7-2010

Improving IRT parameter estimates with small sample sizes: Evaluating the efficacy of a new data augmentation technique

Brett P. Foley
University of Nebraska - Lincoln, brettfoley@gmail.com

IMPROVING IRT PARAMETER ESTIMATES WITH SMALL SAMPLE SIZES: EVALUATING THE EFFICACY OF A NEW DATA AUGMENTATION TECHNIQUE

by

Brett Patrick Foley

A DISSERTATION

Presented to the Faculty of

The Graduate College at the University of Nebraska

In Partial Fulfillment of Requirement

For the Degree of Doctor of Philosophy

Major: Psychological Studies in Education

Under the Supervision of Professor Rafael J. De Ayala

Lincoln, Nebraska

July 2010

IMPROVING IRT PARAMETER ESTIMATES WITH SMALL SAMPLE SIZES: EVALUATING THE EFFICACY OF A NEW DATA AUGMENTATION TECHNIQUE

Brett Patrick Foley, Ph.D.

University of Nebraska, 2010

Advisor: Rafael J. De Ayala

The 3PL model is a flexible and widely used tool in assessment. However, it suffers from limitations due to its need for large sample sizes. This study introduces and evaluates the efficacy of a new sample size augmentation technique called Duplicate, Erase, and Replace (DupER) Augmentation through a simulation study. Data are augmented using several variations of DupER Augmentation (based on different imputation methodologies, deletion rates, and duplication rates), analyzed in BILOG-MG 3, and results are compared to those obtained from analyzing the raw data. Additional manipulated variables include test length and sample size. Estimates are compared using seven different evaluative criteria.

Results are mixed and inconclusive. DupER augmented data tend to result in larger root mean squared errors (RMSEs) and lower correlations between estimates and parameters for both item and ability parameters. However, some DupER variations produce estimates that are much less biased than those obtained from the raw data alone. For one DupER variation, it was found that DupER produced better results for low-ability simulees and worse results for those with high abilities. Findings, limitations, and recommendations for future studies are discussed. Specific recommendations for future studies include the application of

Duper Augmentation (1) to empirical data, (2) with additional IRT models, and (3) the analysis of the efficacy of the procedure for different item and ability parameter distributions.

To Lindy, Addilyn, Mom, and Dad

Acknowledgments

An accomplishment like this would be impossible without the help of many wonderful people. First, I would like to thank my wife, Lindy. We were married only a month and a half before I began graduate school, so at least she had a brief window of time to know me before I became a professional student. Thank you, Lindy, for all these years of love and support. Next, I would like to thank my daughter, Addiyn. Addy, I know there were many nights and weekends that you wished I would have been playing with you instead of sitting downstairs, typing away on Daddy's "work." Believe me, I would much rather have been spending time with you too. I promise to make up the time, with interest.

Next, I would like to thank some of my first teachers: my aunt, Frances Foley, and my kindergarten teacher, Anna Beagle. Thank you, ladies, for helping to start me on the long path that has taken me where I am today. I would like to thank Andy Dwyer, who helped the idea of DupER Augmentation come about through our many discussions of the benefits of deleting data.

I want to thank my advisor, Rafael De Ayala. Thanks Ralph, for all of the help and support. Sorry for stopping by your office so many times, and never making an appointment. I would also like to thank my committee members, Charles Ansorge, James Bovaird, Miles Bryant, and Kurt Geisinger.

Finally, I would like to thank my Dad and Mom, Patrick and Martha Foley. Thank you both for always believing in me and for pushing me to do my best. I hope I can always make you proud.

TABLE OF CONTENTS

List of Tables

List of Figures

Chapter 1. Introduction

Background

In recent decades, item response theory (IRT) models have been growing in popularity. IRT is now a well-known and accepted method that is widely used across a variety of assessment programs. These models provide a way to model the probability of giving a correct answer on an item based on the underlying ability of the examinee. Although IRT models have many advantages, there are also some drawbacks. The three-parameter logistic (3PL) model in particular requires large sample sizes to obtain accurate parameter estimates. However, in many educational settings, large sample sizes are either unavailable or undesirable. The purpose of this study is to evaluate the efficacy of a new sample size augmentation technique for estimating 3PL item parameters with small sample sizes.

The primary benefit of IRT is that estimates of item parameters are (examinee) sample independent, and person ability estimates are independent of items (Hambleton, Swaminathan, & Rogers, 1991). The test-based nature of classical test theory (CTT) does not allow us to predict how an examinee may perform on a test item. However, IRT allows greater flexibility: "A broader range of interpretations may be made at the item level. Thus [IRT] permits the measurement specialist to determine the probability of a particular examinee correctly answering a given item" (Hambleton & Jones, 1993, p. 43). This flexibility allows us to use item response functions (and relatedly, item information functions) to create customized tests.

There are many different IRT models in use today. Two of the most common of these are the Rasch and 3PL models. The Rasch model has the benefit of being a simple model that requires relatively few items and relatively small sample sizes to obtain accurate parameter estimates (~20 items and 200 examinees, Wright & Stone, 1979). However, the Rasch model has very stringent assumptions, including the assumptions that all items have equal discrimination parameters and that all lower asymptotes are equal to zero. The 3PL model relaxes the assumptions of the Rasch model: Item discrimination parameters may vary, and guessing by examinees is accounted for and included explicitly in the model. Although this provides the 3PL model greater flexibility, it also makes parameter estimation more difficult: about 60 items and 1,000 examinees are necessary to obtain adequate estimates (Swaminathan & Gifford, 1983).

Although the 3PL model is clearly an attractive option, there are many cases where obtaining large sample sizes is difficult or undesirable. For example, computerized adaptive testing (CAT) requires that item parameters be known in advance, but pre-testing items with large samples presents test security (i.e., item exposure) problems (Wainer & Eignor, 2000). In other cases, practitioners would like to make use of the flexibility of the 3PL model, but only have access to small populations. Additionally, in order to shorten scoring time, some state K-12 accountability testing programs pre-calibrate and equate their assessments with an early sample of student assessment data (i.e., a calibration sample; Geisinger, Wells, & Foley, 2007). Obtaining accurate results for the entire population assumes accurate item statistics from the calibration sample. However, participating in a calibration sample can be a burden for schools because test data need to be submitted early. Therefore, it is beneficial

to minimize the size of the calibration sample. This study attempts to improve parameter estimates through augmenting the data set by generating additional plausible response vectors based on the original collected data.

Statement of Problem

The 3PL model is a flexible and useful way to score assessment data. However, its use is limited due to its reliance on large sample sizes. Effective methods to improve the accuracy of 3PL parameter estimation could result in an expansion of the model's use into areas of assessment in which it is currently unsuitable due to sample size limitations.

Introducing DupER Augmentation

Although "data augmentation" is discussed most commonly in research regarding the handling of missing data, it is used in a different context here. In this study data augmentation is used to refer to a process of adding additional, plausible response vectors to a data set, a process that might be more appropriately described as "sample size augmentation." Specifically, this study introduces and evaluates the efficacy of a new sample size augmentation technique called Duplicate, Erase, and Replace (hereafter, DupER) Augmentation. A detailed description of DupER Augmentation is included in Chapter 3.

Significance

If DupER Augmentation proves to be successful across a wide range of conditions, it could be valuable to the educational community in several ways. First, it could reduce pre-testing costs, because smaller samples would be sufficient. Secondly, it could help improve test security by reducing item exposure (fewer examinees need to see each item to estimate

the item parameters accurately). Finally, practitioners could use the flexible 3PL model in situations where populations are small or where a smaller calibration sample is desired.

Chapter 2. Review of Literature

Introduction

This chapter begins with a brief introduction to IRT followed by a summary of some of the most common IRT item parameter estimation methods. Next, research on sample size and improving item parameter estimation is reviewed. Then, resampling techniques and imputation methods are introduced. The chapter concludes with a discussion of the research questions and hypotheses of the current study.

A Brief Introduction to IRT

Much has been written about the theoretical foundations, development, and application of IRT (e.g., de Ayala, 2009; Hambleton, Swaminathan, & Rogers, 1991; Yen & Fitzpatrick, 2006). The intent of this section is to provide a consise introduction to IRT and a brief descripion of its benefits and limitations.

At its core, IRT is a group of statistical models used to analyze assessment data. These models, which focus on individual items rather than intact assessments, employ nonlinear functions to relate the properties of an item (e.g., difficulty, discrimination) to the probability of an examinee providing a particular response (e.g., correct, incorrect). Mathematically, this can be defined as:

$$P_i(\theta) \equiv P_i(X_i = x_i|\{\theta\},\{\delta_i\}) \quad (1)$$

This equation, or item response function (IRF), indicates that the probability of an examinee responding x_i on item X_i depends on one or more examinee ability parameters, $\{\theta\}$, and one or more item parameters, $\{\delta_i\}$. This equation illustrates the primary benefits of IRT: Because the probability of a given response is conditional on both the item and examinee

characteristics, estimates of item parameters are (examinee) sample independent, and person estimates are independent of items (Hambleton, Swaminathan, & Rogers, 1991; Yen & Fitzpatrick, 2006). IRFs are displayed graphically using item characteristic curves (ICC; Yen & Fitzpatrick, 2006). Two different ICCs are shown in Figure 1.

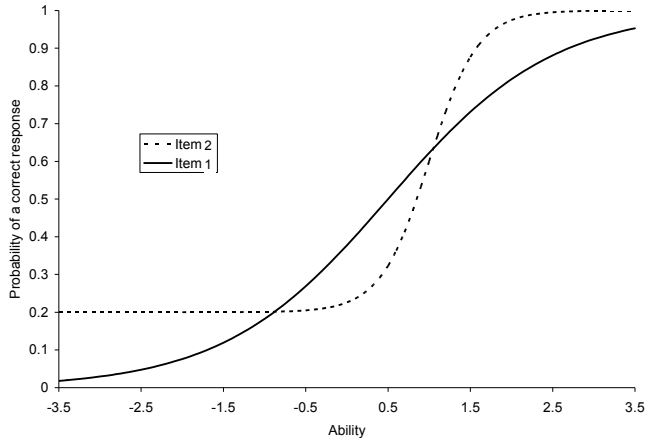

Figure 1. ICCs for two different assessment items

Equation 1 is very general and does not specify that the item responses be either dichotomous or polytomous. This study focuses on an IRT model for dichotomous responses. The responses for dichotomous models are typically coded to either a zero (for an incorrect response) or one (for a correct response). Three of the most common models for dichotomous responses are discussed in more detail below.

One of the simplest dichotomous IRT models is the Rasch or one-parameter logistic (1PL) model:

$$P_i(X_i = 1|\theta) = \frac{1}{1+e^{-(\theta - b_i)}} \qquad (2)$$

This equation indicates that the probability of a correct response is dependent on the ability of the examinee (θ) and the item parameter b_i, which is commonly referred to as item

difficulty. Mathematically, the item difficulty corresponds to the ability level at the point of inflection of the ICC. Thought of another way, an examinee whose ability is equal to the item difficulty will have equal probabilities (.5) of getting the item correct or incorrect. When using the 1PL model, the shape of the ICCs is the same for all items; the ICCs merely shift up or down the θ scale depending on the item difficulty value. Item 1 in Figure 1 is an example of a 1PL item with a difficulty of .5.

An extension of the 1PL model is the two-parameter logistic (2PL) model:

$$P_i(X_i = 1|\theta) = \frac{1}{1+e^{-Da_i(\theta-b_i)}} \quad (3)$$

This model is similar to the 1PL model but adds the additional item parameter, a_i. The item parameter a_i is commonly referred to as the item discrimination parameter and is a measure of the slope of the ICC at its point of inflection. Conceptually, item discrimination is an indication of the strength of the relationship between the item response and ability (Yen & Fitzpatrick, 2006). Item 2 in Figure 1 has a higher item discrimination parameter, and thus steeper slope, than Item 1's. The constant *D* is often set to a value of 1.7 in order to make the model similar to the normal ogive function (Hambleton, Swaminathan, & Rogers, 1991). However, *D*'s value is a matter of individual preference (Yen & Fitzpatrick, 2006) and is not necessary.

For the 1PL and 2PL models it is a tacit assumption that as examinee ability levels become very low (approaching negative infinity), the probability of a correct response approaches zero. For many assessments, however, this may not be appropriate. For example, on multiple-choice assessments a low-ability examinee may get an item correct simply by guessing. The 3PL model allows for this possibility through the inclusion of a guessing

parameter (sometimes referred to as a pseudo-guessing parameter). Therefore, extending the 2PL model results in the three-parameter logistic (3PL) model:

$$P_i(X_i = 1|\theta) = c_i + \frac{1-c_i}{1+e^{-Da_i(\theta-b_i)}} \quad , \qquad (4)$$

where a_i and b_i are defined above and c_i is the pseudo-guessing parameter. Conceptually, the guessing parameter is the probability of a very low ability examinee getting an item correct. Mathematically, the guessing parameter is the value of the lower asymptote of the ICC. For example, Item 2 in Figure 1 is modeled using the 3PL model with a guessing parameter value of .2. In other words, very low ability examinees have a 20% chance of answering the item correctly, as may be the case for a multiple-choice item with four or five answer choices.

Each of the three dichotomous IRT models has its own strengths and weaknesses, as well as practical and theoretical justifications. This study focuses exclusively on the 3PL model, which, although flexible due to permitting different discrimination and guessing parameters for each item, has the drawback of requiring longer tests and larger sample sizes than simpler dichotomous IRT models.

Although Equations 2- 4 are relatively straightforward, calculation of item and examinee parameter estimates can be difficult computationally. Researchers have proposed several different item parameter estimation methods, the most common of which are discussed in the following section.

Summary of IRT Item Parameter Estimation Methods

As stated above, the purpose of this study is to examine a new methodology for improving item parameter estimation with small samples. With this in mind, it is useful to understand the traditional ways in which item parameter estimates are calculated. Three of

the most common item parameter estimation methods are summarized below: joint maximum likelihood (JML), marginal maximum likelihood (MML), and Bayesian estimation. Although these are not the only item parameter estimation methods in use today, other methods tend to be less frequently used (e.g., nonparametric estimation) or specific to only a small number of models (e.g., conditional maximum likelihood). For a comprehensive review of the many IRT estimation methodologies see Baker and Kim (2004).

Joint maximum likelihood.

The solutions to the equations discussed in the previous section are complicated by the fact that in real testing situations parameters for both the items and the examinee abilities are unknown. JML addresses this problem by solving for both sets of parameters simultaneously.

Let U be an $N \times n$ matrix consisting of dichotomously scored assessment results (1 = correct, 0 = incorrect) for an assessment that is n items long and administered to N examinees. Item responses are denoted u_{ij}, where i indicates the item, $i = 1, \ldots, n$, and j indicates the examinee, $j = 1, \ldots, N$. Let θ be a vector of ability parameters ($\theta_1, \ldots, \theta_j, \ldots, \theta_N$). Also, let $P_i(\theta_j)$ equal the probability of a person with ability θ_j getting item i correct, and let $Q_i(\theta_j)$ equal 1-$P_i(\theta_j)$. Therefore, the probability of the observed results matrix, U, given the abilities of the examinees, θ, can be described by the following likelihood function:

$$L = Prob(U|\theta) = \prod_{j=1}^{N} \prod_{i=1}^{n} P_i^{u_{ij}}(\theta_j)\, Q_i^{1-u_{ij}}(\theta_j). \qquad (5)$$

Taking the natural log of Equation 5 yields

$$ln\, L = \sum_{j=1}^{N} \sum_{i=1}^{n} [u_{ij} \ln P_i(\theta_j) + (1 - u_{ij}) \ln Q_i(\theta_j)]. \qquad (6)$$

The likelihood equation for a given parameter of interest, λ, is obtained by setting the first derivative of Equation 6, with respect to λ, equal to zero:

$$\frac{\partial \ln L}{\partial \lambda} = \sum \frac{[u_{ij} - P_i(\theta_j)]}{P_i(\theta_j)Q_i(\theta_j)} \frac{\partial P_i(\theta_j)}{\partial \lambda} = 0. \quad (7)$$

For the parameters of interest in the 3PL model, θ_j, a_i, b_i, and c_i, Equation 7 can be rewritten as

$$\sum_{i=1}^{n} \frac{P_i(\theta_j) - c_i}{(1-c_i)P_i(\theta_j)} [u_{ij} - P_i(\theta_j)] = 0 \quad (8)$$

for θ_j,

$$\frac{1}{1-c_i} \sum_{j=1}^{N} \frac{[\theta_j - b_i][P_i(\theta_j) - c_i]}{P_i(\theta_j)} [u_{ij} - P_i(\theta_j)] = 0 \quad (9)$$

for a_i,

$$\frac{a_i}{1-c_i} \sum_{j=1}^{N} \frac{[P_i(\theta_j) - c_i]}{P_i(\theta_j)} [u_{ij} - P_i(\theta_j)] = 0 \quad (10)$$

for b_i, and,

$$\frac{1}{1-c_i} \sum_{j=1}^{N} \frac{1}{P_i(\theta_j)} [u_{ij} - P_i(\theta_j)] = 0 \quad (11)$$

for c_i (Lord, 1980; Yen & Fitzpatrick, 2006). Equations 8-11 are solved using an iterative procedure with four steps:

Step 1. In the first step, person ability estimates are treated as fixed, set to an initial value, usually based on the examinee's raw score, and estimates are calculated for the item parameters.

Step 2. In the second step, the newly estimated item parameters are treated as fixed, and estimates are calculated for examinee abilities.

Step 3. One benefit of IRT is that the item difficulty and examinee ability parameters are on the same scale. However, this scale does not have an inherent metric. To address this indeterminacy of scale, an anchor point needs to be fixed at a specific (though inherently arbitrary) value for either item difficulty or examinee ability. Several IRT software programs accomplish this by setting the mean of the estimated ability parameters to zero (de Ayala, 2009). Thus, in the third step of the estimation process, the difficulty and ability scales are set.

Step 4. New estimates are calculated for the item parameters while treating the newly estimated and re-centered person ability estimates as fixed.

Steps 2 through 4 are repeated until the change in parameter estimates between iterations becomes smaller than some fixed threshold known as a convergence criterion. The second estimation procedure that will be discussed, MML, separates the estimation of item parameters from that of examinee abilities.

Marginal maximum likelihood.

In JML both the item and examinee parameters are treated as fixed effects. Thus, as the number of examinees increases so do the number of parameters that need to be estimated. MML takes a different approach in that it treats examinees as random effects. The following description of the estimation process is a summary of a more detailed derivation by Baker (1987).

It is assumed that the θ parameters are a random sample from an overarching normal distribution (or some other empirical or user-defined distribution), $g(\theta)$. As before, assume that the assessment is n dichotomous items in length. Let s equal the number of distinct

response patterns and let l be the label of a specific response pattern such that $l = 1, 2, \ldots, s$. Therefore, the data matrix U is an $s \times n$ matrix consisting of one row for each of the s unique response vectors and one column for each of the n items. Therefore, the probability of an examinee with ability θ having the response vector u_l is

$$L_l = Prob(u = u_l) = \int_{-\infty}^{\infty} P(u = u_l|\theta)g(\theta). \quad (12)$$

The integral is approximated by summing the estimated value of the probability function at q quadrature points, where X_k ($k = 1, \ldots, q$) is a specific quadrature point and $A(X_k)$ is the quadrature weight of point X_k. Therefore, Equation 12 above can be approximated as follows:

$$L_l = \sum_{k=1}^{q} P(u = u_l|X_k)A(X_k). \quad (13)$$

Equation 13 is used along with the expectation maximization (EM) algorithm (Bock & Aitkin, 1981) to obtain parameter estimates. In the first (E) step of the algorithm, initial item parameter estimates are used to obtain the expected number of examinees whose θ values correspond with the level of the quadrature point, $\overline{N_k}$, and the expected number of correct responses to item i at that level, $\overline{r_{ik}}$. These values are estimated using the following equations:

$$\overline{N_k} = \frac{\sum_{l=1}^{s} r_l L_l(X_k)A(X_k)}{\sum_{k=1}^{q} L_l(X_k)A(X_k)} \quad (14)$$

$$\overline{r_{ik}} = \frac{\sum_{l=1}^{s} r_l u_{li} L_l(X_k)A(X_k)}{\sum_{k=1}^{q} L_l(X_k)A(X_k).} \quad (15)$$

where u_{li} is the response to item i within pattern l, and $L_l(X_k)$ is the relative density at $\theta = X_k$.

In the second (M) step of the algorithm, $\overline{N_k}$ and $\overline{r_{ik}}$ are treated as observed data and used to obtain improved estimates of item parameters using the following equations:

$$\sum_{k=1}^{q}[\overline{r_{ik}} - \overline{N_k}P_i(X_k)] = 0 \quad (16)$$

$$\sum_{k=1}^{q}\left[\overline{r_{ik}} - \overline{N_k}P_j(X_k)\right]X_k = 0. \quad (17)$$

The improved item estimates from Equations 16 and 17 are used in another E step to obtain improved estimates of $\overline{N_k}$ and $\overline{r_{ik}}$, which in turn are used in the subsequent M step to obtain improved item parameter estimates. This process iterates until the change in parameter estimates becomes smaller than the convergence criterion. After the final item parameter estimates are obtained, they are treated as fixed and some other methodology (e.g., maximum likelihood [ML], Bayesian expected a posteriori [EAP]) is used to obtain ability estimates.

Bayesian methods.

Generally speaking, IRT Bayesian methods are modifications of either JML or MML estimation where a priori assumptions are made about the distribution of item parameters. These assumptions can be applied either formally or informally. For example the LOGIST software program (Wingersky, Barton, & Lord, 1982) uses JML along with an informal method of specifying the item parameter distributions by placing upper and lower limits on the *a* and *c* parameters (Mislevy & Stocking, 1989).

In formal applications of Bayesian methods, a prior distribution is specified and multiplied by the likelihood function to produce a posterior distribution from which parameter estimates are obtained (Baker, 1987). The BILOG software program (Mislevy & Bock, 1997) is based on MML estimation, but by default uses Bayesian methods of estimation for certain item parameters. For the 3PL model, discrimination parameters are assumed to follow a log-normal distribution, the difficulty parameters are assumed to follow a normal distribution, and the guessing parameter is assumed to follow a beta distribution.

The specific parameters describing these prior distributions (i.e., hyperparameters) are either specified by the user or estimated from the data (Mislevy & Stocking, 1989).

Comparison of JML, MML, and Bayesian estimation methods.

Although more detail about the effect of the estimation method on the accuracy of IRT parameter estimation is presented in a subsequent section, this section briefly summarizes the positive and negative aspects of the three estimation methods described above. JML is appealing in its lack of assumptions about item and ability parameter distributions. However, it has several shortcomings. Item estimates cannot be obtained for items where all examinees answered the question correctly (or all answered incorrectly). Similarly, ability estimates cannot be obtained for examinees who answer all items correctly (or all incorrectly). These items and examinees are removed from the data set before estimation begins. Another limitation is that JML tends to produce biased estimates, especially for short tests (Lord, 1983, 1986).

MML has an advantage over JML in that it can produce more accurate item parameter estimates, especially when sample sizes are small, because item parameters are estimated without the need to simultaneously estimate ability parameters (Lord, 1986). However, both MML and JML estimation methodologies suffer from the disadvantage that they can result in either infinite or implausible parameter estimates.

Although researchers have found including prior information about item parameters can improve estimation results over MML and JML (e.g., Kim, 2007; Mislevy, 1986; Swaminathan & Gifford, 1986), in some cases Bayesian estimation methods may result in

estimates regressing to the mean of a specified prior distribution or may cause restriction of range problems (Kim, 2007).

Research on the Necessary Sample Size/Test Length for Item Parameter Estimation

Although there are no hard and fast rules for the minimum sample size and test length for IRT parameter estimation, there are many recommendations and rules of thumb. Jones, Smith, and Talley (2006) provide a comprehensive summary of these recommendations not just for IRT but for many aspects of test development in the context of pretesting. As sample size requirements of the 3PL model tend to be some of the largest of all IRT models, these are the focus of this section.

As early as 1968, Lord suggested using test lengths of at least 50 items and sample sizes of at least 1,000 when using JML to estimate 3PL model parameters in order to control the sampling error of the discrimination parameter estimates. Hulin, Lissak, and Drasgow (1982), conducted a simulation study using LOGIST to determine the accuracy of item parameter estimates obtained using JML estimation. They examined sample sizes of 200, 500, 1000, 2000, test lengths of 15, 30, and 60 items, and two different θ distributions. Using the root mean squared error (RMSE) of the ICC and the correlations between estimates and parameters as evaluative criteria, they concluded that the tests consisting of 30 or 60 items taken by at least 1,000 examinees were sufficient to produce accurate estimates of the ICCs.

In an analysis of the effect of sample size on linear equating, Ree and Jensen (1983) examined several combinations of calibration and equating sample sizes. They suggested a minimum sample size of 500, but recommended administering test items "to the largest samples available" (p. 145). Hambleton and Cook (1983) simulated tests of 10, 20, and 80

items with sample sizes of 50, 200, and 1000 in order to determine the effect of sample size on the standard errors of ability estimation curves. Ability scores were drawn from a standard normal distribution, and item parameters were estimated using heuristic estimation software (Urry, 1974). They concluded that adequate precision could be obtained near the center of the ability continuum under most testing conditions with a test length of 20 and a sample size of 200.

These and other empirical and simulation studies have been reviewed and compiled elsewhere, with recommendations consistently pointing to a minimum sample size of 1000 for the 3PL model using maximum likelihood estimation(either JML or MML; see, for example, de Ayala, 2009, Hambleton, 1993; Jones et. al., 2006; and Yen & Fitzpatrick, 2006). De Ayala, however, argues strongly against the use of hard-and-fast sample size and test length and rules, and emphasizes the need for consideration of additional factors (e.g., test purpose, missing data, estimation method) when making any sample size determination.

Item Parameter Estimation with Small Sample Sizes and/or Reduced Test Length

Various strategies have been proposed for improving the accuracy of IRT parameter estimates. Prominent examples of these strategies include estimation software, simplified/modified IRT models, optimal examinees, and prior distributions/information. In this section, summaries of each of these strategies, as well as their pros and cons, are discussed.

Estimation software.

As mentioned above, there are several methods for estimating IRT parameters. In practice, these estimation methodologies are compared in the context of the software

programs in which they are implemented: “the item parameter estimation issue is inextricably interlinked with the computer software that implements the estimation procedures. The characteristics of the obtained item parameter estimates are critically dependent upon the manner in which the underlying mathematics are implemented in the software” (Baker, 1987, p. 137). Although most of these software packages are able to estimate several IRT models this section focuses on research involving comparisons between programs estimating 3PL item parameters.

Perhaps the earliest published study comparing 3PL estimation software programs was conducted by Jensema (1976). He compared a simple estimation technique based on classical test theory item statistics (e.g., point biserial correlations) for obtaining initial item parameter estimates with a self-written joint maximum likelihood estimation program (Jensema, 1972). The effects of the number of examinees (250, 500, 750, 1000, or 2000), the number of items (25, 50, or 100), the magnitude of the discrimination (0.5, 1.0, 1.5, or 2.0) and item difficulty (-2.4 to 2.4, by 0.2) parameters were examined. Forty-eight simulated data sets were created, one for each sample size/test length/discrimination parameter combination. Jensema evaluated the efficacy of the procedures by calculating the correlation between the estimated and true item parameters. The simple estimates resulted in correlations very similar to those obtained through maximum likelihood estimation.

Another early study was conducted by Ree (1979) using a UNIVAC 1108 computer. This study compared the ANCILLES (Urry, 1978), LOGIST (Wood, Wingersky, & Lord, 1976), and OGIVIA (Urry, 1977) software programs. ANCILLES and OGIVIA are based on estimation heuristic procedures developed by Urry (1974), and LOGIST uses joint maximum

likelihood estimation. The study compared these three programs using simulated data from three ability distributions (uniform, skewed, normal), and was designed to simulate an 80-item multiple-choice test taken by 2,000 examinees per distribution. Item parameters were generated from normal distributions. Results were evaluated using the correlations between the estimated and true item parameters, correlations between estimated and true abilities, the differences between estimated and true scores, and the accuracy of the test characteristic curve (TCC[1]). Although results varied based on the evaluative criteria, LOGIST tended to produce the best results for the uniform θ distribution. ANCILLES and OGIVIA had some difficulty estimating item parameters from the skewed data set. The OGIVIA software produced the best estimates when θ was normally distributed.

The work of Ree (1979) was extended by Swaminathan and Gifford (1983) through the examination of additional test lengths (10, 15, 20, and 80) and sample sizes (50, 200, 1000, and 2000). They also compared the ANCILLES and LOGIST software programs. Data were generated based on uniform distributions for the *a* (U[0.6, 2.0]) and *b* (U[-2.0, 2.0]) parameters; the values of the *c* parameter were fixed at .25. LOGIST produced better estimates of the *a* parameters than did ANCILLES, with ANCILLES having an overestimation bias. LOGIST also produced better estimates for the *b* parameters, *c* parameters, and ability parameters (for the 10 item tests; there was no ability estimation difference between the programs for tests of 15 items or more). The accuracy of ANCILLES's estimates approached those of LOGIST as test length and sample size were increased.

[1] The test characteristic curve is the graphical representation of the test characteristic function (TCF). The TCF is the expected raw test score of an examinee given his/her ability, θ (Yen & Fitzpatrick, 2006).

Swaminathan and Gifford examined the accuracy of LOGIST again in 1986, this time as a baseline for comparison against a Bayesian estimation procedure written by the authors. The simulation study included sample sizes of 100, 200, and 400 and test lengths of 25 and 35 items. The difficulty and ability parameters were drawn from a unit normal distribution, and the discrimination and guessing parameters were drawn from uniform distributions (U[0.5, 2.0] and U[.04, .22], respectively). The procedures were compared using the mean squared differences and the correlations between the parameters and their estimates. The Bayesian procedure tended to result in higher correlations and lower mean squared differences than did LOGIST. Unlike LOGIST sometimes does, the Bayesian procedure did not yield extreme parameter estimates (e.g., discrimination estimates greater than 10 or difficulty estimates outside of [-5, 5]).

LOGIST was again compared to a Bayesian procedure in Yen's comparison of LOGIST and BILOG (cited in Mislevy & Bock, 1984). This study included simulated tests ranging from 10 to 40 items (one 10-item test, four-20 item tests, and four-40 item tests) and a sample size of 1,000. Four ability distributions were examined (standard normal, skewed right, skewed left, and symmetric platykurtic). Item parameters varied by test, with discrimination and guessing parameters fixed at 1.0 and .2, respectively, in some cases. LOGIST and BILOG were used with the default options. The accuracy of the procedures was evaluated using several methods including RMSEs, correlations, and measures of the accuracy of item and TCCs. The BILOG item parameter estimates were consistently better than those obtained using LOGIST. In terms of ability estimates, BILOG performed better for the short (i.e., 10-item) test, whereas LOGIST performed better for the longer (i.e., 40-

item) tests. Although BILOG produced more accurate ICC estimates for the 10-item test, the programs performed similarly for the 20- and 40-item tests.

LOGIST was evaluated again in a 1988 study by Vale and Gialluca. In this case it was compared with ASCAL (Vale & Gialluca, 1985), a version of ANCILLES (Croll & Urry, 1978), and a heuristic estimation method (Jensema, 1976). ASCAL is based on joint maximum likelihood, but includes prior distributions for the ability (standard normal), discrimination (symmetric beta), and guessing (symmetric beta) parameters. Three sets of simulated the item responses were created based on item parameters from real tests: one 50-item test, a second 50-item test based on the first but with the difficulty parameters multiplied by two, and 57-item test with more difficult/less discriminating items. Three sample sizes were examined: 500, 1000, and 2000 (LOGIST was only used in the $N = 2000$ cases). The evaluative criteria included the RMSE of the item parameter estimates, the correlation between the estimated and true parameters, and a calibration efficiency criterion developed by Vale, Maurelli, Gialluca, Weiss, and Ree (1981), which uses "the ratio of the information in the estimated parameters to the information in the true parameters" (Vale & Gialluca, 1988, p. 58). ASCAL produced the best estimates of the parameters in most cases in terms of both the RMSE and correlation. ASCAL also resulted in the highest item calibration efficiency in all but one case. Differences between the procedures became less pronounced as the sample size was increased.

In a small simulation study Mislevy and Stocking (1989) compared LOGIST and BILOG. They evaluated the software performance on 15- and 45-item tests, both with 1,500 examinees. The authors concluded that the programs performed similarly for the 45- item

test, but BILOG appeared to estimate the item parameters more accurately for the 15-item test.

Similar to Vale and Gialluca (1988), Skaggs and Stevenson (1989) compared the performance of ASCAL and LOGIST. They examined simulated data for two test lengths (15 and 35 items) and two sample sizes (500 and 2,000 examinees). All item parameters were generated from uniform distributions ($a \sim U[0.4, 1.2]$, $b \sim U[-2.0, 2.0]$, $c \sim U[.1, .3]$). Three different normal ability distributions were used with means of -1.0, 0.0, and 1.0 (all with standard deviation equal to 1.0). Five replications were used for the 15-item tests, and two replications were used for the 35-item tests. The resulting procedures were evaluated using the average RMSEs of the recovered ICCs, the correlations between the estimated and true parameters, and estimates of the bias and the absolute value of the bias. With respect to the RMSEs, ASCAL outperformed LOGIST when the sample size was 500, but LOGIST performed better on the 15-item test when the sample size was 2000. There was only a negligible difference between the programs for the 35-item/2,000 examinee condition. In terms of correlations, the programs performed similarly for the *b* and *c* parameters, with mixed results for the *a* parameter. LOGIST performed better for the larger sample size and worse for the smaller sample size.

In 1993, Yoes provided a comprehensive review of studies examining item parameter estimation techniques for the 3PL model. He also conducted a comparison of LOGIST (Wingersky, Barton, & Lord, 1985), ASCAL (Assessment Systems Corporation, 1987, 1989), and BILOG (Mislevy & Bock, 1986, 1989). In this large simulation study, several factors were examined including sample size (250, 500, 1000, and 2000), test length (15, 20,

50, 75, and 100 items), ability distribution (standard normal, uniform, and negatively skewed). Four different tests were constructed using two different normal distributions for the discrimination parameter, a normal and a uniform distribution for the difficulty parameter, and a normal distribution for the guessing parameter. Several evaluative criteria were used for the item parameter estimates: the RMSEs of the estimates, the average bias of the estimates, the RMSEs of the recovered ICCs, and the RMSEs of the test information functions[2]. These criteria were examined both descriptively and using analysis of variance (ANOVA). The programs were run using primarily the default settings. For the discrimination parameter, ASCAL tended to produce fewer extreme values than LOGIST or BILOG; BILOG tended to have the lowest average bias and median RMSE. All programs did a good job estimating item difficulty with BILOG doing the best for short tests (n=15) and small sample sizes (N=250). Although all programs had difficulty producing accurate guessing estimates, BILOG tended to do the best job with ASCAL performing as well on longer tests (n≥50). BILOG also did the best job reproducing ICCs on short tests (n=15); differences between the programs were small with long tests (n≥50) and large sample sizes (N≥500).

Several of the previous studies have evaluated the estimation accuracy of BILOG. However, most of them do so using only one set of program specifications. In a study conducted by Abdel-fattah (1994), LOGIST was compared to BILOG using two different sets of program specifications: marginal maximum likelihood and marginal Bayesian

[2] Item information for an item at a particular value of θ is equal to the expected value of the inverse of the error variance (Thissen, 2000). The item information function describes the information for the item across the range of θ values. The test information function is the sum of the item information functions (Wendler & Walker, 2006).

estimation. In this simulation study the other factors that were varied included the test length (20 and 60 items), the sample size (250 and 1,000 examinees), and the ability distribution (normal, truncated normal, and beta). The discrimination, difficulty, and guessing parameters were generated from lognormal, normal, and beta distributions, respectively. For the Bayesian BILOG analyses, the program's default priors were used. The programs were compared using the correlations of the estimates with the true parameters along with the bias, variance, and mean squared error of the estimates. The estimates of the difficulty and discrimination parameters from BILOG with priors were more accurate than those obtained from BILOG without priors or from LOGIST, especially with small sample sizes and short tests. However, LOGIST did as well or better than either of the BILOG variations when estimating the guessing parameters when abilities followed a beta distribution. BILOG estimates were more accurate when the ability distribution was normal.

Although BILOG clearly outperformed LOGIST in most situations in the previous study this was not the case when BILOG was compared to TESTGRAF (Ramsay, 1993). Patsula and Gessaroli (1995) conducted a simulation study comparing BILOG to TESTGRAF's nonparametric IRT estimation approach. The factors investigated were test length (20 and 40 items) and sample size (100, 250, 500, and 1000). Item parameters came from calibrations of the ACT test. Abilities were simulated to follow a standard normal distribution. One hundred replications per cell were conducted. The programs' estimation accuracy was evaluated using the estimates' mean bias and RMSE. Results were analyzed using factorial ANOVAs. Main and interaction effects were considered important if they had a large effect size (i.e., $f^2 > .35$). TESTGRAF produced less biased estimates of all item

parameters. For the discrimination parameter, the TESTGRAF results were more accurate at small sample sizes than those of BILOG, on average. There was little difference between the programs in terms of RMSEs. BILOG had lower RMSEs for the guessing parameter than did TESTGRAF for small sample sizes (n=100 and n=250), but the difference decreased as sample size increased. Overall, for the conditions examined in this study, TESTGRAF tended to produce better results on average than did BILOG. However, the differences between the programs became trivial with larger sample sizes. (It was unclear whether the authors used Bayesian priors with the BILOG software.)

Yoes (1995) expanded on his 1993 study with an evaluation of the ASCAL (Assessment Systems Corporation, 1987), BILOG, XCALIBRE (Yoes, 1996), and LOGIST software programs. XCALIBRE uses MML estimation and uses priors on the item and ability parameter distributions (Gierl & Ackerman, 1996). This simulation study used the same sample sizes and test lengths as Yoes (1993), but used only a standard normal ability distribution and two of the original four sets of item distributions. The default options generally were used for all programs. BILOG and XCALIBRE were allowed to update prior distributions at each stage of estimation, and BILOG was allowed to empirically determine the distribution of abilities. The evaluative criteria were similar to those of Yoes (1993). For the discrimination parameters, XCALIBRE's estimates had the highest correlation with the true parameters; XCALIBRE and ASCAL were better at preventing unrealistic estimates BILOG and LOGIST. BILOG and XCALIBRE did the best job estimating item difficulties for small sample sizes and short tests; all programs performed similarly for larger sample sizes (N=1,000 or N=2000) and longer tests (n=75 or n=100). All programs performed

poorly when estimating the guessing parameters, but BILOG and XCALIBRE had the lowest RMSEs. For the RMSEs of the ICCs, XCALIBRE produced the most accurate estimates. Overall, the two MML-based programs, BILOG and XCALIBRE, performed better than the JML-based programs, ASCAL and LOGIST.

All of the simulation studies discussed have been based on data generated to fit a unidimensional IRT model. Kirisci, Hsu, and Yu (2001) evaluated the performance of BILOG, MULTILOG (Thissen, 1991), and XCALIBRE in the case of multidimensional data. MULTILOG is an extension of BILOG; it uses MML estimation and can provide estimates for polytomous items (Hambleton, 1993). The authors simulated both unidimensional and three-dimensional data (with correlations of .6 between the three factors). They also varied the ability distribution (standard normal, skewed, and platykurtic). All item parameters were generated from uniform distributions ($a \sim$ U[0.4, 2.0], $b \sim$ U [-2.0, 2.0], $c \sim$ U[0, .3]). The sample size, test length, and number of replications per cell were 1000, 40, and 10, respectively. All three programs used similar options, including marginal maximum likelihood estimation; no priors on the item parameters were used. The results were compared using factorial ANOVA analyses on the RMSEs of the item parameter estimates. Main and interaction effects were considered important if they were both significant ($\alpha <$ .001) and had a medium effect size (i.e., $\eta^2 > .09$). There was a significant interaction between the software and the dimensionality of the data for all item parameters except for the guessing parameter. MULTILOG had the lowest average discrimination RMSE when the data were unidimensional, but BILOG was best for the three-dimensional data. BILOG and XCALIBRE had the lowest average difficulty RMSE when the data were unidimensional,

and BILOG was also the best for the three-dimensional data. For the guessing and ability estimates, BILOG produced the best estimates in all conditions.

Many of the studies described above compare software that uses JML estimation to software that implements MML or Bayesian estimation. Any of these estimation approaches may be conjoined with DupER Augmentation.

Simplified/modified models.

Another way researchers have approached the problem of obtaining accurate parameter estimates with smaller sample sizes is to employ simplified/modified IRT models. This section summarizes several studies in which researchers used simplified/modified IRT models to analyze data that were designed/believed to follow a more complex model.

In 1983, Lord argued that when sample sizes are small, simple IRT models may provide more accurate results than more complex models, even when the more complex models theoretically should provide a better fit to the data. He evaluated this claim using item parameters taken from data from 3,000 sixth-grade students who took a 50-item Metropolitan vocabulary test. He concluded that when sample sizes were less than 200 the 1PL model resulted in more accurate ability estimates than did the 2PL model.

In addition to using simplified models, researchers have examined the impact of using modified models. Barnes and Wise (1991) evaluated the efficacy of a 1PL model with a fixed nonzero lower asymptote. In their simulation study they examined this modified 1PL model with *c* fixed at one of two levels, .20 and .25. These models were compared to the 1PL ($c = .0$) and 3PL models across three sample sizes (50, 100, and 200) and two test lengths (25 and 50 items). The simulated data were based on ability and difficulty

parameters generated from a standard normal distribution in the range of -3 to 3, $.5 \leq a \leq 2.0$, discrimination parameters ranged from .50 to 2.0, and guessing parameters that ranged from .10 to .30. Correlations, RMSEs, and bias of both the ability and the difficulty parameter estimates were used to evaluate results. Additionally, the RMSEs of the ICCs were examined. Five replications were carried out per cell. The 3PL model had the greatest problems with convergence. Ability estimates obtained using the modified 1PL models tended to have higher correlations with the true parameters than did estimates obtained using the 1PL and 3PL models. The modified 1PL model with the lower asymptote fixed at .20 produced the most accurate recovery of the ICCs. The authors suggested that a modified 1PL model may be the best choice in testing conditions similar to those simulated (i.e., relatively small sample sizes with tests of moderate length).

Sireci (1992) also examined the utility of modified IRT models, but used real rather than simulated data. The data were obtained from four administrations of a national financial planning certification examination over four years. Sample sizes were 173, 149, 106, and 159 examinees. The primary goal of the study was to evaluate the stability of item parameters for 13 test items that were common across all four test forms. Five IRT models were compared: the 1PL, 2PL, 3PL, modified 1PL (c =.20), and the modified 2PL (c =.20). The fixed value of the discrimination parameter was chosen to be the reciprocal of the number of answer choices (i.e., 4) minus .05. Item parameter estimates were obtained using MULTILOG. None of the models exhibited item parameter stability over the four data sets. Therefore, the author concluded that none of the evaluated models was appropriate for these small data sets.

Parshall, Kromrey, and Chason (1996) compared regular and modified IRT models with respect to model-data fit and stability. Simulated data were generated from 3PL item parameters obtained from a 40-item ACT mathematics assessment. Examinee abilities were generated from a standard normal distribution. Six models were examined: 1PL, 2PL, 3PL, modified 2PL (the discrimination parameter was restricted using a strong prior distribution), and two different modified 3PL models (the discrimination parameter was restricted using a strong prior distribution and one model had a common guessing parameter, which was estimated from the data, but constrained to be equal for all items). Four sample sizes were examined (100, 250, 500, and 1000). One hundred replications were conducted for each experimental condition. The BILOG software program was used for all calibrations. Model-data fit was evaluated using item and person residuals. Stability was evaluated using the standard deviations of the discrimination and difficulty parameters, and the ICCs across replications. The 3PL and modified 3PL (restricted *a*) models had the smallest item and person residuals for most sample sizes. However, these models had the least stable difficulty estimates across replications; the most stable estimates were obtained using the 1PL and modified 2PL models. The most stable discrimination estimates were obtained from the models that constrained the discrimination parameters (i.e., the 1PL and modified models).

Setiadi (1997) compared a modified 1PL model ($c = .20$) with the 1PL model (estimated using MML and several Bayesian variations) and the 3PL model. The 3PL model was estimated using the non-parametric TESTGRAF software program. The 1PL and modified 1PL models were estimated using BILOG. Item parameters for the simulation study were chosen from both real and hypothetical testing situations. Data were generated

based on the 3PL model. The author examined two test lengths (30 and 60 items), three sample sizes (100, 200, and 500), two sets of item parameters (one taken from the Law School Aptitude Test and one created by the author with higher discrimination values) and two ability distributions (normal and uniform). One hundred replications were conducted for each condition. Results were evaluated using correlations, average errors, absolute bias, standard deviation of estimation errors, and RMSEs of item parameters. Setiadi found that the modified 1PL model resulted in more accurate estimation of ability than did the other models when ability was normally distributed. For the uniformly distributed data, the modified 1PL model had the most accurate item parameter estimates.

Parshall, Kromrey, Chason, and Yi (1997) expanded on the earlier work of Parshall, Kromrey, and Chason (1996) by examining the efficacy of modified models in the presence of multidimensional data. As in the earlier study, six models were examined: 1PL, 2PL, 3PL, modified 2PL (the discrimination parameter was restricted using a strong prior distribution), and two different modified 3PL models (the discrimination parameter was restricted using a strong prior distribution one model had a common guessing parameter, which was estimated from the data, but constrained to be equal for all items). Simulated item parameters for an 80-item, 6-dimensional test were generated from archival assessment data. Examinee abilities were generated using independent standard normal distributions for each dimension. Four sample sizes were examined (100, 250, 500, and 1000), and one hundred replications were conducted for each experimental condition. Parameter estimates were obtained using BILOG. The authors used the same evaluative criteria as the earlier Parshall et al. study with the addition of the mean squared error of the expected response probabilities, the RMSE of

the estimated number correct for each examinee, and the Spearman correlation of the estimated number correct score and the true number correct score. Results showed that the 2PL model provided the best fit to the data. However, with respect to estimation accuracy, the best results were obtained from the 3PL model and the modified 3PL model with restricted discrimination values.

In contrast to the studies described above, Stone, Weissman, and Lane (2005) compared competing IRT models with respect to the consistency of student proficiency classifications. That is, rather than examining the accuracy of ability estimates or scale scores, they evaluated the accuracy of classifications based on these estimates. This study used real data from 13,621 11^{th}-grade students from a 1999 state mathematics assessment. The test consisted of 60 multiple-choice items. 1PL and 3PL models were fit using the MULTILOG software program. Using the bookmark standard-setting procedure, the score scale was divided into four categories: Below Basic, Basic, Proficient, and Advanced. A standard-setting panel used an ordered item booklet with the items ordered based on the 1PL model. The four performance categories were identified using the difficulties of three items. The same three items were used to compare student performance classifications based on the competing IRT models. Based on the two competing IRT models, students were classified into different performance categories about 10% of the time. In the same paper, the authors discussed the results of a simulation study based on the same data. That is, the item parameters were the 3PL estimates from the real data and abilities were generated from a standard normal distribution. With the simulated data, comparisons could be made between estimated and true performance classifications. When the 1PL model misclassified students,

it tended to underestimate their ability. However, under the 3PL model misclassifications were more equally balanced between under- and overestimation.

The studies described above provide comparisons of various competing IRT models. In all of these cases, the alternative models can be thought of as special cases of the 3PL model. Although CTT is neither a simplified nor modified version of the 3PL model, it can be thought of as a simpler alternative procedure. Several researchers have compared the item and person statistics generated from CTT and IRT (e.g., DeMars 2001, Fan, 1998, Hwang, 2002, and Macdonald & Paunonen, 2002). In most cases, the conclusions drawn based on the CTT analyses are very similar to those from IRT.

Clearly, simplified/modified IRT models may be viable alternatives to the 3PL model in situations where sample sizes may be small, such as licensure testing. However, these models are less helpful in situations where a relatively small sample is used to obtain item parameter estimates that are then treated as known and used to build and/or administer a test based on the 3PL model (e.g., a large-scale CAT). Additionally, simpler models may result in worse estimates when the data fit a more complex model. For example, Hambleton and Cook (1983) found that for data generated with the 3PL model, the 3PL model resulted in more accurate rank-ordering of examinees than did the 2PL model.

"Optimal" examinees.

Several researchers have attempted to improve item estimates (or obtain equally good estimates using smaller sample sizes) by choosing examinees in such a way as to get the most accurate item estimates possible. Wingersky and Lord (1984) investigated the effect that changing the number of items, number of examinees, and the distribution of examinee

abilities had on the accuracy of item parameter estimates using real data from a regular administration of the Test of English as a Foreign Language (TOEFL). They used the 3PL model LOGIST for estimation (Wingersky et. al., 1982), and either a rectangular distribution with 1,500 examinees and 45 items or bell-shaped distributions of examinee abilities of either 1,500 or 6,000 examinees and either 45 or 90 items.

They found that the standard errors of item parameter estimates became smaller as sample size increased, but were not substantially impacted by increasing the number of items. Conversely, they found that the standard errors of examinee ability estimates decreased as the number of items was increased, but were not substantially impacted by increasing the examinees. Additionally, they found that rectangular distributions of examinee abilities gave smaller standard errors for the item parameter estimates than did the bell-shaped distributions, indicating that better item parameter estimates could be obtained if examinees were selected systematically based on their ability. This recommendation was supported by a simulation study by Hambleton and Cook (1983), who found that the rank-ordering of examinees was more accurate when examinee abilities were generated using a uniform distribution rather than a standard normal distribution.

Stocking (1990) expanded on the work of Wingersky and Lord by evaluating which examinee abilities provide the most information for estimating item parameters for the 1PL and 2PL models, as well as the 3PL model used in Wingersky and Lord (1984). Stocking showed that for the 3PL model:

- Both low and high ability examinees provide little information for estimating item discrimination (as do those with abilities close to the optimal value for

estimating item difficulty); the most informative examinees have abilities just above or just below the item difficulty.

- Examinees provide the most information for estimating item difficulty when their ability is equal to the item's difficulty parameter, but when the guessing parameter is greater than zero the optimal ability level for estimating difficulty is greater than the difficulty parameter and depends on the item's discrimination and guessing parameters.
- Only examinees with very low abilities provide information for estimating guessing parameters.

Thus, the examinee who provides the most information for estimating the difficulty parameter may be very different from the examinee who provides the most information for estimating the discrimination parameter, who also may be different from the examinee who is most useful for estimating the guessing parameter. Stocking concluded that selecting samples of examinees where ability was distributed either uniformly or bimodally would serve as a good compromise for overall item parameter estimation accuracy.

Research like that of Timminga (1995) and Berger, King, and Wong (2000), who examined methods to select examinees in a more systematic way, have expanded the work of Wingersky and Lord and Stocking. Instead of studying distributions of panelist abilities, Timminga used multi-objective programming. This procedure uses mathematical models to obtain optimal results based on a specific set of goals and constraints to choose the number of examinees at given ability levels in such a way as to make the worst-case scenario as acceptable as possible. In other words, one should choose examinees whose abilities

maximize the minimum amount of information for *b* obtained across items (or *a* or *c*). Timminga demonstrated the method using a maximum number of 500 examinees chosen from up to 21 evenly spaced ability levels from -3 to 3 (both with and without limits on the number of examinees at each ability level) and seven items with known parameters ($a = 1$ or 2; $b = -2, -1, 0, 1,$ or 2; $c = 0, .2,$ or .4). The resulting "optimal" design produced better estimates of *c* than typical examinee samples (i.e., abilities distributed U[-3,3], N(0,1), or N(0,2)), but worse estimates of *a* and *b*. The author hypothesized these results likely occurred because the "optimal" design used in the study valued all item parameters equally, but typical samples estimate *c* poorly because of small numbers of very low ability examinees. The author also noted weaknesses of this method: it requires that both examinees' abilities and the item parameters be known in advance, which is seldom, if ever, the case and the resulting sample is only optimal for the specific set of item parameters on which it is based (i.e., locally optimal).

Berger et al. (2000) at least partially overcame these limitations. Their design was similar to Timminga's in its use of mathematical programming to determine an optimal sample of examinees for estimating item parameters. However, their design differed in that although examinee abilities had to be known, the user supplies a range of parameter values (rather than the actual item parameters). Their criterion for an optimal sample was similar in that they sought to minimize the worst possible case over the range of user-supplied item parameter values. They found that their optimal designs tended to be symmetric about the middle of the theta scale, that the number of different ability levels tended to increase as the range of parameter values increased and tended to be more uniformly distributed than locally

optimal designs. Their methodology resulted in samples that produced estimates nearly as accurate as locally optimal designs, but did not require specific knowledge of the item parameters.

The above studies show that parameter estimates can be improved (or can be estimated equally well with a smaller sample size) by choosing examinees in a systematic way, such as according to a particular distribution or a sophisticated mathematically derived sampling design. However, each of these methods suffers from the weakness that (at least) the abilities of the examinees need to be known before the test is given. In most testing situations, only rough estimates of examinee ability may be available based on other measures, at best.

Prior distributions.

In the previous section on estimation software, several studies were discussed in which researchers compared software using maximum likelihood procedures to software using Bayesian procedures where prior distributions for some or all parameters were specified (e.g., Mislevy & Stocking, 1989; Swaminathan & Gifford, 1986; Yen, 1987; Yoes, 1993). This section focuses on research comparing maximum likelihood and Bayesian procedures within a software program as well as comparisons of competing prior distributions.

In 1986, Mislevy published a paper describing the application of a Bayesian framework to MML estimation. This paper presented a numerical comparison of standard MML estimation to Bayesian estimation using the BILOG software program. Data were simulated for 1,000 examinees from a standard normal ability distribution and a 20-item

assessment with ability, discrimination, and guessing parameters all generated from normal distributions. For the Bayesian estimation, BILOG's "floating priors" option, in which the values of priors are re-estimated after each iteration, was used. The different estimation procedures produced similar results for the discrimination and difficulty parameters. However, though the Bayesian estimates of the guessing parameter tended to regress toward the mean of the prior distribution, they tended to be more stable and reasonable than the estimates obtained using MML.

Gifford and Swaminathan (1990) compared the effect of different prior distributions of item parameters on the accuracy and bias of estimation. They examined 1PL, 2PL, and 3PL models individually. The results most germane to the present study are those for the 3PL model. For this model, data were simulated for a single testing situation: a 35-item test taken by 200 simulees. Ability and difficulty values were selected using a uniform distribution ranging from -1.73 to 1.73. The discrimination and guessing parameters were also drawn from uniform distributions (U[0.6, 1.9] and U[.00, .22], respectively). Non-informative priors were used for the ability and difficulty parameters, and a chi prior distribution was used for the discrimination parameters. Nine different beta distributions were selected as possible priors for the guessing parameters. Although varying the prior for *c* had little effect on the estimation of ability and difficulty, discrimination and guessing estimates were affected by the choice of prior. The best discrimination estimates were obtained using a diffuse *c* prior. Unsurprisingly, the best guessing estimates were obtained when the mode of the prior distribution was at the center of the generating guessing distribution (i.e., U[.00, .22]). However, the authors concluded that different prior distributions for the parameters

"do not have any marked effect on the estimation as long as the prior distributions are not too extreme" (p. 43). The authors did not specify the estimation method or software.

The above studies compared the use of different prior distributions for item parameters. With MML estimation there is also an assumption about the population distribution of examinee abilities. Seong (1990) evaluated the effect of varying the prior ability distribution on the estimation of both item and ability parameters. Seong varied the sample size (100 and 1,000), prior ability distribution (normal, positively skewed, and negatively skewed), the number of quadrature points (10 or 20), and the underlying population ability distribution (normal, positively skewed, and negatively skewed). 2PL model data were generated to simulate a 45-item test with item difficulties of -1, 0, or 1 and discrimination parameters ranging from .3 to 1.1. Five replications were performed for each set of experimental conditions. The data were calibrated using the BILOG software program with the default prior distributions for the discrimination parameters and no prior for the difficulty parameters. Ability estimates were obtained using Bayesian EAP estimation. Results were evaluated using the RMSEs and absolute bias in a split-plot factorial design. In most cases, estimates were most accurate when the prior ability distribution matched the underlying ability distribution; this was true for the discrimination, difficulty, and ability estimates. Not surprisingly, estimates were worst when the prior and underlying ability distributions were skewed in opposite directions.

Like Gifford and Swaminathan (1990), Harwell and Janosky (1991) examined the effect of varying prior distributions of item parameters with MML-based estimation. Simulated data were generated based on the 2PL model for six different samples sizes (75,

100, 150, 250, 500, and 1000) and two test lengths (15 and 25). Ability and difficulty parameters were generated using a standard normal distribution restricted to the range of -3 to 3. Discrimination parameters were generated using a uniform distribution ranging from .6 to 1.9. Parameters were estimated in BILOG using four different prior variances for the discrimination parameter distribution ($.75^2$, $.5^2$, $.25^2$, and $.1^2$, all in a lognormal metric). Estimates were also obtained without using priors. Results were evaluated using correlations, RMSEs, and trimmed RMSEs (i.e., the RMSE with the 10 largest errors removed). For sample sizes less than 250, the smaller (i.e., more stringent) prior variances produced the most accurate results. At larger sample sizes, estimation differences based on the different priors were small. Similarly, smaller prior variances resulted in more accurate estimation for the 25-item test at sample sizes of 75 and 100 (but differences were negligible at larger sample sizes).

In addition to evaluating the efficacy of a modified 1PL model, Setiadi (1997) evaluated several different prior distributions for the difficulty parameter of the 1PL model. All item parameters were generated from uniform distributions ($a \sim$ U[0.4, 1.8], $b \sim$ U [-2.0, 2.0], $c \sim$ U[0, .25]), and the ability parameters were generated from a standard normal distribution. A single test length of 30 items was used with three different samples sizes (100, 200, and 500). Five replications were generated for each set of experimental conditions. Seven different prior distributions were examined. The priors varied in their variances and whether they were centered or not centered at the true parameters' values. The best results were obtained with centered priors (i.e., matching the generating difficulty

distribution) and small prior variances. Like Harwell and Janosky (1991), the author found the choice of prior was less important with large sample sizes.

Seong (1990) recommended choosing a prior distributions based on either "theoretical or empirical considerations" (p. 310). Swaminathan, Hambleton, Sireci, Xing, and Rizavi (2003) provided an illustration of an empirical method for determining a prior distribution for the difficulty parameters. The authors derived possible prior distributions based on the judgments of a panel of subject matter specialists and test developers. Specifically, panelists estimated the proportion of examinees who would answer each item correctly. These proportions were converted to the *b* scale using a transformation based on the density function of the normal distribution. Finally, the mean of the prior distribution for the difficulty parameter was set to the average of the panel-estimated *b* values.

To evaluate this methodology, Swaminathan et al. used data from 5,000 examinees who took the 21-item reading comprehension section of the Law School Admissions Council test. The 3PL item parameter estimates obtained from analyzing the full data set were treated as parameters. Six randomly drawn sample sizes (100, 150, 200, 300, 400, and 500) and three prior distributions based on the panelists' judgments were evaluated. These three priors used the average of the panel-estimated *b* values as the mean and a standard deviation equal to 1, 2, or the standard deviation of the panel-estimated *b* values. Six other prior distributions were examined with the mean estimated from the population proportion correct, the sample proportion correct, or simply set to zero (each with a standard deviation of either 1 or 2); a condition without priors was also included. Item parameters for the 1PL, 2PL, and 3PL models were estimated using BILOG. Each experimental condition was replicated 100 times

using repeated samples from the full 5,000-examinee data set. Results were evaluated using RMSEs, bias, and standard error. The difficulty parameters were estimated most accurately for all models (1PL, 2PL, and 3PL) using the priors based on the population proportion correct and the proportion correct estimated by the panelists (both with *SD* = 1). This was true at all sample sizes, but less pronounced for larger sample sizes. The 2PL and 3PL models' discrimination parameters were estimated well using both the standard normal prior for the difficulty parameter and the prior based the proportion correct estimated by the panelists (*SD* = 1). The prior based on the proportion correct estimated by the panelists (*SD* = 1) also resulted in the most accurate estimation of the guessing parameters.

Although research has consistently shown that including prior information about item and/or ability parameters can improve estimation results, in some cases Bayesian estimation methods may result in estimates regressing to the mean of the specified prior distribution or may cause restriction of range problems (Kim, 2007). Additionally, Swaminathan et al. (2003) noted that while item parameter estimates were improved, obtaining prior information regarding items (based on the judgments of panelists) can be "time consuming and costly" (p. 50).

DupER Augmentation avoids some of the limitations of the methods discussed above in that it does not rely on any previous knowledge of the item parameters or examinee abilities (though it does rely on the observed responses being plausible), and it does not require using simplified/modified IRT models.

Resampling Techniques

DupER Augmentation can be seen as an outgrowth of a family of statistical procedures known as resampling techniques. These procedures employ repeated sampling of a given data set to help obtain more accurate inferences. Two widely used resampling techniques are the jackknife and the bootstrap. DupER Augmentation's algorithm resembles these two approaches, however, it differs from these procedures in that the jackknife and the bootstrap are most commonly used for estimating more accurate standard errors of parameter estimates; DupER Augmentation's aim is to improve the parameter estimates themselves.

The jackknife.

It is often necessary to calculate empirical estimates of standard errors because "for most statistical estimators other than the mean there is no formula ... to provide estimated standard errors. In other words, it is hard to assess the accuracy of an estimate other than the mean" (Efron & Tibshirani, 1994, p. 12). At its heart the jackknife is a fairly simple procedure. In order to obtain the jackknifed standard error of an estimate, one performs the following procedure:

1. Remove the first observation from the data set and calculate the estimate of the parameter of interest.
2. Remove the next observation from the data set and, again, calculate the estimate of the parameter of interest.
3. Repeat Step 2 for the rest of the data set.
4. Calculate the standard deviation of the set of leave-one-out parameter estimates.

The standard deviation obtained in Step 4 is the jackknifed estimate of the standard error of the parameter of interest. This nonparametric procedure can be generalized to any estimator and does not rely on any normal theory assumptions (Efron, 1982).

The bootstrap.

Another resampling technique for obtaining empirical estimates of standard errors is the bootstrap. Unlike the related jackknife, the bootstrap does not discard information. Although several variations of bootstrapping exist, the basic procedure is as follows:

1. Sample *n* observations (with *n* equal to the original *n* of the data set) from the data set, with replacement, and calculate the estimate of the parameter of interest.
2. Repeat Step 1 many times.
3. Calculate the standard deviation of the resampled parameter estimates.

The standard deviation obtained in Step 3 is the bootstraped estimate of the standard error of the parameter of interest. This is a nonparametric procedure that can be generalized to any estimator and does not rely on any normal theory assumptions.

As mentioned earlier, although DupER Augmentation is similar to these procedures, the jackknife and the bootstrap are typically used to produce more accurate standard errors while leaving the parameter estimates unchanged. DupER Augmentation, on the other hand, has been developed to improve the accuracy of parameter estimates with small samples. It is expected that any improvement in the parameter estimates may affect the standard errors as well.

Missing Data Mechanisms

Because the DupER Augmentation algorithm calls for the deletion of observations from the data set, it is useful to discuss this missing data mechanism in terms of the existing literature. In the missing data literature, the reasons why data are missing typically are grouped into three categories based on the work of Rubin (1976). These categories are discussed at length elsewhere (e.g., Allison, 2001; Little & Rubin, 2002; Schafer & Graham, 2002), and therefore are summarized only briefly here. If the reasons for the missing values for a variable are unrelated to that variable and other variables in the analysis, the missing data are said to be missing completely at random (MCAR). If the reasons for the missing values for a variable are unrelated to that variable but are related to other variables in the analysis, the missing data are said to be missing at random (MAR). If the reasons for the missing values for a variable are related to that variable, the missing data are said to be missing not at random (MNAR). In DupER Augmentation observations are deleted based on values created using a random number generator. These values are used to delete observations without regard to any properties of the items or simulees. Therefore these missing data introduced by DupER Augmentation can be said to be MCAR, which is the type of missing data amenable to the greatest number of missing data handling techniques. These techniques include imputation, which is discussed in the following section.

Imputation

One of the steps in DupER Augmentation involves imputing data for the observations that have been deleted at random. Imputation is one of a family of tools used to deal with item nonresponse in data sets. Item nonresponse occurs when some, but not all, information

is available from a respondent (Schafer & Graham, 2002). Imputation involves using the available data in the data set to determine plausible values for the data that are missing. These plausible values replace the missing data, and these "complete" data are analyzed. Several methods exist for determining plausible values. These methods can be divided into two main categories: single and multiple imputation.

Single imputation.

The simplest single imputation method, mean imputation (a.k.a. unconditional mean imputation), uses the mean of the observed values for a variable to replace the missing values for that variable (Allison, 2001). Another single imputation method is known as regression imputation (a.k.a. conditional mean imputation). With this method, regression is used to predict examinees' missing values using their observed responses to other variables. The predicted values are used in lieu of the variable's missing values. Both of these methods result in an underestimation of variability and, in turn, an overestimation of the strengths of relationships between variables (Schafer & Graham, 2002). This problem can partially be overcome using stochastic regression imputation (a.k.a. imputation from a conditional distribution). Stochastic regression imputation is similar to regression imputation with the exception that instead of using only a predicted value to replace the missing datum, an error component (chosen at random from a normal distribution with a mean of 0 and a variance equal to the residual variance of the estimated regression function) is added to the predicted value (Little & Rubin, 2002). This helps to recapture some of the variability lost through the imputation process.

One variation of regression imputation is EM imputation. A detailed explanation of how the EM algorithm is used for imputation and missing data handling is given in Enders (2010). EM imputation utilizes the EM algorithm in much the same way it is used for marginal maximum likelihood estimation. Initial estimates of the means and covariances are obtained using a simple missing data handing technique (e.g., estimation based only on complete cases). In the first (E) step of the algorithm, regression equations are developed from the initial estimates to predict the missing data based on the observed data values. These regression equations are used to create a "complete" data set. In the second (M) step of the algorithm, improved estimates of the means and covariances are obtained using the new "complete" data set. The improved estimates of the means and covariances then are used in another E step to create new regression equations and "fill in" the data set, which in turn is used in the subsequent M step to obtain improved estimates of means and covariances. This process iterates until the change in parameter estimates becomes smaller than a convergence criterion. After obtaining the final item parameter estimates, the mean vector and covariance matrix may be used in subsequent analyses, or, in the case of EM imputation, may be used to impute a final raw data set based on the parameter estimates from the final EM cycle.

Enders (2010) notes that,

> The only difference between EM imputation and regression imputation is that the EM approach uses a maximum likelihood estimate of the mean vector and the covariance matrix to generate the regression equations, whereas standard regression imputation schemes tend to use listwise

> deletion estimates of [the mean vector and the covariance matrix] to build the regressions. (p. 113)

This leads EM imputation to have the same limitations as regression imputation (i.e., an underestimation of variability and an overestimation of the strengths of relationships between variables).

Multiple imputation.

Multiple imputation (Rubin, 1987) expands on single imputation by creating multiple imputed data sets. This provides the benefit of allowing the researcher to separate variability into two components: variability in the data and variability due to imputation. Multiple imputation has been described as one of the best methods available for dealing with missing data (Schaefer & Graham, 2002). One variation of multiple imputation uses Markov chain Monte Carlo [MCMC] estimation in the imputation process (see Allison, 2001; Enders, 2005, 2010; Schafer, 1997).

MCMC imputation is an iterative procedure with two steps, an I- (i.e., imputation) and P- (i.e. posterior) step, which ultimately results in a distribution of parameters. The process begins by obtaining initial values for the mean vector and the covariance matrix; often, these are obtained using the EM algorithm (Allison, 2001). In the I-step, these means and covariances are used as input for a stochastic regression procedure (as described above), which imputes estimates of the missing values. In keeping with the goal of using multiple imputation to create several different complete data sets, "the purpose of the P-step is to generate alternate estimates of the mean vector and the covariance matrix (the building blocks of the I-step regression equations)" (Enders, 2010, p. 190). In the P-step, the filled in

data set created in the I-step is used to estimate a posterior distributions for the parameter. New estimates of the values for the means and the covariance matrix are estimated by sampling from this estimated posterior distribution. These new parameter estimates are used as input to a new I-step in which revised imputed values are calculated. This process repeats over a series of "burn-in" iterations until convergence is achieved.

Convergence is conceptualized differently for MCMC imputation than for the EM algorithm. Rather than evaluating the change in item parameters, convergence in MCMC imputation is characterized by the stabilization of the posterior distributions. Two graphical convergence diagnostic tools are time-series plots and autocorrelation function plots. Time-series plots compare parameter estimates across iterations. Convergence is assumed when systematic trends are no longer visible. Three time-series plots are shown in Figure 2. The plotted parameter is the mean of item 7 (a=0.57, b=-0.66, c=0.22). The plot extends across 500 iterations and is based on one replication of DupER augmented data using 50 duplications per simulee and a 40% deletion rate. The three plots represent results for three different testing conditions (i.e., N=500/n=10, N=500/n=30, and N=500/n=60). For these three graphs, it is difficult to identify any trends and, subsequently, a specific iteration where convergence has been achieved. In this case, it is easier to identify convergence using autocorrelation function plots.

In autocorrelation function plots the correlation of parameters separated by a given number of iterations (or lags) is plotted. Convergence is presumed to have occurred when the number of lags is sufficient that there are no longer significant correlations between item parameters, in other words, the point where values between iterations become independent.

Three autocorrelation function plots are shown in Figure 3. The plotted values are the autocorrelations for item 3 (a=1.31, b=0.57, c=.05) with lag values ranging from 0 to 200. Again, the three plots represent results for three different testing conditions (i.e., N=500/n=10, N=500/n=30, and N=500/n=60) based on one replication of DupER augmented data using 50 duplications per simulee and a 40% deletion rate. It appears from the graphs that convergence is achieved after approximately three iterations for the N=500/n=10 testing condition, after approximately 25 iterations for the N=500/n=30 testing condition, and after approximately 75 iterations for the N=500/n=60 testing condition.

The number of iterations conducted before an imputed data set is extracted (a.k.a. “burn-in” iterations), should be sufficiently large so that convergence is achieved for all parameters. This random draw procedure is repeated several times (either sequentially or as parallel processes) resulting in several "complete" data sets. These data sets are analyzed separately and their results combined to determine the final parameter and standard error estimates.

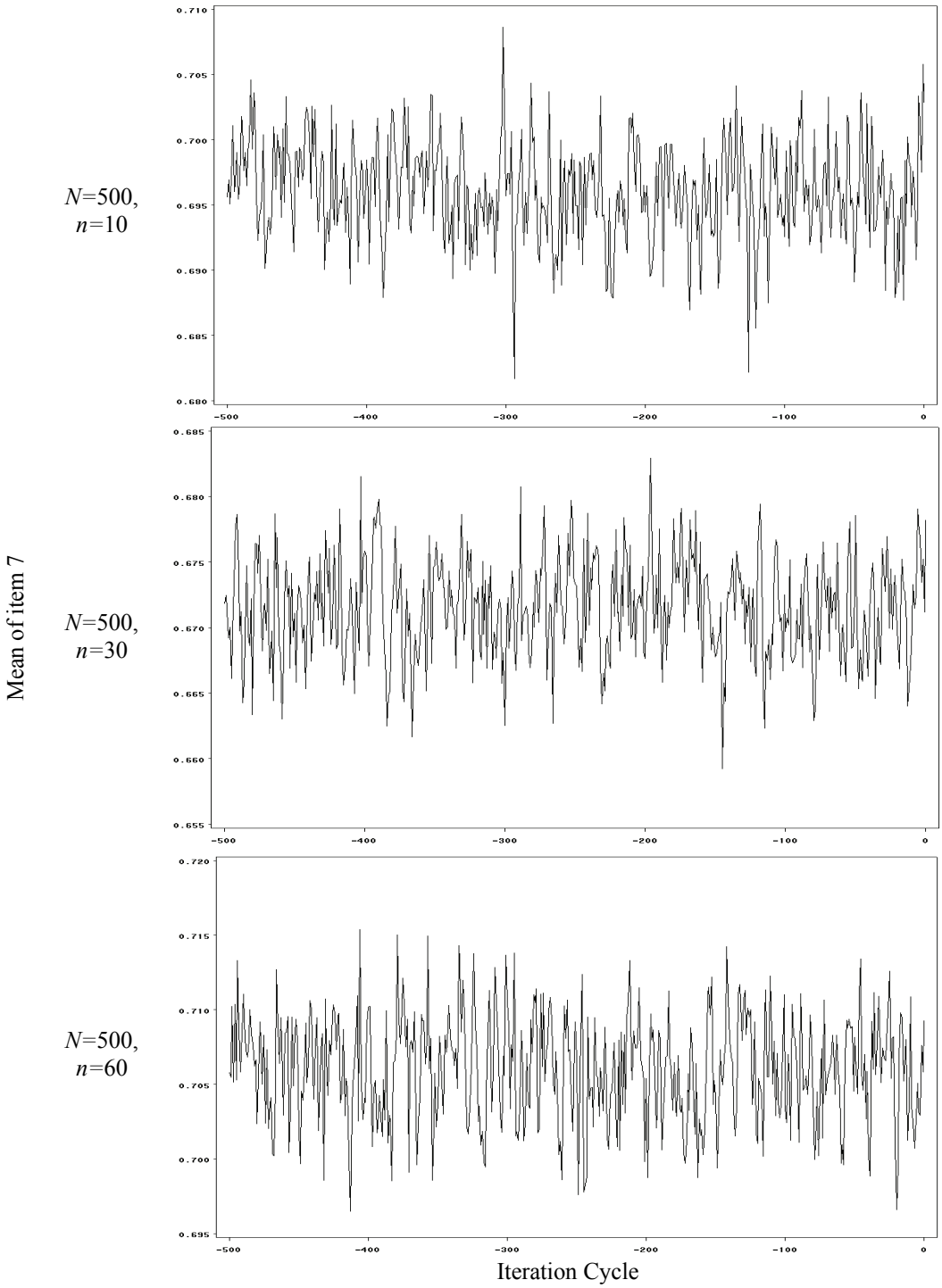

Figure 2. Time-series plots for item 7 using DupER with 50 duplications per subject and a 40% deletion rate for 3 testing conditions

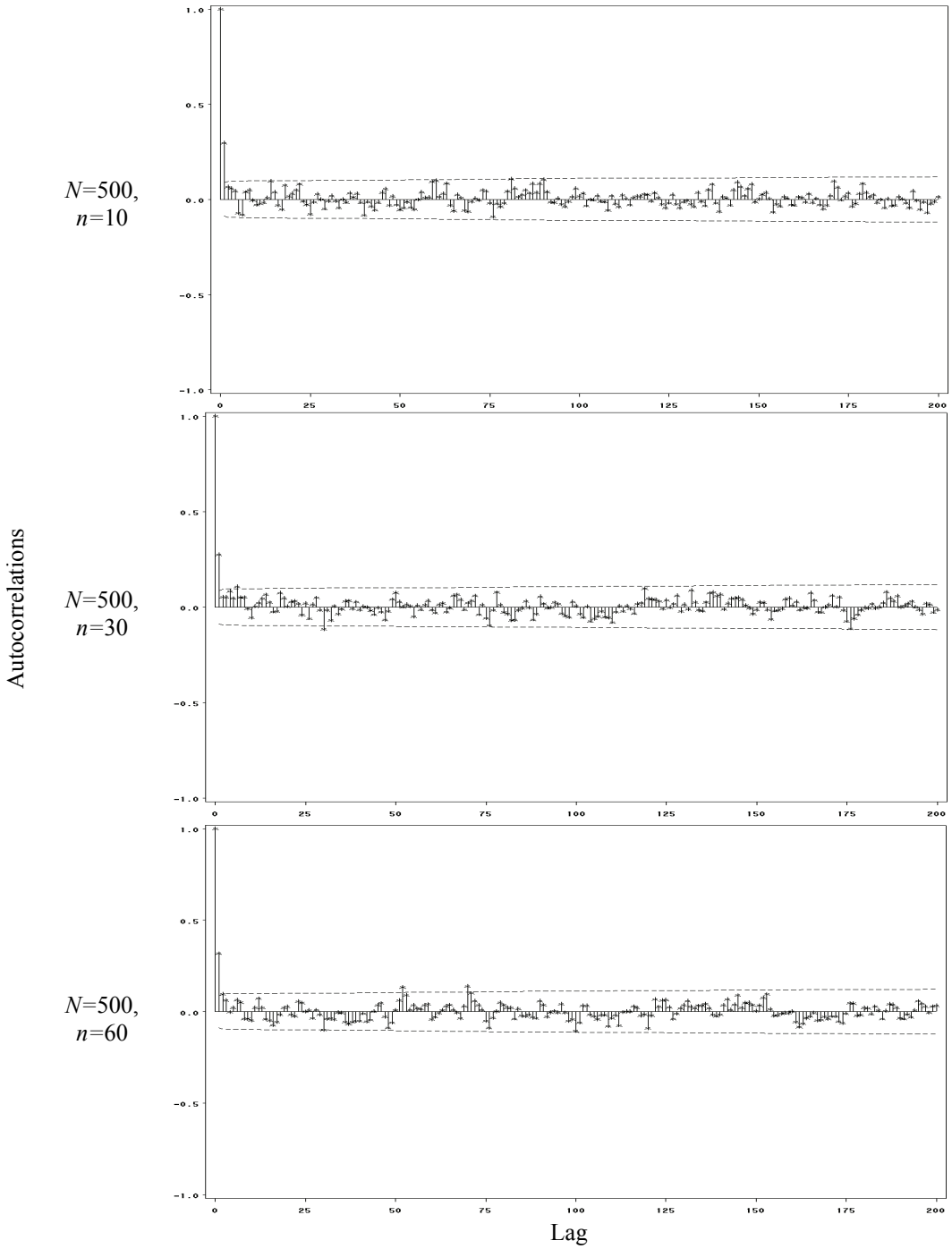

Figure 3. Autocorrelation plots with 95% confidence intervals for item 3 using DupER with 50 duplications per subject and a 40% deletion rate for 3 testing conditions.

Pilot Study Results

In a pilot study, Foley (2009) examined the efficacy of DupER Augmentation. Simulated data were randomly generated according to the 3PL model. Item parameters were selected randomly from a uniform distribution: for the *a* parameter U[0.5, 2.0] for the *b* parameter U[-2.5, 2.5], and U[0.0, 0.3] for the *c* parameter. Examinee abilities were generated from a standard normal distribution. Four different samples sizes (100, 250, 500, and 1000) and three different test lengths (10-, 30-, and 60-items) were examined. Nine DupER Augmentation variations based on three duplication rates (10, 25, and 50 duplications per simulee) and three deletion rates (20%, 40%, and 60%) were evaluated. Only a single imputation methodology was included (i.e., singly imputed data based on the MCMC imputation). Data (both raw and augmented) were analyzed in PARSCALE (Muraki & Bock, 2003) using the program's default settings (e.g., no prior distributions specified for item parameters); convergence of the item parameter estimation process was not evaluated. DupER results were compared to those from the raw data using five evaluative criteria: RMSEs, bias, and correlations of item parameter estimates; RMSEs of the ICC estimates; and, correlations of ability parameter estimates.

DupER Augmentation variations using Duplication rates of 50 consistently outperformed the DupER Augmentation variations using duplication rates of 10 or 25. In most cases DupER Augmented data resulted in lower median RMSEs for item parameter estimates than analyzing the raw data. Bias results for item estimates were mixed. DupER showed less bias when n=100 and n=250 across all test lengths for discrimination estimates, and but results were inconsistent for the other item parameters. DupER with either 20% or

40% deletion resulted in higher correlations between estimates and parameters for both item and ability estimates. Correlation results were more erratic when DupER was used with 60% deletion. For the RMSEs of ICCs, DupER variations tended to outperform the raw data for 60-item test length, but results were mixed for other conditions.

The current study builds on the pilot study in that it evaluates an additional imputation methodology, utilizes additional evaluative criteria, uses different software package, and incorporates prior distributions for the item parameters. The current study is also more rigorous in that all item parameter estimation procedures are checked for convergence, and nonconverging replications are removed from subsequent analyses.

Research Questions and Hypotheses

The questions that this study was designed to answer are as follows:

1. How does the precision and accuracy of item parameter estimates obtained using DupER Augmentation compare to those obtained analyzing raw data alone?
2. How does the precision and accuracy of person ability estimates obtained using DupER Augmentation compare to those obtained analyzing raw data alone?

It is hypothesized that estimates (both item and ability) obtained using DupER Augmentation will be more accurate than those obtained by analyzing raw data alone across most testing conditions (especially those with small sample sizes). It is also hypothesized that DupER Augmentation will perform better with high duplication rates (i.e., 50 duplications per examinee) and moderate deletion rates (i.e. 40% deletion). Finally, because

DupER Augmentation artificially increases the sample size rather than the test length, it is hypothesized that DupER Augmentation will not have as great of an effect on the accuracy of person ability estimates as on item parameter estimates.

In his 1982 book on resampling methods, Bradley Efron wrote that "*Good* simple ideas ... are our most precious intellectual commodity" (p. 1). Clearly, DupER Augmentation is a simple idea. The goal of this study is to determine if it is "*good*," at least under the conditions evaluated here.

Chapter 3. Methodology

Introduction

This chapter begins with a detailed description of the DupER Augmentation Procedure followed by a discussion of the study conditions and descriptions of the variables of interest. Next, the imputation methods used in the DupER Augmentation process are described. Following this is a description of how simulated data are generated and the calibration process by which the item parameters are estimated. The chapter concludes with a description of the data analysis process, which includes information about the evaluative criteria used to compare methodologies.

DupER Augmentation

To understand the logic behind DupER Augmentation, it is useful to think of the right/wrong responses to a given test. For example, consider a 60-item, multiple-choice test. For any given test of this length, there are more than one quintillion (2^{60}) possible right/wrong response patterns. Even if only one one-hundreth of one percent of these response patterns were actually plausible (given the characteristics of the test), that still leaves more than 100 billion plausible response patterns. DupER Augmentation attempts to take advantage of this by using existing test data to generate additional, plausible response vectors.

DupER Augmentation works as follows:

1. Given a data set of examinees, each with their own response vector, each response vector is duplicated several times. For example, a test might be given to 100 examinees and then scored (e.g., each response coded as either “1” for a correct response or “0” for an

incorrect response). Each response vector is duplicated, say 10 times, so the data set now consists of 1,000 "pseudo response vectors," each based on one of the original examinees.

2. Next, observations in the new data set (consisting of the duplicated response vectors), are deleted with a given probability. Continuing with the above example, assume the assessment in question consists of 30 items. Using a randomized process, observations from the 30,000 item/pseudo-examinee observations (i.e., 30 items for each of the 1,000 pseudo-examinees) are deleted at a rate of, say, 40%. In this case, each of the 1,000 pseudo-examinees would have approximately 12 observations missing from their response vectors. Note: The original complete response vectors are not used in estimating the item parameters, although, they are used for person parameter estimation.

3. The missing observations are then replaced using imputation (Note: while any imputation methodology may be used, two specific methodologies were chosen for comparison this study). Again, using the same example, imputation software uses the 1,000 pseudo-examinees data set (with 40% missing data) to generate plausible values to replace the missing data. This process results in a data set of 1,000 pseudo-examinees that has no missing data, is based on the original 100 examinees, and contains response vectors different from the original data (because some of the observations may change due to the deletion and imputation process), yet is still plausible. A simplified illustration of this process is shown in Figure 4.

Raw response patterns (1=right, 0=wrong)

	1	2	3	4	5	6	7	8	9	10
A	1	1	0	1	0	0	0	1	1	1
B	1	0	1	0	1	0	1	1	0	0
C	1	1	1	1	0	0	1	1	1	0

Duplicate response patterns

	1	2	3	4	5	6	7	8	9	10
A1	1	1	0	1	0	0	0	1	1	1
A2	1	1	0	1	0	0	0	1	1	1
A3	1	1	0	1	0	0	0	1	1	1
B1	1	0	1	0	1	0	1	1	0	0
B2	1	0	1	0	1	0	1	1	0	0
B3	1	0	1	0	1	0	1	1	0	0
C1	1	1	1	1	0	0	1	1	1	0
C2	1	1	1	1	0	0	1	1	1	0
C3	1	1	1	1	0	0	1	1	1	0

Randomly delete observations (at a given rate, say 40%)

	1	2	3	4	5	6	7	8	9	10
A1m	1		0	1		0	0			1
A2m	1	1	0	1		0		1		1
A3m	1	1		1	0	0		1	1	1
B1m		0			1		1	1	0	0
B2m	1		1		1		1		0	
B3m	1	0		0	1		1		0	0
C1m			1	1		0	1	1		0
C2m	1	1	1	1	0	0				
C3m		1		1	0		1	1	1	0

Impute missing values

	1	2	3	4	5	6	7	8	9	10
A1i	1	**0**	0	1	**0**	0	0	**1**	**1**	1
A2i	1	1	0	1	**1**	0	**1**	1	**0**	1
A3i	1	1	**1**	1	0	0	**0**	1	1	1
B1i	**1**	0	**1**	**0**	1	**0**	1	1	0	0
B2i	1	**1**	1	**0**	1	**0**	1	**1**	0	**1**
B3i	1	0	**0**	0	1	**0**	1	**1**	0	0
C1i	**0**	**1**	1	1	**1**	0	1	1	**1**	0
C2i	1	1	1	1	0	0	**1**	**1**	**1**	**0**
C3i	**0**	1	**1**	1	0	**0**	1	1	1	0

Figure 4. Illustration of DupER Augmentation for a 10-item test with three examinees (A, B, and C), using three duplications per examinee and a 40% deletion rate

Study Conditions and Variables

There are five study conditions: (a) the number of items (i.e., test length, *n*), (b) the number of simulees (i.e., sample size, *N*), (c) the number of duplications of each simulee (i.e., the number of pseudo-simulees), (d) the deletion rate for observations, and (e) the imputation method. The duplication rate/deletion rate/imputation method combinations comprise the different variations of DupER Augmentation. Each DupER variation, in addition to being compared to the other conditions, is compared to a control condition where the data are not augmented (i.e., the raw data). For each set of study conditions 1,000 replications are performed.

Test length.

The test lengths examined are 10-, 30-, and 60-items. These lengths were selected in order to provide a range that is representative of assessments being used in the field. For example, Hambleton and Cook (1983) reported that "a test with 10 items is about as short a test as is ever used in practice" (p. 41). Sixty items is generally believed to be adequate for most 3PL model applications (see earlier section on necessary sample size). Thirty items was chosen as an intermediate assessment length. The test lengths chosen here are all also similar to test lengths used in other IRT simulation studies (Harwell, Stone, Hsu, & Kirisci, 1996).

Sample size.

Four different samples sizes are used in the study: 100, 250, 500, and 1000. A sample size of 1,000 is usually considered the minimum for use with the 3PL model. Therefore, it was included in order to serve as a benchmark for the smaller sample sizes. Again, the

sample sizes chosen here are similar to those used in other IRT simulation studies (Harwell et al., 1996). The three test lengths and four sample sizes described above combined to form 12 "testing situations."

Duplications per simulee.

Three duplication rates are used in the study: 10, 25, and 50. Different rates are examined to determine if the rates have a differential impact and, if so, is there a point of diminishing returns. The maximum duplication rate of 50 duplications per simulee was chosen for pragmatic reasons: in the experimental condition of 60 items, 1,000 simulee, and 50 duplications, pilot work showed that each replication took between 2 and 3 hours. Therefore, completing all 1,000 replications for this one cell takes more than 3 months (on an Intel Core2 Duo-class computer). The primary factor influencing the execution time is the imputation procedure in SAS version 9.1 (SAS Institute Inc., 2004).

Deletion rate.

Three deletion rates are selected for study: 20%, 40%, and 60%. As a new procedure there is no guidance in the literature as to what are appropriate deletion rates. Therefore, rates are chosen to represent a range of values that might represent low, moderate, and high deletion rates. It is hypothesized that if deletion rates are too low, insufficient variability would be added to the data set and DupER Augmentation would not be effective in improving parameter estimation. If deletion rates are too high, it is hypothesized that the imputed response patterns would lose their plausibility, resulting in unrealistic response vectors and poor parameter estimates.

Imputation methods.

For this study, two different imputation methods are used to fill in the missing values created in the DupER Augmentation process. Both methods are implemented using the MI procedure in SAS version 9.1 (SAS Institute Inc., 2004).

The first imputation methodology is a variation of MCMC imputation. MCMC is used because of its stochastic properties. That is, because imputed values are selected randomly from a distribution, two response vectors with the same missing data pattern may result in different response patterns after imputation. Therefore, MCMC imputation was chosen because it may result in more plausible response vectors being added to the data set than might have been added with more deterministic imputation methods.

This procedure varies from the typical MCMC imputation in that rather than multiple imputations, only a single imputed data set is created for each replication. The reasons for this are twofold. First, multiple imputation is typically used to obtain more accurate parameter and standard error estimates from a incomplete data set (i.e., a data set with some data that are both missing and unknown); this situation differs from the simulated situation examined in the current study. That is, a complete data set is available, but it is possibly of insufficient size to produce accurate 3PL parameter estimates. Imputation serves as a means to generate additional plausible response vectors to augment the data set, and it is believed that single imputation is sufficient to accomplish this goal.

The second imputation methodology is EM imputation. Like regression imputation, this methodology uses observed data to predict values of missing observations. Although it is known that this methodology underestimates variability, it is included in order to

determine if results obtained from DupER Augmentation are dependent on the imputation method. Additionally, it is desired to compare the results of DupER using MCMC imputation to results obtained from a less computationally complex imputation procedure. Appendix A contains SAS syntax templates for the imputation procedures.

The three duplication rates, three deletion rates, and two imputation methods combined to produce 18 DupER Augmentation variations. These variants are compared to each other and to results obtained from analyzing the raw data across the 12 testing conditions described above. Therefore, there are a total of 228 ([18*12]+12) combinations of experimental conditions. The complete research design is summarized in Figure 5.

Definition of Terms

In order to facilitate description and discussion of results, the following notation and terminology is used:

- RAW - the original data set of item responses
- DupER/XX[YY/ZZ] - the DupER augmented data set of item responses where XX is replaced by the imputation method (EM or MCMC), YY is replaced by the number of duplications (10, 25, or 50), and ZZ is replaced by the deletion rate (20%, 40%, or 60%). For example, DupER/MCMC[25/40] represents the variation of DupER Augmentation using MCMC imputation, 25 duplications per subject, and a 40% deletion rate.

DupER Variation (EM Imputation)
(Duplication rate and deletion rate)

Testing Condition (Sample size and test length)	RAW	10 duplications 20%	10 duplications 40%	10 duplications 60%	25 duplications 20%	25 duplications 40%	25 duplications 60%	50 duplications 20%	50 duplications 40%	50 duplications 60%
$N=100$ $n=10$										
$N=100$ $n=30$										
$N=100$ $n=60$										
$N=250$ $n=10$										
$N=250$ $n=30$										
$N=250$ $n=60$										
$N=500$ $n=10$										
$N=500$ $n=30$										
$N=500$ $n=60$										
$N=1000$ $n=10$										
$N=1000$ $n=30$										
$N=1000$ $n=60$										

DupER Variation (MCMC Imputation)
(Duplication rate and deletion rate)

Testing Condition (Sample size and test length)	RAW	10 duplications 20%	10 duplications 40%	10 duplications 60%	25 duplications 20%	25 duplications 40%	25 duplications 60%	50 duplications 20%	50 duplications 40%	50 duplications 60%
$N=100$ $n=10$										
$N=100$ $n=30$										
$N=100$ $n=60$										
$N=250$ $n=10$										
$N=250$ $n=30$										
$N=250$ $n=60$										
$N=500$ $n=10$										
$N=500$ $n=30$										
$N=500$ $n=60$										
$N=1000$ $n=10$										
$N=1000$ $n=30$										
$N=1000$ $n=60$										

Figure 5 . Summary of research design.

Data Generation and Calibration

Each simulated item's parameters are randomly selected from a uniform distribution: for the *a* parameter U[0.5, 2.0], for the *b* parameter U[-2.5, 2.5], and U[0.0, 0.3] for the *c* parameter. A uniform distribution is chosen to ensure a greater likelihood of obtaining a full range of values as opposed to, say, a normal distribution where one expects values to be concentrated around the mean. These distributions are similar those used in other IRT simulation studies (see Harwell et al., 1996).

The abilities of the simulated examinees are drawn at random from a normal distribution with a mean of 0 and a standard deviation of 1. A standard normal ability distribution is chosen because "in the absence of any strong a priori beliefs, a normal distribution of ability is a good approximation to the usually encountered distributions of ability measures" (Hulin et al., 1982, p. 254). Additionally, one possible use for DupER is for the calibration of a subsample of pretest data for large testing programs. For programs such as large K-12 accountability testing, which may employ pretesting, a normal distribution of student abilities is probably a reasonable assumption. Example SAS code for generating item parameters, ability parameters, and response vectors is included in Appendix B. The IRT parameters and corresponding CTT approximations for all items at each test length are summarized in Appendix C.

For the calibration of the data (with and without DupER Augmentation) BILOG-MG 3 (Zimowski, Muraki, Mislevy, & Bock, 2003) is used. For item parameter estimation, the number of EM and Newton iterations are set to 200 and 25, respectively. This change to the default BILOG parameters was made in order to increase the number of useful replications

by allowing slower-converging replications a better chance to arrive at the convergence criteria. The Ridge option is also used to help facilitate convergence. Priors are specified for all three item parameters.

Ability estimates are obtained using EAP estimation. For the scoring step, the number of quadrature points is increased 80 based on De Ayala's (2009, p. 79) recommendation. (Note: This recommendation is primarily geared towards estimating accurate standard errors; although standard errors are not examined in this study, it was felt this option might be beneficial for future secondary analyses of these data.) Templates for the BILOG syntax are included in Appendix D. In order to compare estimated item parameters and ICCs to the true parameters, the estimates are placed on the parameter scale using the characteristic curve equating software program EQUATE (Baker, 1993) to account for scale indeterminacy.

Data Analysis

Diagnostics.

In order to help ensure stable estimates, convergence of item parameter estimates are examined for each replication. Results from non-converging replications are excluded from subsequent analyses. The percent of converging replications are reported by testing condition and DupER variation.

Data for this study are generated to fit the 3PL model. Therefore, response patterns should exhibit relatively good fit to that model. However, there was some concern that the DupER Augmentation process might result in good-fitting response patterns erroneously appearing aberrant. Although several indices exist that measure person-fit (De Ayala, 2009),

the only measure available in BILOG is the marginal probabilities of the response vectors. For the n=30 and n=60 testing conditions, the values of these probabilities are sufficiently small that is unlikely that any value other than zero (given the number of decimal places displayed in the BILOG output) will be displayed. Therefore, differences in person fit between the DupER augmented and the raw data are evaluated using the marginal probabilities of the response vectors for the n=10 conditions. Results are compared using mean differences and effect sizes (standardized based on the standard deviation of the marginal probabilities from the raw data).

Evaluative criteria.

The seven criteria used to evaluate the efficacy of DupER Augmentation are described in Table 1.

Table 1

Evaluative criteria

Criteria #	Description
1	Root mean squared error (RMSE) of the item parameter estimates
2	Bias of the item parameter estimates
3	Pearson correlation between the estimated item parameters and their corresponding parameters
4	RMSE of the estimated ICCs
5	RMSE of the ability parameter estimates
6	Bias of the item parameter estimates
7	Pearson correlation between the estimated person ability parameters and their corresponding parameters

The RMSE of the item parameter estimates for item *i* is defined as

$$RMSE_{\pi;\, i} = \sqrt{\frac{\sum_{k=1}^{r}(\widehat{\pi_{ik}} - \pi_{ik})^2}{r}} \qquad (18)$$

where π_{ik} is the item parameter (either *a*, *b*, or *c*) of item *i* for replication *k;* $\widehat{\pi_{ik}}$ is the estimate of that parameter. The number of replications is *r*. $RMSE_{\pi;\, i}$ is used as a measure of the precision of the item parameter estimates.

Although the RMSEs of the item parameter estimates are useful for estimating precision, they do not tell us whether estimates are consistently too high or too low. To determine this, the bias of the parameter estimate is calculated for each item. The bias of the item parameter estimates for item *i* is defined as

$$Bias_{\pi;\, i} = \frac{\sum_{k=1}^{r}(\widehat{\pi_{ik}} - \pi_{ik})}{r} \qquad . \qquad (19)$$

As a measure of the strength of the linear relationship between the estimates and parameters, the Pearson correlation between the estimated item parameters and the item parameters for replication *k*, $r_{\hat{\pi},\pi;\, k}$, is calculated across all items for each replication and then the median of the correlations is taken across replications.

Because several different sets of item parameters can produce very similar ICCs (Yen & Fitzpatrick, 2006), the accuracy of individual item parameters may not tell the whole story. Therefore, the RMSEs of the estimated ICCs for each item are calculated. This calculation uses 201 different equally spaced θ values from z = -3 to z = 3. The RMSE of the estimated ICC for item *i* is defined as

$$RMSE_{icc;\, i} = \sqrt{\frac{\sum_{z=1}^{201}\left[\widehat{P(\theta_z)} - P(\theta_z)\right]^2}{201}} \qquad (20)$$

where $P(\theta_z)$ is the value of the true ICC when $\theta_z = z$, and $\widehat{P(\theta_z)}$ is the estimate of that value based on the item parameter estimates. This value can be thought of as the average absolute distance between the estimated and true ICCs.

The RMSE of the ability parameter estimates is defined as

$$RMSE_\theta = \sqrt{\frac{\sum_{i=1}^{N^*}(\widehat{\theta_i} - \theta_i)^2}{N^*}} \quad (21)$$

where θ_j is the ability parameter of original, non-duplicated simulee *j* (ability estimates are not calculated for the duplicated simulee, only the original simulee)*;* $\widehat{\theta_j}$ is the estimate of that parameter. The total number of simulees across all (converging) replications is *N**. $RMSE_\theta$ is used as a measure of the precision of the ability parameter estimates.

The bias of the ability parameter estimates is defined as

$$Bias_\theta = \frac{\sum_{j=1}^{N^*} r(\widehat{\theta_J} - \theta_j)}{N^*} \quad . \quad (22)$$

Finally, the Pearson correlation between the estimated ability parameters and the ability parameters, $r_{\hat{\theta},\theta}$, is calculated across all (non-duplicated) simulees and replications.

Summary.

In order to produce an overall summary statistic for each experimental condition, for Criteria 1, 2, 3, and 4 the RMSE/bias is estimated for each item and then the median is calculated across test items. For example, for a test with 30 items, there are 30 different RMSE estimates for the *a* parameter (1 per item). The median of these 30 estimates provides a one-number summary of the of these RMSE estimates. The median is used, rather than the mean, to minimize the effect of outliers, providing a better summary of the "typical" value of

the evaluative criteria across items. To summarize the variability of the RMSE/bias across items, the interquartile range (IRQ) is reported as well.

It is felt that the number of replications is sufficiently large that the sampling error of the summary statistics is negligible. Therefore, the results can be evaluated using descriptive statistics rather than significance tests. Even so, there are 60 RMSE comparisons (i.e., RMSEs for discrimination, difficulty, guessing, ICC, and ability estimates), 48 bias comparisons (i.e., bias for discrimination, difficulty, guessing, and ability estimates), and 48 correlation comparisons (i.e., correlations for discrimination, difficulty, guessing, and ability estimates) between results obtained from data subjected to DupER Augmentation and the raw data for each DupER variation. This being the case, even if data subjected to DupER Augmentation and raw data performed equally well, one would expect that the results from DupER Augmentation would be better than those from raw data about 50% of the time (and, consequently, worse about 50% of the time). To evaluate if data subjected to DupER Augmentation outperforms the raw data more often than would be expected by chance, binomial tests are conducted to determine if the percentage of time that DupER Augmentation outperformed the raw data for the comparisons is significantly greater than 50%. These tests are conducted for comparisons between DupER and RAW and between DupER/EM and DupER/MCMC .

Chapter 4. Results

Introduction

This section begins with a presentation of diagnostic information. Next, results are presented for the discrimination, difficulty, guessing, ICC, and ability estimates. Finally, results are summarized across the evaluative criteria.

Diagnostics

Convergence.

A summary of BILOG item estimation convergence results is shown in Table 2. Convergence rates based on analyses of RAW data sets are very high across all testing conditions, ranging from 96.1% to 100%.

Analyses of data sets produced using variations of DupER/EM tended to converge less frequently than did analyses based on RAW. In some cases, the DupER/EM convergence rate is substantially worse than that of RAW. For example, in the N=100/n=60 testing condition, analyses based on RAW converges 96.1% of the time whereas analyses based on DupER/EM[50/20] converges only in 36.6% of replications.

The convergence rates for DupER/MCMC are very close to, although usually lower than, those obtained in analyses using RAW. However, there are some DupER/MCMC variations and test condition combinations that result in low convergence rates. The worst of these is DupER/MCMC[50/20], which has a convergence rate of 62.2% in the N=100/n=60 testing condition.

Table 2

Convergence rates for item parameter estimation

Testing Condition (Sample size and test length)		DupER Variation (EM Imputation) (Duplication rate and deletion rate)									
			10 duplications			25 duplications			50 duplications		
		RAW	20%	40%	60%	20%	40%	60%	20%	40%	60%
N= 100	*n*= 10	98.9	87.7	78.3	65.4	76.7	67.3	56.7	62.3	54.8	52.4
	n= 30	98.6	94.9	96.8	91.3	78.3	86.6	83.1	65.5	78.2	81.1
	n=60	96.1	82.4	93.4	96.1	52.8	72.2	86.4	36.6	65.7	81.2
N= 250	*n*=10	99.7	99.9	94.2	79.6	98.1	89.7	76.1	94.0	84.2	67.9
	n=30	99.9	99.6	99.7	97.8	98.4	99.2	98.6	95.9	99.1	98.5
	n=60	99.8	99.3	99.5	99.9	97.7	99.5	99.5	94.9	98.7	99.8
N= 500	*n*=10	99.9	100.0	96.8	86.0	99.1	93.3	82.3	96.1	86.1	74.6
	n=30	100.0	100.0	99.9	99.8	99.9	99.9	99.8	99.4	99.8	99.9
	n=60	100.0	100.0	99.9	99.9	99.7	99.9	100.0	99.7	100.0	99.9
N=1000	*n*=10	100.0	100.0	98.0	89.2	98.3	92.2	83.0	95.8	81.8	69.8
	n=30	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0
	n=60	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0

Testing Condition (Sample size and test length)		DupER Variation (MCMC Imputation) (Duplication rate and deletion rate)									
			10 duplications			25 duplications			50 duplications		
		RAW	20%	40%	60%	20%	40%	60%	20%	40%	60%
N= 100	*n*= 10	98.9	97.2	97.5	97.0	94.5	97.3	98.5	88.3	94.4	97.0
	n= 30	98.6	98.7	99.2	98.6	89.1	94.4	95.4	84.4	92.7	95.4
	n=60	96.1	92.1	96.8	99.1	72.1	88.0	95.4	62.2	83.6	94.8
N= 250	*n*=10	99.7	100.0	100.0	100.0	99.6	99.8	99.8	98.9	98.8	99.1
	n=30	99.9	99.5	99.7	99.3	99.4	99.7	99.4	98.7	99.3	99.5
	n=60	99.8	99.8	99.9	99.5	99.0	99.6	99.3	98.3	99.3	98.9
N= 500	*n*=10	99.9	99.8	99.7	99.8	99.9	99.8	99.6	99.2	99.9	99.8
	n=30	100.0	99.9	99.9	100.0	99.7	99.4	99.3	99.6	99.7	99.8
	n=60	100.0	100.0	99.8	99.7	99.4	99.5	99.4	99.5	99.9	99.9
N=1000	*n*=10	100.0	99.9	99.9	100.0	99.8	99.7	99.8	99.6	99.5	99.9
	n=30	100.0	99.8	100.0	99.8	99.8	99.8	99.6	99.7	99.5	99.6
	n=60	100.0	99.9	99.7	99.7	99.9	99.8	99.8	99.9	100.0	99.8

Person fit.

Mean differences and effect sizes (standardized based on the standard deviation of the marginal probabilities from the raw data) for the marginal probabilities of the response vectors for the n=10 conditions are summarized in Table 3. Mean differences range from -0.0008 (DupER/MCMC[50/60], N=1000) to 0.0152 (DupER/EM[10/60], N=100). Effect sizes range from -0.07 (DupER/MCMC[50/60], N=1000) to 1.22 (DupER/EM[10/60], N=100).

All DupER/EM variations have higher marginal probabilities than the corresponding RAW data. This unexpected result may indicate that DupER/EM reduces the sensitivity of BILOG to identify aberrant response vectors. The effect size of these differences could be described as small, medium, and large for the DupER/EM conditions with 20%, 40%, and 60% deletion rates, respectively (Cohen, 1988).

Marginal probabilities calculated using DupER/MCMC-based estimates are very similar to RAW across all DupER/MCMC variations and all sample sizes. The effect sizes of the mean differences are all very small (all less than |0.1|). This may be indicative of BILOG having a similar sensitivity for detecting aberrant response vectors for DupER/MCMC and RAW.

Table 3

Differences in response vector marginal probabilities, by DupER variation and testing condition

Mean Difference* [(DupER-RAW) X 1,000]

DupER Variation (EM Imputation)
(Duplication rate and deletion rate)

Testing Condition (Sample size and test length)	10 duplications 20%	10 duplications 40%	10 duplications 60%	25 duplications 20%	25 duplications 40%	25 duplications 60%	50 duplications 20%	50 duplications 40%	50 duplications 60%
N=100 n=10	4.80	9.90	15.20	5.20	10.10	14.70	5.10	9.90	14.60
N=250 n=10	3.60	7.70	12.20	3.60	7.70	12.60	3.60	7.90	12.30
N=500 n=10	3.30	7.10	11.50	3.40	7.20	11.50	3.40	7.20	11.60
N=1000 n=10	3.30	7.10	11.30	3.30	7.10	11.40	3.30	7.10	11.50

DupER Variation (MCMC Imputation)
(Duplication rate and deletion rate)

Testing Condition (Sample size and test length)	10 duplications 20%	10 duplications 40%	10 duplications 60%	25 duplications 20%	25 duplications 40%	25 duplications 60%	50 duplications 20%	50 duplications 40%	50 duplications 60%
N=100 n=10	0.80	0.40	0.02	0.90	0.50	-0.30	0.90	0.40	-0.40
N=250 n=10	0.10	-0.15	-0.47	0.20	-0.15	-0.47	0.20	-0.14	-0.47
N=500 n=10	-0.10	-0.35	-0.67	-0.09	-0.35	-0.66	-0.09	-0.35	-0.67
N=1000 n=10	-0.19	-0.45	-0.75	-0.20	-0.45	-0.76	-0.20	-0.45	-0.77

Effect Size** (standarized mean difference)

DupER Variation (EM Imputation)
(Duplication rate and deletion rate)

Testing Condition (Sample size and test length)	10 duplications 20%	10 duplications 40%	10 duplications 60%	25 duplications 20%	25 duplications 40%	25 duplications 60%	50 duplications 20%	50 duplications 40%	50 duplications 60%
N=100 n=10	0.38	0.79	1.22	0.42	0.81	1.18	0.41	0.79	1.17
N=250 n=10	0.32	0.68	1.08	0.32	0.68	1.11	0.32	0.70	1.09
N=500 n=10	0.30	0.64	1.03	0.30	0.64	1.03	0.30	0.64	1.04
N=1000 n=10	0.30	0.64	1.02	0.30	0.64	1.03	0.30	0.64	1.04

DupER Variation (MCMC Imputation)
(Duplication rate and deletion rate)

Testing Condition (Sample size and test length)	10 duplications 20%	10 duplications 40%	10 duplications 60%	25 duplications 20%	25 duplications 40%	25 duplications 60%	50 duplications 20%	50 duplications 40%	50 duplications 60%
N=100 n=10	0.06	0.03	0.00	0.07	0.04	-0.02	0.07	0.03	-0.03
N=250 n=10	0.01	-0.01	-0.04	0.02	-0.01	-0.04	0.02	-0.01	-0.04
N=500 n=10	-0.01	-0.03	-0.06	-0.01	-0.03	-0.06	-0.01	-0.03	-0.06
N=1000 n=10	-0.02	-0.04	-0.07	-0.02	-0.04	-0.07	-0.02	-0.04	-0.07

*All differences are statistically significant at the $\alpha < .001$ level. Samples sizes range from 52,000 to 1,000,000.

**Standardized based on the standard deviation of the RAW marginal probabilities for the given testing condition

Discrimination Estimates

RMSE.

Detailed RMSE results for the item discrimination estimates are shown in Table 4. The median RMSEs for the RAW conditions tend to decrease as text length and sample size increased. These median RMSEs across items range from 0.253 (N=1000, n=30) to 0.447 (N=100, n=30).

All DupER/EM variations have higher median RMSEs than do the corresponding RAW data. DupER/EM median RMSEs range from 33% to 552% greater than (median=163% greater than) than the corresponding RAW median RMSEs. The best performing DupER/EM variations across testing conditions are those with a deletion rate of 20%. The RMSE IQRs are larger for DupER/EM than for RAW across all DupER/EM variations testing conditions and are much larger when n=10.

Median RMSEs for DupER/MCMC are higher than those for RAW across all three test lengths at sample sizes N=100 and N=250. For the larger two sample sizes, results are mixed, with DupER/MCMC sometimes resulting in lower median RMSEs than the RAW data. DupER/MCMC median RMSEs range from 18% less than to 393% greater than (median=39% greater than) the corresponding RAW RMSEs. The best-performing DupER variations are DupER/MCMC[25/60] and DupER/MCMC[50/60]; each of these variations outperforms the RAW data in several testing conditions. The RMSE IQRs are larger for DupER/MCMC than for RAW across all DupER/MCMC variations when N=100 and N=250, but do not show a consistent pattern when N=500 and N=1000.

Median RMSEs for DupER/MCMC are lower than those for DupER/EM for every DupER variation and every testing condition. DupER/MCMC median RMSEs range from 12% to 77% less than (median=40% less than) those from the corresponding DupER/EM variation.

Table 4

RMSE of item discrimination estimates [median (interquartile range)]

Testing Condition (Sample size and test length)		RAW	DupER Variation (EM Imputation) (Duplication rate and deletion rate)								
			10 duplications			25 duplications			50 duplications		
			20%	40%	60%	20%	40%	60%	20%	40%	60%
N= 100	*n*=10	0.402 (0.123)	1.328 (0.847)	1.595 (1.704)	1.935 (1.642)	1.917 (1.200)	2.054 (1.953)	2.294 (1.904)	2.285 (1.141)	2.519 (2.207)	2.620 (1.639)
	n=30	0.447 (0.096)	1.253 (0.334)	1.237 (0.397)	1.442 (0.714)	1.706 (0.486)	1.465 (0.467)	1.555 (0.773)	1.857 (0.471)	1.622 (0.547)	1.652 (0.754)
	n=60	0.412 (0.122)	1.189 (0.389)	1.074 (0.253)	1.057 (0.237)	1.496 (0.743)	1.355 (0.530)	1.215 (0.308)	1.572 (0.796)	1.404 (0.643)	1.273 (0.450)
N= 250	*n*=10	0.366 (0.096)	1.020 (1.490)	1.228 (2.635)	1.697 (2.627)	1.384 (1.891)	1.580 (2.606)	1.959 (2.458)	1.538 (1.969)	1.766 (2.526)	2.012 (2.254)
	n=30	0.382 (0.101)	0.758 (0.339)	0.786 (0.494)	0.971 (0.879)	0.845 (0.441)	0.872 (0.468)	1.022 (0.920)	0.843 (0.495)	0.945 (0.565)	1.008 (0.960)
	n=60	0.346 (0.111)	0.648 (0.292)	0.683 (0.295)	0.733 (0.285)	0.687 (0.378)	0.726 (0.369)	0.735 (0.348)	0.695 (0.442)	0.711 (0.387)	0.753 (0.372)
N= 500	*n*=10	0.331 (0.092)	0.758 (1.516)	1.013 (2.871)	1.505 (3.130)	0.954 (1.797)	1.306 (2.823)	1.548 (2.853)	0.998 (1.805)	1.225 (2.791)	1.512 (2.457)
	n=30	0.323 (0.135)	0.527 (0.381)	0.555 (0.748)	0.659 (1.178)	0.532 (0.463)	0.622 (0.774)	0.778 (1.182)	0.538 (0.487)	0.650 (0.785)	0.807 (1.214)
	n=60	0.291 (0.109)	0.421 (0.282)	0.483 (0.415)	0.551 (0.517)	0.436 (0.287)	0.493 (0.452)	0.612 (0.611)	0.440 (0.296)	0.498 (0.474)	0.635 (0.637)
N=1000	*n*=10	0.290 (0.128)	0.561 (1.353)	0.840 (2.756)	1.329 (3.255)	0.627 (1.442)	0.888 (2.747)	1.183 (3.044)	0.687 (1.485)	0.934 (2.620)	0.934 (3.046)
	n=30	0.253 (0.102)	0.430 (0.383)	0.442 (0.952)	0.711 (1.470)	0.430 (0.375)	0.488 (0.969)	0.778 (1.765)	0.427 (0.367)	0.490 (0.984)	0.788 (1.895)
	n=60	0.227 (0.089)	0.302 (0.294)	0.394 (0.570)	0.553 (0.807)	0.305 (0.295)	0.396 (0.584)	0.597 (0.902)	0.309 (0.293)	0.397 (0.588)	0.612 (0.950)

Testing Condition (Sample size and test length)		RAW	DupER Variation (MCMC Imputation) (Duplication rate and deletion rate)								
			10 duplications			25 duplications			50 duplications		
			20%	40%	60%	20%	40%	60%	20%	40%	60%
N= 100	*n*=10	0.402 (0.123)	1.035 (0.420)	1.016 (0.428)	1.029 (0.595)	1.534 (0.548)	1.431 (0.626)	1.192 (0.739)	1.981 (0.732)	1.734 (0.828)	1.486 (0.762)
	n=30	0.447 (0.096)	1.000 (0.270)	0.813 (0.330)	0.820 (0.543)	1.247 (0.502)	0.947 (0.240)	0.823 (0.333)	1.375 (0.597)	0.971 (0.396)	0.800 (0.261)
	n=60	0.412 (0.122)	1.042 (0.429)	0.842 (0.396)	0.767 (0.350)	1.304 (0.743)	0.981 (0.607)	0.816 (0.414)	1.357 (0.902)	1.001 (0.739)	0.812 (0.429)
N= 250	*n*=10	0.366 (0.096)	0.738 (0.292)	0.744 (0.395)	0.749 (0.386)	1.022 (0.481)	0.863 (0.419)	0.751 (0.265)	1.277 (0.526)	0.986 (0.409)	0.716 (0.315)
	n=30	0.382 (0.101)	0.593 (0.200)	0.502 (0.177)	0.529 (0.131)	0.593 (0.307)	0.486 (0.174)	0.461 (0.207)	0.584 (0.328)	0.490 (0.193)	0.414 (0.178)
	n=60	0.346 (0.111)	0.523 (0.307)	0.489 (0.257)	0.495 (0.234)	0.535 (0.399)	0.462 (0.289)	0.443 (0.221)	0.539 (0.404)	0.453 (0.299)	0.406 (0.216)
N= 500	*n*=10	0.331 (0.092)	0.578 (0.245)	0.558 (0.214)	0.492 (0.223)	0.703 (0.283)	0.593 (0.258)	0.406 (0.156)	0.751 (0.359)	0.585 (0.263)	0.372 (0.155)
	n=30	0.323 (0.135)	0.370 (0.119)	0.349 (0.096)	0.372 (0.125)	0.364 (0.137)	0.331 (0.091)	**0.303** (0.121)	0.368 (0.150)	**0.321** (0.095)	**0.287** (0.108)
	n=60	0.291 (0.109)	0.337 (0.169)	0.326 (0.157)	0.342 (0.147)	0.334 (0.184)	0.307 (0.160)	0.300 (0.128)	0.332 (0.192)	0.293 (0.151)	**0.280** (0.116)
N=1000	*n*=10	0.290 (0.128)	0.413 (0.159)	0.374 (0.155)	0.304 (0.104)	0.407 (0.156)	0.352 (0.151)	**0.274** (0.097)	0.405 (0.149)	0.337 (0.134)	**0.282** (0.107)
	n=30	0.253 (0.102)	**0.248** (0.064)	**0.244** (0.083)	0.260 (0.110)	**0.243** (0.070)	**0.235** (0.074)	**0.219** (0.119)	**0.240** (0.075)	**0.227** (0.080)	**0.208** (0.139)
	n=60	0.227 (0.089)	0.230 (0.101)	0.229 (0.100)	0.244 (0.104)	**0.224** (0.103)	**0.217** (0.092)	**0.217** (0.095)	**0.223** (0.108)	**0.221** (0.090)	**0.213** (0.109)

Note: Values in bold indicate that the value based on DupER Augmentation is lower than the corresponding RAW value

Bias.

Detailed bias results for the item discrimination estimates are shown in Table 5. Median bias for the RAW conditions tends to decrease as sample size increases, but does not consistently vary with test length. These median RAW discrimination estimates are uniformly overestimated across testing conditions with a range across items from 0.062 (N=1000, n=30) to 0.236 (N=100, n=30).

Nearly all DupER/EM variations are more biased than the corresponding RAW data. Exceptions include DupER/EM[25/20] and DupER/EM[50/20], which are less biased in some cases when N=1000. Overall, median bias based on DupER/EM data ranges from 39% less than to 920% greater than (median=175% greater than) the corresponding RAW median bias (in absolute terms). The best performing DupER/EM variation across testing conditions is DupER/EM[10/20]. The bias IQRs are larger for DupER/EM than for RAW across most DupER/EM variations and testing conditions and are much larger when n=10.

DupER/MCMC-based estimates are more biased than RAW across all three test lengths at sample size N=100. For the larger three sample sizes, results are mixed. DupER/MCMC median bias ranges from 97% less than to 554% greater than (median=29% greater than) the corresponding RAW median bias. The best-performing DupER/MCMC variations are those using 25 or 50 duplications combined with 40% or 60% deletion rates; each of these variations outperform the RAW data in several testing conditions when sample sizes are 250 or greater. The bias IQRs are smaller for DupER/MCMC than for RAW across nearly all DupER/MCMC variations and testing conditions.

DupER/MCMC estimates are less biased than those for DupER/EM for every DupER variation and every testing condition when deletion rates are 40% or 60%; results are mixed for the 20% DupER variations. DupER/MCMC median bias ranges from 99% less to 100% more than (median=42% less than) those from the corresponding DupER/EM variation.

Table 5

Bias of item discrimination estimates [median (interquartile range)]

Testing Condition (Sample size and test length)			DupER Variation (EM Imputation) (Duplication rate and deletion rate)								
				10 duplications			25 duplications			50 duplications	
		RAW	20%	40%	60%	20%	40%	60%	20%	40%	60%
N= 100	*n*= 10	0.127 (0.662)	0.451 (0.911)	0.558 (1.598)	1.035 (1.509)	0.767 (1.218)	0.814 (2.010)	1.251 (1.687)	0.982 (1.209)	0.982 (2.196)	1.296 (1.493)
	n= 30	0.236 (0.488)	0.489 (0.353)	0.526 (0.513)	0.689 (0.606)	0.680 (0.350)	0.627 (0.506)	0.687 (0.799)	0.736 (0.394)	0.593 (0.569)	0.686 (0.962)
	n =60	0.162 (0.413)	0.517 (0.243)	0.479 (0.231)	0.522 (0.254)	0.658 (0.367)	0.647 (0.280)	0.602 (0.287)	0.669 (0.369)	0.667 (0.302)	0.609 (0.291)
N= 250	*n* =10	0.130 (0.423)	0.252 (1.318)	0.306 (2.342)	0.832 (2.444)	0.431 (1.534)	0.418 (2.316)	0.909 (2.349)	0.487 (1.575)	0.473 (2.298)	0.853 (2.165)
	n =30	0.128 (0.362)	0.270 (0.389)	0.239 (0.748)	0.382 (1.176)	0.294 (0.396)	0.221 (0.815)	0.354 (1.453)	0.314 (0.386)	0.237 (0.779)	0.349 (1.560)
	n =60	0.130 (0.287)	0.308 (0.213)	0.328 (0.326)	0.443 (0.451)	0.355 (0.231)	0.315 (0.336)	0.437 (0.522)	0.360 (0.233)	0.321 (0.349)	0.412 (0.554)
N= 500	*n* =10	0.086 (0.212)	0.125 (1.298)	0.159 (2.454)	0.606 (2.749)	0.175 (1.407)	0.190 (2.450)	0.533 (2.456)	0.200 (1.388)	0.207 (2.403)	0.500 (2.287)
	n =30	0.092 (0.269)	0.158 (0.503)	0.129 (0.983)	0.284 (1.649)	0.136 (0.485)	0.108 (1.036)	0.276 (1.821)	0.142 (0.485)	0.110 (1.052)	0.266 (1.921)
	n =60	0.096 (0.217)	0.184 (0.287)	0.224 (0.535)	0.339 (0.749)	0.176 (0.290)	0.210 (0.565)	0.317 (0.861)	0.169 (0.289)	0.203 (0.575)	0.298 (0.919)
N=1000	*n* =10	0.084 (0.166)	**0.051** (1.222)	0.124 (2.483)	0.443 (2.924)	**0.058** (1.224)	0.132 (2.493)	0.404 (2.827)	**0.062** (1.223)	0.133 (2.402)	0.347 (2.812)
	n =30	0.062 (0.169)	0.091 (0.559)	0.063 (1.290)	0.257 (1.937)	0.064 (0.548)	0.065 (1.297)	0.247 (2.230)	**0.061** (0.551)	0.070 (1.315)	0.241 (2.389)
	n =60	0.069 (0.154)	0.100 (0.406)	0.148 (0.779)	0.283 (1.086)	0.086 (0.388)	0.127 (0.797)	0.255 (1.193)	0.073 (0.385)	0.119 (0.796)	0.246 (1.240)

Testing Condition (Sample size and test length)			DupER Variation (MCMC Imputation) (Duplication rate and deletion rate)								
				10 duplications			25 duplications			50 duplications	
		RAW	20%	40%	60%	20%	40%	60%	20%	40%	60%
N= 100	*n*= 10	0.127 (0.662)	0.409 (0.219)	0.402 (0.236)	0.408 (0.286)	0.661 (0.361)	0.620 (0.379)	0.439 (0.391)	0.830 (0.494)	0.708 (0.464)	0.532 (0.496)
	n= 30	0.236 (0.488)	0.422 (0.254)	0.362 (0.236)	0.376 (0.289)	0.562 (0.220)	0.411 (0.289)	0.350 (0.284)	0.597 (0.185)	0.426 (0.220)	0.310 (0.227)
	n =60	0.162 (0.413)	0.401 (0.225)	0.299 (0.238)	0.271 (0.275)	0.572 (0.342)	0.425 (0.244)	0.299 (0.239)	0.568 (0.402)	0.422 (0.268)	0.287 (0.217)
N= 250	*n* =10	0.130 (0.423)	0.253 (0.225)	0.215 (0.230)	0.198 (0.272)	0.348 (0.297)	0.248 (0.291)	**0.129** (0.254)	0.415 (0.313)	0.250 (0.268)	**0.076** (0.225)
	n =30	0.128 (0.362)	0.210 (0.160)	0.166 (0.232)	0.176 (0.272)	0.237 (0.104)	0.162 (0.189)	**0.101** (0.200)	0.235 (0.088)	0.153 (0.151)	**0.053** (0.243)
	n =60	0.130 (0.287)	0.212 (0.163)	0.161 (0.179)	0.135 (0.236)	0.247 (0.135)	0.168 (0.153)	**0.117** (0.214)	0.237 (0.143)	0.154 (0.139)	**0.093** (0.213)
N= 500	*n* =10	0.086 (0.212)	0.181 (0.135)	0.125 (0.144)	**0.046** (0.195)	0.216 (0.149)	0.106 (0.131)	**-0.055** (0.116)	0.213 (0.147)	0.095 (0.129)	-0.119 (0.149)
	n =30	0.092 (0.269)	0.113 (0.132)	**0.087** (0.180)	**0.058** (0.235)	0.124 (0.105)	**0.080** (0.149)	**0.009** (0.274)	0.120 (0.102)	**0.063** (0.135)	**-0.008** (0.286)
	n =60	0.096 (0.217)	0.124 (0.112)	**0.093** (0.179)	**0.058** (0.233)	0.136 (0.095)	**0.084** (0.152)	**0.030** (0.217)	0.134 (0.095)	**0.071** (0.133)	**0.015** (0.211)
N=1000	*n* =10	0.084 (0.166)	0.102 (0.098)	**0.047** (0.122)	**-0.080** (0.153)	0.095 (0.097)	**0.029** (0.113)	-0.153 (0.193)	0.089 (0.083)	**0.018** (0.110)	-0.214 (0.214)
	n =30	0.062 (0.169)	0.072 (0.103)	**0.034** (0.154)	**-0.013** (0.217)	0.067 (0.078)	**0.020** (0.128)	**-0.037** (0.244)	0.062 (0.066)	**0.010** (0.127)	**-0.053** (0.257)
	n =60	0.069 (0.154)	**0.068** (0.094)	**0.042** (0.171)	**0.002** (0.217)	**0.066** (0.079)	**0.021** (0.148)	**-0.028** (0.201)	**0.063** (0.075)	**0.012** (0.142)	**-0.042** (0.207)

Note: Values in bold indicate that the absolute value based on DupERAugmentation is less than the corresponding RAW value

Correlations.

Detailed correlation results for the item discrimination estimates are shown in Table 6. Median correlations for the RAW conditions increase as test length and sample size increased with a range across items from .432 (N=100, n=10) to .881 (N=1000, n=60).

All DupER/EM variations have lower median correlations than the corresponding RAW data. Overall, correlations based on DupER/EM data range from .09 to .78 less than (median=.30 less than) the corresponding RAW median correlations. The best performing DupER variations across testing conditions are those with a deletion rate of 20%. The correlation IQRs are larger for DupER/EM than for RAW across all DupER/EM variations and testing conditions.

DupER/MCMC-based estimates tend to have lower median correlations than do RAW across most testing conditions, but are much closer to the RAW median correlations then are the DupER/EM estimates. DupER/MCMC variations sometimes outperform RAW at the N=100/n=10 testing condition. Additionally, DupER/MCMC[50/60] outperforms RAW for the three smallest sample sizes when n=10. DupER/MCMC median correlations range from .18 less than to .07 greater than (median=.05 less than) the corresponding RAW correlations. The performance of different DupER/MCMC variations relative to the RAW data is similar across variations. The correlation IQRs are larger for DupER/MCMC than for RAW across nearly all DupER/MCMC variations and testing conditions.

DupER/MCMC estimates are more highly correlated with the parameters than those of DupER/EM for every DupER variation and testing condition. DupER/MCMC

correlations range from .03 to .75 greater than (median=.28 greater than) those from the corresponding DupER/EM variation.

Table 6

Correlation of item discrimination estimates with the item parameters [median (interquartile range)]

Testing Condition (Sample size and test length)			DupER Variation (EM Imputation) (Duplication rate and deletion rate)								
				10 duplications			25 duplications			50 duplications	
		RAW	20%	40%	60%	20%	40%	60%	20%	40%	60%
$N=100$	$n=10$	0.432 (0.364)	0.248 (0.376)	0.133 (0.360)	0.132 (0.369)	0.221 (0.379)	0.117 (0.353)	0.153 (0.379)	0.220 (0.363)	0.130 (0.370)	0.203 (0.379)
	$n=30$	0.518 (0.141)	0.431 (0.181)	0.423 (0.181)	0.387 (0.191)	0.375 (0.212)	0.396 (0.186)	0.366 (0.187)	0.350 (0.224)	0.370 (0.190)	0.324 (0.197)
	$n=60$	0.526 (0.108)	0.404 (0.154)	0.437 (0.138)	0.436 (0.137)	0.334 (0.181)	0.383 (0.155)	0.412 (0.151)	0.319 (0.179)	0.360 (0.165)	0.386 (0.153)
$N=250$	$n=10$	0.643 (0.222)	0.229 (0.291)	0.079 (0.254)	0.108 (0.280)	0.205 (0.301)	0.090 (0.249)	0.117 (0.308)	0.199 (0.314)	0.088 (0.264)	0.129 (0.302)
	$n=30$	0.698 (0.099)	0.565 (0.153)	0.482 (0.143)	0.388 (0.138)	0.535 (0.168)	0.467 (0.148)	0.324 (0.142)	0.521 (0.177)	0.447 (0.153)	0.298 (0.133)
	$n=60$	0.711 (0.069)	0.581 (0.127)	0.555 (0.116)	0.496 (0.117)	0.546 (0.148)	0.536 (0.120)	0.440 (0.117)	0.530 (0.158)	0.522 (0.121)	0.417 (0.115)
$N=500$	$n=10$	0.747 (0.167)	0.224 (0.266)	0.058 (0.210)	0.084 (0.231)	0.196 (0.260)	0.054 (0.208)	0.093 (0.255)	0.200 (0.285)	0.050 (0.225)	0.120 (0.275)
	$n=30$	0.793 (0.074)	0.644 (0.114)	0.481 (0.107)	0.325 (0.104)	0.639 (0.117)	0.465 (0.109)	0.264 (0.102)	0.635 (0.123)	0.455 (0.113)	0.247 (0.089)
	$n=60$	0.810 (0.049)	0.689 (0.087)	0.576 (0.091)	0.427 (0.089)	0.683 (0.095)	0.564 (0.094)	0.370 (0.090)	0.679 (0.098)	0.560 (0.095)	0.355 (0.088)
$N=1000$	$n=10$	0.829 (0.116)	0.231 (0.213)	0.053 (0.146)	0.066 (0.178)	0.218 (0.228)	0.049 (0.155)	0.085 (0.179)	0.220 (0.234)	0.049 (0.163)	0.080 (0.187)
	$n=30$	0.865 (0.049)	0.698 (0.085)	0.454 (0.086)	0.244 (0.078)	0.698 (0.088)	0.439 (0.083)	0.212 (0.066)	0.697 (0.091)	0.430 (0.080)	0.201 (0.062)
	$n=60$	0.881 (0.032)	0.757 (0.063)	0.544 (0.070)	0.331 (0.063)	0.758 (0.067)	0.534 (0.069)	0.299 (0.062)	0.759 (0.065)	0.530 (0.069)	0.287 (0.062)

Testing Condition (Sample size and test length)			DupER Variation (MCMC Imputation) (Duplication rate and deletion rate)								
				10 duplications			25 duplications			50 duplications	
		RAW	20%	40%	60%	20%	40%	60%	20%	40%	60%
$N=100$	$n=10$	0.432 (0.364)	**0.462** (0.362)	**0.446** (0.353)	**0.467** (0.319)	0.403 (0.374)	0.422 (0.365)	**0.501** (0.320)	0.372 (0.387)	0.367 (0.372)	**0.482** (0.319)
	$n=30$	0.518 (0.141)	0.480 (0.181)	0.505 (0.167)	0.470 (0.158)	0.420 (0.189)	0.483 (0.167)	0.481 (0.149)	0.391 (0.203)	0.462 (0.179)	0.471 (0.177)
	$n=60$	0.526 (0.108)	0.437 (0.152)	0.483 (0.130)	0.478 (0.129)	0.374 (0.168)	0.451 (0.141)	0.481 (0.128)	0.347 (0.178)	0.426 (0.152)	0.472 (0.141)
$N=250$	$n=10$	0.643 (0.222)	0.585 (0.286)	0.562 (0.280)	0.616 (0.228)	0.540 (0.323)	0.554 (0.305)	0.641 (0.221)	0.516 (0.328)	0.530 (0.315)	**0.656** (0.205)
	$n=30$	0.698 (0.099)	0.640 (0.142)	0.661 (0.130)	0.621 (0.135)	0.613 (0.169)	0.654 (0.136)	0.635 (0.135)	0.598 (0.170)	0.647 (0.144)	0.641 (0.135)
	$n=60$	0.711 (0.069)	0.629 (0.112)	0.652 (0.104)	0.634 (0.093)	0.597 (0.137)	0.643 (0.104)	0.648 (0.099)	0.582 (0.146)	0.640 (0.108)	0.653 (0.101)
$N=500$	$n=10$	0.747 (0.167)	0.691 (0.228)	0.673 (0.235)	0.713 (0.180)	0.656 (0.272)	0.666 (0.268)	0.744 (0.155)	0.648 (0.264)	0.671 (0.252)	**0.757** (0.135)
	$n=30$	0.793 (0.074)	0.757 (0.102)	0.760 (0.092)	0.720 (0.100)	0.749 (0.114)	0.765 (0.095)	0.738 (0.099)	0.740 (0.114)	0.766 (0.093)	0.742 (0.096)
	$n=60$	0.810 (0.049)	0.756 (0.076)	0.762 (0.068)	0.735 (0.069)	0.746 (0.085)	0.766 (0.074)	0.758 (0.075)	0.744 (0.084)	0.766 (0.074)	0.768 (0.074)
$N=1000$	$n=10$	0.829 (0.116)	0.793 (0.182)	0.784 (0.178)	0.788 (0.112)	0.772 (0.193)	0.784 (0.174)	0.799 (0.092)	0.769 (0.204)	0.798 (0.175)	0.804 (0.079)
	$n=30$	0.865 (0.049)	0.848 (0.064)	0.841 (0.066)	0.797 (0.075)	0.846 (0.066)	0.853 (0.061)	0.813 (0.069)	0.845 (0.069)	0.853 (0.063)	0.819 (0.064)
	$n=60$	0.881 (0.032)	0.856 (0.044)	0.850 (0.043)	0.822 (0.052)	0.855 (0.047)	0.857 (0.042)	0.841 (0.044)	0.854 (0.045)	0.859 (0.042)	0.848 (0.042)

Difficulty Estimates

RMSE.

Detailed RMSE results for the item difficulty estimates are shown in Table 7. Median RMSEs for the RAW conditions tend to decrease as sample size increases, but do not consistently vary with test length. The median RMSEs across items range from 0.202 (N=1000, n=30) to 0.451 (N=100, n=60).

All DupER/EM variations have higher median RMSEs than the corresponding RAW data. DupER/EM median RMSEs range from 25% to 349% greater than (median=114% greater than) the corresponding RAW median RMSEs. The best performing DupER variations across testing conditions are those with a deletion rate of 20%. The RMSE IQRs are larger for DupER/EM than for RAW across all DupER/EM variations and testing conditions.

With the exception of DupER/MCMC[10/60], median RMSEs for DupER/MCMC are higher than those for RAW across all testing conditions. DupER/MCMC[10/60] has a lower median RMSE in testing condition N=1000/n=10. DupER/MCMC RMSEs range from 4% less than to 151% greater than (median=31% greater than) the corresponding RAW RMSEs. The best performing DupER variation across testing conditions is DupER/MCMC[25/60]. The RMSE IQRs are larger for DupER/MCMC than for RAW across nearly all DupER/MCMC variations and testing conditions and are much larger when n=10.

Median RMSEs for DupER/MCMC are lower than those for DupER/EM for nearly all DupER variations and every testing condition. DupER/MCMC median RMSEs range

from 3% greater to 71% less than (median=29% less than) those from the corresponding DupER/EM variation.

Table 7

RMSE of item difficulty estimates [median (interquartile range)]

Testing Condition (Sample size and test length)		DupER Variation (EM Imputation) (Duplication rate and deletion rate)									
			10 duplications			25 duplications			50 duplications		
		RAW	20%	40%	60%	20%	40%	60%	20%	40%	60%
N=100	*n*=10	0.436 (0.588)	1.181 (1.084)	1.521 (1.332)	1.144 (0.514)	1.527 (1.841)	1.832 (2.278)	1.111 (0.824)	1.207 (2.558)	1.957 (3.188)	1.454 (0.853)
	n=30	0.447 (0.417)	0.733 (0.827)	0.818 (1.255)	0.987 (1.769)	0.870 (1.599)	0.945 (1.761)	1.207 (2.386)	0.957 (1.635)	1.120 (2.277)	1.422 (2.720)
	n=60	0.451 (0.258)	0.649 (0.661)	0.680 (0.709)	0.757 (0.853)	0.780 (0.948)	0.822 (1.353)	0.897 (1.649)	0.908 (1.290)	0.850 (1.658)	0.938 (2.003)
N=250	*n*=10	0.310 (0.246)	0.580 (1.131)	1.159 (1.766)	0.891 (1.163)	0.661 (1.084)	0.976 (2.299)	0.771 (1.285)	0.731 (0.932)	0.984 (2.347)	0.957 (1.284)
	n=30	0.338 (0.205)	0.428 (0.428)	0.574 (0.770)	0.914 (1.454)	0.443 (0.547)	0.591 (0.861)	0.861 (1.386)	0.454 (0.645)	0.591 (0.931)	0.846 (1.521)
	n=60	0.340 (0.172)	0.465 (0.352)	0.508 (0.450)	0.554 (0.730)	0.492 (0.400)	0.553 (0.533)	0.613 (0.856)	0.492 (0.408)	0.539 (0.582)	0.652 (0.922)
N=500	*n*=10	0.248 (0.224)	0.477 (0.697)	0.828 (0.970)	0.694 (1.350)	0.516 (0.698)	0.807 (0.940)	0.785 (1.503)	0.538 (0.690)	0.778 (1.044)	0.868 (1.643)
	n=30	0.263 (0.150)	0.337 (0.346)	0.530 (0.631)	0.794 (1.054)	0.372 (0.343)	0.551 (0.648)	0.799 (0.939)	0.374 (0.340)	0.535 (0.634)	0.802 (0.911)
	n=60	0.260 (0.122)	0.325 (0.246)	0.446 (0.486)	0.609 (0.761)	0.341 (0.261)	0.485 (0.481)	0.718 (0.765)	0.355 (0.272)	0.495 (0.427)	0.755 (0.735)
N=1000	*n*=10	0.208 (0.058)	0.295 (0.514)	0.439 (0.892)	0.694 (1.438)	0.349 (0.646)	0.454 (0.969)	0.799 (1.584)	0.387 (0.740)	0.468 (1.021)	0.876 (1.645)
	n=30	0.202 (0.119)	0.281 (0.324)	0.499 (0.609)	0.758 (0.850)	0.305 (0.354)	0.481 (0.592)	0.751 (0.818)	0.323 (0.371)	0.482 (0.605)	0.741 (0.823)
	n=60	0.208 (0.088)	0.280 (0.225)	0.546 (0.489)	0.747 (0.740)	0.316 (0.229)	0.535 (0.496)	0.751 (0.758)	0.328 (0.252)	0.522 (0.520)	0.775 (0.766)

Testing Condition (Sample size and test length)		DupER Variation (MCMC Imputation) (Duplication rate and deletion rate)									
			10 duplications			25 duplications			50 duplications		
		RAW	20%	40%	60%	20%	40%	60%	20%	40%	60%
N=100	*n*=10	0.436 (0.588)	0.915 (0.594)	0.912 (1.063)	0.632 (0.560)	1.043 (1.126)	0.986 (1.020)	0.565 (0.411)	1.095 (1.932)	0.989 (1.789)	0.542 (0.228)
	n=30	0.447 (0.417)	0.662 (0.738)	0.684 (0.679)	0.726 (0.991)	0.789 (1.230)	0.782 (1.202)	0.734 (1.152)	0.894 (1.278)	0.811 (1.441)	0.816 (1.609)
	n=60	0.451 (0.258)	0.670 (0.578)	0.681 (0.558)	0.714 (0.533)	0.763 (0.908)	0.728 (0.923)	0.773 (0.904)	0.868 (1.244)	0.859 (1.237)	0.827 (1.273)
N=250	*n*=10	0.310 (0.246)	0.477 (0.727)	0.473 (0.708)	0.381 (0.115)	0.587 (0.751)	0.540 (0.683)	0.340 (0.095)	0.668 (0.797)	0.577 (0.902)	0.330 (0.111)
	n=30	0.338 (0.205)	0.406 (0.372)	0.431 (0.384)	0.520 (0.564)	0.433 (0.508)	0.414 (0.480)	0.469 (0.582)	0.444 (0.558)	0.428 (0.588)	0.460 (0.495)
	n=60	0.340 (0.172)	0.445 (0.311)	0.455 (0.357)	0.512 (0.394)	0.474 (0.372)	0.460 (0.434)	0.482 (0.428)	0.490 (0.467)	0.477 (0.466)	0.466 (0.435)
N=500	*n*=10	0.248 (0.224)	0.369 (0.553)	0.387 (0.535)	0.254 (0.091)	0.408 (0.525)	0.371 (0.547)	0.254 (0.067)	0.445 (0.484)	0.395 (0.526)	0.298 (0.126)
	n=30	0.263 (0.150)	0.293 (0.236)	0.301 (0.247)	0.353 (0.307)	0.308 (0.240)	0.300 (0.245)	0.311 (0.288)	0.316 (0.229)	0.309 (0.233)	0.308 (0.250)
	n=60	0.260 (0.122)	0.301 (0.186)	0.319 (0.186)	0.353 (0.245)	0.319 (0.223)	0.318 (0.238)	0.323 (0.261)	0.332 (0.216)	0.308 (0.208)	0.308 (0.257)
N=1000	*n*=10	0.208 (0.058)	0.257 (0.289)	0.242 (0.204)	**0.199** (0.067)	0.278 (0.352)	0.220 (0.226)	0.272 (0.144)	0.299 (0.403)	0.221 (0.241)	0.328 (0.205)
	n=30	0.202 (0.119)	0.218 (0.132)	0.206 (0.137)	0.248 (0.245)	0.235 (0.149)	0.212 (0.138)	0.230 (0.199)	0.241 (0.159)	0.214 (0.159)	0.233 (0.169)
	n=60	0.208 (0.088)	0.223 (0.148)	0.220 (0.159)	0.235 (0.197)	0.233 (0.171)	0.215 (0.145)	0.224 (0.178)	0.237 (0.175)	0.220 (0.143)	0.223 (0.164)

Note: Values in bold indicate that the value based on DupER Augmentation is lower than the corresponding RAW value

Bias.

Detailed bias results for the item difficulty estimates are shown in Table 8. Median bias for the RAW conditions tends to decrease as sample size increases, but does not consistently vary with test length. The median RAW difficulty estimates are uniformly overestimated across testing conditions with a range across items from 0.100 (N=1000, n=30) to 0.195 (N=100, n=30).

Nearly all DupER/EM variations are less biased than the corresponding RAW data for the 20% deletion DupER/EM variations. Results are mixed for the DupER/EM variations using 40% and 60% deletion rates. Overall, median bias based on DupER/EM data ranges from 99% less than to 316% greater than (median=11% less than) the corresponding RAW median bias (in absolute terms). Across testing conditions, the 20% deletion DupER/EM variations performed the best. The bias IQRs for DupER/EM tend to increase as the deletion rate increases, but do not show a consistent pattern when compared with RAW.

Like DupER/EM, nearly all DupER/MCMC variations are less biased than the corresponding RAW data for the 20% deletion DupER/MCMC variations. Additionally, DupER/MCMC tends to outperform RAW for the 40% and 60% deletion rates as well with the exception of the N=100 conditions, where results are mixed. Overall, median bias based on DupER/MCMC data ranges from 99% less than to 117% greater than (median=27% less than) the corresponding RAW median bias (in absolute terms). Across testing conditions, all DupER/MCMC variations perform similarly. The bias IQRs for DupER/MCMC do not show a consistent pattern across DupER/MCMC variations, nor do they show a consistent pattern when compared with RAW.

DupER/MCMC estimates tend to be less biased than those for DupER/EM for most DupER variations and testing conditions when deletion rates are 60%. The situation is reversed when deletion rates are 20% with DupER/EM tending to outperform DupER/MCMC across testing conditions. Results are mixed for the 40% DupER variations. DupER/MCMC median bias ranges from 100% less than to 11,100% greater than (median=7% less than) those from the corresponding DupER/EM variation.

Table 8

Bias of item difficulty estimates [median (interquartile range)]

Testing Condition (Sample size and test length)			DupER Variation (EM Imputation) (Duplication rate and deletion rate)									
				10 duplications			25 duplications			50 duplications		
		RAW	20%	40%	60%	20%	40%	60%	20%	40%	60%
N= 100	*n*=10	0.154 (0.246)	**0.069** (0.310)	**0.099** (0.505)	-0.260 (0.812)	**0.122** (0.319)	**0.095** (0.563)	-0.208 (0.667)	**0.107** (0.412)	**0.071** (0.322)	-0.160 (0.841)
	n=30	0.195 (0.272)	**0.127** (0.192)	**0.070** (0.364)	**0.072** (0.354)	**0.124** (0.185)	**0.069** (0.315)	**0.027** (0.402)	**0.123** (0.250)	**0.030** (0.240)	**0.019** (0.393)
	n=60	0.189 (0.250)	**0.182** (0.200)	**0.161** (0.208)	**0.148** (0.239)	**0.199** (0.220)	**0.169** (0.210)	**0.105** (0.229)	0.211 (0.251)	**0.169** (0.217)	**0.081** (0.262)
N= 250	*n*=10	0.143 (0.200)	**-0.025** (0.194)	**-0.089** (0.386)	-0.255 (0.836)	**0.005** (0.271)	**-0.129** (0.577)	-0.307 (0.892)	**0.011** (0.240)	**-0.118** (0.529)	-0.350 (0.943)
	n=30	0.136 (0.166)	**0.012** (0.255)	**-0.076** (0.524)	-0.166 (0.694)	**0.014** (0.206)	**-0.079** (0.556)	-0.150 (0.773)	**0.013** (0.199)	**-0.080** (0.557)	-0.167 (0.800)
	n=60	0.131 (0.223)	**0.095** (0.195)	**0.021** (0.333)	**-0.072** (0.482)	**0.092** (0.186)	**-0.007** (0.345)	-0.144 (0.588)	**0.087** (0.175)	**-0.023** (0.329)	-0.177 (0.618)
N= 500	*n*=10	0.136 (0.134)	**-0.001** (0.237)	-0.164 (0.577)	-0.340 (0.806)	**-0.002** (0.255)	-0.149 (0.517)	-0.380 (0.879)	**-0.009** (0.253)	-0.140 (0.483)	-0.412 (0.936)
	n=30	0.127 (0.149)	**-0.011** (0.341)	-0.189 (0.622)	-0.295 (0.819)	**-0.018** (0.290)	-0.222 (0.569)	-0.329 (0.840)	**-0.026** (0.278)	-0.229 (0.567)	-0.327 (0.862)
	n=60	0.115 (0.182)	**0.008** (0.209)	-0.134 (0.499)	-0.267 (0.731)	**-0.007** (0.175)	-0.181 (0.512)	-0.352 (0.768)	**-0.015** (0.183)	-0.214 (0.539)	-0.332 (0.806)
N=1000	*n*=10	0.121 (0.066)	**-0.004** (0.248)	-0.162 (0.501)	-0.375 (0.833)	**-0.006** (0.265)	-0.151 (0.460)	-0.421 (0.901)	**-0.011** (0.275)	-0.149 (0.453)	-0.457 (0.976)
	n=30	0.100 (0.128)	**-0.056** (0.335)	-0.242 (0.648)	-0.337 (0.881)	**-0.063** (0.329)	-0.254 (0.662)	-0.341 (0.838)	**-0.060** (0.318)	-0.257 (0.648)	-0.348 (0.841)
	n=60	0.104 (0.168)	**-0.090** (0.289)	-0.271 (0.681)	-0.386 (0.860)	**-0.100** (0.304)	-0.294 (0.678)	-0.422 (0.905)	**-0.102** (0.315)	-0.308 (0.700)	-0.433 (0.889)

Testing Condition (Sample size and test length)			DupER Variation (MCMC Imputation) (Duplication rate and deletion rate)									
				10 duplications			25 duplications			50 duplications		
		RAW	20%	40%	60%	20%	40%	60%	20%	40%	60%
N= 100	*n*=10	0.154 (0.246)	0.171 (0.145)	0.220 (0.168)	0.182 (0.154)	0.209 (0.244)	0.242 (0.233)	**0.148** (0.148)	0.195 (0.293)	0.188 (0.360)	**0.118** (0.194)
	n=30	0.195 (0.272)	**0.169** (0.211)	**0.192** (0.209)	0.247 (0.371)	**0.189** (0.208)	0.204 (0.240)	0.202 (0.270)	**0.171** (0.248)	0.208 (0.327)	0.173 (0.371)
	n=60	0.189 (0.250)	**0.186** (0.210)	0.199 (0.240)	0.245 (0.278)	0.227 (0.202)	0.215 (0.220)	0.242 (0.289)	0.229 (0.282)	0.263 (0.268)	0.253 (0.294)
N= 250	*n*=10	0.143 (0.200)	**0.134** (0.092)	0.164 (0.083)	**0.105** (0.101)	**0.139** (0.135)	**0.126** (0.115)	**0.038** (0.106)	**0.129** (0.164)	**0.108** (0.131)	**-0.054** (0.103)
	n=30	0.136 (0.166)	**0.110** (0.178)	**0.114** (0.177)	**0.134** (0.189)	**0.106** (0.143)	**0.092** (0.170)	**0.078** (0.156)	**0.091** (0.164)	**0.076** (0.173)	**0.057** (0.136)
	n=60	0.131 (0.223)	**0.112** (0.216)	**0.106** (0.224)	0.132 (0.241)	**0.120** (0.168)	**0.106** (0.182)	**0.094** (0.189)	**0.124** (0.181)	**0.117** (0.160)	**0.063** (0.180)
N= 500	*n*=10	0.136 (0.134)	**0.112** (0.169)	**0.113** (0.163)	**0.038** (0.079)	**0.100** (0.174)	**0.079** (0.155)	**-0.087** (0.097)	**0.087** (0.151)	**0.047** (0.153)	-0.160 (0.111)
	n=30	0.127 (0.149)	**0.080** (0.140)	**0.073** (0.156)	**0.068** (0.154)	**0.064** (0.126)	**0.050** (0.140)	**0.021** (0.134)	**0.053** (0.135)	**0.034** (0.126)	**-0.014** (0.135)
	n=60	0.115 (0.182)	**0.099** (0.197)	**0.064** (0.199)	**0.066** (0.240)	**0.088** (0.190)	**0.054** (0.176)	**0.011** (0.167)	**0.078** (0.177)	**0.051** (0.142)	**-0.008** (0.149)
N=1000	*n*=10	0.121 (0.066)	**0.075** (0.156)	**0.065** (0.170)	**-0.094** (0.063)	**0.059** (0.154)	**0.026** (0.141)	-0.200 (0.096)	**0.042** (0.130)	**-0.008** (0.112)	-0.263 (0.090)
	n=30	0.100 (0.128)	**0.051** (0.123)	**0.042** (0.141)	**0.025** (0.138)	**0.033** (0.087)	**0.014** (0.131)	**-0.023** (0.136)	**0.024** (0.086)	**-0.005** (0.099)	**-0.041** (0.147)
	n=60	0.104 (0.168)	**0.070** (0.151)	**0.031** (0.161)	**0.005** (0.170)	**0.060** (0.133)	**0.016** (0.135)	**-0.020** (0.165)	**0.047** (0.123)	**0.001** (0.115)	**-0.029** (0.153)

Note: Values in bold indicate that the absolute value based on DupER Augmentation is less than the corresponding RAW value

Correlations.

Detailed correlation results for the item difficulty estimates are shown in Table 9. Median correlations for the RAW conditions are uniformly high, increase as sample size increases, and tend to be slightly higher when n=30 with a range across items from .948 (N=100, n=10) to .992 (N=1000, n=30).

All DupER/EM variations have lower correlations than the corresponding RAW data. Overall, median correlations based on DupER/EM data range from .01 to .16 less than (median=.03 less than) the corresponding RAW correlations. The best performing DupER variations across testing conditions are those with a deletion rate of 20%. The correlation IQRs are larger for DupER/EM than for RAW across all DupER/EM variations and testing conditions.

Like DupER/EM, all DupER/MCMC-based variations have lower median correlations than RAW across all testing conditions. Overall, median correlations based on DupER/MCMC data range from .02 to .03 less than (median=.02 less than) the corresponding RAW correlations. Across testing conditions, all DupER variations were similarly close to the RAW correlations. Like those for DupER/EM, the correlation IQRs are larger for DupER/MCMC than for RAW across all DupER/MCMC variations and testing conditions.

DupER/MCMC estimates have higher median correlations than those for DupER/EM for nearly every DupER variation and testing condition. DupER/MCMC median correlations range from .003 less to .074 greater than (median=.014 greater than) those from the corresponding DupER/EM variation.

Table 9

Correlation of item difficulty estimates with the item parameters [median (interquartile range)]

Testing Condition (Sample size and test length)			DupER Variation (EM Imputation) (Duplication rate and deletion rate)								
				10 duplications			25 duplications			50 duplications	
		RAW	20%	40%	60%	20%	40%	60%	20%	40%	60%
N= 100	*n*= 10	0.948 (0.035)	0.909 (0.078)	0.883 (0.099)	0.874 (0.093)	0.884 (0.106)	0.864 (0.118)	0.864 (0.122)	0.864 (0.137)	0.849 (0.134)	0.851 (0.137)
	n= 30	0.956 (0.017)	0.920 (0.052)	0.913 (0.059)	0.895 (0.075)	0.901 (0.087)	0.892 (0.105)	0.867 (0.176)	0.888 (0.133)	0.871 (0.171)	0.847 (0.238)
	n=60	0.952 (0.013)	0.909 (0.043)	0.903 (0.046)	0.890 (0.063)	0.874 (0.083)	0.871 (0.105)	0.852 (0.133)	0.835 (0.156)	0.827 (0.180)	0.796 (0.212)
N= 250	*n*=10	0.970 (0.024)	0.948 (0.047)	0.929 (0.066)	0.913 (0.087)	0.938 (0.058)	0.923 (0.083)	0.909 (0.105)	0.930 (0.072)	0.915 (0.094)	0.907 (0.123)
	n=30	0.975 (0.011)	0.961 (0.023)	0.953 (0.028)	0.937 (0.041)	0.954 (0.030)	0.948 (0.035)	0.933 (0.047)	0.950 (0.033)	0.945 (0.040)	0.934 (0.049)
	n=60	0.973 (0.008)	0.954 (0.020)	0.950 (0.024)	0.941 (0.033)	0.946 (0.027)	0.946 (0.033)	0.940 (0.047)	0.941 (0.034)	0.942 (0.044)	0.938 (0.056)
N= 500	*n*=10	0.982 (0.018)	0.963 (0.035)	0.949 (0.048)	0.937 (0.067)	0.957 (0.040)	0.946 (0.053)	0.936 (0.065)	0.949 (0.045)	0.941 (0.061)	0.934 (0.069)
	n=30	0.985 (0.007)	0.975 (0.014)	0.966 (0.016)	0.954 (0.022)	0.970 (0.017)	0.963 (0.018)	0.954 (0.023)	0.967 (0.018)	0.962 (0.019)	0.953 (0.023)
	n=60	0.984 (0.005)	0.972 (0.011)	0.969 (0.012)	0.962 (0.016)	0.968 (0.012)	0.967 (0.013)	0.961 (0.016)	0.965 (0.013)	0.965 (0.014)	0.959 (0.017)
N=1000	*n*=10	0.990 (0.010)	0.973 (0.024)	0.963 (0.033)	0.952 (0.040)	0.967 (0.028)	0.960 (0.033)	0.947 (0.039)	0.964 (0.031)	0.957 (0.037)	0.950 (0.044)
	n=30	0.992 (0.004)	0.980 (0.009)	0.971 (0.012)	0.961 (0.015)	0.976 (0.010)	0.967 (0.012)	0.959 (0.015)	0.974 (0.011)	0.965 (0.012)	0.958 (0.015)
	n=60	0.990 (0.003)	0.981 (0.006)	0.976 (0.007)	0.967 (0.010)	0.978 (0.007)	0.973 (0.008)	0.964 (0.010)	0.976 (0.008)	0.971 (0.008)	0.962 (0.011)

Testing Condition (Sample size and test length)			DupER Variation (MCMC Imputation) (Duplication rate and deletion rate)								
				10 duplications			25 duplications			50 duplications	
		RAW	20%	40%	60%	20%	40%	60%	20%	40%	60%
N= 100	*n*= 10	0.948 (0.035)	0.923 (0.072)	0.919 (0.085)	0.926 (0.070)	0.908 (0.090)	0.905 (0.092)	0.926 (0.072)	0.890 (0.108)	0.894 (0.115)	0.925 (0.078)
	n= 30	0.956 (0.017)	0.926 (0.045)	0.923 (0.048)	0.910 (0.057)	0.907 (0.076)	0.905 (0.088)	0.899 (0.083)	0.891 (0.107)	0.889 (0.115)	0.892 (0.115)
	n=60	0.952 (0.013)	0.911 (0.039)	0.911 (0.039)	0.903 (0.053)	0.879 (0.083)	0.881 (0.084)	0.879 (0.097)	0.849 (0.142)	0.846 (0.146)	0.851 (0.160)
N= 250	*n*=10	0.970 (0.024)	0.960 (0.039)	0.956 (0.043)	0.958 (0.043)	0.952 (0.050)	0.955 (0.050)	0.962 (0.043)	0.944 (0.058)	0.948 (0.056)	0.962 (0.048)
	n=30	0.975 (0.011)	0.964 (0.023)	0.963 (0.022)	0.953 (0.029)	0.956 (0.027)	0.957 (0.027)	0.955 (0.033)	0.951 (0.031)	0.954 (0.031)	0.954 (0.032)
	n=60	0.973 (0.008)	0.955 (0.019)	0.954 (0.019)	0.948 (0.022)	0.946 (0.024)	0.947 (0.026)	0.947 (0.028)	0.939 (0.031)	0.942 (0.030)	0.945 (0.031)
N= 500	*n*=10	0.982 (0.018)	0.975 (0.026)	0.973 (0.027)	0.974 (0.029)	0.969 (0.032)	0.973 (0.030)	0.977 (0.029)	0.965 (0.037)	0.969 (0.037)	0.975 (0.032)
	n=30	0.985 (0.007)	0.978 (0.014)	0.977 (0.013)	0.972 (0.017)	0.973 (0.018)	0.976 (0.017)	0.975 (0.016)	0.969 (0.020)	0.974 (0.018)	0.975 (0.015)
	n=60	0.984 (0.005)	0.971 (0.012)	0.971 (0.012)	0.968 (0.013)	0.965 (0.014)	0.968 (0.015)	0.969 (0.014)	0.962 (0.015)	0.966 (0.016)	0.968 (0.015)
N=1000	*n*=10	0.990 (0.010)	0.985 (0.016)	0.985 (0.017)	0.984 (0.022)	0.983 (0.021)	0.986 (0.018)	0.982 (0.024)	0.980 (0.025)	0.985 (0.020)	0.980 (0.025)
	n=30	0.992 (0.004)	0.987 (0.010)	0.987 (0.009)	0.984 (0.009)	0.983 (0.013)	0.987 (0.010)	0.986 (0.008)	0.981 (0.014)	0.986 (0.011)	0.986 (0.009)
	n=60	0.990 (0.003)	0.982 (0.009)	0.983 (0.008)	0.980 (0.008)	0.978 (0.010)	0.981 (0.010)	0.983 (0.008)	0.976 (0.011)	0.980 (0.010)	0.983 (0.009)

Guessing Estimates

RMSE.

Detailed RMSE results for the item guessing estimates are shown in Table 10. Median RMSEs for the RAW conditions tend to decrease as sample size increases but does not consistently vary with test length. Specifically, median RMSEs across items ranges from 0.042 (N=1000, n=30) to 0.068 (N=100, n=30).

All DupER/EM variations have higher median RMSEs than the corresponding RAW data. DupER/EM median RMSEs range from 13% to 196% greater than (median=74% greater than) the corresponding RAW RMSEs. Across testing conditions, the DupER/EM median RMSEs tend to get worse as the DupER duplication rate increases; within duplication rates, there is much less variation in the DupER/EM median RMSEs. The RMSE IQRs are larger for DupER/EM than for RAW across nearly all DupER/EM variations when N=250, N=500, and N=1000, but do not show a consistent pattern when N=100.

Like DupER/EM, all DupER/MCMC-based variations have higher median RMSEs than does RAW across all testing conditions. DupER/MCMC median RMSEs range from range from 20% to 137% greater than (median=54% greater than) the corresponding RAW median RMSEs. Across testing conditions, the DupER/MCMC and RMSEs tend to increase as the DupER duplication rate increases and decrease as the deletion rate increases. The best-performing DupER/MCMC variation is DupER/MCMC[10/60]. The RMSE IQRs for DupER/MCMC do not show a consistent pattern across DupER/MCMC variations, nor do they show a consistent pattern when compared with RAW.

Median RMSE comparisons between DupER/MCMC and DupER/EM yield mixed results. Specifically, median RMSEs for DupER/MCMC tend to be lower than those for DupER/EM for DupER variations with 40% and 60% deletion rates. Conversely, median RMSEs for DupER/MCMC tend to be greater than those for DupER/EM for DupER variations with 40% deletion rates. DupER/MCMC median RMSEs range from 34% greater to 52% less than (median=4% less than) those from the corresponding DupER/EM variation.

Table 10

RMSE of item guessing estimates [median (interquartile range)]

Testing Condition (Sample size and test length)			DupER Variation (EM Imputation) (Duplication rate and deletion rate)									
				10 duplications			25 duplications			50 duplications		
		RAW	20%	40%	60%	20%	40%	60%	20%	40%	60%
$N=100$	$n=10$	0.060 (0.046)	0.075 (0.019)	0.079 (0.032)	0.068 (0.061)	0.098 (0.035)	0.100 (0.038)	0.077 (0.037)	0.131 (0.042)	0.130 (0.057)	0.096 (0.057)
	$n=30$	0.068 (0.066)	0.085 (0.042)	0.083 (0.041)	0.088 (0.043)	0.119 (0.078)	0.103 (0.060)	0.109 (0.064)	0.132 (0.097)	0.121 (0.085)	0.122 (0.098)
	$n=60$	0.064 (0.059)	0.110 (0.053)	0.107 (0.053)	0.102 (0.045)	0.136 (0.095)	0.132 (0.087)	0.126 (0.074)	0.151 (0.098)	0.149 (0.095)	0.145 (0.093)
$N=250$	$n=10$	0.056 (0.035)	0.075 (0.034)	0.067 (0.054)	0.065 (0.043)	0.091 (0.048)	0.111 (0.065)	0.076 (0.062)	0.132 (0.064)	0.135 (0.089)	0.089 (0.078)
	$n=30$	0.059 (0.052)	0.074 (0.053)	0.076 (0.056)	0.068 (0.073)	0.103 (0.086)	0.094 (0.079)	0.111 (0.103)	0.112 (0.102)	0.109 (0.099)	0.124 (0.112)
	$n=60$	0.059 (0.050)	0.096 (0.066)	0.086 (0.054)	0.094 (0.066)	0.126 (0.095)	0.111 (0.079)	0.112 (0.082)	0.137 (0.107)	0.125 (0.095)	0.120 (0.097)
$N=500$	$n=10$	0.051 (0.030)	0.071 (0.044)	0.073 (0.065)	0.069 (0.053)	0.108 (0.064)	0.115 (0.085)	0.084 (0.080)	0.137 (0.082)	0.137 (0.110)	0.109 (0.090)
	$n=30$	0.051 (0.043)	0.068 (0.060)	0.071 (0.059)	0.073 (0.088)	0.084 (0.097)	0.088 (0.094)	0.108 (0.109)	0.085 (0.113)	0.091 (0.108)	0.122 (0.131)
	$n=60$	0.053 (0.041)	0.086 (0.070)	0.083 (0.064)	0.086 (0.074)	0.102 (0.095)	0.092 (0.085)	0.106 (0.085)	0.109 (0.105)	0.095 (0.102)	0.110 (0.096)
$N=1000$	$n=10$	0.047 (0.025)	0.071 (0.046)	0.084 (0.069)	0.077 (0.085)	0.106 (0.075)	0.111 (0.097)	0.106 (0.098)	0.125 (0.093)	0.139 (0.115)	0.127 (0.125)
	$n=30$	0.042 (0.038)	0.057 (0.070)	0.072 (0.068)	0.085 (0.119)	0.058 (0.098)	0.085 (0.108)	0.107 (0.142)	0.060 (0.117)	0.086 (0.130)	0.109 (0.144)
	$n=60$	0.050 (0.033)	0.070 (0.065)	0.074 (0.080)	0.087 (0.095)	0.073 (0.093)	0.075 (0.097)	0.101 (0.109)	0.077 (0.104)	0.078 (0.116)	0.108 (0.113)

Testing Condition (Sample size and test length)			DupER Variation (MCMC Imputation) (Duplication rate and deletion rate)									
				10 duplications			25 duplications			50 duplications		
		RAW	20%	40%	60%	20%	40%	60%	20%	40%	60%
$N=100$	$n=10$	0.060 (0.046)	0.082 (0.030)	0.084 (0.032)	0.072 (0.036)	0.108 (0.029)	0.104 (0.021)	0.086 (0.032)	0.123 (0.041)	0.125 (0.021)	0.092 (0.026)
	$n=30$	0.068 (0.066)	0.091 (0.042)	0.088 (0.033)	0.087 (0.044)	0.119 (0.072)	0.102 (0.049)	0.099 (0.039)	0.132 (0.085)	0.126 (0.076)	0.113 (0.060)
	$n=60$	0.064 (0.059)	0.107 (0.045)	0.101 (0.040)	0.093 (0.043)	0.132 (0.085)	0.126 (0.069)	0.117 (0.050)	0.145 (0.096)	0.139 (0.088)	0.130 (0.069)
$N=250$	$n=10$	0.056 (0.035)	0.081 (0.033)	0.078 (0.041)	0.067 (0.034)	0.097 (0.051)	0.094 (0.028)	0.076 (0.017)	0.117 (0.054)	0.111 (0.032)	0.083 (0.023)
	$n=30$	0.059 (0.052)	0.079 (0.043)	0.072 (0.042)	0.078 (0.038)	0.107 (0.080)	0.090 (0.052)	0.081 (0.043)	0.114 (0.097)	0.108 (0.079)	0.090 (0.061)
	$n=60$	0.059 (0.050)	0.097 (0.056)	0.089 (0.047)	0.087 (0.039)	0.120 (0.088)	0.106 (0.064)	0.094 (0.051)	0.131 (0.099)	0.116 (0.084)	0.103 (0.071)
$N=500$	$n=10$	0.051 (0.030)	0.075 (0.032)	0.073 (0.030)	0.061 (0.021)	0.090 (0.050)	0.079 (0.029)	0.068 (0.030)	0.121 (0.060)	0.090 (0.037)	0.084 (0.048)
	$n=30$	0.051 (0.043)	0.075 (0.062)	0.063 (0.047)	0.063 (0.042)	0.092 (0.081)	0.081 (0.056)	0.065 (0.044)	0.096 (0.101)	0.081 (0.073)	0.073 (0.050)
	$n=60$	0.053 (0.041)	0.089 (0.065)	0.079 (0.049)	0.073 (0.037)	0.102 (0.088)	0.091 (0.067)	0.078 (0.056)	0.108 (0.098)	0.094 (0.085)	0.084 (0.073)
$N=1000$	$n=10$	0.047 (0.025)	0.063 (0.037)	0.060 (0.022)	0.060 (0.030)	0.092 (0.044)	0.066 (0.023)	0.080 (0.056)	0.109 (0.054)	0.067 (0.029)	0.096 (0.066)
	$n=30$	0.042 (0.038)	0.071 (0.060)	0.058 (0.036)	0.053 (0.032)	0.078 (0.084)	0.064 (0.056)	0.056 (0.041)	0.080 (0.096)	0.064 (0.066)	0.060 (0.064)
	$n=60$	0.050 (0.033)	0.077 (0.068)	0.068 (0.052)	0.062 (0.041)	0.084 (0.085)	0.074 (0.067)	0.062 (0.051)	0.090 (0.097)	0.077 (0.075)	0.065 (0.057)

Bias.

Detailed bias results for the item guessing estimates are shown in Table 11. Median bias for the RAW conditions tends to decrease as sample size increases, but does not consistently vary with test length. The median RAW guessing estimates are uniformly overestimated across testing conditions. Median bias across items ranges from 0.029 (N=1000, n=30) to 0.056 (N=100, n=30).

All DupER/EM variations are less biased than the corresponding RAW data for the 20% and 40% deletion DupER/EM variations. Results are mixed for the DupER/EM variations using the 60% deletion rate. Overall, median bias based on DupER/EM data ranges from 100% less than to 320% greater than (median=54% less than) the corresponding RAW median bias (in absolute terms). Across testing conditions, the 40% deletion DupER/EM variations perform the best. The bias IQRs for DupER/EM tend to increase as the duplication rate increases. The IQRs are smaller for DupER/EM than for RAW across nearly all DupER/EM variations when N=100 and N=250, but do not show a consistent pattern when N>250.

All DupER/MCMC variations are less biased than the corresponding RAW data for a majority of the testing conditions. Overall, median bias based on DupER/MCMC data ranges from 95% less than to 200% greater than (median=32% less than) the corresponding RAW bias (in absolute terms). Across testing conditions, there is not a consistent pattern to DupER/MCMC results; the best performing variation is DupER/MCMC[50/60], and the worst performing variation is DupER/MCMC[50/40]. The bias IQRs are smaller for DupER/MCMC than for RAW when DupER/MCMC is used with 20% and 40% deletion

rates and for all DupER/MCMC variations when *n*=60. For other DupER/MCMC variations and test length combinations, there is not a consistent pattern.

DupER/MCMC estimates tend to be more biased than those for DupER/EM for most DupER variations and testing conditions when deletion rates are 20% or 40%. Results are mixed at 60% deletion rate. DupER/MCMC median bias ranges from 94% less than to 3,200% greater than (median=44% greater than) the bias from the corresponding DupER/EM variation.

Table 11

Bias of item guessing estimates [median (interquartile range)]

Testing Condition (Sample size and test length)			DupER Variation (EM Imputation) (Duplication rate and deletion rate)									
				10 duplications			25 duplications			50 duplications		
		RAW	20%	40%	60%	20%	40%	60%	20%	40%	60%
	n=10	0.040 (0.078)	**0.020** (0.062)	**0.005** (0.041)	**-0.016** (0.055)	**0.026** (0.045)	**0.004** (0.041)	**-0.033** (0.056)	**0.022** (0.031)	**0.005** (0.046)	-0.043 (0.063)
N=100	n=30	0.056 (0.112)	**0.035** (0.066)	**0.025** (0.069)	**0.019** (0.069)	**0.036** (0.080)	**0.022** (0.073)	**0.013** (0.064)	**0.035** (0.080)	**0.026** (0.071)	**0.013** (0.072)
	n=60	0.050 (0.109)	**0.046** (0.088)	**0.044** (0.087)	**0.042** (0.086)	**0.043** (0.100)	**0.040** (0.101)	**0.030** (0.108)	**0.040** (0.098)	**0.038** (0.100)	**0.028** (0.109)
	n=10	0.035 (0.075)	**0.017** (0.057)	**-0.010** (0.042)	-0.045 (0.064)	**0.016** (0.040)	**-0.010** (0.046)	-0.057 (0.079)	**0.015** (0.036)	**-0.001** (0.048)	-0.071 (0.077)
N=250	n=30	0.046 (0.096)	**0.015** (0.050)	**0.000** (0.054)	**-0.006** (0.060)	**0.022** (0.052)	**0.002** (0.056)	**-0.017** (0.067)	**0.019** (0.062)	**-0.004** (0.060)	**-0.025** (0.068)
	n=60	0.047 (0.093)	**0.027** (0.081)	**0.014** (0.079)	**0.002** (0.088)	**0.026** (0.085)	**0.017** (0.075)	**0.000** (0.078)	**0.022** (0.081)	**0.012** (0.074)	**-0.007** (0.074)
	n=10	0.033 (0.074)	**0.015** (0.056)	**-0.015** (0.055)	-0.057 (0.087)	**0.013** (0.059)	**-0.007** (0.072)	-0.076 (0.083)	**0.017** (0.072)	**0.002** (0.088)	-0.097 (0.094)
N=500	n=30	0.037 (0.084)	**0.008** (0.043)	**-0.006** (0.064)	**-0.022** (0.076)	**0.006** (0.048)	**-0.009** (0.084)	**-0.030** (0.104)	**0.002** (0.067)	**-0.016** (0.104)	**-0.036** (0.132)
	n=60	0.038 (0.077)	**0.015** (0.061)	**0.000** (0.063)	**-0.015** (0.074)	**0.012** (0.060)	**-0.013** (0.073)	**-0.032** (0.089)	**0.005** (0.065)	**-0.019** (0.072)	-0.042 (0.099)
	n=10	0.030 (0.073)	**0.015** (0.067)	**-0.010** (0.070)	-0.072 (0.082)	**0.010** (0.086)	**0.002** (0.088)	-0.104 (0.101)	**0.013** (0.103)	**0.006** (0.113)	-0.126 (0.128)
N=1000	n=30	0.029 (0.066)	**0.004** (0.053)	**-0.015** (0.090)	-0.030 (0.118)	**-0.004** (0.069)	**-0.023** (0.107)	-0.041 (0.124)	**-0.005** (0.084)	**-0.023** (0.121)	-0.042 (0.138)
	n=60	0.032 (0.067)	**0.002** (0.060)	**-0.022** (0.080)	-0.038 (0.103)	**-0.006** (0.053)	**-0.026** (0.092)	-0.053 (0.115)	**-0.009** (0.055)	**-0.031** (0.094)	-0.054 (0.120)

Testing Condition (Sample size and test length)			DupER Variation (MCMC Imputation) (Duplication rate and deletion rate)									
				10 duplications			25 duplications			50 duplications		
		RAW	20%	40%	60%	20%	40%	60%	20%	40%	60%
	n=10	0.040 (0.078)	**0.038** (0.073)	0.041 (0.078)	**0.023** (0.095)	0.046 (0.066)	0.050 (0.080)	**0.009** (0.099)	0.051 (0.058)	0.050 (0.069)	**-0.003** (0.102)
N=100	n=30	0.056 (0.112)	**0.044** (0.079)	**0.047** (0.089)	**0.053** (0.096)	**0.041** (0.077)	**0.043** (0.072)	**0.050** (0.083)	**0.042** (0.081)	**0.043** (0.083)	**0.038** (0.083)
	n=60	0.050 (0.109)	0.052 (0.080)	0.053 (0.082)	0.057 (0.083)	0.047 (0.095)	0.052 (0.083)	0.052 (0.081)	**0.047** (0.096)	**0.048** (0.090)	**0.048** (0.086)
	n=10	0.035 (0.075)	0.036 (0.074)	0.038 (0.084)	**0.004** (0.089)	0.041 (0.063)	0.039 (0.078)	**-0.024** (0.077)	0.041 (0.051)	**0.033** (0.073)	-0.053 (0.084)
N=250	n=30	0.046 (0.096)	**0.033** (0.065)	**0.036** (0.074)	**0.041** (0.085)	**0.030** (0.065)	**0.025** (0.065)	**0.030** (0.071)	**0.028** (0.061)	**0.021** (0.055)	**0.021** (0.074)
	n=60	0.047 (0.093)	**0.039** (0.070)	**0.043** (0.068)	0.049 (0.072)	**0.035** (0.074)	**0.036** (0.068)	**0.033** (0.064)	**0.032** (0.071)	**0.028** (0.066)	**0.025** (0.060)
	n=10	0.033 (0.074)	**0.030** (0.067)	**0.028** (0.071)	**-0.020** (0.063)	**0.028** (0.055)	**0.019** (0.066)	-0.053 (0.072)	**0.020** (0.043)	**0.011** (0.062)	-0.072 (0.079)
N=500	n=30	0.037 (0.084)	**0.022** (0.053)	**0.025** (0.063)	**0.024** (0.070)	**0.015** (0.049)	**0.016** (0.049)	**0.011** (0.060)	**0.013** (0.043)	**0.010** (0.044)	**0.002** (0.052)
	n=60	0.038 (0.077)	**0.031** (0.060)	**0.028** (0.053)	**0.031** (0.061)	**0.027** (0.055)	**0.019** (0.051)	**0.015** (0.052)	**0.021** (0.049)	**0.015** (0.042)	**0.009** (0.048)
	n=10	0.030 (0.073)	**0.018** (0.058)	**0.014** (0.054)	-0.045 (0.064)	**0.010** (0.045)	**0.002** (0.042)	-0.073 (0.073)	**0.004** (0.039)	**-0.007** (0.051)	-0.090 (0.074)
N=1000	n=30	0.029 (0.066)	**0.014** (0.039)	**0.015** (0.047)	**0.007** (0.059)	**0.007** (0.033)	**0.003** (0.038)	**-0.006** (0.047)	**0.006** (0.030)	**-0.003** (0.041)	**-0.013** (0.049)
	n=60	0.032 (0.067)	**0.017** (0.047)	**0.016** (0.042)	**0.013** (0.052)	**0.014** (0.040)	**0.011** (0.037)	**0.003** (0.046)	**0.011** (0.033)	**0.004** (0.035)	**-0.005** (0.044)

Note: Values in bold indicate that the absolute value based on DupER Augmentation is less than the corresponding RAW value

Correlations.

Detailed correlation results for the item guessing estimates are shown in Table 12. Median correlations for the RAW increase as sample size increases and are higher when n=30 with a range across items from .394 (N=100, n=10) to .735 (N=1000, n=30).

All DupER/EM variations have lower median correlations than the corresponding RAW data. Overall, correlations based on DupER/EM data range from .04 to .63 less than (median=.28 less than) the corresponding RAW correlations. Across testing conditions, DupER/EM variations with 20% deletion rates tend to perform best. The correlation IQRs are larger for DupER/EM than for RAW across all DupER/EM variations and testing conditions.

Like DupER/EM, all DupER/MCMC-based variations have lower median correlations than RAW across all testing conditions. Overall, median correlations based on DupER/MCMC data range from .02 to .28 less than (median=.15 less than) the corresponding RAW median correlations. Across testing conditions, DupER/MCMC median correlations tend to increase as the duplication rates decreased. The correlation IQRs are larger for DupER/MCMC than for RAW across all DupER/MCMC variations and testing conditions.

DupER/MCMC estimates have higher median correlations than those for DupER/EM for all but one DupER variation and testing condition (i.e., 50 duplications/40% deletion for the N=100/n=10 testing condition). DupER/MCMC median correlations range from .01 less to .05 greater than (median=.11 greater than) those from the corresponding DupER/EM variation.

Table 12

Correlation of item guessing estimates with the item parameters [median (interquartile range)]

Testing Condition (Sample size and test length)			DupER Variation (EM Imputation) (Duplication rate and deletion rate)									
				10 duplications			25 duplications			50 duplications		
		RAW	20%	40%	60%	20%	40%	60%	20%	40%	60%
N=100	*n*=10	0.394 (0.285)	0.351 (0.360)	0.310 (0.338)	0.290 (0.353)	0.273 (0.413)	0.275 (0.384)	0.241 (0.366)	0.233 (0.424)	0.243 (0.400)	0.236 (0.455)
	n=30	0.561 (0.090)	0.478 (0.192)	0.482 (0.175)	0.451 (0.170)	0.407 (0.204)	0.402 (0.207)	0.376 (0.204)	0.349 (0.216)	0.343 (0.211)	0.282 (0.199)
	n=60	0.535 (0.072)	0.409 (0.135)	0.412 (0.136)	0.415 (0.123)	0.349 (0.135)	0.349 (0.131)	0.345 (0.138)	0.307 (0.137)	0.311 (0.142)	0.294 (0.145)
N=250	*n*=10	0.553 (0.211)	0.377 (0.368)	0.319 (0.338)	0.272 (0.314)	0.288 (0.425)	0.221 (0.384)	0.253 (0.360)	0.230 (0.429)	0.198 (0.392)	0.252 (0.440)
	n=30	0.642 (0.080)	0.530 (0.173)	0.512 (0.174)	0.443 (0.162)	0.429 (0.203)	0.386 (0.220)	0.248 (0.179)	0.383 (0.196)	0.322 (0.199)	0.198 (0.158)
	n=60	0.605 (0.064)	0.461 (0.140)	0.458 (0.130)	0.441 (0.129)	0.395 (0.149)	0.375 (0.140)	0.305 (0.142)	0.364 (0.148)	0.333 (0.150)	0.255 (0.137)
N=500	*n*=10	0.639 (0.168)	0.387 (0.370)	0.293 (0.337)	0.263 (0.352)	0.294 (0.411)	0.233 (0.374)	0.283 (0.478)	0.247 (0.432)	0.181 (0.403)	0.259 (0.526)
	n=30	0.689 (0.079)	0.534 (0.180)	0.477 (0.172)	0.285 (0.202)	0.435 (0.200)	0.327 (0.217)	0.142 (0.117)	0.395 (0.201)	0.274 (0.199)	0.133 (0.106)
	n=60	0.652 (0.063)	0.485 (0.123)	0.454 (0.124)	0.347 (0.135)	0.415 (0.141)	0.358 (0.141)	0.203 (0.122)	0.383 (0.141)	0.314 (0.144)	0.175 (0.113)
N=1000	*n*=10	0.690 (0.142)	0.411 (0.374)	0.296 (0.346)	0.328 (0.392)	0.336 (0.402)	0.260 (0.391)	0.318 (0.485)	0.298 (0.399)	0.218 (0.390)	0.248 (0.538)
	n=30	0.735 (0.072)	0.530 (0.170)	0.415 (0.183)	0.116 (0.081)	0.441 (0.190)	0.270 (0.200)	0.110 (0.072)	0.403 (0.195)	0.229 (0.177)	0.112 (0.072)
	n=60	0.694 (0.061)	0.489 (0.128)	0.391 (0.126)	0.177 (0.111)	0.418 (0.137)	0.288 (0.130)	0.108 (0.081)	0.392 (0.138)	0.249 (0.129)	0.089 (0.074)

Testing Condition (Sample size and test length)			DupER Variation (MCMC Imputation) (Duplication rate and deletion rate)									
				10 duplications			25 duplications			50 duplications		
		RAW	20%	40%	60%	20%	40%	60%	20%	40%	60%
N=100	*n*=10	0.394 (0.285)	0.375 (0.396)	0.351 (0.411)	0.310 (0.426)	0.294 (0.418)	0.291 (0.417)	0.290 (0.412)	0.243 (0.440)	0.231 (0.404)	0.254 (0.415)
	n=30	0.561 (0.090)	0.495 (0.187)	0.499 (0.180)	0.468 (0.179)	0.419 (0.221)	0.434 (0.201)	0.430 (0.213)	0.363 (0.216)	0.373 (0.214)	0.362 (0.216)
	n=60	0.535 (0.072)	0.415 (0.134)	0.433 (0.134)	0.432 (0.133)	0.352 (0.136)	0.364 (0.143)	0.373 (0.141)	0.317 (0.144)	0.321 (0.143)	0.328 (0.149)
N=250	*n*=10	0.553 (0.211)	0.446 (0.341)	0.447 (0.325)	0.410 (0.399)	0.344 (0.361)	0.368 (0.371)	0.376 (0.429)	0.297 (0.384)	0.298 (0.392)	0.353 (0.431)
	n=30	0.642 (0.080)	0.564 (0.166)	0.579 (0.157)	0.550 (0.158)	0.472 (0.196)	0.490 (0.201)	0.464 (0.226)	0.424 (0.198)	0.434 (0.214)	0.403 (0.230)
	n=60	0.605 (0.064)	0.482 (0.128)	0.505 (0.129)	0.502 (0.124)	0.414 (0.151)	0.433 (0.139)	0.432 (0.136)	0.380 (0.146)	0.386 (0.150)	0.382 (0.152)
N=500	*n*=10	0.639 (0.168)	0.492 (0.326)	0.504 (0.310)	0.488 (0.375)	0.402 (0.341)	0.431 (0.353)	0.507 (0.381)	0.369 (0.360)	0.408 (0.341)	0.505 (0.367)
	n=30	0.689 (0.079)	0.604 (0.151)	0.625 (0.146)	0.591 (0.176)	0.520 (0.188)	0.551 (0.199)	0.473 (0.269)	0.481 (0.192)	0.488 (0.210)	0.417 (0.253)
	n=60	0.652 (0.063)	0.534 (0.118)	0.555 (0.114)	0.539 (0.126)	0.466 (0.134)	0.483 (0.136)	0.458 (0.140)	0.436 (0.138)	0.445 (0.138)	0.427 (0.143)
N=1000	*n*=10	0.690 (0.142)	0.555 (0.279)	0.580 (0.275)	0.618 (0.294)	0.509 (0.308)	0.567 (0.287)	0.629 (0.282)	0.488 (0.321)	0.566 (0.287)	0.594 (0.225)
	n=30	0.735 (0.072)	0.649 (0.146)	0.675 (0.129)	0.623 (0.223)	0.584 (0.172)	0.621 (0.179)	0.533 (0.336)	0.550 (0.185)	0.582 (0.200)	0.459 (0.318)
	n=60	0.694 (0.061)	0.584 (0.107)	0.600 (0.108)	0.552 (0.129)	0.531 (0.130)	0.548 (0.123)	0.494 (0.150)	0.503 (0.132)	0.512 (0.128)	0.463 (0.153)

Item Characteristic Curve Estimates

Detailed RMSE results for the ICC estimates are shown in Table 13. Median RMSEs for the RAW conditions tend to decrease as sample size and test length increase. These median RMSEs range from 0.030 (N=1000, n=60) to 0.075 (N=100, n=10).

All DupER/EM variations have higher median RMSEs than do the corresponding RAW data. DupER/EM median RMSEs range from 24% to 357% greater than (median=99% greater than) the corresponding RAW RMSEs. Across testing conditions, DupER/EM median RMSEs decrease as deletion and duplication rates decrease. The RMSE IQRs are larger for DupER/EM than for RAW across nearly all DupER/EM variations and testing conditions.

Like DupER/EM, all DupER/MCMC-based variations have larger median RMSEs than do RAW across all testing conditions. Overall, median RMSEs based on DupER/MCMC data range from 9% to 51% greater than (median=23% greater than) the corresponding RAW median RMSEs. DupER/MCMC median RMSEs do not very consistently across variations; DupER/MCMC[10/20] has the lowest median RMSE and DupER/MCMC[10/60] has the greatest. The RMSE IQRs are larger for DupER/MCMC than for RAW across nearly all DupER/EM variations when n=30 and n=60, but do not show a consistent pattern when n=10.

Median RMSEs for DupER/MCMC are lower than those for DupER/EM for every DupER variation and every testing condition. DupER/MCMC median RMSEs range from 2% to 75% less than (median=36% less than) those from the corresponding DupER/EM variation.

Table 13

RMSE of ICC estimates [median (interquartile range)]

Testing Condition (Sample size and test length)		DupER Variation (EM Imputation) (Duplication rate and deletion rate)									
			10 duplications			25 duplications			50 duplications		
		RAW	20%	40%	60%	20%	40%	60%	20%	40%	60%
N=100	*n*=10	0.075 (0.028)	0.116 (0.008)	0.138 (0.024)	0.165 (0.047)	0.128 (0.011)	0.148 (0.018)	0.170 (0.048)	0.133 (0.011)	0.150 (0.009)	0.167 (0.035)
	n=30	0.073 (0.020)	0.093 (0.025)	0.105 (0.025)	0.124 (0.026)	0.098 (0.025)	0.109 (0.024)	0.131 (0.024)	0.100 (0.025)	0.111 (0.027)	0.134 (0.028)
	n=60	0.069 (0.021)	0.086 (0.020)	0.090 (0.024)	0.101 (0.029)	0.092 (0.022)	0.093 (0.024)	0.105 (0.029)	0.095 (0.024)	0.095 (0.022)	0.108 (0.031)
N=250	*n*=10	0.058 (0.015)	0.095 (0.027)	0.128 (0.043)	0.164 (0.047)	0.101 (0.029)	0.130 (0.039)	0.163 (0.037)	0.103 (0.029)	0.129 (0.033)	0.158 (0.033)
	n=30	0.055 (0.011)	0.068 (0.016)	0.091 (0.018)	0.120 (0.025)	0.070 (0.015)	0.094 (0.024)	0.128 (0.026)	0.072 (0.017)	0.095 (0.025)	0.132 (0.026)
	n=60	0.051 (0.012)	0.063 (0.016)	0.078 (0.022)	0.100 (0.027)	0.065 (0.016)	0.080 (0.021)	0.108 (0.027)	0.066 (0.016)	0.081 (0.021)	0.111 (0.028)
N=500	*n*=10	0.047 (0.012)	0.084 (0.037)	0.124 (0.059)	0.161 (0.045)	0.088 (0.038)	0.124 (0.056)	0.158 (0.035)	0.090 (0.038)	0.124 (0.052)	0.157 (0.031)
	n=30	0.041 (0.008)	0.058 (0.014)	0.092 (0.021)	0.130 (0.028)	0.061 (0.014)	0.094 (0.020)	0.138 (0.029)	0.061 (0.016)	0.095 (0.020)	0.140 (0.036)
	n=60	0.039 (0.009)	0.054 (0.016)	0.082 (0.024)	0.115 (0.031)	0.055 (0.016)	0.083 (0.025)	0.120 (0.032)	0.056 (0.016)	0.083 (0.026)	0.123 (0.032)
N=1000	*n*=10	0.035 (0.008)	0.076 (0.053)	0.124 (0.067)	0.159 (0.045)	0.078 (0.052)	0.122 (0.066)	0.158 (0.043)	0.080 (0.052)	0.121 (0.062)	0.160 (0.039)
	n=30	0.031 (0.005)	0.055 (0.013)	0.094 (0.018)	0.136 (0.032)	0.056 (0.013)	0.096 (0.018)	0.139 (0.037)	0.057 (0.015)	0.098 (0.019)	0.140 (0.038)
	n=60	0.030 (0.007)	0.050 (0.017)	0.088 (0.028)	0.127 (0.036)	0.051 (0.017)	0.089 (0.028)	0.131 (0.037)	0.051 (0.017)	0.089 (0.028)	0.132 (0.038)

Testing Condition (Sample size and test length)		DupER Variation (MCMC Imputation) (Duplication rate and deletion rate)									
			10 duplications			25 duplications			50 duplications		
		RAW	20%	40%	60%	20%	40%	60%	20%	40%	60%
N=100	*n*=10	0.075 (0.028)	0.100 (0.025)	0.102 (0.026)	0.105 (0.011)	0.108 (0.011)	0.107 (0.027)	0.102 (0.017)	0.113 (0.030)	0.111 (0.009)	0.103 (0.019)
	n=30	0.073 (0.020)	0.086 (0.027)	0.085 (0.031)	0.091 (0.038)	0.090 (0.025)	0.089 (0.032)	0.089 (0.033)	0.092 (0.030)	0.092 (0.027)	0.090 (0.036)
	n=60	0.069 (0.021)	0.084 (0.023)	0.085 (0.024)	0.089 (0.028)	0.090 (0.022)	0.087 (0.024)	0.089 (0.027)	0.092 (0.025)	0.089 (0.022)	0.091 (0.025)
N=250	*n*=10	0.058 (0.015)	0.071 (0.014)	0.074 (0.015)	0.076 (0.012)	0.076 (0.029)	0.075 (0.016)	0.071 (0.009)	0.080 (0.017)	0.078 (0.033)	0.069 (0.010)
	n=30	0.055 (0.011)	0.060 (0.015)	0.061 (0.018)	0.068 (0.022)	0.063 (0.015)	0.062 (0.020)	0.065 (0.024)	0.065 (0.019)	0.064 (0.025)	0.065 (0.023)
	n=60	0.051 (0.012)	0.059 (0.012)	0.061 (0.013)	0.066 (0.015)	0.062 (0.016)	0.061 (0.015)	0.064 (0.016)	0.064 (0.016)	0.062 (0.021)	0.064 (0.014)
N=500	*n*=10	0.047 (0.012)	0.055 (0.011)	0.060 (0.012)	0.059 (0.010)	0.056 (0.038)	0.058 (0.012)	0.055 (0.009)	0.058 (0.015)	0.058 (0.052)	0.051 (0.007)
	n=30	0.041 (0.008)	0.046 (0.011)	0.049 (0.014)	0.053 (0.019)	0.048 (0.014)	0.049 (0.015)	0.051 (0.020)	0.050 (0.015)	0.049 (0.020)	0.049 (0.024)
	n=60	0.039 (0.009)	0.046 (0.010)	0.048 (0.009)	0.053 (0.010)	0.047 (0.016)	0.048 (0.010)	0.051 (0.011)	0.049 (0.012)	0.049 (0.026)	0.050 (0.011)
N=1000	*n*=10	0.035 (0.008)	0.041 (0.008)	0.044 (0.009)	0.045 (0.003)	0.041 (0.052)	0.042 (0.005)	0.041 (0.008)	0.042 (0.009)	0.042 (0.062)	0.040 (0.011)
	n=30	0.031 (0.005)	0.034 (0.009)	0.036 (0.011)	0.043 (0.017)	0.036 (0.013)	0.036 (0.012)	0.040 (0.020)	0.037 (0.009)	0.036 (0.019)	0.039 (0.019)
	n=60	0.030 (0.007)	0.034 (0.007)	0.037 (0.008)	0.042 (0.009)	0.035 (0.017)	0.036 (0.008)	0.039 (0.010)	0.036 (0.008)	0.036 (0.028)	0.039 (0.009)

Ability Estimates

Note: As mentioned in Chapter 3, RMSEs, bias and correlations for ability estimates are calculated based on the total number of simulees across all (converging) replications, as opposed to calculating the statistics for each replication (or some other subset of results).

RMSE.

Detailed RMSE results for the person ability estimates are shown in Table 14. RMSEs for the RAW conditions decrease as text length and sample size increase. RMSEs calculated across all replications range from 0.336 (N=1000, n=60) to 0.705 (N=100, n=10).

All DupER/EM variations have higher RMSEs than do the corresponding RAW data. DupER/EM RMSEs range from 0.5% to 19.0% greater than (median=4.9% greater than) the corresponding RAW RMSEs. The best performing DupER/EM variations across testing conditions are those with a deletion rate of 20%.

RMSEs for DupER/MCMC are higher than those for RAW across all three test lengths at the smallest three sample sizes. For the N=1000 sample sizes, results are mixed, with DupER/MCMC[25/20] resulting in a lower RMSE than the RAW data when n=30 and DupER/MCMC[50/20] resulting in lower RMSEs than the RAW data at n=30 and n=60. DupER/MCMC RMSEs range from 0.1% less than to 8.2% greater than (median=2.0% greater than) the corresponding RAW RMSEs. The best-performing DupER variations are DupER/MCMC variations with a deletion rate of 20%. These variations perform similarly to (and sometimes outperforming) RAW.

RMSE comparisons between DupER/MCMC and DupER/EM yield mixed results. RMSEs for DupER/EM tend to be lower than those of DupER/MCMC for most DupER

variations for the N=100/n=60 and N=250/n=60 testing conditions. In the other testing conditions DupER/MCMC tends to outperform DupER/EM across DupER variations. DupER/MCMC RMSEs range from 5.8% greater to 14.5% less than (median=2.9% less than) those from the corresponding DupER/EM variation.

Table 14

RMSE of ability estimates

Testing Condition (Sample size and test length)		DupER Variation (EM Imputation) (Duplication rate and deletion rate)									
			10 duplications			25 duplications			50 duplications		
		RAW	20%	40%	60%	20%	40%	60%	20%	40%	60%
N= 100	*n*= 10	0.705	0.744	0.769	0.798	0.752	0.775	0.811	0.755	0.784	0.813
	n= 30	0.463	0.470	0.475	0.491	0.476	0.480	0.499	0.486	0.483	0.503
	n=60	0.356	0.368	0.366	0.364	0.373	0.369	0.367	0.377	0.369	0.371
N= 250	*n*=10	0.684	0.709	0.735	0.778	0.713	0.736	0.782	0.715	0.738	0.801
	n=30	0.446	0.449	0.460	0.482	0.450	0.462	0.490	0.451	0.464	0.493
	n=60	0.344	0.347	0.350	0.360	0.348	0.352	0.368	0.349	0.353	0.370
N= 500	*n*=10	0.677	0.696	0.714	0.765	0.698	0.715	0.773	0.700	0.722	0.783
	n=30	0.439	0.443	0.459	0.490	0.443	0.461	0.498	0.443	0.462	0.501
	n=60	0.339	0.340	0.353	0.375	0.341	0.355	0.381	0.341	0.355	0.383
N=1000	*n*=10	0.673	0.689	0.704	0.757	0.690	0.704	0.768	0.690	0.705	0.777
	n=30	0.435	0.440	0.462	0.502	0.441	0.464	0.507	0.441	0.465	0.509
	n=60	0.336	0.339	0.362	0.393	0.340	0.363	0.398	0.340	0.363	0.399

Testing Condition (Sample size and test length)		DupER Variation (MCMC Imputation) (Duplication rate and deletion rate)									
			10 duplications			25 duplications			50 duplications		
		RAW	20%	40%	60%	20%	40%	60%	20%	40%	60%
N= 100	*n*= 10	0.705	0.719	0.724	0.723	0.726	0.729	0.721	0.728	0.731	0.717
	n= 30	0.463	0.467	0.474	0.495	0.471	0.471	0.483	0.473	0.472	0.482
	n=60	0.356	0.368	0.371	0.385	0.372	0.372	0.377	0.374	0.371	0.375
N= 250	*n*=10	0.684	0.693	0.701	0.700	0.695	0.701	0.697	0.696	0.701	0.695
	n=30	0.446	0.448	0.454	0.471	0.448	0.451	0.465	0.448	0.450	0.462
	n=60	0.344	0.351	0.356	0.369	0.351	0.353	0.360	0.351	0.352	0.356
N= 500	*n*=10	0.677	0.683	0.691	0.690	0.684	0.690	0.689	0.684	0.689	0.689
	n=30	0.439	0.440	0.446	0.461	0.439	0.443	0.456	0.439	0.442	0.454
	n=60	0.339	0.342	0.347	0.357	0.341	0.344	0.350	0.341	0.342	0.347
N=1000	*n*=10	0.673	0.678	0.686	0.687	0.678	0.684	0.688	0.678	0.683	0.689
	n=30	0.435	0.436	0.441	0.454	**0.435**	0.439	0.451	**0.435**	0.438	0.449
	n=60	0.336	0.337	0.340	0.348	0.336	0.338	0.343	**0.335**	0.337	0.341

Note: Values in bold indicate that the value based on DupER Augmentation is lower than the corresponding RAW value

Bias.

Detailed bias results for the person ability estimates are shown in Table 15. Bias for the RAW conditions tends to decrease as sample size increases, but does not consistently

vary with test length. The RAW ability estimates are uniformly overestimated across testing conditions. Bias calculated across all replications ranges from 0.027 (N=1000, n=10) to 0.102 (N=100, n=10).

DupER/EM variations tend to be less biased than the corresponding RAW data across most testing conditions. For the N=500/n=10 and N=1000/n=10 testing conditions, RAW outperforms DupER/EM when deletion rates are 20% or 60% but not when 40% percent deletion is used. Overall, bias based on DupER/EM data ranges from 99% less than to 511% greater than (median=41% less than) the corresponding RAW bias (in absolute terms). The best performing DupER/EM variations across testing conditions are those with a deletion rate of 40%.

Nearly all DupER/MCMC variations are less biased than the corresponding RAW data for the 20% and 40% deletion DupER/MCMC variations. RAW outperforms all DupER/MCMC variations for the N=250/n=60 testing condition. Overall, bias based on DupER/MCMC data ranges from 100% less than to 307% greater than (median=26% less than) the corresponding RAW bias (in absolute terms). Across testing conditions, the 20% and 40% deletion DupER/MCMC variations performed the best.

DupER/MCMC estimates tend to be less biased than those for DupER/EM for most DupER variations and testing conditions when n=10 and for the N=1000/n=30 testing condition. The situation is reversed for the remaining testing conditions. DupER/MCMC bias ranges from 99% less than to 10,056% greater than (median=29% greater than) those from the corresponding DupER/EM variation.

Table 15

Bias of ability estimates

Testing Condition (Sample size and test length)			DupER Variation (EM Imputation) (Duplication rate and deletion rate)								
				10 duplications			25 duplications			50 duplications		
		RAW	20%	40%	60%	20%	40%	60%	20%	40%	60%
N= 100	*n*= 10	0.102	**0.039**	**0.011**	**-0.051**	**0.028**	**0.006**	**-0.074**	**0.020**	**-0.015**	**-0.072**
	n= 30	0.097	**0.052**	**0.029**	**-0.015**	**0.050**	**0.024**	**-0.023**	**0.045**	**0.022**	**-0.015**
	n=60	0.085	**0.071**	**0.052**	**0.020**	**0.076**	**0.049**	**0.002**	**0.070**	**0.043**	**-0.005**
N= 250	*n*=10	0.061	**0.035**	**0.010**	-0.099	**0.034**	**0.011**	-0.111	**0.033**	**0.009**	-0.119
	n=30	0.070	**0.046**	**0.030**	**-0.004**	**0.043**	**0.029**	**0.008**	**0.042**	**0.029**	**0.018**
	n=60	0.064	**0.045**	**0.014**	**-0.032**	**0.043**	**0.009**	**-0.039**	**0.040**	**0.006**	**-0.040**
N= 500	*n*=10	0.041	0.043	**0.023**	-0.113	0.043	**0.023**	-0.126	0.043	**0.019**	-0.132
	n=30	0.057	**0.043**	**0.037**	**0.016**	**0.041**	**0.038**	**0.038**	**0.039**	**0.041**	**0.044**
	n=60	0.057	**0.031**	**0.004**	**-0.040**	**0.027**	**0.001**	**-0.038**	**0.024**	**-0.001**	**-0.037**
N=1000	*n*=10	0.027	0.044	**0.021**	-0.133	0.043	**0.017**	-0.153	0.042	**0.018**	-0.163
	n=30	0.047	**0.041**	**0.042**	**0.042**	**0.039**	**0.046**	0.053	**0.038**	0.048	0.056
	n=60	0.049	**0.023**	**0.006**	**-0.033**	**0.018**	**0.004**	**-0.029**	**0.015**	**0.004**	**-0.028**

Testing Condition (Sample size and test length)			DupER Variation (MCMC Imputation) (Duplication rate and deletion rate)								
				10 duplications			25 duplications			50 duplications		
		RAW	20%	40%	60%	20%	40%	60%	20%	40%	60%
N= 100	*n*= 10	0.102	**0.034**	**0.030**	**0.012**	**0.021**	**0.023**	**-0.008**	**0.018**	**0.020**	**-0.018**
	n= 30	0.097	**0.068**	**0.072**	**0.079**	**0.066**	**0.061**	**0.066**	**0.064**	**0.057**	**0.059**
	n=60	0.085	**0.084**	0.090	0.095	0.086	0.087	0.087	**0.082**	**0.082**	**0.082**
N= 250	*n*=10	0.061	**0.028**	**0.035**	**-0.017**	**0.027**	**0.028**	**-0.046**	**0.024**	**0.022**	-0.066
	n=30	0.070	**0.059**	**0.063**	0.074	**0.054**	**0.052**	**0.058**	**0.050**	**0.047**	**0.049**
	n=60	0.064	0.074	0.081	0.089	0.071	0.073	0.074	0.067	0.068	0.064
N= 500	*n*=10	0.041	**0.024**	**0.030**	-0.045	**0.021**	**0.019**	-0.077	**0.017**	**0.011**	-0.098
	n=30	0.057	**0.047**	**0.051**	0.058	**0.041**	**0.039**	0.045	**0.038**	**0.033**	**0.034**
	n=60	0.057	0.062	0.067	0.072	0.056	0.058	0.055	**0.052**	**0.052**	**0.045**
N=1000	*n*=10	0.027	**0.013**	**0.014**	-0.077	**0.008**	**0.000**	-0.108	**0.005**	**-0.008**	-0.125
	n=30	0.047	**0.035**	**0.035**	**0.040**	**0.028**	**0.023**	**0.025**	**0.024**	**0.017**	**0.017**
	n=60	0.049	**0.047**	0.051	0.051	**0.041**	**0.041**	**0.036**	**0.037**	**0.035**	**0.028**

Note: Values in bold indicate that the absolute value based on DupER Augmentation is less than the corresponding RAW value

Correlations.

Detailed correlation results for the person ability estimates are shown in Table 16. Correlations for the RAW conditions increase as sample size and test length increase. Median correlations calculated across all replications range from .717 (N=100, n=10) to .944 (N=1000, n=60).

All DupER/EM variations have lower correlations than the corresponding RAW data. Overall, correlations based on DupER/EM data range from .002 to .060 less than (median=.014 less than) the corresponding RAW correlations. Across testing conditions, DupER/EM variations with 20% deletion rates tend to have correlations that are closer to the RAW correlations than the other DupER/EM variations.

Like DupER/EM, all DupER/MCMC-based variations have lower correlations than RAW across all testing conditions. Overall, correlations based on DupER/MCMC data range from .0003 to .034 less than (median=.003 less than) the corresponding RAW correlations. All DupER/MCMC variations were similarly close to the RAW correlations.

DupER/MCMC estimates have higher correlations than those for DupER/EM for every DupER variation and every testing condition. DupER/MCMC correlations range from .001 to .039 greater than (median=.011 greater than) those from the corresponding DupER/EM variation.

Table 16

Correlation of ability estimates with the ability parameters

Testing Condition (Sample size and test length)		DupER Variation (EM Imputation) (Duplication rate and deletion rate)									
			10 duplications			25 duplications			50 duplications		
		RAW	20%	40%	60%	20%	40%	60%	20%	40%	60%
N= 100	*n*= 10	0.717	0.685	0.667	0.666	0.679	0.662	0.664	0.676	0.656	0.670
	n= 30	0.893	0.884	0.881	0.872	0.881	0.879	0.870	0.876	0.877	0.867
	n=60	0.939	0.932	0.932	0.932	0.931	0.931	0.931	0.929	0.931	0.930
N= 250	*n*=10	0.731	0.710	0.691	0.688	0.707	0.690	0.690	0.705	0.688	0.683
	n=30	0.899	0.895	0.889	0.878	0.894	0.888	0.873	0.893	0.887	0.871
	n=60	0.942	0.939	0.938	0.935	0.938	0.937	0.934	0.938	0.937	0.933
N= 500	*n*=10	0.737	0.721	0.707	0.699	0.719	0.705	0.698	0.718	0.698	0.695
	n=30	0.901	0.898	0.890	0.873	0.897	0.889	0.868	0.897	0.888	0.867
	n=60	0.943	0.941	0.938	0.933	0.941	0.938	0.931	0.941	0.938	0.930
N=1000	*n*=10	0.741	0.728	0.715	0.707	0.728	0.715	0.705	0.727	0.713	0.704
	n=30	0.902	0.899	0.889	0.867	0.899	0.888	0.864	0.899	0.888	0.863
	n=60	0.944	0.942	0.937	0.928	0.942	0.936	0.926	0.942	0.936	0.925

Testing Condition (Sample size and test length)		DupER Variation (MCMC Imputation) (Duplication rate and deletion rate)									
			10 duplications			25 duplications			50 duplications		
		RAW	20%	40%	60%	20%	40%	60%	20%	40%	60%
N= 100	*n*= 10	0.717	0.696	0.690	0.690	0.689	0.685	0.693	0.688	0.682	0.697
	n= 30	0.893	0.887	0.886	0.880	0.885	0.886	0.884	0.883	0.885	0.884
	n=60	0.939	0.934	0.934	0.933	0.932	0.933	0.934	0.931	0.932	0.933
N= 250	*n*=10	0.731	0.721	0.716	0.716	0.719	0.716	0.720	0.717	0.716	0.722
	n=30	0.899	0.897	0.897	0.894	0.896	0.897	0.895	0.896	0.896	0.896
	n=60	0.942	0.940	0.940	0.939	0.939	0.940	0.940	0.939	0.940	0.940
N= 500	*n*=10	0.737	0.731	0.729	0.728	0.730	0.729	0.731	0.729	0.729	0.732
	n=30	0.901	0.900	0.900	0.898	0.900	0.900	0.899	0.900	0.900	0.899
	n=60	0.943	0.942	0.942	0.941	0.942	0.942	0.942	0.942	0.942	0.942
N=1000	*n*=10	0.741	0.738	0.737	0.737	0.738	0.738	0.738	0.738	0.738	0.739
	n=30	0.902	0.902	0.902	0.901	0.902	0.902	0.901	0.902	0.902	0.901
	n=60	0.944	0.943	0.943	0.943	0.943	0.943	0.943	0.943	0.943	0.943

Binomial Tests

To help summarize results for the evaluative criteria, RAW/DupER comparisons are tabulated for the three main "families" of evaluative statistics (i.e., RMSE, bias, and correlation). There are 60 RMSE comparisons (i.e., RMSEs for discrimination, difficulty, guessing, ICC, and ability estimates), 48 bias comparisons (i.e., bias for discrimination, difficulty, guessing, and ability estimates), and 48 correlation comparisons (i.e., correlations for discrimination, difficulty, guessing, and ability estimates) between results obtained from data subjected to DupER Augmentation and the raw data for each DupER variation. If RAW and DupER performed equally well, one would expect DupER to outperform RAW about 50% of the time simply do to chance. Therefore, for each family of evaluative statistics, a binomial test is conducted to determine if the percent of comparisons in which DupER outperformed RAW is significantly greater than 50%. One-tailed tests are used because of the a priori hypothesis that DupER would outperform RAW.

Detailed results are shown in Table 17. DupER/EM does not outperform RAW in any of the median RMSE or correlation comparisons and DupER/MCMC does so only occasionally. However, many DupER variations frequently outperform RAW in terms of bias. For example, DupER/MCMC[50/40] results in less median bias than RAW in 75.0% of comparisons; this percentage is significantly greater than 50% ($p < .001$). All DupER/MCMC variations except for DupER/MCMC[10/60] are significantly better performing in terms of median bias than RAW. Additionally, all DupER/EM variations with 20% deletion rates as well as DupER/EM[10/40] and DupER/EM[25/40] are significantly

Table 17

Results summary: Percent of comparisons where DupER outperformed RAW, by statistic type and DupER variation

RMSE (60 comparisons)[#]

Duplication Rate	Deletion Rate	Imputation type: EM	Imputation type: MCMC
10	20%	0.0	1.7
	40%	0.0	1.7
	60%	0.0	1.7
25	20%	0.0	5.0
	40%	0.0	3.3
	60%	0.0	6.7
50	20%	0.0	6.7
	40%	0.0	5.0
	60%	0.0	8.3

[#]None of the values are significantly greater than 50% at the α=.05 level

Bias (48 comparisons)

Duplication Rate	Deletion Rate	Imputation type: EM	Imputation type: MCMC
10	20%	72.9**	66.7*
	40%	62.5	64.6*
	60%	39.6	58.3
25	20%	72.9**	62.5
	40%	62.5	66.7*
	60%	35.4	70.8**
50	20%	72.9**	66.7*
	40%	60.4	75.0***
	60%	31.3	68.8**

* significantly greater than 50% at the α=.05 level
** significantly greater than 50% at the α=.01 level
*** significantly greater than 50% at the α=.001 level

Correlation (48 comparisons)[#]

Duplication Rate	Deletion Rate	Imputation type: EM	Imputation type: MCMC
10	20%	0.0	2.1
	40%	0.0	2.1
	60%	0.0	2.1
25	20%	0.0	0.0
	40%	0.0	0.0
	60%	0.0	2.1
50	20%	0.0	0.0
	40%	0.0	0.0
	60%	0.0	6.3

[#]None of the values are significantly greater than 50% at the α=.05 level

Table 18. Results summary: Percent of comparisons where DupER/MCMC outperformed DupER/EM, by statistic type and DupER variation

RMSE (60 comparisons)

Duplication Rate	Deletion Rate	%
10	20%	80.0***
	40%	88.3***
	60%	90.0***
25	20%	88.3***
	40%	95.0***
	60%	96.7***
50	20%	93.3***
	40%	95.0***
	60%	98.3***

*** significantly different from 50% at the α=.001 level

Bias (48 comparisons)

Duplication Rate	Deletion Rate	%
10	20%	31.3*
	40%	45.8
	60%	64.6
25	20%	33.3*
	40%	52.1
	60%	72.9**
50	20%	37.5
	40%	58.3
	60%	79.2***

* significantly different from 50% at the α=.05 level
*** significantly different from 50% at the α=.001 level

Correlation (48 comparisons)

Duplication Rate	Deletion Rate	%
10	20%	97.9***
	40%	100.0***
	60%	100.0***
25	20%	93.8***
	40%	100.0***
	60%	100.0***
50	20%	93.8***
	40%	95.8***
	60%	100.0***

*** significantly different from 50% at the α=.001 level

better performing in terms of median bias than RAW. (Note: For bias, DupER values of 66.7% or greater are significantly greater than 50%, and DupER values of 33.3% or less are significantly less than 50% when using a two-tailed test.)

Similar comparisons are made between DupER/EM and DupER/MCMC. Binomial tests are conducted to determine if the percent of comparisons where DupER/MCMC outperforms DupER/EM is significantly different from 50%. Two-tailed tests are used because there is not an a priori hypothesis as to which DupER imputation model is superior.

Detailed results are shown in Table 18. All DupER/MCMC variations are significantly better in terms of median RMSE and correlation than the corresponding DupER/EM variations, with DupER/MCMC outperforming DupER/EM in 80.0% to 100.0% of the comparisons. Results are mixed for median bias. DupER/EM significantly outperforms DupER/MCMC for the [10/20] and [25/20] conditions. DupER/MCMC significantly outperforms DupER/EM for the [25/60] and [50/60] conditions. For the other DupER variations, differences are not statistically significant.

Chapter 5. Discussion

Discussion of Findings

It was hypothesized that item and ability estimates obtained using DupER Augmentation would be more accurate than those obtained by analyzing raw data alone across most testing conditions (especially those with small sample sizes). It is difficult to draw strong conclusions regarding the efficacy of DupER Augmentation due to the mixed and sometimes contradictory nature of the results. Deciding whether or not DupER Augmentation is a valuable procedure likely will depend on one's judgment as to the relative importance of the evaluative criteria. It is clear that all DupER Augmentation variations tend to result in higher median RMSEs and lower correlations for both item and ability estimates. In this respect it could be said that DupER augmented data tend to result in less precise estimates than analyzing raw data alone. On the other hand, many DupER variations produce less biased estimates of both item and ability parameters. In this respect it could be said that DupER augmented data tend to result in more accurate estimates than analyzing raw data alone. However, it should also be noted that the aggregation of results across items might oversimplify the effect of DupER on precision and accuracy. That is, the conflicting RMSE/bais conclusions may be the result of taking medians of the results across items and examinees. For example, median bias may appear low if some items with strongly negatively biased estimates “cancel out” other items with strongly positively biased estimates. In some ways, the absolute nature (due to squaring deviations) of RMSE calculations helps avoid this situation.

If one's primary concern is estimating item parameters with the minimum amount of error, DupER Augmentation should not be used. When estimating the median RMSEs of the ICCs (perhaps the best measure of item estimation because it incorporates all 3PL item parameters) all DupER variations produced less precise estimates than analyzing the raw data alone. The best performing DupER variation for this evaluative index was DupER/MCMC[10/20].

However, if one's primary concern is accurately estimating ability parameters, DupER Augmentation may be a viable alternative. Consider DupER/MCMC[50/20]: This DupER variation tends to result in ability RMSEs that are greater than analyzing the raw data alone. Similarly, correlations between the estimated and true ability values are consistently lower for DupER/MCMC[50/20] than for RAW. However, in both cases, the magnitudes of the differences are minimal. The difference in correlation is at most .029 (N=100/n=10). Contrast this small difference to the large reductions in bias seen by implementing DupER/MCMC[50/20]. In absolute terms, bias is reduced by up to 82% (N=100/n=10). For the one testing condition where bias increases (N=250/n=60), the bias is only 4% greater for DupER/MCMC than for RAW. Similar results are seen for DupER/MCMC[50/40]. The case could be made that these reductions in bias are sufficiently large to justify the small increases in RMSEs and small decreases in correlations.

An easier case to make is that for DupER/MCMC over DupER/EM. Although DupER/EM often produces excellent results in terms of median bias/bias (sometimes superior to DupER/MCMC), median RMSE/RMSE and median correlation/correlation (recall, for item statistics the median of the evaluative criteria is taken across items) results

are always worse than with RAW and usually worse than with DupER/MCMC. Additionally, while DupER/EM produces ability estimates that were similar to those found using RAW, item parameter estimates are uniformly poor across testing conditions as are convergence rates.

The second hypothesis dealt with the effects of duplication and deletion rates. It was hypothesized that DupER Augmentation would perform better with high duplication rates (i.e., 50 duplications per examinee) and moderate deletion rates (i.e. 40% deletion). Although DupER/MCMC[50/40] does prove to be one of the better performing DupER variations, there is not a clearly identifiable best DupER variant across the evaluative criteria.

To better understand the differences in ability estimation between DupER and RAW the RMSEs of ability estimates are plotted as a function of their true values. Figure 6 illustrates a comparison of the RMSEs of ability estimates for DupER/MCMC[50/20] and RAW across the ability range for all testing conditions. For both analyses, ability estimation is worse for low and high abilities, and is best near the mean of zero. One general trend that can be observed in the graphs is that DupER/MCMC[50/20] tends to result in lower RMSEs for low ability levels than RAW, while RAW tends to perform better for high ability levels (the two procedures are very similar near the mean ability level). This trend is more pronounced at the n=100 and n=250 sample sizes.

Figure 7 illustrates a comparison of the bias of ability estimates for DupER/MCMC[50/20] and RAW across ability range for all testing conditions. Again, the graphs show that ability estimation is worse for low and high abilities, and is best near the mean. More visible in these graphs is the tendency for estimates to regress toward the mean.

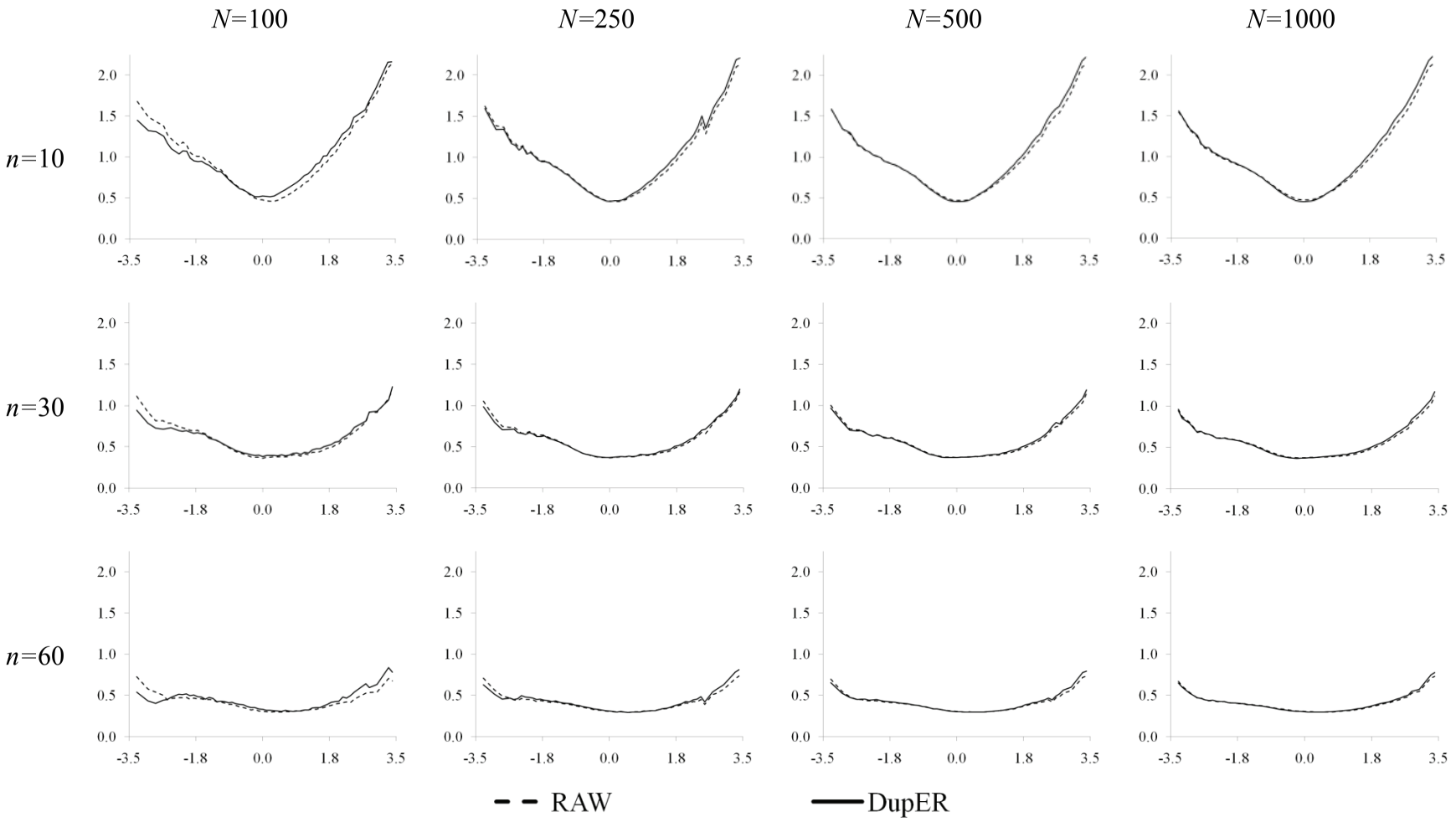

Figure 6. RMSE of ability estimates for DupER/MCMC[50/20] and RAW across ability range for all testing conditions

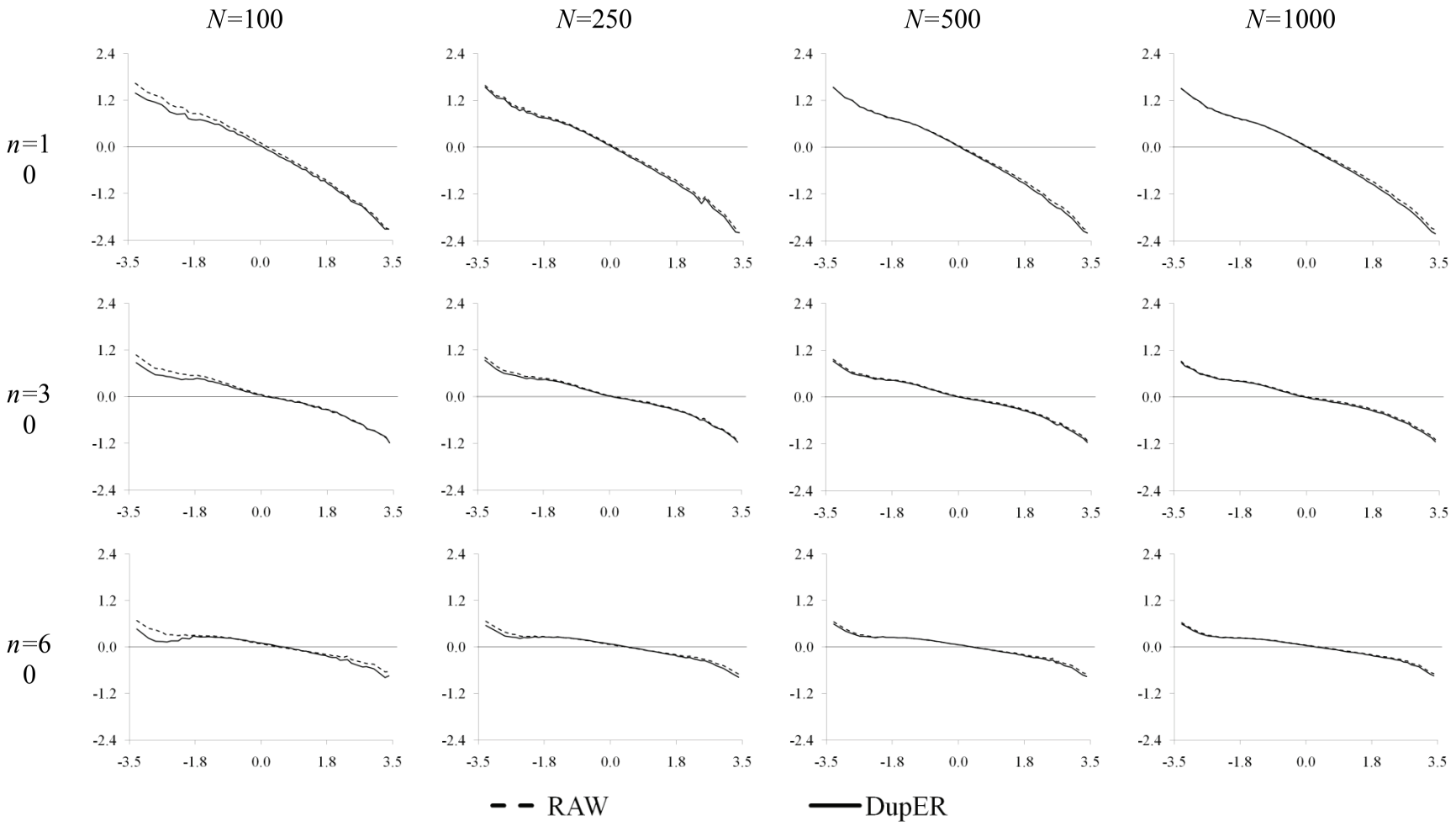

Figure 7. Bias of ability estimates for DupER/MCMC[50/20] and RAW across ability range for all testing conditions

That is, abilities are overestimated at the low end of the scale and underestimated at he high end of the scale. DupER/MCMC[50/20] tends to result in less biased estimates at low ability levels than RAW, while RAW tends to perform better for high ability levels. Again, the two procedures are very similar near the mean ability level. This trend is most visible at the n=100 sample sizes.

The final hypothesis was that DupER Augmentation would not have as great of an effect on the accuracy of person ability estimates as on item parameter estimates. This is not borne out in the data. Several variations of DupER Augmentation are effective in reducing bias for both item and ability estimates. However, DupER Augmentation is not effective in reducing RMSEs or increasing estimate-parameter correlations for either item or ability estimates (although RAW and DupER results are very similar in several situations).

Integration and Implication of Findings

The promising results of an earlier pilot study (i.e., Foley, 2009) are not completely reproduced in this expanded and more comprehensive study. In the pilot study, DupER consistently outperformed RAW in terms of RMSEs and correlations. In the current study, the reverse is true. One possible explanation for this is differences in the estimation software and settings. The current study used BILOG with priors on the item parameters; the pilot study used PARSCALE with no priors. Without priors, parameter estimates sometimes converge to unrealistic values (e.g., discrimination estimates greater than 10 or difficulties less than -4). It may be that in the pilot study (i.e., without priors) DupER reduced the frequency of these unrealistic estimates, but in the current study (i.e., with priors) unrealistic estimates were already being prevented through the use of priors. Another possible

explanation for differences in results is that in the current study nonconverging replications are removed from subsequent analyses; no such constraints were applied in the pilot study. Because convergence information was not maintained for the pilot study, is difficult to draw firm conclusions. However, judging by the convergence rates seen in the present study (see Table 2), it is likely that there were many nonconverging replications that may have tainted the overall conclusions of the pilot study.

One result from the pilot study that was replicated was the fact that several variations of DupER result in less biased item parameter estimates. Additionally, it was found that DupER resulted in less biased ability estimates (a criterion not evaluated in the pilot study). There are several methodologies in the literature that have been shown to improve IRT estimates (e.g., simplified/modified models, optimal examinees, and prior information about items or examinees). While DupER Augmentation may not be a revolutionary new estimation procedure, the reductions in bias it provides may contribute evolutionary gains in estimation accuracy, especially when used in concert with other established techniques.

Limitations

It should be noted that this study has several important limitations. First, like all simulation studies, care should be taken in generalizing these results to other testing conditions. These results are based on one set of assessment items chosen from a single set of distributions. Different items may have behaved differently. Similarly, only one distribution of examinees was examined. A normal distribution may be an accurate portrayal of some populations, but definitely not all. For example, the population of test takers for a licensure test may be skewed left, that is, many high ability candidates and only a few low

ability candidates. Different distributions of subject ability may affect the results of DupER Augmentation. In this study, all of the items, by design, fit the 3PL model well. In real-world situations, data tend to be messier, with some items being a poor fit to the 3PL model. If this is the case, again, DupER Augmentation may perform differently. Additionally, this study may not generalize to other IRT models. Computationally, no boundaries were placed on item or person parameter estimates, so the results may be subject to distortion due to the presence of extreme values. However, attempts were made to limit the influence of outliers through the use of prior distributions and by using the median (as opposed to the mean) where possible.

A barrier to replicating and expanding this research is the amount of computational power necessary to conduct the simulations. Data generation and analyses used for this study took between four and five computer-years on dedicated machines using high-end consumer processors (i.e., from the Intel Pentium 4, Core2, and i7 processor families) that were available during the window of time in which the study was conducted. Although it is doubtless that the amount of computational time could have been reduced somewhat through coding efficiencies, the vast majority of computational time was spent on the imputation of missing values created through the DupER Augmentation process. The MI procedure in SAS version 9.1 does not appear to take advantage of processors with multiple processing cores. It is hoped that future versions of the software are optimized to take advantage of the processing power of multi-cored of processors (to the extent that the estimation algorithms allow) in order to make large-scale simulation studies using the imputation procedures more practicable.

Recommendations for Further Study

Additional work is needed in the following areas:

- This study examined only simulated data. Applying DupER Augmentation to subsamples of real data sets could show how DupER performs in real data situations.
- It would be beneficial to examine a wider and/or finer range of experimental conditions. For example, it would be interesting to see the effect (if any) of very large duplication rates combined with very small deletion rates. Also, DupER may prove to be effective when applied to different IRT models such as the 2PL or polytomous models.
- It may be possible to determine better prior distributions for item parameters based on the original raw data and thereby improve the efficacy of DupER.
- Since it has been shown that the accuracy of item parameter estimates is affected by the distribution of examinee abilities, the effects of DupER Augmentation on estimation when ability distributions are non-normal (e.g., skewed left, bimodal, etc.). These distributions should be examined because they may be more typical for testing situations were sample sizes are small (e.g., licensure testing with a high-ability population of examinees). Different distributions of item parameters should be examined as well. In testing situations where a pass/fail decisions are made, items are often chosen to obtain the most accurate estimates near the cut-point (rather than being uniformly distributed, as was the case in this study).

- In this study, DupER Augmentation was applied to simulated assessment data after it had been "scored" (i.e., coded either right or wrong). A more general framework for DupER would involve applying the augmentation to the original, unscored responses (e.g., specific answer choices on a multiple-choice test).
- This study utilized and found very different results across two imputation methods; many others imputation methods exist (e.g., mean, hot-deck, etc.). Other imputation methods should be examined to determine their effect on the efficacy of DupER Augmentation.
- While this study examined the effect of DupER Augmentation on item and ability estimates, it is also important to understand its effect on standard errors, confidence intervals, and fit statistics.
- The original, complete response vectors were not used in estimating the item parameters for any of the DupER variations examined in this study. It may be that including the original data along with the augmented data would result in improved estimates.
- Using auxiliary variables (e.g., demographics, etc.) in the imputation process has been shown to "mitigate (or eliminate) bias and … improve power" (Enders, 2010, p. 128). While these benefits typically are found in the context of traditional missing data problems, the effect of auxiliary variables on the performance of DupER should be examined.

- Finally, as stated earlier, there are many existing strategies for improving item estimation with small sample sizes (e.g., optimal examinees, prior information about items, etc.). DupER Augmentation should be used with these strategies to examine whether DupER provides an additive effect.

The current study shows that the performance of DupER Augmentation is mixed. DupER augmented data tend to result in IRT estimates (for both for items and people) with higher RMSEs but less bias. Given these results, the future prospects for DupER are unclear. However, if a variation of DupER Augmentation can be found that proves to be successful across a wide range of testing conditions, it could be valuable to the educational community in several ways. First it could reduce pre-testing costs, since smaller samples would be sufficient. Next, it could help improve test security by reducing item exposure (fewer examinees need to see each item to estimate the item parameters accurately). Finally, practitioners could use the flexible 3PL IRT model in situations with small populations or calibration samples.

References

Abdel-fattah, A.-f. A. (April, 1994). *Comparing BILOG and LOGIST estimates for normal, truncated normal and beta ability distributions.* Paper presented at the Annual Meeting of the American Educational Research Association, New Orleans, LA.

Allison, P. D. (2001). *Missing data* (Vol. 136). Thousand Oaks, CA: Sage Publications, Inc.

Assessment Systems Corporation. (1987). MICROCAT: A computer program for computerized adaptive testing (2nd ed.) [Computer Software]: Assessment Systems Corporation.

Assessment Systems Corporation (1989). MICROCAT: A computer program for computerized adaptive testing (3rd ed.) [Computer Software]: Assessment Systems Corporation.

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

Baker, F. B. (1987). Methodology review: Item parameter estimation under the one-, two-, and three-parameter logistic models. *Applied Psychological Measurement, 11*, 111-141.

Baker, F. B. (1993). EQUATE 2.0: A computer program for the characteristic curve method of IRT equating. *Applied Psychological Measurement, 17*, 20.

Barnes, L. L. B., & Wise, S. L. (1991). The utility of a modified one-parameter IRT model with small samples. *Applied Measurement in Education, 4*, 143-157.

Berger, M., King, C.Y.J., & Wong, W. (2000). Minimax d-optimal designs for item response theory models. *Psychometrika, 65*, 377-390.

Bock, R., & Aitkin, M. (1981). Marginal maximum likelihood estimation of item parameters: Application of an EM algorithm. *Psychometrika, 46*, 443-459.

Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd edition). Hillsdale, NJ: Erlbaum.

Croll, P. R., & Urry, V. W. (1978). ANCILLES: Item parameter estimation program with normal ogive and logistic three-parameter model options - Version 78.5 [Computer Software]. Washington, D.C.: U.S. Civil Service Commission, Personnel Research and Development Center.

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

DeMars, C. (2001). Group differences based on IRT scores: Does the model matter? *Educational and Psychological Measurement, 61*, 60-70.

Efron, B. (1982). *The jackknife, the bootstrap and other resampling plans*. Philadelphia, PA: Society for Industrial and Applied Mathematics.

Efron, B., & Tibshirani, R. J. (1994). *An introduction to the bootstrap*. London: CRC Press.

Enders, C.K. (2005, April 20). Multiple imputation. Lecture presented in Educational Psychology 995. University of Nebraska, Lincoln, NE.

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

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

Foley, B. F. (2009, April). *Improving IRT item parameter estimates with small sample sizes: Evaluating the efficacy of a new data augmentation technique*. Paper presented at the annual meeting of the National Council on Measurement in Education. San Diego, CA.

Geisinger, K. F., Wells, C. S., & Foley, B. P. (2007). *Initial report to the Florida Department of Education: Recommendations on FCAT for 2007-2008*. (Technical report). Lincoln, NE: Buros Center for Testing.

Gifford, J., & Swaminathan, H. (1990). Bias and the effect of priors in Bayesian estimation of parameters of item response models. *Applied Psychological Measurement, 14*, 33-43.

Gierl, M. J., & Ackerman, T. (1996). Software review: XCALIBRE™ marginal maximum-likelihood estimation program, Windows version 1.10. *Applied Psychological Measurement, 20*, 303-307.

Hambleton, R. K. (1993). Principles and selected applications of item response theory. In R. Linn (Ed.), *Educational measurement* (3rd ed., pp. 147-200). Washington, D.C.: American Council on Education.

Hambleton, R. K., & Cook, L. L. (1983). Robustness of item response models and effects of test length and sample size on the precision of ability estimates. In D. J. Weiss & R. D. Bock (Eds.), *New horizons in testing: Latent trait test theory and computerized adaptive testing* (pp. 31-49). New York: Academic Press.

Hambleton, R. K., & Jones, R. W. (1993). An NCME instructional module on comparison of classical test theory and item response theory and their applications to test development. *Educational Measurement: Issues and Practice, 12*, 38-47.

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

Harwell, M. R., & Janosky, J. E. (1991). An empirical study of the effects of small datasets and varying prior variances on item parameter estimation in BILOG. *Applied Psychological Measurement, 15*, 279-291.

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

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

Hwang, D.-Y. (2002). *Classical test theory and item response theory: Analytical and empirical comparisons*. Paper presented at the Annual Meeting of the Southwest Educational Research Association, Austin, TX.

Jensema, C. (1972). *An application of latent trait mental test theory to the Washington Pre-College Testing Battery (Research Bulletin)*. Seattle, WA: University of Washington, Bureau of Testing.

Jensema, C. (1976). A simple technique for estimating latent trait mental test parameters. *Educational and Psychological Measurement, 36*, 705-715.

Jones, P., Smith, R. W., & Talley, D. (2006). Developing test forms for small-scale achievement testing systems. In S. M. Downing & T. M. Haladyna (Eds.), *Handbook of test development* (pp. 487-525). Mahwah, N.J.: L. Erlbaum.

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

Kirisci, L., Hsu, T.-c., & Yu, L. (2001). Robustness of item parameter estimation programs to assumptions of unidimensionality and normality. *Applied Psychological Measurement, 25*, 146-162.

Little, R.J.A., Rubin, D.B. (2002). *Statistical analysis with missing data* (2nd ed.). New York, NY: Wiley.

Lord, F. M. (1968). An analysis of the Verbal Scholastic Aptitude Test using Birnbaum's three-parameter logistic model. *Educational and Psychological Measurement, 28*, 989-1020.

Lord, F. (1980). *Applications of item response theory to practical testing problems*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Lord, F. (1983). Statistical bias in maximum likelihood estimators of item parameters. *Psychometrika, 48*, 425-435.

Lord, F. M. (1986). Maximum likelihood and Bayesian parameter estimation in item response theory. *Journal of Educational Measurement, 23*, 157-162.

Macdonald, P., & Paunonen, S. V. (2002). A Monte Carlo comparison of item and person statistics based on item response theory versus classical test theory. *Educational and Psychological Measurement, 62*, 921-943.

Mislevy, R. (1986). Bayes modal estimation in item response models. *Psychometrika, 51*, 177-195.

Mislevy, R. J., & Bock, R. D. (1984). BILOG: Maximum likelihood item analysis and test scoring with logistic models for binary items [Computer software]. Mooresville, IN: Scientific Software, Inc.

Mislevy, R. J., & Bock, R. D. (1986). PC-BILOG: Item analysis and test scoring with binary logistic models [Computer software]. Mooresville, IN: Scientific Software, Inc.

Mislevy, R. J., & Bock, R. D. (1989). PC-BILOG 3: Item analysis and test scoring with binary logistic models [Computer software]. Mooresville, IN: Scientific Software, Inc.

Mislevy, R. J., & Bock, R. D. (1997). BILOG 3: Item analysis and test scoring with binary logistic models [Computer software]. Mooresville, IN: Scientific Software.

Mislevy, R. J., & Stocking, M. L. (1989). A consumer's guide to LOGIST and BILOG. *Applied Psychological Measurement, 13*, 57-75.

Muraki, E., & Bock, R. D. (2003). PARSCALE 4: IRT item analysis and test scoring for rating-scale data [Computer software]. Chicago: Scientific Software.

Parshall, C. G., Kromrey, J. D., Chason, W. M., & Yi, Q. (1997). *Evaluation of parameter estimation under modified IRT models and small samples.* Paper presented at the Annual Meeting of the Psychometric Society, Gatlinburg, TN

Parshall, C. G., Kromrey, J. D., & Chason, W. M. (1996). *Comparison of alternative models for item parameter estimation with small samples.* Paper presented at the Annual Meeting of the Psychometric Society, Banff, Alberta, Canada.

Patsula, L. N., & Gessaroli, M. E. (1995). *A comparison of item parameter estimates and ICCs produced with TESTGRAF and BILOG under different test lengths and sample sizes.* Paper presented at the Annual Meeting of the National Council on Measurement in Education, San Francisco, CA.

Ramsay, J. O. (1993). TESTGRAF: A program for the graphical analysis of multiple choice test data [Computer software]. Montreal: McGill University.

Ree, M. J. (1979). Estimating item characteristic curves. *Applied Psychological Measurement, 3*, 371-385.

Ree, M. J., & Jensen, H. E. (1983). Effects of sample size on linear equating of item characteristic curve parameters. In D. J. Weiss & R. D. Bock (Eds.), *New horizons in testing: Latent trait test theory and computerized adaptive testing* (pp. 135-146). New York: Academic Press.

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

Rubin, D. B. (1987). *Multiple imputation for nonresponse in surveys*. New York, NY: Wiley.

SAS Institute Inc. (2004). *SAS/STAT® 9.1 user's guide*. Cary, NC: SAS Institute Inc.

Schafer, J. L. (1997). *Analysis of incomplete multivariate data*. New York, NY: J. Wiley & Sons.

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

Seong, T.-J. (1990). Sensitivity of marginal maximum likelihood estimation of item and ability parameters to the characteristics of the prior ability distributions. *Applied Psychological Measurement, 14*, 299-311.

Setiadi, H. (1997). *Small sample IRT item parameter estimates.* Unpublished Ed.D., University of Massachusetts Amherst, United States, Massachusetts.

Sireci, S. G. (1992). *The utility of IRT in small-sample testing applications.* Paper presented at the Annual Meeting of the American Psychological Association, Washington, DC.

Skaggs, G., & Stevenson, J. (1989). A comparison of pseudo-Bayesian and joint maximum likelihood procedures for estimating item parameters in the three-parameter IRT model. *Applied Psychological Measurement, 13*, 391-402.

Stone, C. A., Weissman, A., & Lane, S. (2005). The consistency of student proficiency classifications under competing IRT models. *Educational Assessment, 10*, 125-146.

Stocking, M. (1990). Specifying optimum examinees for item parameter estimation in item response theory. *Psychometrika, 55*, 461-475.

Swaminathan, H., & Gifford, J. (1983). Estimation of parameters in the three-parameter latent trait model. In D. J. Weiss & R. D. Bock (Eds.), *New horizons in testing: latent trait test theory and computerized adaptive testing* (pp. 13-30). New York: Academic Press.

Swaminathan, H., & Gifford, J. (1986). Bayesian estimation in the three-parameter logistic model. *Psychometrika, 51*, 589-601.

Swaminathan, H., Hambleton, R. K., Sireci, S. G., Xing, D., & Rizavi, S. M. (2003). Small sample estimation in dichotomous item response models: Effect of priors based on

judgmental information on the accuracy of item parameter estimates. *Applied Psychological Measurement, 27*, 27-51.

Thissen, D. (1991). MULTILOG version 6.0 user's guide [Computer software]. Chicago: Scientific Software International.

Thissen, D. (2000). Reliability and measurement precision. In H. Wainer (Ed.), *Computer adaptive testing: A primer* (2nd ed., pp. 159-184). Westport, CT: Praeger Publishers.

Timminga, E. (1995). Optimum examinee samples for item parameter estimation in item response theory: A multi-objective programming approach. *Psychometrika, 60*, 137-154.

Urry, V. W. (1974). Approximations to item parameters of mental test models and their uses. *Educational and Psychological Measurement, 34*, 253-269.

Urry, V. W. (1977). OGIVIA: Item parameter estimation program with normal ogive and logistic three-parameter model options [Computer software]. Washington, D.C.: U.S. Civil Service Commission, Personnel Research and Development Center.

Urry, V. W. (1978). ANCILLES: Item parameter estimation program with normal ogive and logistic three-parameter model options [Computer software]. Washington, D.C.: U.S. Civil Service Commission, Personnel Research and Development Center.

Vale, C. D., Maurelli, V.A., Gialluca, K. A., Weiss, D.J., & Ree, M.J. (1981). *Methods for linking item parameters* (AFHRL-TR-81-10). Brooks Air Force Base TX: U.S. Air Force Human Resources Laboratory, Manpower and Personnel Division.

Vale, C. D., & Gialluca, K. A. (1985). ASCAL: A microcomputer program for estimating logistic IRT item parameters (ONR-85-4) [Computer software]. St. Paul, MN: Assessment Systems Corporation.

Vale, C. D., & Gialluca, K. A. (1988). Evaluation of the efficiency of item calibration. *Applied Psychological Measurement, 12*, 53-67.

Wainer, H. & Eignor, D. (2000). Caveats, pitfalls, and unexpected consequences of implementing large-scale computerized testing. In H. Wainer (Ed.), *Computer adaptive testing: A primer* (2nd ed., pp. 271-299). Westport, CT: Praeger Publishers.

Wendler, C.L & Walker, M.E. (2006). Practical issues in designing and maintaining multiple test forms for large-scale programs. In S. M. Downing & T. M. Haladyna (Eds.), *Handbook of test development* (pp. 445-467). Mahwah, N.J.: L. Erlbaum.

Wingersky, M. S., Barton, M. A., & Lord, F. M. (1982). LOGIST user's guide [Computer software]. Princeton, NJ: Educational Testing Service.

Wingersky, M. S., Barton, M. A., & Lord, F. M. (1985). LOGIST user's guide (LOGIST 5 version 2.1) [Computer software]. Princeton, NJ: Educational Testing Service.

Wingersky, M. S., & Lord, F. M. (1984). An investigation of methods for reducing sampling error in certain IRT procedures. *Applied Psychological Measurement, 8*, 347-364.

Wood, R., Wingersky, M. S., & Lord, F. (1976). LOGIST: A computer program for estimating examinee ability and item characteristic curve parameters (Research Memorandum 76-6) [Computer Software]. Princeton, NJ: Educational Testing Service.

Wright, B., & Stone, M. H. (1979). *Best test design*. Chicago: MESA Press.

Yen, W. (1987). A comparison of the efficiency and accuracy of BILOG and LOGIST. *Psychometrika, 52*(2), 275-291.

Yen, W., & Fitzpatrick, A. R. (2006). Item response theory. In R. L. Brennan (Ed.), *Educational measurement* (4th ed., pp. 111-153). Westport, CT: Praeger Publishers.

Yoes, M. E. (1993). *A comparison of the effectiveness of item parameter estimation techniques used with the three-parameter logistic item response theory model. (Volumes I and II)*. Unpublished Ph.D., University of Minnesota, Minneapolis/St. Paul, MN.

Yoes, M. E. (1995). *An updated comparison of micro-computer based item parameter estimation procedures used with the 3-parameter IRT model (ASC Technical Rep. 95-1R)*. Saint Paul, MN: Assessment Systems Corporation.

Yoes, M. E. (1996). User's manual for the XCALIBRE marginal maximum-likelihood estimation program [Computer software]. St. Paul, MN: Assessment Systems Corp.

Zimowski, M. F., Muraki, E., Mislevy, R. J., & Bock, R.D. (2003). BILOG-MG 3 for Windows: Multiple-group IRT analysis and test maintenance for binary items [Computer software]. Lincolnwood, IL: Scientific Software International, Inc

Appendix A: Example SAS Imputation Syntax Templates

EM Imputation

```
proc mi data=<data with holes>
    seed=<random number seed>
    nimpute= 0
    round = 1 noprint;
    EM out=<filled-in data set> maxiter=100000;
    var item1 – item60;
run;
```

MCMC Imputation

```
proc mi data=<data with holes>
    seed=<random number seed>
    OUT=<filled-in data set>
    nimpute= 1
    round = 1 noprint;
    mcmc nbiter = 500;
    var item1 – item60;
run;
```

Appendix B: Example SAS code for generating item parameters, ability parameters, and response vectors

```
*********************************************************
* Set options so the log works correctly, and SAS waits  *
*    for the equate program to finish before moving on   *
*********************************************************;
options nonotes nodate nonumber LINESIZE=MAX PAGESIZE=MAX label nocenter NOXWAIT XSYNC;
proc datasets lib=work memtype=data kill; run;


*********************************************************
* This suppresses the output and log and sends them to a *
*    file                                                *
*********************************************************;
proc printto new print='c:\outputtestu.lst'; run;
proc printto new log='c:\logtestu.log'; run;


*****************************************************
* Universal variables                               *
*****************************************************;

*****Item information;
%let num_items = 10;       *number of items;
%let lowest_a = .5;        *minimum value of 'a';
%let range_a = 1.5;        *range of 'a' values (i.e., maximum-minimum);
%let lowest_b = -2.5;      *minimum value of 'b';
%let range_b = 5;          *range of 'b values (i.e., maximum-minimum);
%let lowest_c = 0;         *minimum value of 'c';
%let range_c = .3;         *range of 'c' values (i.e., maximum-minimum);

*****Number of real subjects;
%let num_real_subs = 250;
```

```
*****Pseudo-Subject information;
%let num_dups    = 25;                                  *number of pseudo-people per person;
%let prob_del    = .20;                                 *probability of randomly deleting each observation;
%let num_pseudo  = %eval(&num_real_subs * &num_dups); *total number of pseudo-people;

*****Number of replications;
%let num_reps = 1000;

*****Random number seeds;
%let seed1 = 475743;
%let seed2 = 654287;
%let seed3 = 985429;
%let seed4 = 421355;
%let seed5 = 785461;
%let seed6 = 867539;
%let seed7 = 477863;
%let seed8 = 847589;


*******************************************************
* Generate items and output parameters to a file       *
*******************************************************;

*Generate items;
data items;
   do item = 1 to &num_items;
      a = &range_a*(ranuni(&seed1))+ &lowest_a;
        b = &range_b*(ranuni(&seed2))+ &lowest_b;
      c = &range_c*(ranuni(&seed3))+ &lowest_c;
      output;
   end;
run;

*Output to a file;
data _null_;
    set items;
      file "z:\items.dat";
    put @1 a @20 b @40 c;
```

```
run;
*******************************************************
* Generate data sets                                  *
*******************************************************;

%macro generate_datasets;
   %do i=1 %to &num_reps;
      *******************************************************
      * Generate subjects and output abilities to a file    *
      *******************************************************;

      *delete the temporary files so I do not fill up the hard drive;
        proc datasets kill; run;

      *Generate items (this is here again so I can delete temp files at the end of the loop);
      data items;
         do item = 1 to &num_items;
            a = &range_a*(ranuni(&seed1))+ &lowest_a;
              b = &range_b*(ranuni(&seed2))+ &lowest_b;
            c = &range_c*(ranuni(&seed3))+ &lowest_c;
            output;
         end;
      run;


      *Generate subjects;
      data subjects&i;
         do subject=1 to &num_real_subs;
            seed=&seed4+subject+&i;
            call rannor(seed,theta);
            sub_id = put (subject, z10.);
            output;
         end;
         keep sub_id theta;
      run;
```

```
*Output to a file;
data _null_;
    set subjects&i;
      file "z:\subjects&i..dat";
    put @1 sub_id @20 theta;
run;


*******************************************************
* Merge subject and items into file, simulate         *
*   responses, create response vectors, and output    *
*   vectors to a file                                 *
*******************************************************;

*Create a line for each item for every subject;
data temp1;
   set subjects&i;
   do item = 1 to &num_items;
     output;
   end;
run;

*Sort by item;
proc sort data=temp1; by item; run;

*Add item information;
data temp2;
   merge temp1 items; by item;
run;

*Sort by subject;
proc sort data=temp2; by sub_id; run;
```

```
*Simulate item responses;
data subjects_and_items&i;
   set temp2;
   prob = c + (1-c)*(exp(a*(theta-b))/(1+exp(a*(theta-b))));  *calculate prob of correct answer;
   seed=&seed5+&i;
   y = ranbin(seed, 1, prob);                                  *generate responses based on probs;
run;

*put data in wide (response vector) format;
proc transpose data=subjects_and_items&i out= subjects_and_items_wide&i(DROP=_NAME_) prefix=item;
   by sub_id theta;
   id item;
   var y;
run;

*output to a file;
data _null_;
    set subjects_and_items_wide&i;
      file "z:\real_response_vectors&i..dat" DLM=' ';
    put sub_id item1-item&num_items;
run;


*********************************************************
* Create duplicates of subjects and output response    *
*    vectors                                            *
*********************************************************;

*generate duplicates of subjects;
data subjects_items_and_dups&i;
   set subjects_and_items&i;
   do dup=1 to &num_dups;
      dup_id = put (dup, z10.);
        sub_dup_id = sub_id || dup_id;
      output;
   end;
run;
```

```
*put data in wide (response vector) format;
proc sort data=subjects_items_and_dups&i; by sub_id dup_id item; run;
proc transpose data=subjects_items_and_dups&i
   out=subjects_items_and_dups_wide&i(DROP=_NAME_) prefix=item;
   by sub_id dup_id theta;
   id item;
   var y;
run;

*add a subject/dup id variable;
data subjects_items_and_dups_wide&i; set subjects_items_and_dups_wide&i;
   sub_dup_id = sub_id || dup_id;
run;

*output to a file;
data _null_;
    set subjects_items_and_dups_wide&i;
      file "z:\real_response_vectors_with_dups&i..dat" DLM=' ';
    put sub_dup_id item1-item&num_items;
run;


*******************************************************
* Randomly delete observations from data with dups     *
*    and put data set in wide format                   *
*******************************************************;
data data_with_holes&i;
   set subjects_items_and_dups&i;
   delete_test = ranuni(&seed7+&i);
   if (delete_test < &prob_del) then y= .;
run;

*put data in wide (response vector) format;
proc sort data=data_with_holes&i; by sub_id dup_id item; run;
proc transpose data=data_with_holes&i out= data_with_holes_wide&i(DROP=_NAME_) prefix=item;
   by sub_id dup_id theta;
   id item;
   var y;
```

```
run;
*add a subject/dup id variable;
data data_with_holes_wide&i; set data_with_holes_wide&i;
   sub_dup_id = sub_id || dup_id;
run;

*******************************************************
* Impute randomly deleted observations, clean up the   *
*    imputed data, and output to file (EM Imputation)  *
*******************************************************;

%let seed9 = %eval(&seed8 + &i);

*impute the data;
proc mi data=data_with_holes_wide&i
        seed=&seed9
        nimpute= 0
        round = 1 noprint;
        EM out=imputed_data_em&i  maxiter=100000;
        var item1 - item&num_items;
run;

*clean up the data set;
data imputed_data_clean_em&i;
    set imputed_data_em&i;
    array X[&num_items] item1-item&num_items;

      do i = 1 to &num_items;
          if X[i] LT .5 then X[i] = 0;
        if X[i] GE .5 then X[i] = 1;
      end;

      drop i;
run;
```

```
*output to a file;
data _null_;
    set imputed_data_clean_em&i;
      file "z:\imputed_data_clean_em&i..dat" DLM=' ';
    put sub_dup_id item1-item&num_items;
run;

*********************************************************
* Impute randomly deleted observations, clean up the   *
*    imputed data, and output to file (MCMC imputation)*
*********************************************************;


*impute the data;
proc mi data=data_with_holes_wide&i
        seed=&seed9
        OUT=imputed_data&i
        nimpute= 1
        round = 1 noprint;
        mcmc nbiter = 500 niter=500;
        var item1 - item&num_items;
run;

*clean up the data set;
data imputed_data_clean&i;
    set imputed_data&i;
    array X[&num_items] item1-item&num_items;

      do i = 1 to &num_items;
          if X[i] LT .5 then X[i] = 0;
        if X[i] GE .5 then X[i] = 1;
      end;

      drop i;
run;
```

```
      *output to a file;
      data _null_;
          set imputed_data_clean&i;
            file "z:\imputed_data_clean&i..dat" DLM=' ';
          put sub_dup_id item1-item&num_items;
      run;


   %end; *end of do loop;
%mend generate_datasets;   *end of macro;
%generate_datasets;
```

Appendix C. Item parameter information

Table 19

Descriptive statistics for the item parameters for each test length

		parameter		
Test length	Statistic	*b*	*a*	*c*
10	Mean	-0.19	1.25	0.16
	Median	-0.10	1.30	0.16
	SD	1.67	0.43	0.07
	Minimum	-2.13	0.57	0.05
	Maximum	2.38	1.92	0.27
30	Mean	-0.07	1.32	0.13
	Median	0.39	1.39	0.12
	SD	1.61	0.43	0.08
	Minimum	-2.46	0.54	0.01
	Maximum	2.50	1.92	0.29
60	Mean	0.00	1.25	0.14
	Median	0.23	1.30	0.13
	SD	1.55	0.43	0.08
	Minimum	-2.46	0.51	0.00
	Maximum	2.50	1.95	0.29

Table 20

Item parameters and CTT approximations

Test length			*Parameter* [sampling distribution]			CTT approximations[#]	
			b	*a*	*c*		
10 items	30 items	60 items	[U(-2.5, 2.5)]	[U(0.5, 2)]	[U(0.0, 0.3)]	*p*	*r*
	*	*	-2.46	1.43	0.11	0.98	0.28
	*	*	-2.45	0.72	0.23	0.94	0.27
		*	-2.34	1.79	0.15	0.98	0.28
	*	*	-2.31	1.86	0.03	0.98	0.31
		*	-2.29	1.76	0.02	0.98	0.31
*	*	*	-2.13	1.36	0.22	0.97	0.32
		*	-2.12	1.50	0.14	0.97	0.33
	*	*	-2.09	1.81	0.21	0.97	0.32
	*	*	-2.09	1.87	0.10	0.97	0.34
*	*	*	-2.08	1.79	0.14	0.97	0.34
*	*	*	-1.75	1.92	0.06	0.94	0.43
		*	-1.52	1.33	0.14	0.90	0.44
*	*	*	-1.45	1.28	0.17	0.90	0.44
		*	-1.30	1.18	0.10	0.86	0.48
		*	-1.21	1.05	0.07	0.82	0.48
		*	-1.21	0.78	0.12	0.80	0.41
		*	-1.18	1.78	0.07	0.86	0.55
		*	-1.16	1.70	0.10	0.86	0.54
	*	*	-1.12	0.54	0.21	0.77	0.31
		*	-1.01	0.51	0.24	0.76	0.29
		*	-0.97	1.13	0.28	0.83	0.44
		*	-0.94	1.48	0.16	0.82	0.53
	*	*	-0.89	0.98	0.06	0.75	0.50
		*	-0.81	0.71	0.16	0.73	0.39
*	*	*	-0.67	0.57	0.22	0.71	0.32
		*	-0.54	1.45	0.27	0.76	0.51
	*	*	-0.28	1.53	0.28	0.71	0.51
		*	-0.20	0.72	0.22	0.65	0.38
		*	0.08	1.16	0.14	0.55	0.52
		*	0.16	1.34	0.22	0.57	0.50
	*	*	0.29	1.63	0.09	0.45	0.61
		*	0.33	0.90	0.11	0.48	0.46
	*	*	0.39	1.73	0.01	0.37	0.67
	*	*	0.39	1.02	0.09	0.45	0.50
*	*	*	0.46	1.05	0.12	0.45	0.48
*	*	*	0.58	1.31	0.05	0.36	0.57
		*	0.63	0.55	0.06	0.42	0.35
	*	*	0.64	1.78	0.02	0.30	0.64
*	*	*	0.70	1.15	0.12	0.39	0.47
	*	*	0.71	1.49	0.07	0.33	0.55
	*	*	0.73	0.83	0.06	0.36	0.45
		*	1.04	0.83	0.08	0.31	0.41
		*	1.06	0.84	0.14	0.35	0.37
	*	*	1.27	1.66	0.18	0.29	0.34
	*	*	1.31	0.75	0.29	0.44	0.25
		*	1.31	1.25	0.00	0.15	0.51
	*	*	1.39	1.04	0.21	0.33	0.29
		*	1.46	1.85	0.19	0.27	0.28
		*	1.55	0.81	0.29	0.41	0.22
		*	1.71	1.25	0.23	0.30	0.21
	*	*	1.87	1.55	0.19	0.23	0.19
*	*	*	2.08	1.47	0.18	0.22	0.15
		*	2.10	1.63	0.08	0.11	0.20
	*	*	2.13	1.13	0.02	0.08	0.30
		*	2.15	1.39	0.05	0.09	0.23
		*	2.34	1.95	0.04	0.05	0.17
		*	2.37	0.61	0.13	0.22	0.20
*	*	*	2.38	0.63	0.27	0.35	0.14
		*	2.47	0.53	0.22	0.31	0.16
	*	*	2.50	1.63	0.05	0.07	0.13

[#]The CTT approximations are the estimated p-values and point-biserial correlations assuming a N(0,1) ability distribution. Values are calclated using equations in Urry (1974).

Appendix D: Example BILOG Syntax Templates

Item Calibration

```
TITLE1
TITLE2
>GLOBAL DFNAME=<RAW or DupER DATA FILE>,,
        NPARM = 3, LOGISTIC, SAVE;
>SAVE PARM=<ITEM PARAMETER ESTIMATE OUTPUT FILE;
>LENGTH NITEMS = (60);
>INPUT NTOTAL=60, NIDCH=20;
>ITEMS ;
>TEST1 TNAME='TEST0001', INUMBER = (1(1)60);
(20A1,1X,60(A1,1X));
>CALIB CYCLES=200, NEWTON=25, RIDGE=(2,.01,0.2), SPRIOR,
TPRIOR, GPRIOR;
>SCORE;
```

Ability Estimation

```
TITLE1
TITLE2
>GLOBAL DFNAME=<RAW DATA FILE>,,
        IFNAME=<RAW or DupER ITEM PARAMETER ESTIMATES FILE>,,
        NPARM = 3, LOGISTIC, SAVE;
>SAVE SCORE=<ABILITY ESTIMATE OUTPUT FILE;
>LENGTH NITEMS = (60);
>INPUT NTOTAL=60, NALT=5, NIDCH=10;
>ITEMS ; for him
>TEST1 TNAME='TEST0001', INUMBER = (1(1)60);
(10A1,1X,60(A1,1X));
>CALIB SELECT=0;
>SCORE METHOD=2, NQPT=(80), FIT, NOPRINT;
```