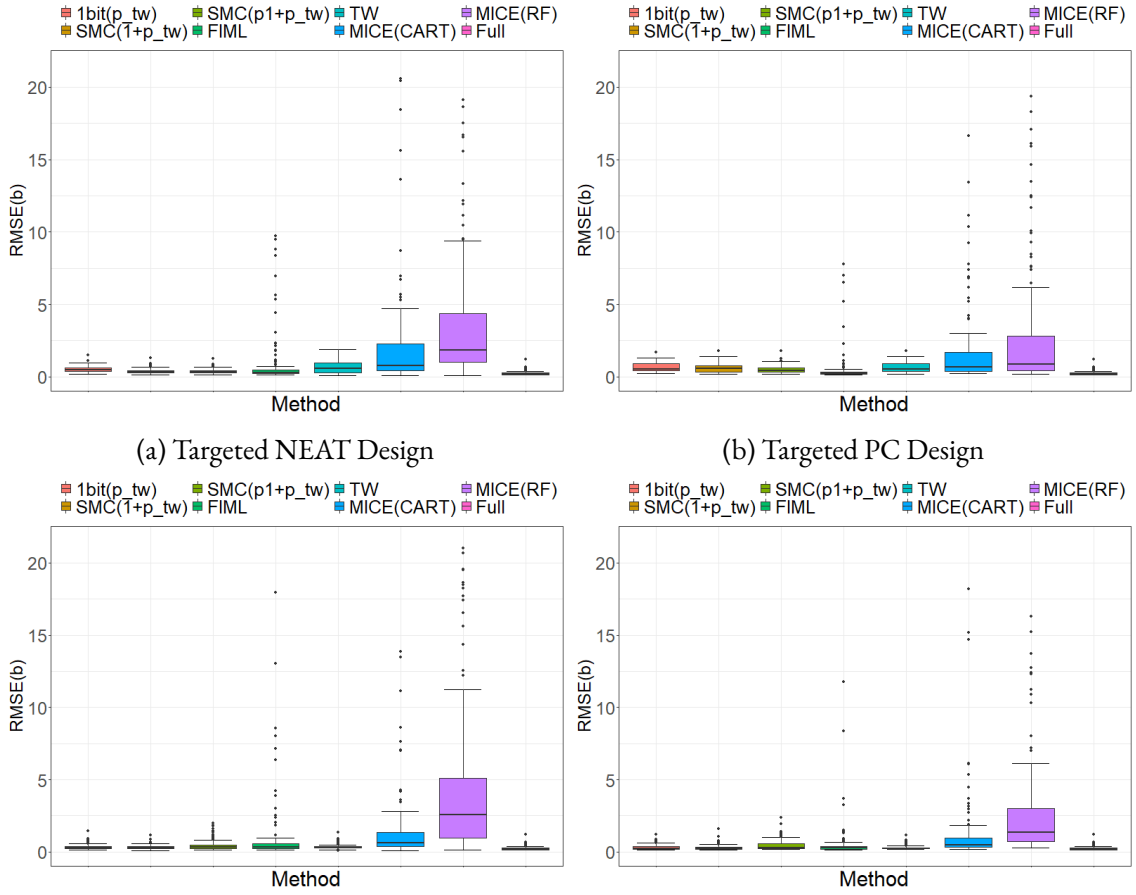
(a) Targeted NEAT Design

(b) Targeted PC Design

(c) Random NEAT Design

(d) Random PC Design

Figure C.19: RMSE(b) of Selected Methods in Uniform Item Bank with $N = 500$



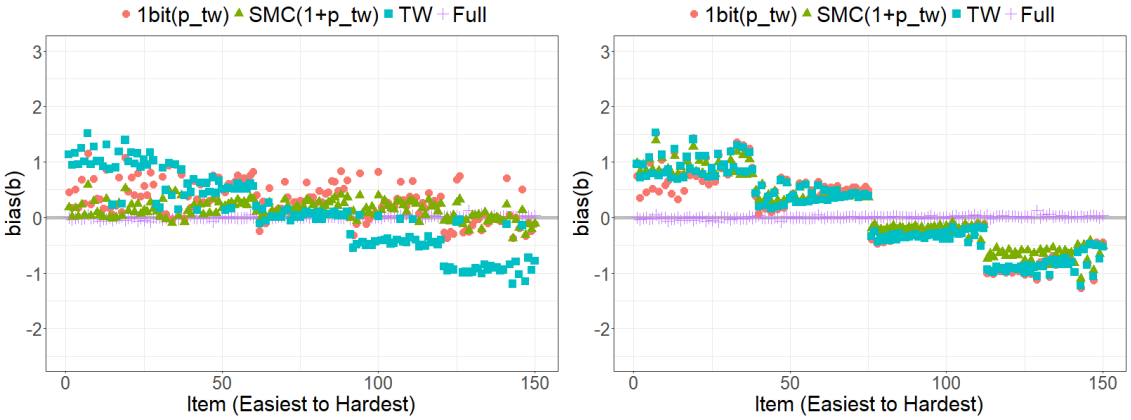
(a) Targeted NEAT Design

(b) Targeted PC Design

(c) Random NEAT Design

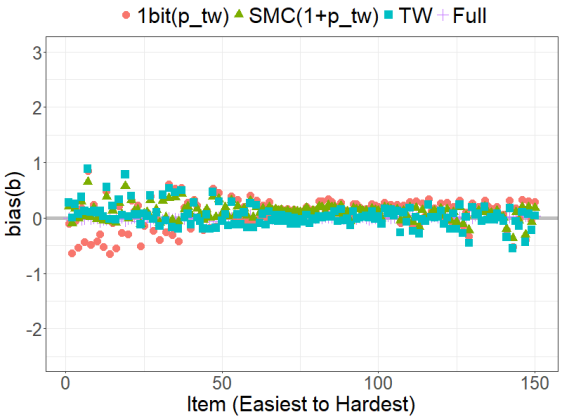
(d) Random PC Design

Figure C.20: RMSE(b) of Selected Methods in Normal Item Bank with $N = 500$

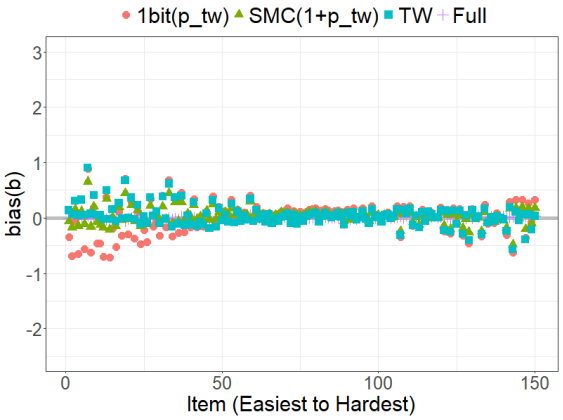


(a) Targeted NEAT Design

(b) Targeted PC Design

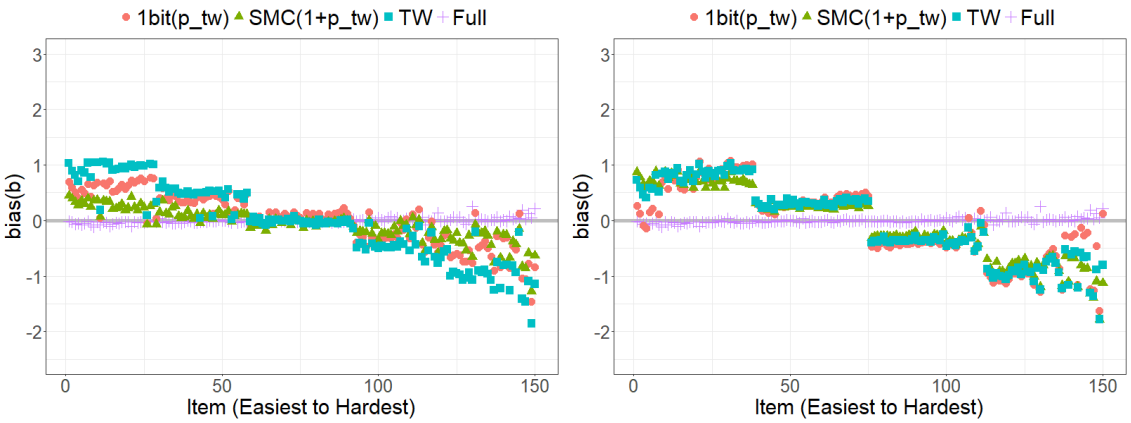


(c) Random NEAT Design



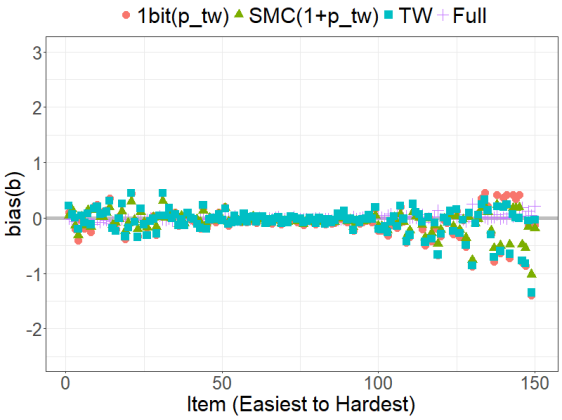
(d) Random PC Design

Figure C.21: Bias(b) of Selected Methods in Uniform Item Bank with $N = 500$

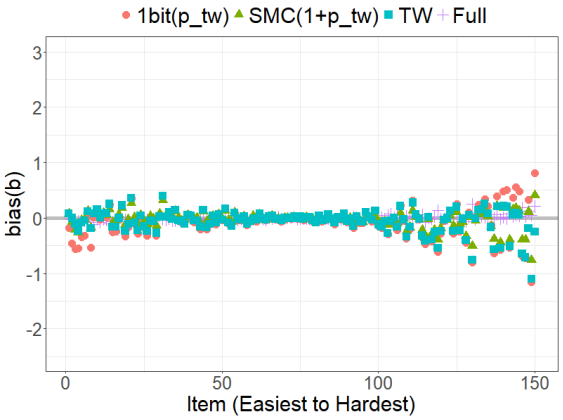


(a) Targeted NEAT Design

(b) Targeted PC Design

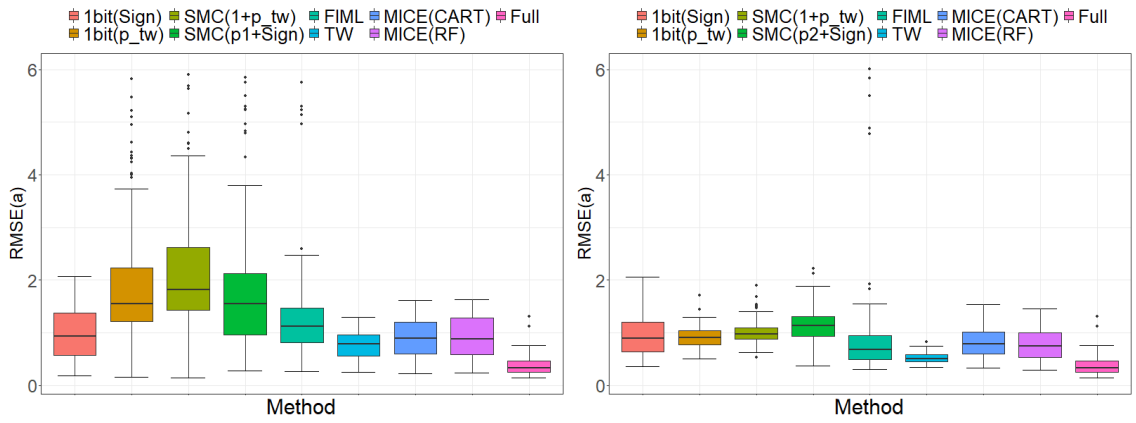


(c) Random NEAT Design



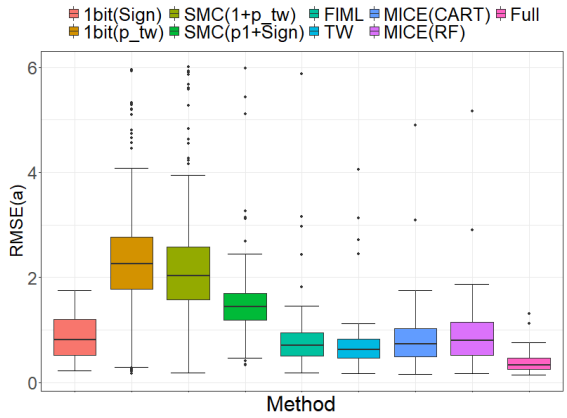
(d) Random PC Design

Figure C.22: Bias(b) of Selected Methods in Normal Item Bank with $N = 500$

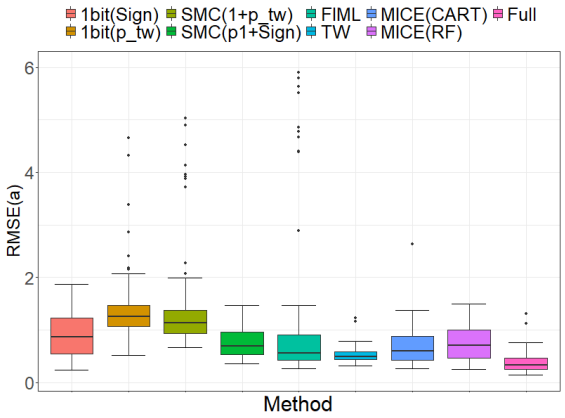


(a) Targeted NEAT Design

(b) Targeted PC Design

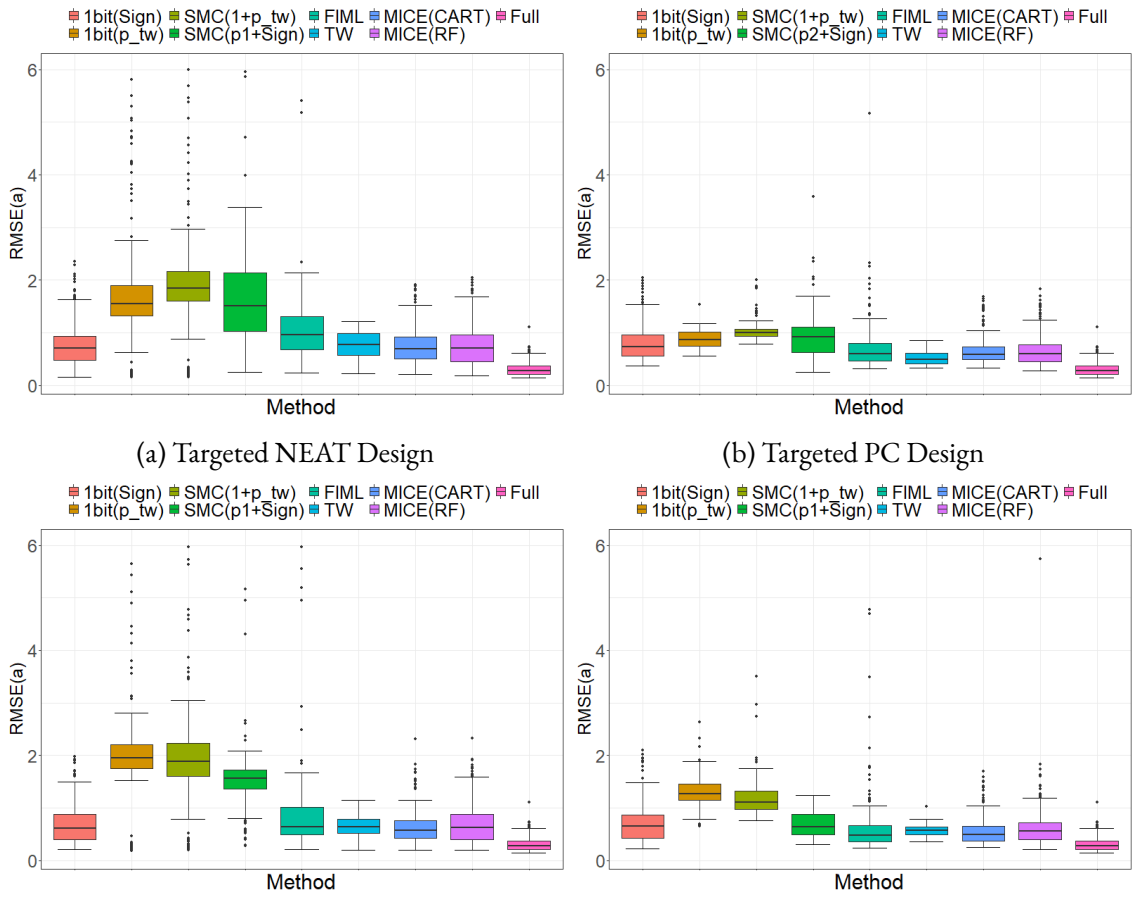


(c) Random NEAT Design



(d) Random PC Design

Figure C.23: RMSE(a) of Selected Methods in Uniform Item Bank with Sample Size of 125



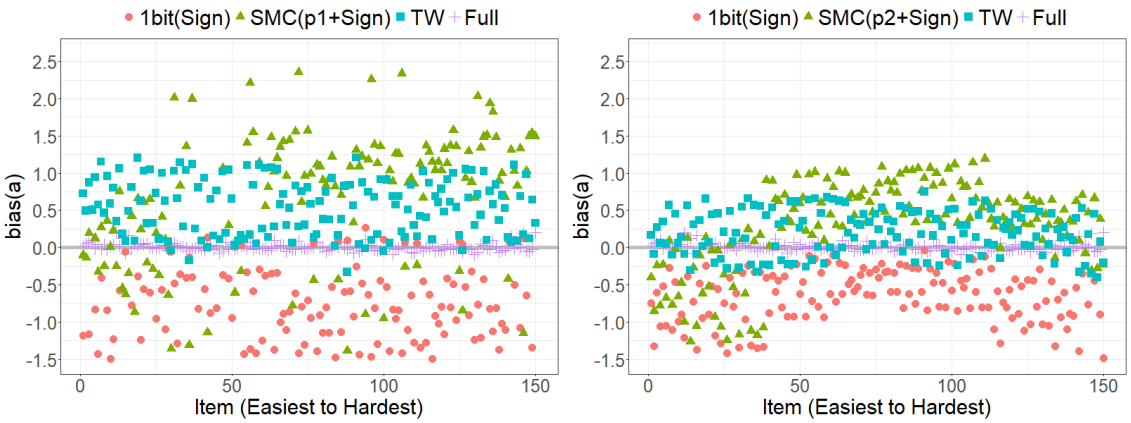
(a) Targeted NEAT Design

(b) Targeted PC Design

(c) Random NEAT Design

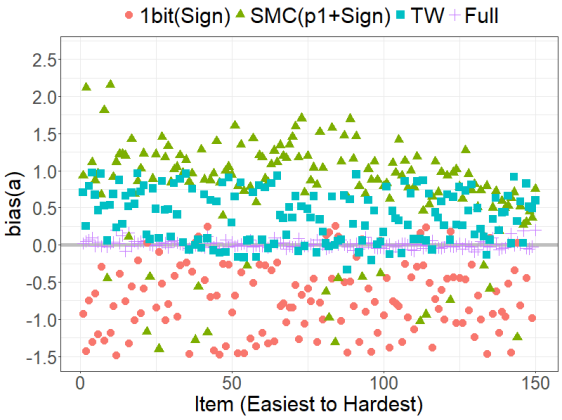
(d) Random PC Design

Figure C.24: RMSE(a) of Selected Methods in Normal Item Bank with Sample Size of 125

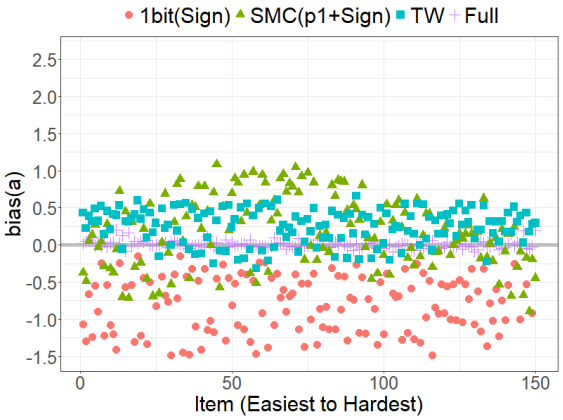


(a) Targeted NEAT Design

(b) Targeted PC Design

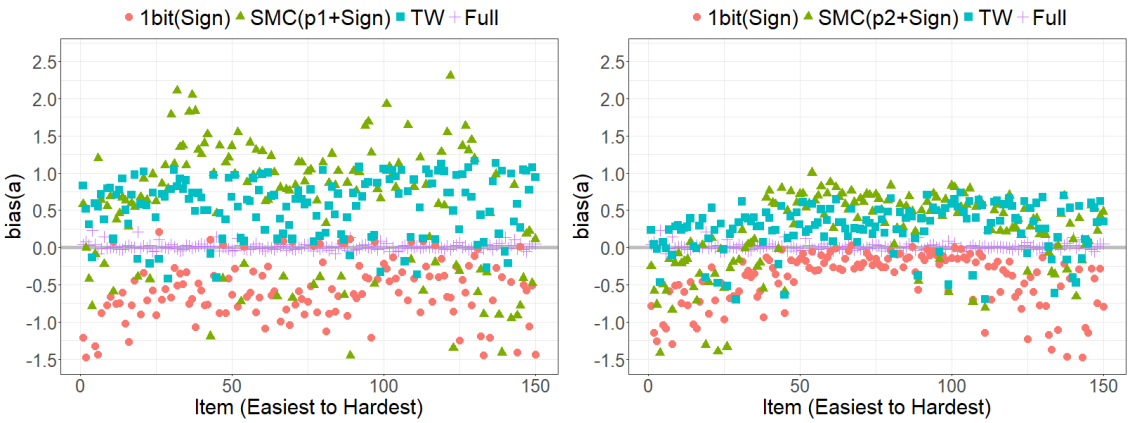


(c) Random NEAT Design



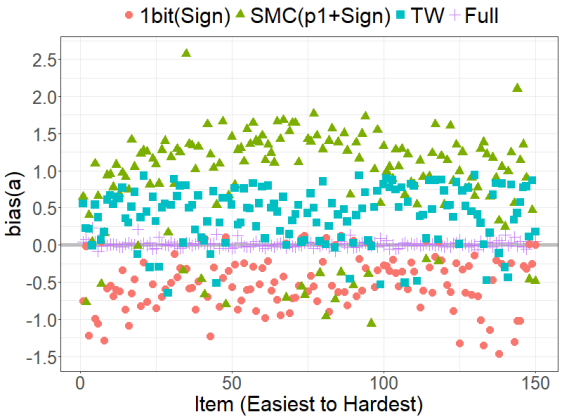
(d) Random PC Design

Figure C.25: Bias(a) of Selected Methods in Uniform Item Bank with Sample Size of 125

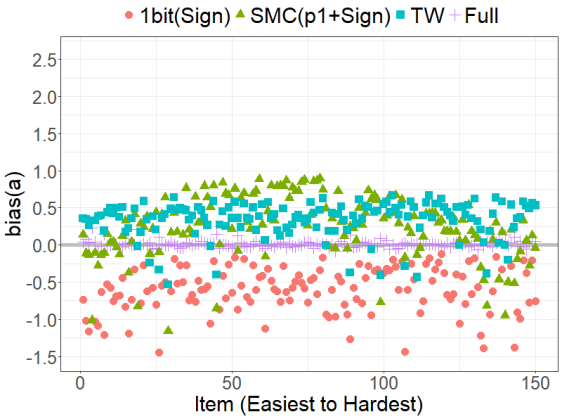


(a) Targeted NEAT Design

(b) Targeted PC Design

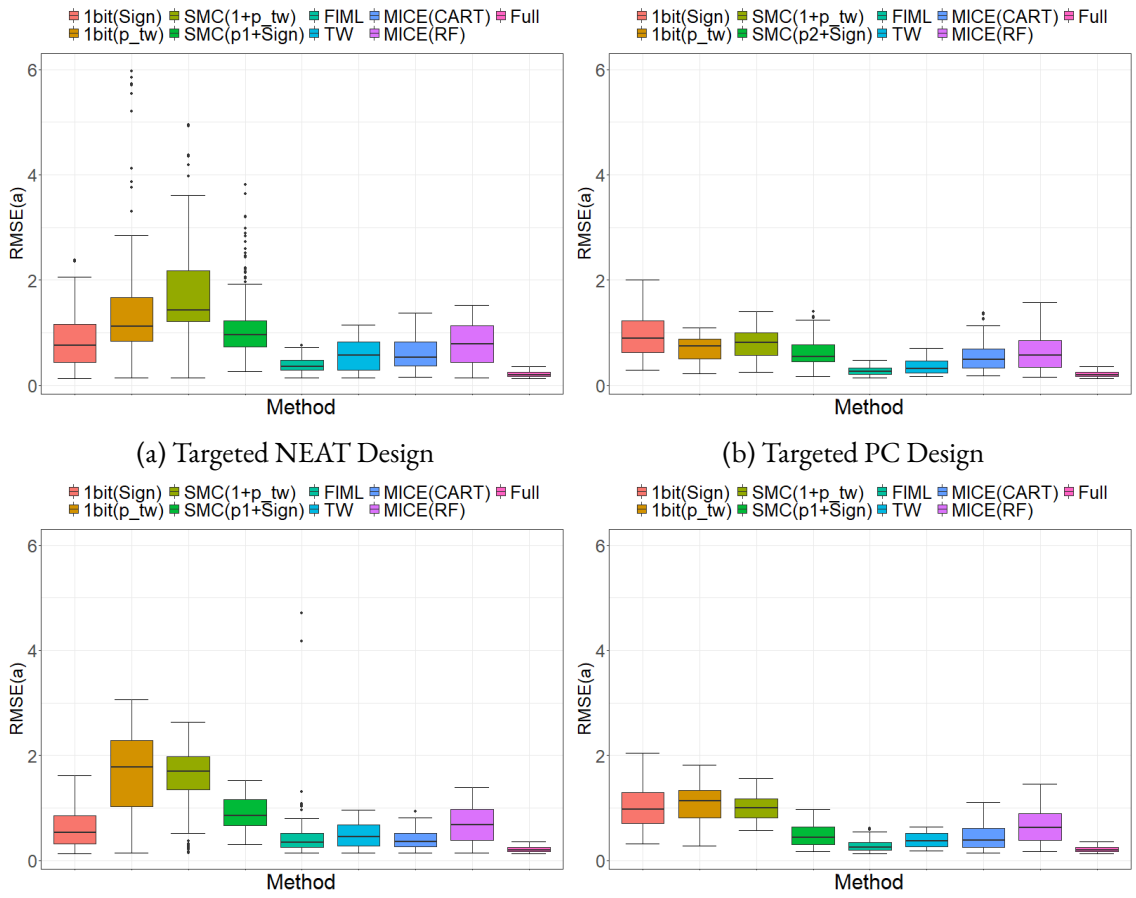


(c) Random NEAT Design



(d) Random PC Design

Figure C.26: Bias(a) of Selected Methods in Normal Item Bank with Sample Size of 125



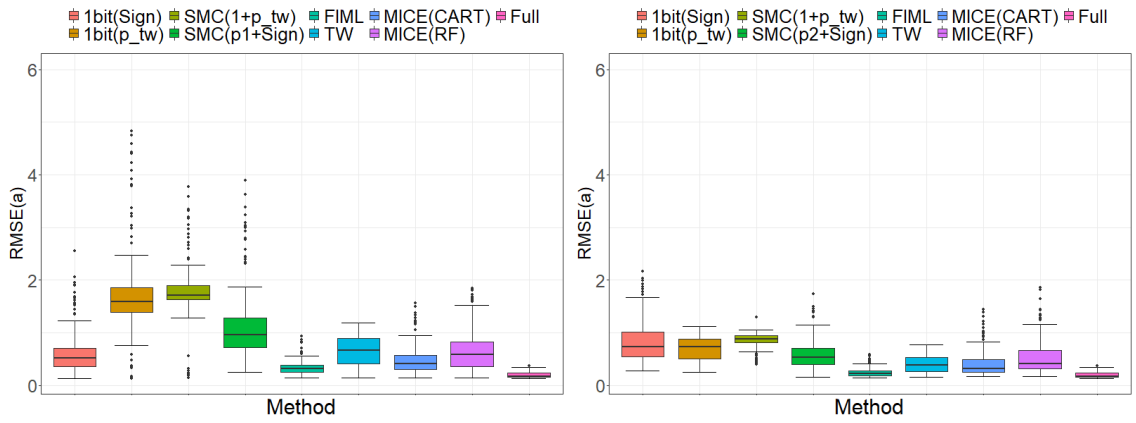
(a) Targeted NEAT Design

(b) Targeted PC Design

(c) Random NEAT Design

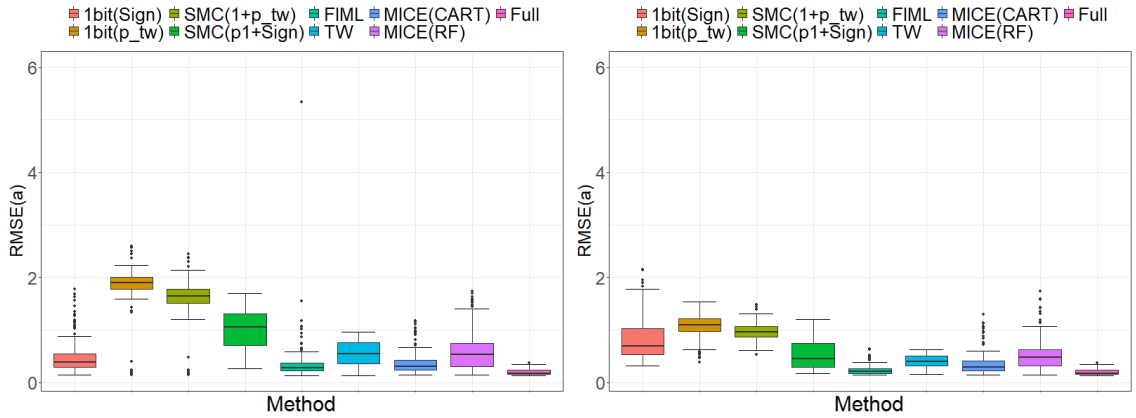
(d) Random PC Design

Figure C.27: RMSE(a) of Selected Methods in Uniform Item Bank with Sample Size of 500



(a) Targeted NEAT Design

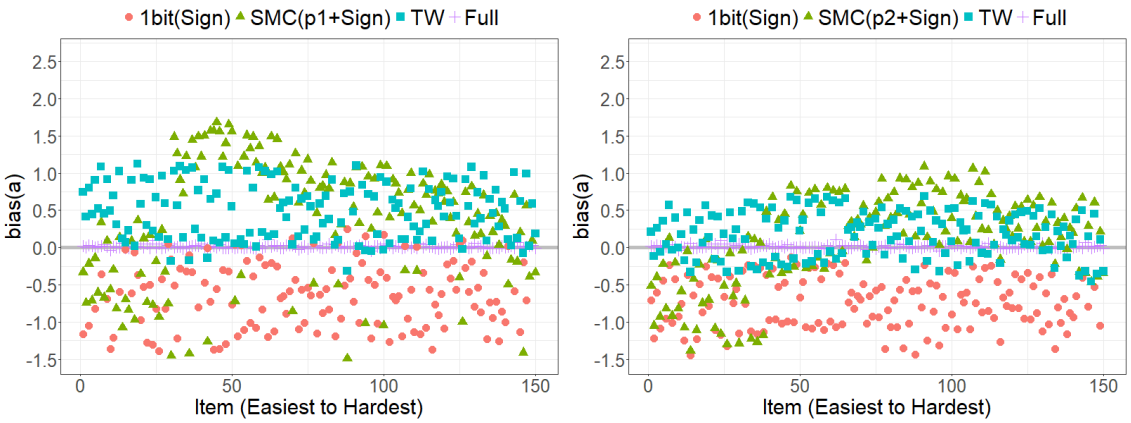
(b) Targeted PC Design



(c) Random NEAT Design

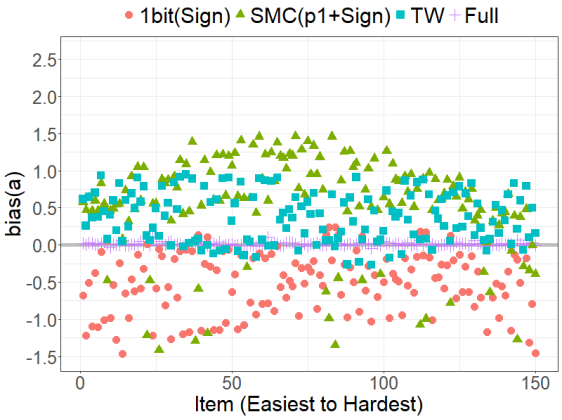
(d) Random PC Design

Figure C.28: RMSE(a) of Selected Methods in Normal Item Bank with Sample Size of 500

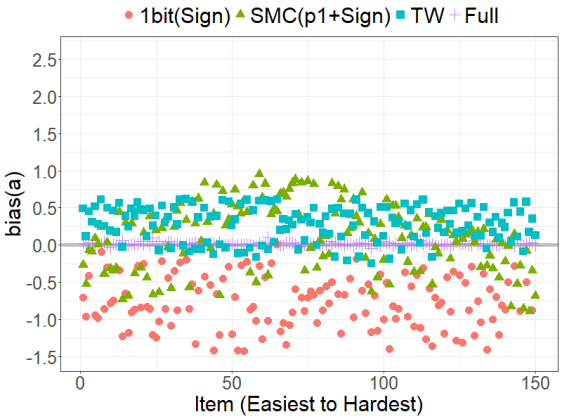


(a) Targeted NEAT Design

(b) Targeted PC Design

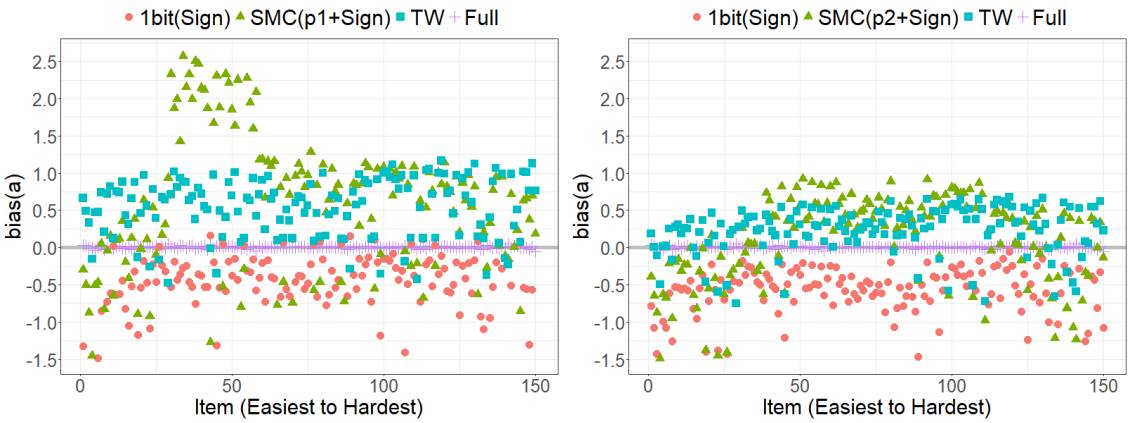


(c) Random NEAT Design



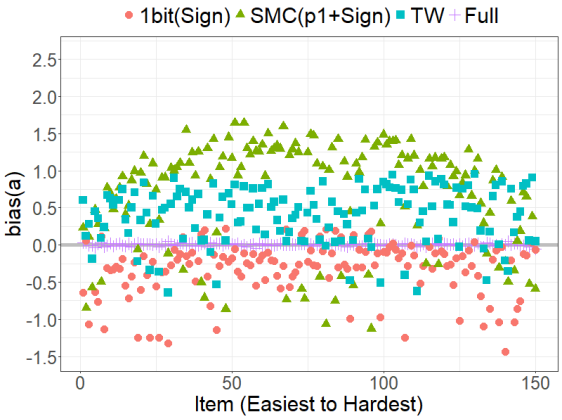
(d) Random PC Design

Figure C.29: Bias(a) of Selected Methods in Uniform Item Bank with Sample Size of 500

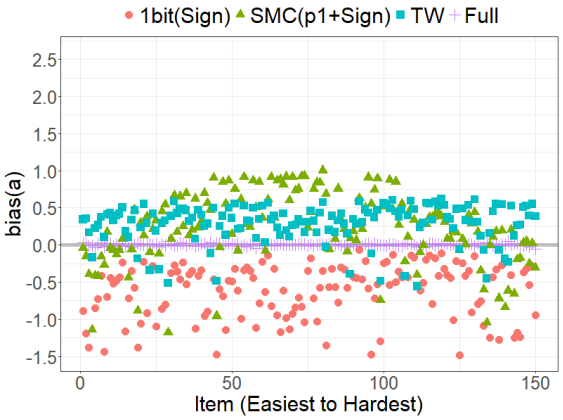


(a) Targeted NEAT Design

(b) Targeted PC Design



(c) Random NEAT Design



(d) Random PC Design

Figure C.30: Bias(a) of Selected Methods in Normal Item Bank with Sample Size of 500

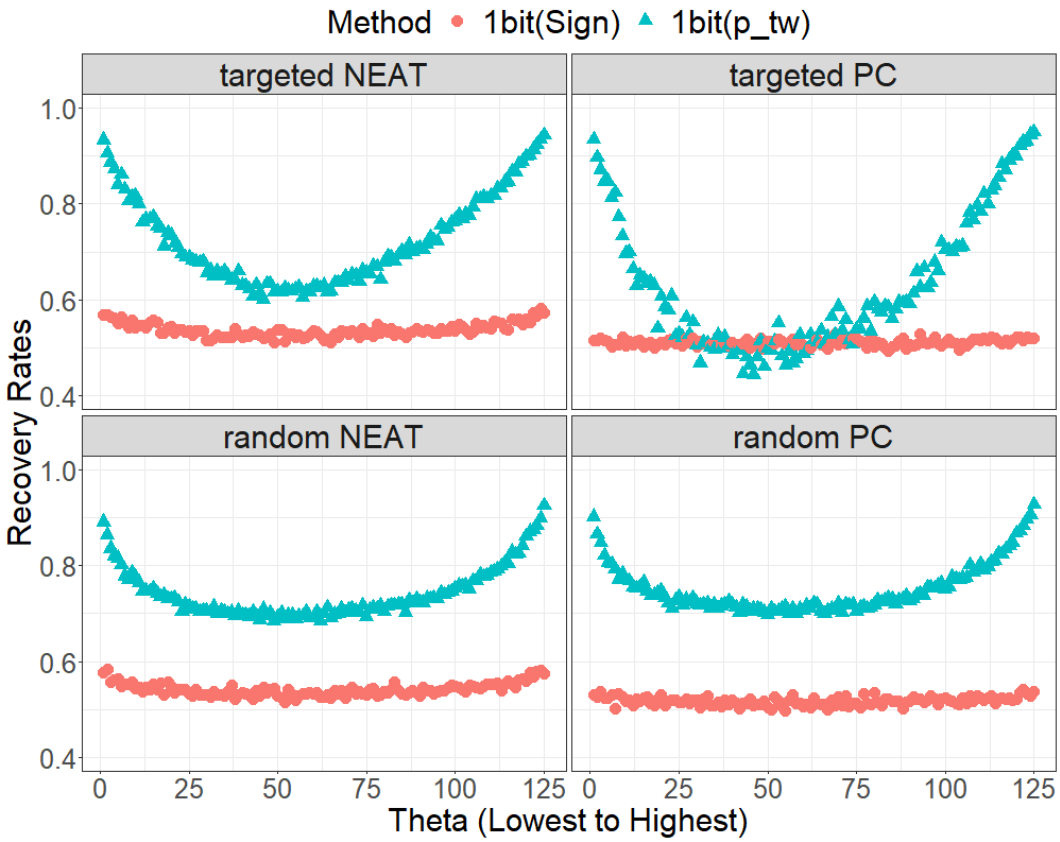


Figure C.31: Recovery Rates Per Examinees Using *1bit(p_tw)* and *1bit(Sign)* in Uniform Item Bank and $N = 125$

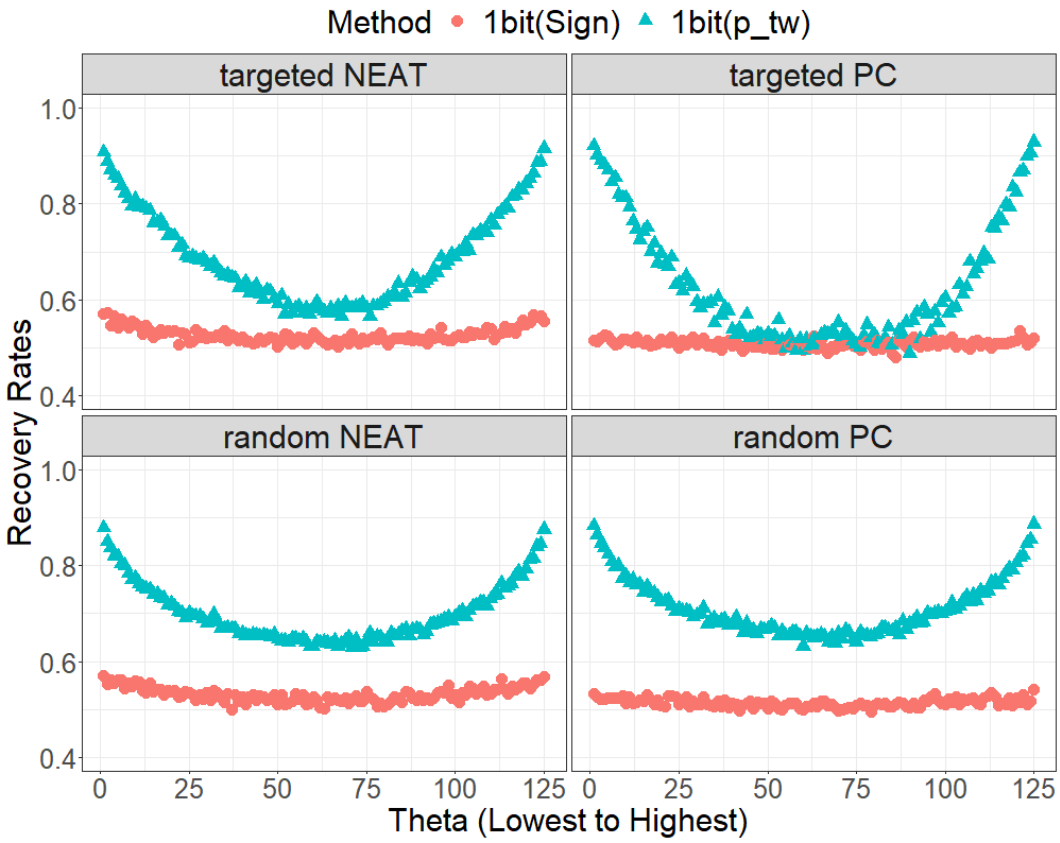


Figure C.32: Recovery Rates Per Examinees Using *1bit(p_tw)* and *1bit(Sign)* in Normal Item Bank and $N = 125$

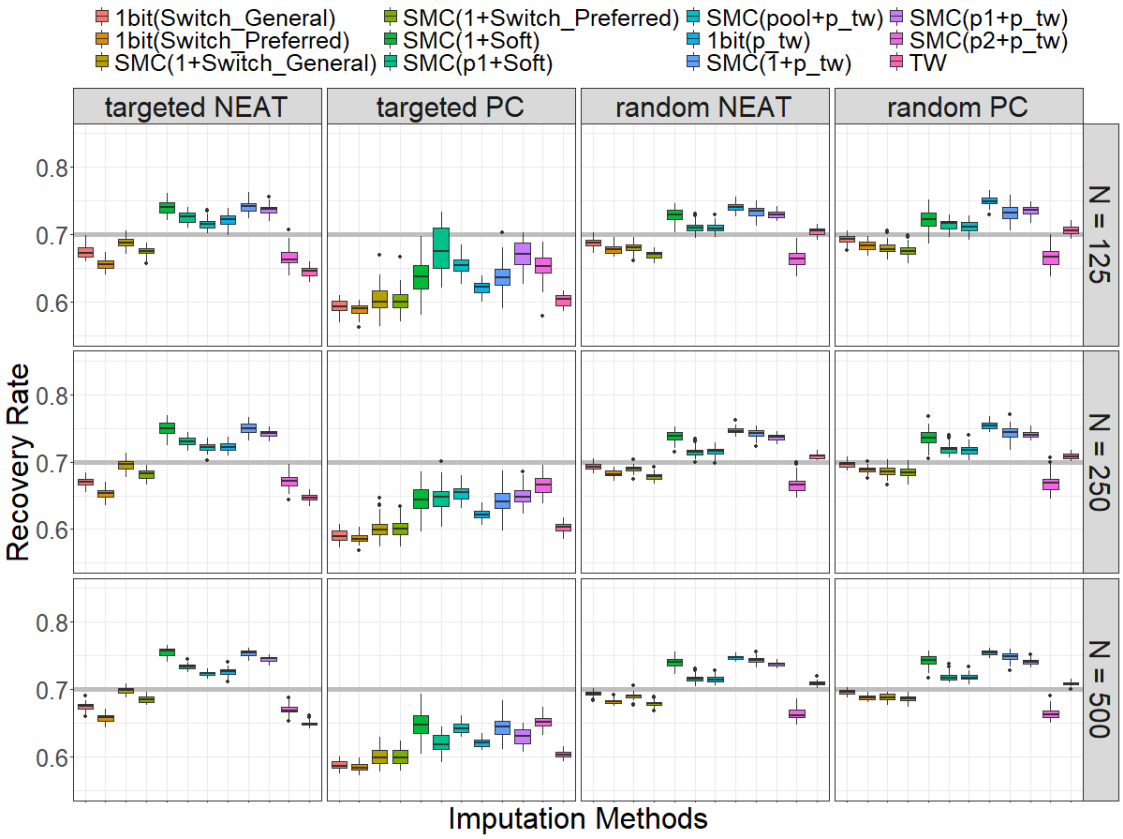


Figure C.33: Recovery Rates of All Methods in the Uniform Item Bank Across Incomplete Calibration Designs and Sample Sizes

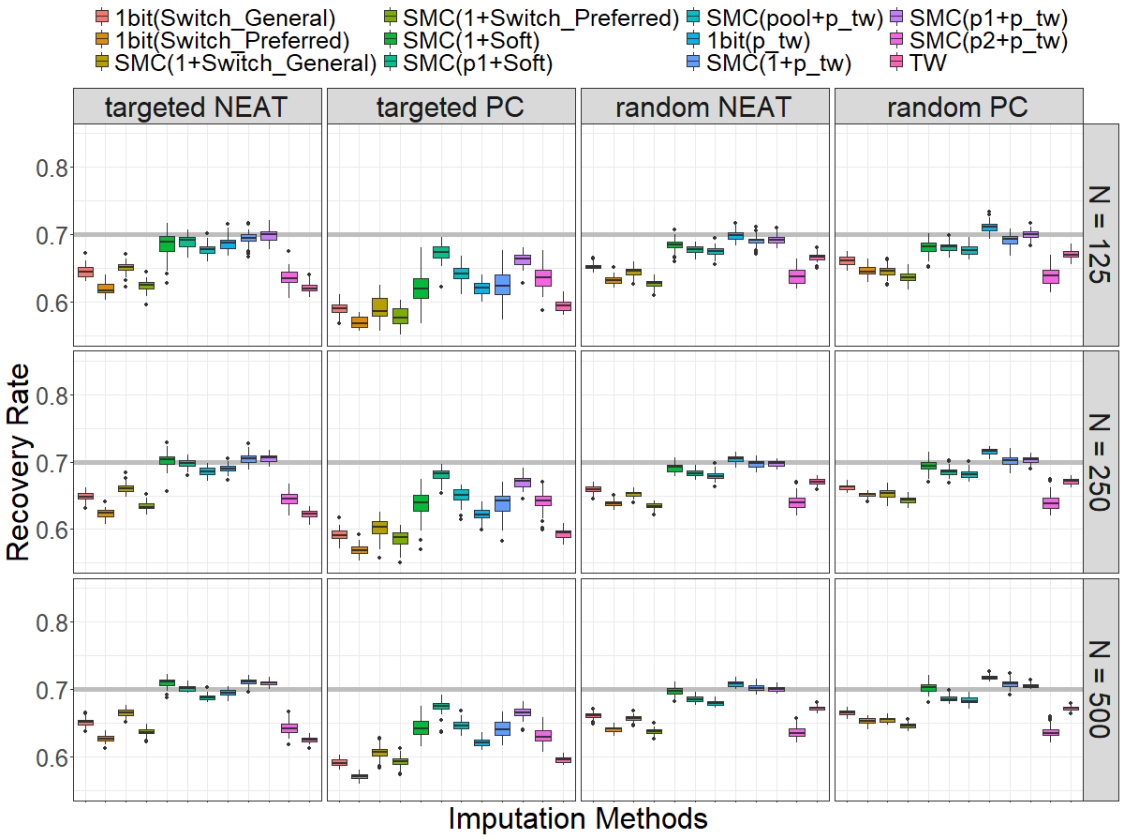


Figure C.34: Recovery Rates of All Methods in the Normal Item Bank Across Incomplete Calibration Designs and Sample Sizes

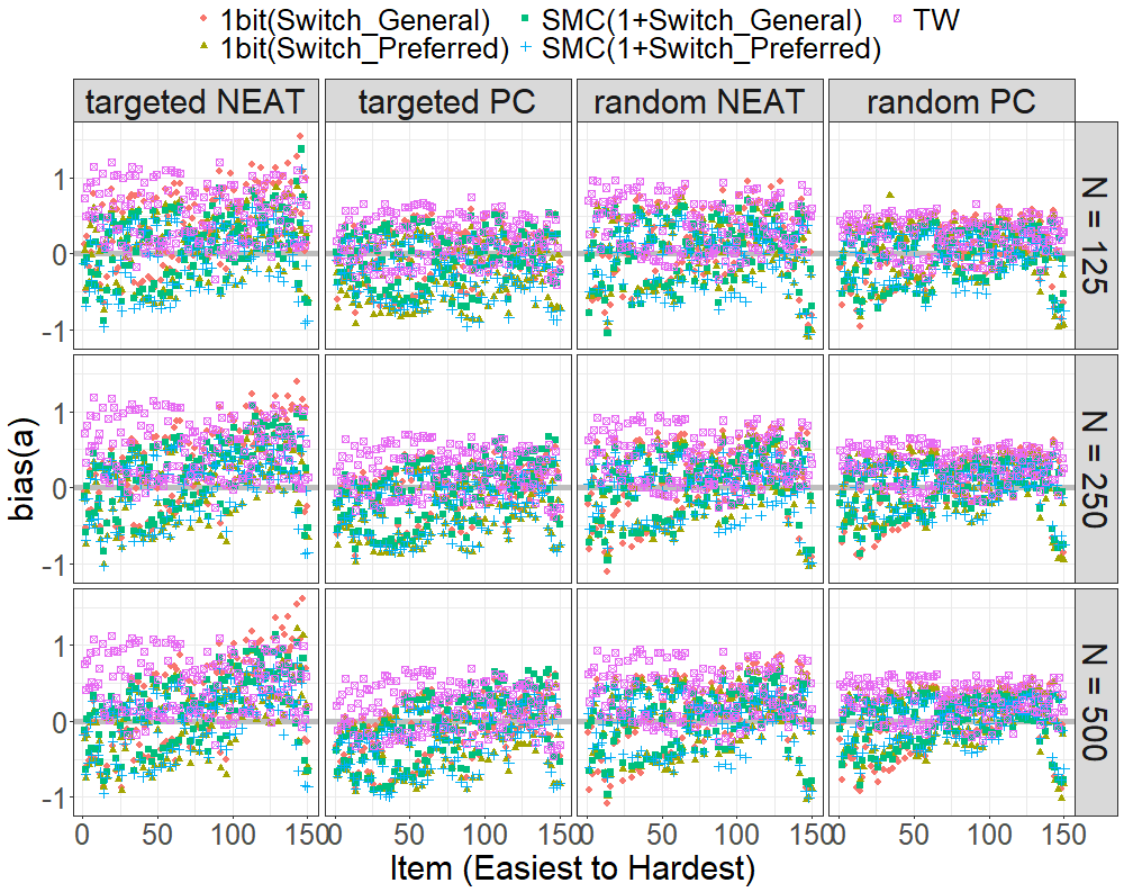


Figure C.35: Bias(a) Per Item of Switching Methods in Uniform Item Bank Across Incomplete Calibration Designs and Sample Sizes.

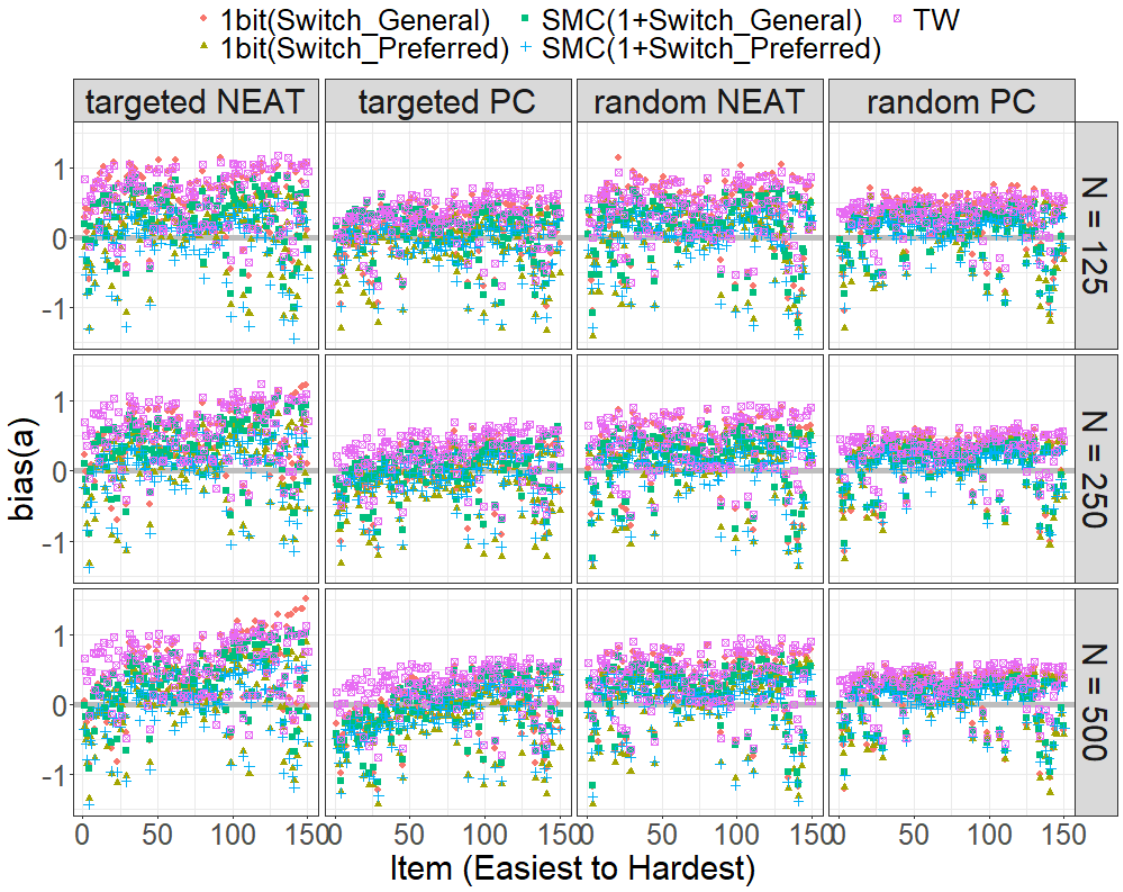


Figure C.36: Bias(a) Per Item of Switching Methods in Normal Item Bank Across Incomplete Calibration Designs and Sample Sizes.

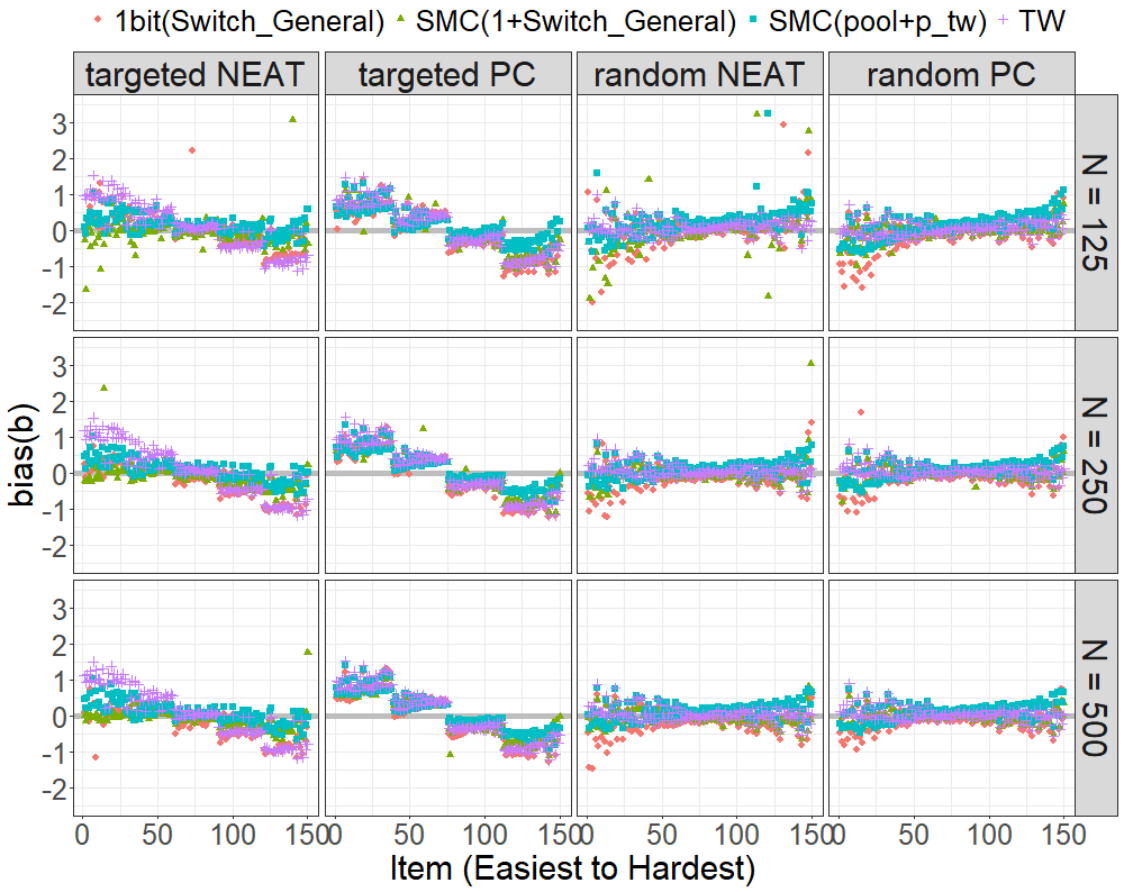


Figure C.37: Bias(b) Per Item of Selected Methods in Uniform Item Bank with $N = 125$.

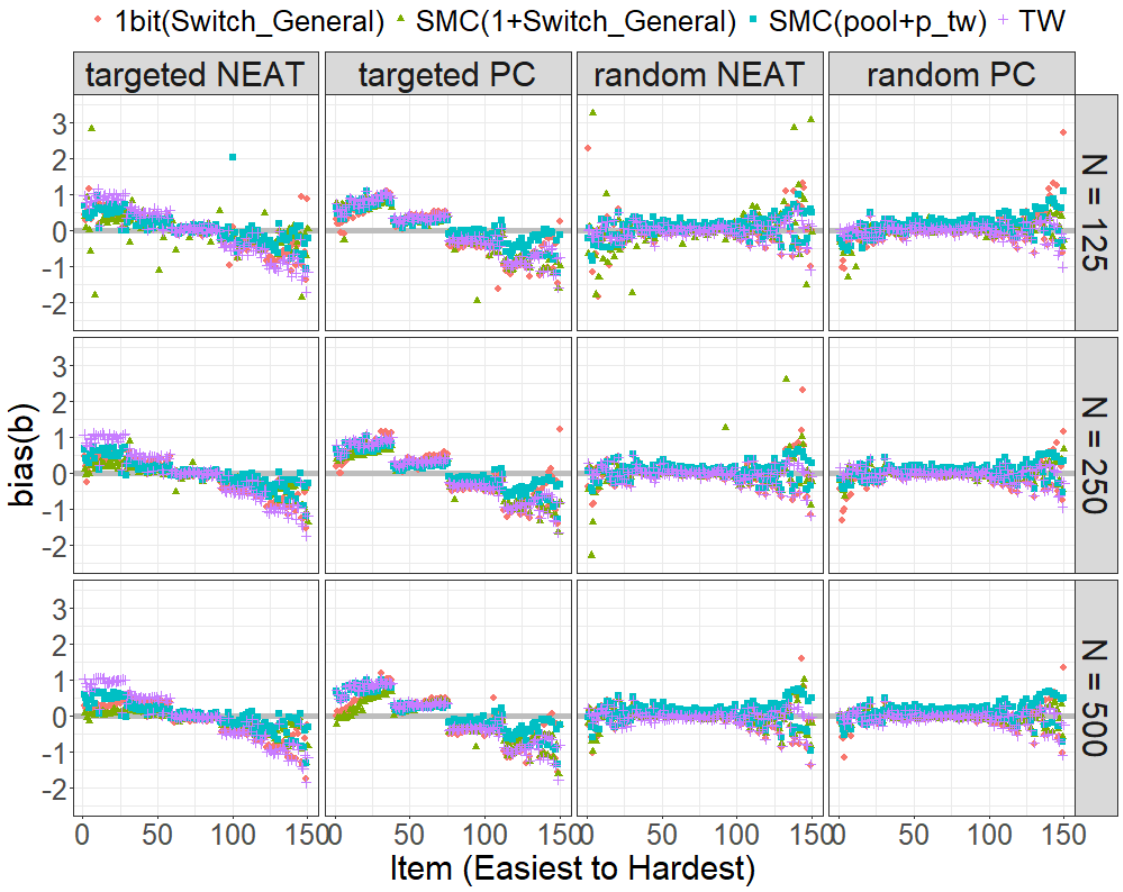
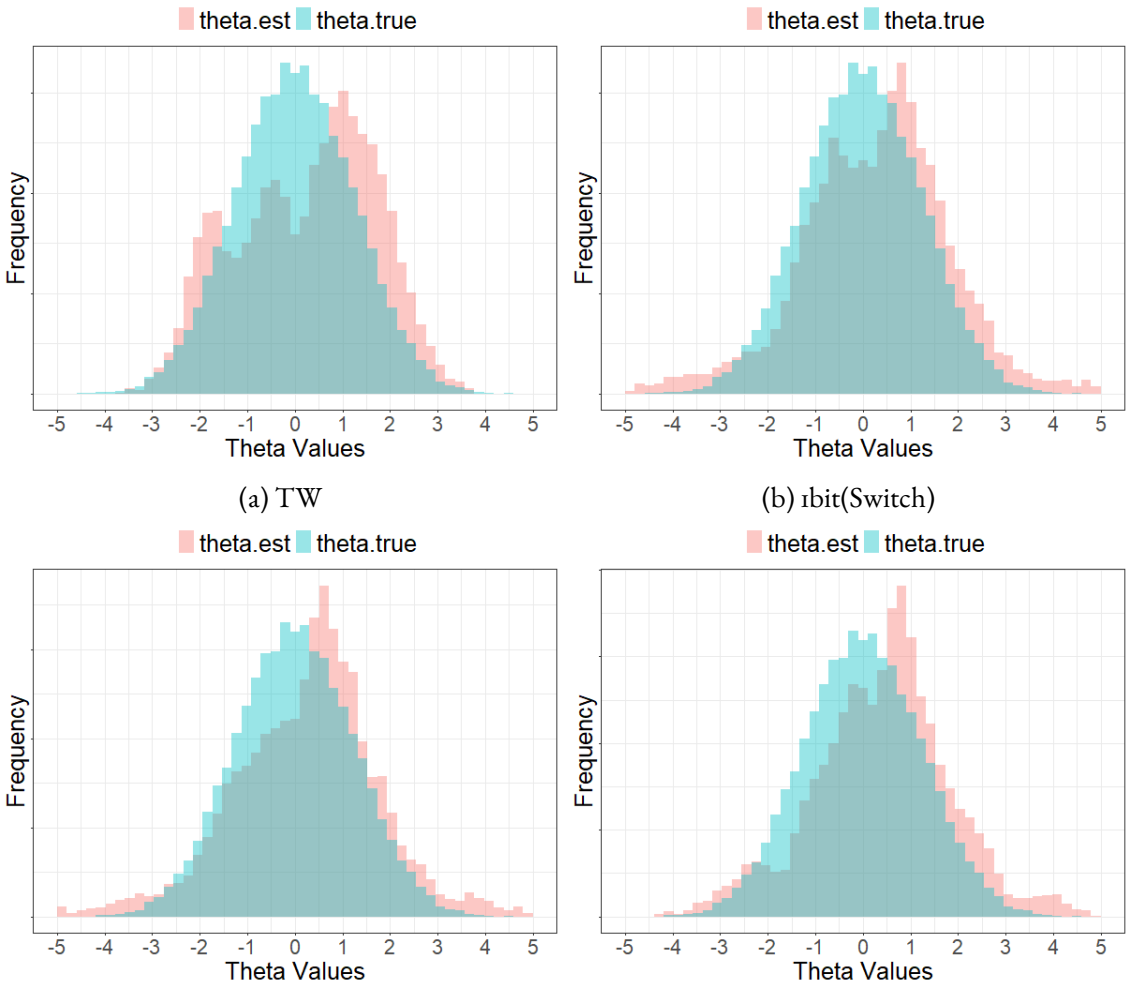


Figure C.38: Bias(b) Per Item of Selected Methods in Normal Item Bank with $N = 125$.



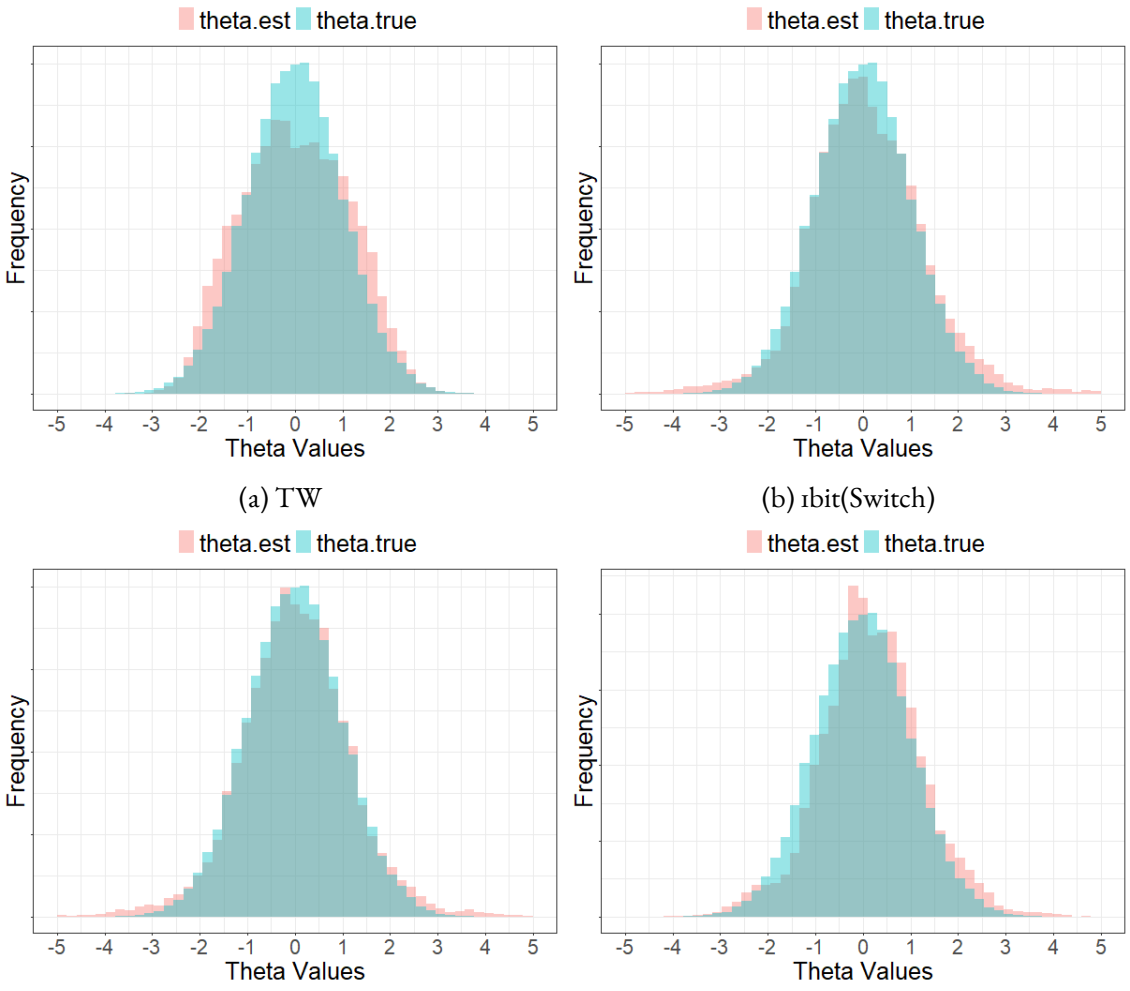
(a) TW

(b) 1bit(Switch)

(c) SMC(1+Switch)

(d) SMC(pool+p_tw)

Figure C.39: Histogram of Estimated θ Versus True θ with Test Length of 20 When $\theta \sim N(\mu, 1)$ Using Item Bank Calibrated in Random NEAT Designs



(a) TW

(b) 1bit(Switch)

(c) SMC(1+Switch)

(d) SMC(pool+p_tw)

Figure C.40: Histogram of Estimated θ Versus True θ with Test Length of 20 When $\theta \sim N(0, 1)$ Using Item Bank Calibrated in Random NEAT Designs

Appendix D

Tables

Table D.1: Recovery Rates of *1bit(Sign)* and *1bit(p_tw)* Across Sample Sizes in Targeted PC Designs

	Method	N = 125	N = 250	N = 500
Uniform	1bit(p_tw)	0.621	0.623	0.621
		(0.010)	(0.008)	(0.007)
	1bit(Sign)	0.510	0.510	0.515
		(0.004)	(0.003)	(0.002)
Normal	1bit(p_tw)	0.620	0.622	0.621
		(0.010)	(0.010)	(0.006)
	1bit(Sign)	0.507	0.508	0.511
		(0.003)	(0.003)	(0.002)
Practical	1bit(p_tw)	0.602	0.602	0.601
		(0.011)	(0.009)	(0.007)
	1bit(Sign)	0.511	0.512	0.517
		(0.005)	(0.003)	(0.002)

Note. The numbers in parentheses are standard deviations.

Table D.2: Recovery Rates of *1bit(Sign)* and *1bit(p_tw)* Across Sample Sizes in Random NEAT Designs

	Method	N = 125	N = 250	N = 500
Uniform	1bit(p_tw)	0.741 (0.006)	0.747 (0.005)	0.747 (0.003)
	1bit(Sign)	0.542 (0.004)	0.554 (0.003)	0.580 (0.003)
Normal	1bit(p_tw)	0.699 (0.007)	0.705 (0.005)	0.708 (0.004)
	1bit(Sign)	0.532 (0.004)	0.542 (0.003)	0.562 (0.003)
Practical	1bit(p_tw)	0.767 (0.005)	0.773 (0.004)	0.775 (0.002)
	1bit(Sign)	0.541 (0.005)	0.554 (0.003)	0.581 (0.003)

Note. The numbers in parentheses are standard deviations.

Table D.3: Recovery Rates of *1bit(Sign)* and *1bit(p_tw)* Across Sample Sizes in Random PC Designs

	Method	N = 125	N = 250	N = 500
Uniform	1bit(p_tw)	0.750 (0.007)	0.754 (0.005)	0.754 (0.004)
	1bit(Sign)	0.517 (0.004)	0.519 (0.004)	0.526 (0.002)
Normal	1bit(p_tw)	0.711 (0.008)	0.714 (0.005)	0.717 (0.004)
	1bit(Sign)	0.513 (0.004)	0.515 (0.003)	0.521 (0.002)
Practical	1bit(p_tw)	0.775 (0.006)	0.780 (0.004)	0.781 (0.003)
	1bit(Sign)	0.518 (0.005)	0.521 (0.003)	0.528 (0.003)

Note. The numbers in parentheses are standard deviations.

Table D.4: Bias(b) and RMSE(b) of *1bit(Sign)* and *1bit(p_tw)* Across Sample Sizes in Uniform Item Bank

		N = 125		N = 250		N = 500	
Design	Method	Bias(b)	RMSE(b)	Bias(b)	RMSE(b)	Bias(b)	RMSE(b)
Targeted NEAT	1bit(Sign)	10.523	98.441	-0.377	11.104	66.487	479.083
	1bit(p_tw)	0.204	0.790	0.292	0.663	0.312	0.558
Targeted PC	1bit(Sign)	-1.821	21.953	2.150	24.307	-0.219	8.581
	1bit(p_tw)	0.043	0.788	0.069	0.715	0.037	0.663
Random NEAT	1bit(Sign)	4.760	50.267	0.101	5.661	0.490	15.137
	1bit(p_tw)	0.139	1.223	0.102	0.506	0.084	0.398
Random PC	1bit(Sign)	-1.961	28.730	0.386	11.691	0.729	22.368
	1bit(p_tw)	-0.092	1.490	0.056	0.435	0.002	0.341

Table D.5: Bias(b) and RMSE(b) of *1bit(Sign)* and *1bit(p_tw)* Across Sample Sizes in Normal Item Bank

		N = 125		N = 250		N = 500	
Design	Method	Bias(b)	RMSE(b)	Bias(b)	RMSE(b)	Bias(b)	RMSE(b)
Targeted NEAT	1bit(Sign)	-0.663	16.391	-0.427	17.566	-0.436	15.507
	1bit(p_tw)	-0.056	0.764	-0.023	0.625	0.026	0.524
Targeted PC	1bit(Sign)	0.059	9.571	1.367	24.151	-0.115	7.524
	1bit(p_tw)	-0.035	0.816	-0.044	0.705	-0.043	0.650
Random NEAT	1bit(Sign)	-1.554	22.193	-0.875	13.696	0.111	5.151
	1bit(p_tw)	-0.038	1.139	-0.065	0.474	-0.076	0.359
Random PC	1bit(Sign)	0.898	28.735	0.720	24.972	-0.238	19.176
	1bit(p_tw)	-0.085	0.808	-0.088	0.408	-0.067	0.328

Table D.6: Bias(a) and RMSE(a) of *1bit(Sign)* and *1bit(p_tw)* Across Sample Sizes in Uniform Item Bank

		N = 125		N = 250		N = 500	
Design	Method	Bias(a)	RMSE(a)	Bias(a)	RMSE(a)	Bias(a)	RMSE(a)
Targeted NEAT	1bit(Sign)	-0.874	0.984	-0.832	0.920	-0.743	0.828
	1bit(p_tw)	1.568	2.502	1.478	1.900	1.484	1.738
Targeted PC	1bit(Sign)	-0.669	0.932	-0.681	0.856	-0.772	0.931
	1bit(p_tw)	0.673	0.908	0.689	0.806	0.613	0.689
Random NEAT	1bit(Sign)	-0.741	0.859	-0.668	0.768	-0.496	0.608
	1bit(p_tw)	2.564	7.466	1.574	1.812	1.526	1.635
Random PC	1bit(Sign)	-0.869	0.915	-0.857	0.884	-0.879	1.018
	1bit(p_tw)	1.052	1.453	1.004	1.139	0.986	1.075

Table D.7: Bias(a) and RMSE(a) of *1bit(Sign)* and *1bit(p_tw)* Across Sample Sizes in Normal Item Bank

		N = 125		N = 250		N = 500	
Design	Method	Bias(a)	RMSE(a)	Bias(a)	RMSE(a)	Bias(a)	RMSE(a)
targeted NEAT	1bit(Sign)	-0.692	0.811	-0.634	0.749	-0.544	0.656
	1bit(p_tw)	1.629	2.375	1.580	1.944	1.654	1.815
targeted PC	1bit(Sign)	-0.502	0.837	-0.531	0.706	-0.664	0.836
	1bit(p_tw)	0.681	0.874	0.650	0.754	0.609	0.695
random NEAT	1bit(Sign)	-0.608	0.716	-0.510	0.632	-0.349	0.503
	1bit(p_tw)	1.701	2.523	1.639	1.834	1.611	1.705
random PC	1bit(Sign)	-0.692	0.740	-0.696	0.717	-0.712	0.840
	1bit(p_tw)	1.078	1.391	1.056	1.163	1.007	1.076

Table D.8: RMSE(b) from Eight Modified Structured Matrix Completion When $N = 125$ in Uniform Item Bank

	Targeted NEAT		Targeted PC		Random NEAT		Random PC	
Method	Bias(b)	RMSE(b)	Bias(b)	RMSE(b)	Bias(b)	RMSE(b)	Bias(b)	RMSE(b)
1+Sign	0.210	3.471	**0.047**	5.840	0.260	5.708	0.121	2.627
1+p_tw	0.157	1.010	0.122	0.836	0.291	3.099	0.041	0.758
p0+Sign	3.237	47.795	1.742	24.860	-0.321	22.097	1.985	14.194
p0+p_tw	-0.803	3.576	-1.604	21.931	-1.151	11.929	-1.099	2.736
p1+Sign	2.055	17.187	-2.694	28.426	2.590	21.910	0.391	7.512
p1+p_tw	0.129	**0.681**	0.094	**0.795**	**0.039**	2.736	**0.007**	1.026
p2+Sign	**0.083**	2.423	0.224	4.566	0.377	2.691	0.465	**0.725**
p2+p_tw	0.155	2.6	0.253	1.014	0.320	**0.848**	0.394	0.741

Note. The bold numbers indicated the bias(a) that is closest to 0 or the smallest RMSE(a) in each incomplete calibration design.

Table D.9: RMSE(b) from 8 Adaptations for Structured Matrix Completion When $N = 125$ in Normal Item Bank

	Targeted NEAT		Targeted PC		Random NEAT		Random PC	
Method	Bias(b)	RMSE(b)	Bias(b)	RMSE(b)	Bias(b)	RMSE(b)	Bias(b)	RMSE(b)
SMC(1+Sign)	**0.008**	3.841	-0.360	4.633	0.133	5.396	-0.204	5.865
SMC(1+p_tw)	-0.092	1.449	**-0.009**	**0.932**	0.117	3.244	**-0.005**	1.048
SMC(p0+Sign)	1.920	12.712	1.171	14.373	0.425	36.346	-51.075	389.643
SMC(p0+p_tw)	-0.407	2.414	-2.095	12.559	-2.030	19.722	-1.058	2.168
SMC(p1+Sign)	0.210	4.455	0.563	5.705	-0.361	9.778	0.698	31.001
SMC(p1+p_tw)	0.009	**0.646**	0.159	1.161	**-0.089**	3.025	-0.178	2.636
SMC(p2+Sign)	0.411	3.407	0.366	1.595	0.180	3.638	0.252	**0.596**
SMC(p2+p_tw)	0.113	1.172	0.428	3.662	0.140	**0.894**	0.231	0.745

Note. The bold numbers indicated the bias(a) that is closest to 0 or the smallest RMSE(a) in each incomplete calibration design.

Table D.10: Bias(a) and RMSE(a) from Eight Modified Structured Matrix Completion When $N = 125$ in Uniform Item Bank

	Targeted NEAT		Targeted PC		Random NEAT		Random PC	
Method	Bias(a)	RMSE(a)	Bias(a)	RMSE(a)	Bias(a)	RMSE(a)	Bias(a)	RMSE(a)
1+Sign	1.734	3.538	0.776	2.295	4.956	25.616	0.622	1.292
1+p_tw	1.534	2.364	0.712	1.002	2.701	8.776	0.877	1.402
p0+Sign	1.566	10.179	0.921	3.296	1.350	10.256	-0.256	4.528
p0+p_tw	0.961	2.029	0.424	1.122	0.874	1.783	0.382	1.048
p1+Sign	1.019	2.900	0.900	3.955	**0.778**	1.762	**0.198**	**0.753**
p1+p_tw	1.355	**1.859**	0.446	0.790	1.395	2.220	0.618	1.020
p2+Sign	**0.925**	2.477	**0.299**	1.094	0.791	**1.593**	0.457	0.787
p2+p_tw	1.116	2.168	0.430	**0.756**	1.102	1.925	0.575	0.827

Note. The bold numbers indicated the bias(a) that is closest to 0 or the smallest RMSE(a) in each incomplete calibration design.

Table D.11: Bias(a) and RMSE(a) from Eight Modified Structured Matrix Completion When $N = 125$ in Normal Item Bank

	Targeted NEAT		Targeted PC		Random NEAT		Random PC	
Method	Bias(a)	RMSE(a)	Bias(a)	RMSE(a)	Bias(a)	RMSE(a)	Bias(a)	RMSE(a)
1+Sign	0.858	5.354	0.865	2.158	1.489	3.727	0.564	1.092
1+p_tw	1.558	2.258	0.797	1.034	3.342	15.068	0.880	1.206
p0+Sign	3.094	14.532	1.232	5.406	1.438	9.448	3546.846	25085.071
p0+p_tw	1.313	2.074	0.515	0.950	1.121	1.828	0.657	1.125
p1+Sign	**0.65**	3.299	0.711	2.552	**0.891**	1.681	**0.245**	**0.689**
p1+p_tw	1.439	**1.969**	0.587	0.848	1.362	2.045	0.734	1.016
p2+Sign	1.405	3.082	**0.194**	0.963	0.974	**1.663**	0.623	0.888
p2+p_tw	1.386	2.372	0.368	**0.707**	1.212	1.903	0.680	0.896

Note. The bold numbers indicated the bias(a) that is closest to 0 or the smallest RMSE(a) in each incomplete calibration design.

Table D.12: Recovery Rates from Eight Modified Structured Matrix Completion Across Sample Sizes in Uniform Item Bank Under Targeted NEAT Designs

Method	N = 125	N = 250	N = 500
SMC(1+p_tw)	0.742	0.750	0.754
SMC(1+Sign)	0.730	0.745	0.754
SMC(p0+p_tw)	0.686	0.676	0.675
SMC(p0+Sign)	0.572	0.557	0.541
SMC(p1+p_tw)	0.737	0.742	0.745
SMC(p1+Sign)	0.665	0.673	0.676
SMC(p2+p_tw)	0.665	0.671	0.670
SMC(p2+Sign)	0.593	0.607	0.607

Table D.13: Recovery Rates from Eight Modified Structured Matrix Completion Across Sample Sizes in Normal Item Bank Under Targeted NEAT Designs

Method	N = 125	N = 250	N = 500
SMC(1+p_tw)	0.694	0.705	0.711
SMC(1+Sign)	0.654	0.690	0.702
SMC(p0+p_tw)	0.664	0.666	0.665
SMC(p0+Sign)	0.520	0.523	0.518
SMC(p1+p_tw)	0.699	0.705	0.710
SMC(p1+Sign)	0.622	0.639	0.640
SMC(p2+p_tw)	0.635	0.645	0.642
SMC(p2+Sign)	0.570	0.584	0.579

Table D.14: Bias(b) and RMSE(b) from Eight Modified Structured Matrix Completion Across Sample Sizes in Practical Item Bank and Targeted PC Designs

	N = 125		N = 250		N = 500	
Method	Bias(b)	RMSE(b)	Bias(b)	RMSE(b)	Bias(b)	RMSE(b)
SMC(1+Sign)	0.485	5.954	0.098	1.710	0.157	0.835
SMC(1+p_tw)	0.074	1.277	0.101	0.855	0.087	0.793
SMC(p0+Sign)	2.322	11.592	2.353	3.867	1.732	4.271
SMC(p0+p_tw)	-7.001	54.459	-4.482	37.977	-0.404	10.038
SMC(p1+Sign)	0.993	18.687	-0.789	21.291	-0.182	2.893
SMC(p1+p_tw)	0.951	7.391	0.064	0.668	0.119	0.610
SMC(p2+Sign)	0.229	1.778	0.240	0.514	0.341	0.531
SMC(p2+p_tw)	0.237	0.947	0.203	0.559	0.257	0.511

Table D.15: Bias(b) and RMSE(b) from Eight Modified Structured Matrix Completion Across Sample Sizes in Practical Item Bank and Random PC Designs

	N = 125		N = 250		N = 500	
Method	Bias(b)	RMSE(b)	Bias(b)	RMSE(b)	Bias(b)	RMSE(b)
SMC(1+Sign)	0.165	3.745	0.007	0.955	-0.023	0.407
SMC(1+p_tw)	-0.065	2.457	-0.011	0.499	-0.025	0.369
SMC(p0+Sign)	3.198	16.696	1.833	7.055	2.435	6.676
SMC(p0+p_tw)	1.542	23.918	-1.092	1.492	-1.011	1.191
SMC(p1+Sign)	0.572	8.302	0.092	2.343	0.198	1.007
SMC(p1+p_tw)	-0.673	9.178	-0.073	1.042	-0.042	0.910
SMC(p2+Sign)	0.411	0.794	0.322	0.676	0.385	0.626
SMC(p2+p_tw)	0.418	0.865	0.307	0.490	0.346	0.455

Table D.16: Bias(a) and RMSE(a) from Eight Modified Structured Matrix Completion Across Sample Sizes in Practical Item Bank and Targeted PC Designs

	N = 125		N = 250		N = 500	
Method	Bias(a)	RMSE(a)	Bias(a)	RMSE(a)	Bias(a)	RMSE(a)
SMC(1+Sign)	0.639	1.595	0.702	1.065	0.726	0.861
SMC(1+p_tw)	0.736	1.051	0.759	0.887	0.706	0.770
SMC(p0+Sign)	1.281	3.221	1.215	2.839	0.857	2.269
SMC(p0+p_tw)	0.621	1.203	0.507	0.945	0.466	0.865
SMC(p1+Sign)	0.807	3.484	0.530	1.628	0.261	1.327
SMC(p1+p_tw)	0.331	0.651	0.340	0.551	0.231	0.408
SMC(p2+Sign)	0.459	1.426	0.198	0.679	0.136	0.501
SMC(p2+p_tw)	0.445	0.743	0.442	0.607	0.348	0.482

Table D.17: Bias(a) and RMSE(a) from Eight Modified Structured Matrix Completion Across Sample Sizes in Practical Item Bank and Random PC Designs

	N = 125		N = 250		N = 500	
Method	Bias(a)	RMSE(a)	Bias(a)	RMSE(a)	Bias(a)	RMSE(a)
SMC(1+Sign)	0.705	1.353	0.668	0.946	0.688	0.852
SMC(1+p_tw)	0.931	1.462	0.865	1.057	0.822	0.928
SMC(p0+Sign)	-0.030	4.767	-0.466	3.640	-0.068	3.691
SMC(p0+p_tw)	0.676	1.239	0.523	0.872	0.502	0.804
SMC(p1+Sign)	0.174	0.777	0.081	0.455	0.090	0.404
SMC(p1+p_tw)	0.608	1.082	0.448	0.687	0.418	0.612
SMC(p2+Sign)	0.619	0.908	0.627	0.754	0.575	0.667
SMC(p2+p_tw)	0.685	0.960	0.663	0.773	0.635	0.700

Table D.18: Summarized Difference Between Imputed Response Matrices from *SMC(1+p_tw)* and Full Response Matrix with Regards to Proportion of Wrongly Imputed Correct Responses

	item 1 - 30	item 31 - 60	item 61 - 90	item 91 - 120	item 121 - 150
Group 1 (Highest θ)	0.027	0.045	**0.085**	**0.138**	0.151
Group 2	0.057	**0.100**	**0.142**	0.140	0.061
Group 3	**0.091**	**0.150**	0.122	0.061	0.016
Group 4	**0.133**	0.153	0.063	0.018	0.004
Group 5 (Lowest θ)	0.150	0.085	0.017	0.004	0.001

Note. The bold number indicates that number is larger than the corresponding value from *TW*.

Table D.19: Summarized Difference Between Imputed Response Matrices from *SMC(1+p_tw)* and Full Response Matrix with Regards to Proportion of Wrongly Imputed Incorrect Responses

	item 1 - 30	item 31 - 60	item 61 - 90	item 91 - 120	item 121 - 150
Group 1 (Highest θ)	**0.002**	0.006	0.028	0.083	0.151
Group 2	0.008	0.030	0.096	**0.174**	**0.138**
Group 3	0.023	0.084	**0.149**	**0.147**	**0.082**
Group 4	0.070	0.161	**0.141**	**0.088**	**0.045**
Group 5 (Lowest θ)	0.171	**0.161**	**0.081**	**0.035**	0.016

Note. The bold number indicates that number is larger than the corresponding value from *TW*.

Table D.20: Average Recovery Rates Across Sample Sizes in Uniform Item Banks

Condition	Method	N = 125	N = 250	N = 500
Targeted NEAT	1bit(p_tw)	0.722	0.723	0.726
	1bit(Switch_General)	0.674	0.670	0.674
	1bit(Switch_Preferred)	0.656	0.653	0.657
	TW	0.645	0.647	0.649
Targeted PC	1bit(p_tw)	0.621	0.623	0.621
	1bit(Switch_General)	0.593	0.590	0.587
	1bit(Switch_Preferred)	0.589	0.587	0.584
	TW	0.603	0.603	0.603
Random NEAT	1bit(p_tw)	0.741	0.747	0.747
	1bit(Switch_General)	0.688	0.693	0.693
	1bit(Switch_Preferred)	0.678	0.683	0.682
	TW	0.704	0.709	0.709
Random PC	1bit(p_tw)	0.750	0.754	0.754
	1bit(Switch_General)	0.693	0.696	0.696
	1bit(Switch_Preferred)	0.683	0.688	0.688
	TW	0.706	0.709	0.708

Table D.21: Average Recovery Rates Across Sample Sizes in Normal Item Banks

Condition	Method	N = 125	N = 250	N = 500
Targeted NEAT	1bit(p_tw)	0.687	0.691	0.695
	1bit(Switch_General)	0.645	0.649	0.651
	1bit(Switch_Preferred)	0.620	0.624	0.626
	TW	0.621	0.622	0.625
Targeted PC	1bit(p_tw)	0.620	0.622	0.621
	1bit(Switch_General)	0.590	0.591	0.591
	1bit(Switch_Preferred)	0.568	0.570	0.570
	TW	0.594	0.594	0.595
Random NEAT	1bit(p_tw)	0.699	0.705	0.708
	1bit(Switch_General)	0.653	0.659	0.661
	1bit(Switch_Preferred)	0.632	0.638	0.641
	TW	0.666	0.671	0.672
Random PC	1bit(p_tw)	0.711	0.714	0.717
	1bit(Switch_General)	0.661	0.662	0.665
	1bit(Switch_Preferred)	0.646	0.651	0.653
	TW	0.670	0.671	0.672

Table D.22: Average Recovery Rates Across Sample Sizes in Uniform Item Bank

Condition	Method	N = 125	N = 250	N = 500
Targeted NEAT	SMC(1+p_tw)	0.694	0.705	0.711
	SMC(1+Switch_General)	0.650	0.661	0.665
	SMC(1+Switch_Preferred)	0.624	0.634	0.637
	TW	0.621	0.622	0.625
Targeted PC	SMC(1+p_tw)	0.624	0.638	0.641
	SMC(1+Switch_General)	0.588	0.600	0.606
	SMC(1+Switch_Preferred)	0.576	0.586	0.593
	TW	0.594	0.594	0.595
Random NEAT	SMC(1+p_tw)	0.690	0.697	0.702
	SMC(1+Switch_General)	0.645	0.652	0.657
	SMC(1+Switch_Preferred)	0.627	0.634	0.637
	TW	0.666	0.671	0.672
Random PC	SMC(1+p_tw)	0.691	0.702	0.708
	SMC(1+Switch_General)	0.645	0.652	0.655
	SMC(1+Switch_Preferred)	0.636	0.643	0.646
	TW	0.670	0.671	0.672

Table D.23: Average Recovery Rates Across Sample Sizes in Normal Item Bank

Condition	Method	N = 125	N = 250	N = 500
Targeted NEAT	SMC(1+p_tw)	0.742	0.750	0.754
	SMC(1+Switch_General)	0.688	0.696	0.698
	SMC(1+Switch_Preferred)	0.675	0.682	0.685
	TW	0.645	0.647	0.649
Targeted PC	SMC(1+p_tw)	0.638	0.642	0.643
	SMC(1+Switch_General)	0.603	0.602	0.599
	SMC(1+Switch_Preferred)	0.602	0.601	0.599
	TW	0.603	0.603	0.603
Random NEAT	SMC(1+p_tw)	0.734	0.742	0.743
	SMC(1+Switch_General)	0.681	0.690	0.690
	SMC(1+Switch_Preferred)	0.670	0.678	0.679
	TW	0.704	0.709	0.709
Random PC	SMC(1+p_tw)	0.732	0.743	0.748
	SMC(1+Switch_General)	0.679	0.686	0.688
	SMC(1+Switch_Preferred)	0.676	0.684	0.686
	TW	0.706	0.709	0.708

Table D.24: Average Recovery Rates of Softened Threshold Methods Across Sample Sizes in the Uniform Item Bank

Condition	Method	N = 125	N = 250	N = 500
Targeted NEAT	SMC(1+p_tw)	0.742	0.750	0.754
	SMC(1+Soft)	0.740	0.751	0.756
	SMC(p1+p_tw)	0.737	0.742	0.745
	SMC(p1+Soft)	0.726	0.731	0.733
	TW	0.645	0.647	0.649
Targeted PC	SMC(1+p_tw)	0.638	0.642	0.643
	SMC(1+Soft)	0.637	0.643	0.646
	SMC(p1+p_tw)	0.670	0.650	0.630
	SMC(p1+Soft)	0.679	0.648	0.620
	TW	0.603	0.603	0.603
Random NEAT	SMC(1+p_tw)	0.734	0.742	0.743
	SMC(1+Soft)	0.729	0.738	0.740
	SMC(p1+p_tw)	0.730	0.736	0.737
	SMC(p1+Soft)	0.710	0.715	0.717
	TW	0.704	0.709	0.709
Random PC	SMC(1+p_tw)	0.732	0.743	0.748
	SMC(1+Soft)	0.722	0.736	0.742
	SMC(p1+p_tw)	0.736	0.741	0.741
	SMC(p1+Soft)	0.715	0.719	0.719
	TW	0.706	0.709	0.708

Table D.25: Average Recovery Rates of Softened Threshold Methods Across Sample Sizes in the Normal Item Bank

Condition	Method	N = 125	N = 250	N = 500
Targeted NEAT	SMC(1+p_tw)	0.694	0.705	0.711
	SMC(1+Soft)	0.684	0.702	0.710
	SMC(p1+p_tw)	0.699	0.705	0.710
	SMC(p1+Soft)	0.689	0.697	0.701
	TW	0.621	0.622	0.625
Targeted PC	SMC(1+p_tw)	0.624	0.638	0.641
	SMC(1+Soft)	0.618	0.637	0.643
	SMC(p1+p_tw)	0.663	0.669	0.665
	SMC(p1+Soft)	0.673	0.681	0.673
	TW	0.594	0.594	0.595
Random NEAT	SMC(1+p_tw)	0.690	0.697	0.702
	SMC(1+Soft)	0.684	0.691	0.697
	SMC(p1+p_tw)	0.692	0.698	0.701
	SMC(p1+Soft)	0.678	0.683	0.685
	TW	0.666	0.671	0.672
Random PC	SMC(1+p_tw)	0.691	0.702	0.708
	SMC(1+Soft)	0.681	0.694	0.702
	SMC(p1+p_tw)	0.700	0.703	0.705
	SMC(p1+Soft)	0.681	0.685	0.686
	TW	0.670	0.671	0.672

Table D.26: Average Recovery Rates of the Pooling Method Across Sample Sizes in the Uniform Item Bank

Condition	Method	N = 125	N = 250	N = 500
Targeted NEAT	SMC(1+p_tw)	0.742	0.75	0.754
	SMC(p1+p_tw)	0.737	0.742	0.745
	SMC(p2+p_tw)	0.665	0.671	0.670
	SMC(pool+p_tw)	0.716	0.721	0.723
	TW	0.645	0.647	0.649
Targeted PC	SMC(1+p_tw)	0.638	0.642	0.643
	SMC(p1+p_tw)	0.670	0.650	0.630
	SMC(p2+p_tw)	0.652	0.666	0.652
	SMC(pool+p_tw)	0.653	0.652	0.642
	TW	0.603	0.603	0.603
Random NEAT	SMC(1+p_tw)	0.734	0.742	0.743
	SMC(p1+p_tw)	0.730	0.736	0.737
	SMC(p2+p_tw)	0.664	0.667	0.665
	SMC(pool+p_tw)	0.710	0.716	0.715
	TW	0.704	0.709	0.709
Random PC	SMC(1+p_tw)	0.732	0.743	0.748
	SMC(p1+p_tw)	0.736	0.741	0.741
	SMC(p2+p_tw)	0.666	0.669	0.665
	SMC(pool+p_tw)	0.711	0.718	0.717
	TW	0.706	0.709	0.708

Table D.27: Average Recovery Rates of the Pooling Method Across Sample Sizes in the Normal Item Bank

Condition	Method	N = 125	N = 250	N = 500
Targeted NEAT	SMC(1+p_tw)	0.694	0.705	0.711
	SMC(p1+p_tw)	0.699	0.705	0.710
	SMC(p2+p_tw)	0.635	0.645	0.642
	SMC(pool+p_tw)	0.678	0.685	0.688
	TW	0.621	0.622	0.625
Targeted PC	SMC(1+p_tw)	0.624	0.638	0.641
	SMC(p1+p_tw)	0.663	0.669	0.665
	SMC(p2+p_tw)	0.635	0.641	0.632
	SMC(pool+p_tw)	0.641	0.65	0.646
	TW	0.594	0.594	0.595
Random NEAT	SMC(1+p_tw)	0.690	0.697	0.702
	SMC(p1+p_tw)	0.692	0.698	0.701
	SMC(p2+p_tw)	0.638	0.641	0.637
	SMC(pool+p_tw)	0.675	0.679	0.68
	TW	0.666	0.671	0.672
Random PC	SMC(1+p_tw)	0.691	0.702	0.708
	SMC(p1+p_tw)	0.700	0.703	0.705
	SMC(p2+p_tw)	0.640	0.641	0.637
	SMC(pool+p_tw)	0.677	0.682	0.684
	TW	0.670	0.671	0.672

Table D.28: Summary Statistics of $\hat{\theta}$ when $\theta \sim N(0,1)$ using Item Parameters Calibrated in Targeted NEAT Design with Test Length of 40

θ Ranges	Method	Deviation	SD	MSE	RAD	Correlation
$\theta \leq -1$	True	-0.004	0.536	0.083	0.158	0.842
	FIML	0.425	0.443	0.261	0.293	0.797
	TW	0.740	**0.311**	0.614	0.489	**0.826**
	1bit(Switch)	0.526	**0.386**	0.342	0.351	**0.821**
	SMC(1+Switch)	**0.108**	0.549	**0.105**	**0.179**	**0.830**
	SMC(pool+p_tw)	0.482	**0.383**	0.300	0.321	**0.813**
$-1 \leq \theta < 1$	True	-0.004	0.596	0.063	5.713	0.907
	FIML	0.046	0.483	0.094	4.806	0.830
	TW	**0.008**	**0.315**	**0.091**	**1.889**	**0.881**
	1bit(Switch)	0.061	**0.426**	**0.064**	**4.725**	**0.897**
	SMC(1+Switch)	**-0.002**	0.542	**0.059**	**2.360**	**0.899**
	SMC(pool+p_tw)	0.130	0.501	**0.078**	**3.820**	**0.889**
$\theta > 1$	True	0.006	0.52	0.061	0.134	0.879
	FIML	-0.432	0.377	0.300	0.282	0.777
	TW	-0.577	**0.372**	0.382	0.383	**0.869**
	1bit(Switch)	**-0.300**	0.448	**0.140**	**0.214**	**0.874**
	SMC(1+Switch)	**-0.061**	0.631	**0.126**	**0.160**	**0.858**
	SMC(pool+p_tw)	**-0.166**	0.446	**0.079**	**0.148**	**0.871**

Note. The bold numbers indicated better θ estimation from the imputation methods than *FIML* method.

Table D.29: Summary Statistics of $\hat{\theta}$ when $\theta \sim N(0,1)$ using Item Parameters Calibrated in Targeted NEAT Design with Test Length of 20

θ Ranges	Method	Deviation	SD	MSE	RAD	Correlation
$\theta \leq -1$	True	-0.011	0.601	0.145	0.207	0.773
	FIML	0.488	0.504	0.378	0.351	0.697
	TW	0.735	**0.380**	0.632	0.492	**0.745**
	1bit(Switch)	**0.454**	**0.492**	**0.325**	**0.328**	**0.733**
	SMC(1+Switch)	**0.033**	0.711	**0.229**	**0.253**	**0.752**
	SMC(pool+p_tw)	**0.429**	**0.493**	**0.308**	**0.313**	**0.723**
$-1 \leq \theta < 1$	True	-0.008	0.629	0.107	7.348	0.854
	FIML	0.048	0.494	0.136	4.718	0.754
	TW	**0.017**	**0.319**	**0.117**	**3.502**	**0.801**
	1bit(Switch)	0.077	**0.472**	**0.097**	5.537	**0.830**
	SMC(1+Switch)	**0.005**	0.580	**0.104**	**3.641**	**0.837**
	SMC(pool+p_tw)	0.122	0.546	**0.122**	5.352	**0.819**
$\theta > 1$	True	0.011	0.559	0.104	0.174	0.818
	FIML	-0.442	0.413	0.306	0.300	0.704
	TW	-0.600	**0.402**	0.438	0.401	**0.791**
	1bit(Switch)	**-0.231**	0.516	**0.148**	**0.211**	**0.804**
	SMC(1+Switch)	**0.033**	0.763	**0.246**	**0.217**	**0.798**
	SMC(pool+p_tw)	**-0.087**	0.528	**0.116**	**0.180**	**0.783**

Note. The bold numbers indicated better θ estimation from the imputation methods than *FIML* method.

Table D.30: Summary Statistics of $\hat{\theta}$ when $\theta \sim N(\mu, 1)$ using Item Parameters Calibrated in Random NEAT Design with Test Length of 20

θ Ranges	Method	Deviation	SD	MSE	RAD	Correlation
$\theta \leq -1$	True	-0.062	0.827	0.290	0.267	0.769
	FIML	-0.021	0.983	0.638	0.350	0.600
	TW	-0.047	**0.661**	**0.224**	**0.256**	**0.714**
	1bit(Switch)	-0.093	1.085	**0.561**	**0.334**	**0.777**
	SMC(1+Switch)	-0.153	1.101	**0.606**	**0.326**	**0.771**
	SMC(pool+p_tw)	0.169	**0.922**	**0.398**	**0.335**	**0.771**
$-1 \leq \theta < 1$	True	0.329	0.732	0.300	5.919	0.803
	FIML	-0.035	0.781	0.365	5.974	0.641
	TW	0.320	0.886	0.419	7.310	**0.790**
	1bit(Switch)	0.341	**0.770**	**0.337**	6.105	**0.796**
	SMC(1+Switch)	0.315	**0.739**	**0.306**	6.270	**0.790**
	SMC(pool+p_tw)	0.440	**0.703**	0.376	**5.553**	**0.794**
$\theta > 1$	True	0.267	0.702	0.257	0.255	0.789
	FIML	-0.143	1.166	1.021	0.364	0.537
	TW	0.340	**0.557**	**0.249**	**0.284**	**0.786**
	1bit(Switch)	0.538	**0.964**	**0.774**	0.399	**0.705**
	SMC(1+Switch)	0.445	**0.988**	**0.688**	0.363	**0.725**
	SMC(pool+p_tw)	0.518	**0.860**	**0.583**	0.378	**0.769**

Note. The bold numbers indicated better θ estimation from the imputation methods than *FIML* method.

Table D.31: Summary Statistics of $\hat{\theta}$ when $\theta \sim N(0,1)$ using Item Parameters Calibrated in Random NEAT Design with Test Length of 20

θ Ranges	Method	Deviation	SD	MSE	RAD	Correlation
$\theta \leq -1$	True	-0.014	0.594	0.143	0.207	0.771
	FIML	0.02	0.711	0.321	0.288	0.612
	TW	**-0.006**	**0.487**	**0.133**	**0.208**	**0.698**
	1bit(Switch)	**-0.009**	0.795	**0.319**	**0.281**	**0.722**
	SMC(1+Switch)	-0.077	0.800	0.322	**0.275**	**0.732**
	SMC(pool+p_tw)	0.205	**0.671**	**0.256**	0.289	**0.727**
$-1 \leq \theta < 1$	True	-0.012	0.626	0.107	4.354	0.853
	FIML	-0.184	0.639	0.265	7.706	0.679
	TW	**0.006**	0.742	**0.179**	**4.438**	**0.827**
	1bit(Switch)	**0.058**	0.672	**0.144**	**3.966**	**0.830**
	SMC(1+Switch)	**0.014**	0.643	**0.128**	**3.270**	**0.831**
	SMC(pool+p_tw)	**0.171**	**0.615**	**0.153**	**6.480**	**0.823**
$\theta > 1$	True	0.012	0.564	0.103	0.174	0.822
	FIML	-0.379	0.775	0.562	0.350	0.570
	TW	**0.093**	**0.472**	**0.103**	**0.183**	**0.778**
	1bit(Switch)	**0.308**	**0.825**	**0.396**	**0.288**	**0.787**
	SMC(1+Switch)	**0.152**	**0.762**	**0.265**	**0.237**	**0.792**
	SMC(pool+p_tw)	**0.202**	**0.665**	**0.216**	**0.244**	**0.787**

Note. The bold numbers indicated better θ estimation from the imputation methods than *FIML* method.

Table D.32: Summary Statistics of $\hat{\theta}$ when $\theta \sim N(0,1)$ using Item Parameters Calibrated in Random NEAT Design with Test Length of 40

θ Ranges	Method	Deviation	SD	MSE	RAD	Correlation
$\theta \leq -1$	True	-0.011	0.538	0.084	0.158	0.841
	FIML	-0.022	0.626	0.202	0.233	0.707
	TW	0.032	**0.397**	**0.077**	**0.152**	**0.792**
	1bit(Switch)	0.122	**0.589**	**0.133**	**0.198**	**0.814**
	SMC(1+Switch)	0.013	**0.600**	**0.118**	**0.182**	**0.822**
	SMC(pool+p_tw)	0.285	**0.537**	**0.178**	0.245	**0.814**
$-1 \leq \theta < 1$	True	-0.006	0.592	0.062	2.422	0.907
	FIML	-0.105	0.626	0.164	6.642	0.785
	TW	**0.025**	0.697	**0.106**	**3.184**	**0.893**
	1bit(Switch)	**0.078**	**0.606**	**0.079**	**3.663**	**0.895**
	SMC(1+Switch)	**0.024**	**0.603**	**0.071**	**2.485**	**0.897**
	SMC(pool+p_tw)	0.185	**0.568**	**0.105**	**5.550**	**0.886**
$\theta > 1$	True	0.009	0.520	0.060	0.133	0.882
	FIML	-0.255	0.691	0.370	0.246	0.650
	TW	**0.073**	**0.407**	**0.058**	**0.137**	**0.859**
	1bit(Switch)	**0.247**	**0.675**	**0.195**	**0.215**	**0.865**
	SMC(1+Switch)	**0.137**	**0.646**	**0.134**	**0.174**	**0.871**
	SMC(pool+p_tw)	**0.140**	**0.543**	**0.096**	**0.166**	**0.862**

Note. The bold numbers indicated better θ estimation from the imputation methods than *FIML* method.