A Constrained Metropolis–Hastings Robbins–Monro Algorithm for Q Matrix Estimation in DINA Models — Source link ↗

Chen Wei Liu, Björn Andersson, Anders Skrondal, Anders Skrondal ...+1 more authors

Institutions: National Taiwan Normal University, University of Oslo, University of California, Berkeley, Norwegian Institute of Public Health

Related papers:

- Stochastic approximation EM for exploratory item factor analysis

- A Deep Learning Algorithm for High-Dimensional Exploratory Item Factor Analysis

- Bayesian estimation of multidimensional item response models. A comparison of analytic and simulation algorithms

- An Application of Adaptive Independent Chain Metropolis–Hastings Algorithms in Bayesian Hazard Rate Estimation

- Efficient sampling schemes for Bayesian MARS models with many predictors

Chen-Wei Liu

Department of Educational Psychology and Counseling, National Taiwan Normal University, Taiwan

Björn Andersson

Centre for Educational Measurement, University of Oslo, Norway

Anders Skrondal

Centre for Fertility and Health, Norwegian Institute of Public Health & Centre for Educational

Measurement, University of Oslo & Graduate School of Education, University of California, Berkeley

Correspondence:

Chen-Wei Liu

Department of Educational Psychology and Counseling, National Taiwan Normal University

162, Section 1, Heping E. Road, 10610 Taipei, Taiwan.

Email: cwliu@ntnu.edu.tw

A Constrained Metropolis-Hastings Robbins-Monro Algorithm for Q Matrix Estimation in DINA Models

Abstract

In diagnostic classification models (DCMs) the Q matrix encodes which attributes are required for each item. The Q matrix is usually predetermined by the researcher but may in practice be misspecified which yields incorrect statistical inference. Instead of using a predetermined Q matrix it is possible to estimate it simultaneously with the item and structural parameters of the DCM. Unfortunately, current methods are computationally intensive when there are many attributes and items. In addition, the identification constraints necessary for DCMs are not always enforced in the estimation algorithms which can lead to non-identified models being considered. We address these problems by simultaneously estimating the item, structural and Q matrix parameters of the Deterministic Input Noisy "And" gate model (DINA) using a constrained Metropolis-Hastings Robbins-Monro (cMHRM) algorithm. Simulations show that the new method is computationally efficient and can outperform previously proposed Bayesian Markov chain Monte-Carlo algorithms in terms of Q matrix recovery, and item and structural parameter estimation. We also illustrate our approach using Tatsuoka's fraction-subtraction data and Certificate of Proficiency in English data.

Keywords: diagnostic classification models, Q matrix, stochastic algorithm

Introduction

Diagnostic classification models (DCMs) for item responses, also called cognitive diagnostic models, have been developed to detect the presence or absence of multiple fine-grained attributes. Areas of application have mainly been educational measurement (e.g., to infer whether specific math skills have been mastered by students) and psychiatry (e.g., to infer if a patient has specific mental problems). Several DCMs serve this diagnostic purpose, such as the Deterministic Input Noisy "And" gate model (DINA; Haertel, 1989; Junker & Sijtsma, 2001; Macready & Dayton, 1977), the Deterministic Input Noisy "Or" gate model (DINO; Templin & Henson, 2006) and the reduced Reparameterized Unified model (r-RUM; Hartz, 2002). General frameworks of DCMs that subsume the above models have been

developed via the log-linear cognitive diagnosis model (LCDM; Henson, Templin, & Willse, 2009) and the generalized DINA model (G-DINA; de la Torre, 2011). Parameter estimation for DCMs has in previous studies mainly focused on marginal maximum likelihood implemented via the expectation maximization (EM) algorithm and Bayesian Markov chain Monte Carlo (MCMC) (for a review, see Rupp, Templin, & Henson, 2010, p. 250). Xu (2017) established a set of sufficient conditions for the identification of the above models. However, the Q matrix is typically assumed known and correctly specified. In practice, Q matrices may be misspecified which could lead to misleading classification of individuals and lack of fit to the data (Rupp & Templin, 2008; Wang, 2017). Estimating the Q matrix from data could provide information regarding the validity of expert-defined Q matrices.

Research on the identification and consistent estimation of Q matrices has recently been presented in the cognitive diagnostic literature (e.g., Chen, Liu, Xu, & Ying, 2015; Liu, Xu, & Ying, 2012; Xu & Shang, 2017). Liu, Xu, and Ying (2013) showed that the Q matrix is identifiable under certain conditions given that the guessing parameters of the DINA model are known. However, the values of the guessing parameters are usually unknown. Chen et al. (2015) further proved that the Q matrices of DINA and DINO models are identifiable given unknown item parameters and structural parameters under certain conditions. Xu and Shang (2017) extended the work of Chen et al. (2015) and established a set of identification results for various DCMs. Their theoretical results enhance the understanding of Q matrix estimation and can be used in any estimation procedure.

Several studies have developed validation approaches to detect misspecified elements of Q matrices (e.g., de la Torre, 2008; de la Torre & Chiu, 2016; DeCarlo, 2012; Liu et al., 2012; Wang et al., 2018). Unfortunately, many of them do not or only partially address the estimation problem, rely on partial knowledge of Q, require provisional starting values of Q, need expensive computations, or depend on particular cutoff values. Furthermore, incorrect partial knowledge of the Q matrix could compromise the consistency of Q validation approaches (Liu, 2017). In this study, we propose a data-driven approach that mitigates the above problems.

Data-driven approaches to estimating the Q matrix have been proposed previously. For

example, a method based on regularization that penalizes appropriate parameters to zero to identify the best model with a range of regularization values has been proposed (Chen et al., 2015). A second approach is a Bayesian MCMC algorithm that uses either Metropolis or Gibbs sampling given appropriate identification (Chen, Culpepper, Chen, & Douglas, 2017). Chen et al. (2017) developed a Bayesian approach that respects the identification conditions for Q matrices developed by Chen et al. (2015). However, both approaches have disadvantages. The regularization approach does not directly impose the identification constraints on Q, relies on the EM algorithm and a regularization algorithm that is time-consuming when using a wide range of regularization parameter values, and requires a two-stage estimation procedure where Q must be estimated first and the remaining parameters are estimated given the estimated Q (Chen et al., 2015). Meanwhile, the MCMC algorithm (see Chen et al., 2017) performed unsatisfactorily regarding Q matrix recovery for the DINA model with large sample sizes (i.e., 2000 and 4000). Conversely, the regularization approach (Chen et al., 2015) performed badly for small sample sizes (e.g., 500). This prompted us to develop a new algorithm that is both computationally efficient and can simultaneously estimate the Q matrix, and item and structural parameters.

Inspired by the successful Metropolis-Hastings Robbins-Monro (MHRM) algorithm for item response models with multidimensional continuous latent variables (Cai, 2010), and the proposal distribution developed for the Q matrix in the MCMC algorithm (Chen et al., 2017), we propose a constrained Metropolis-Hastings Robbins-Monro (cMHRM) algorithm that enjoys the computational efficiency of the MHRM and can estimate the item and structural parameters and the Q matrix simultaneously without any prior knowledge of Q.

The basic idea of cMHRM is straightforward. First, we treat the binary latent attributes and the binary Q matrix as missing data (i.e., latent variables) and sample them by means of MCMC algorithms with appropriate model constraints. Second, the item parameters and structural parameters are estimated by maximum likelihood estimation (MLE) given the known missing data. Third, the Monte Carlo errors from the MCMC algorithm are averaged out by means of the Robbins-Monro (RM) algorithm (Robbins & Monro, 1951). Cai (2010) conducted a range of simulation studies for multidimensional item response models and

showed that the parameter recovery of MHRM was almost identical to the EM algorithm with numerical quadrature. However, MHRM is not directly applicable to DCMs and our new cMHRM is particularly designed for DCMs with unknown model parameters and an unknown Q matrix. In this study, we use the DINA model to illustrate use of the cMHRM algorithm. Importantly, we also demonstrate that existing Bayesian algorithms (including the cGibbs sampler proposed by Chen et al., 2017) require multiple sets of starting values to perform well.

The outline of this article is as follows. First, the general DCM (GDCM) and the DINA model are introduced. Second, the identification conditions for Q matrix estimation are described. Third, the cMHRM algorithm is introduced and described in detail, and relevant issues pertaining to computation, standard error estimation, latent attribute estimation, and Q estimation, are summarized. Fourth, a series of Monte Carlo studies of parameter recovery are conducted. Fifth, we use cMHRM to analyze two datasets: (i) Tatsuoka's fraction-subtraction data and (ii) Examination for the Certificate of Proficiency in English data. Finally, we close the article with some concluding remarks.

Model Formulation

Let N denote the number of individuals, J denote the number of items, K denote the number of attributes, and y_{ij} denote the dichotomous responses in the $N \times J$ matrix for individual i and item j. With DCMs, there are binary attribute parameters of individuals $\alpha = (\alpha_1, ..., \alpha_K)^T \in (0, 1)^K$, item parameters ξ, structural parameters π, and a $J \times K$ Q matrix $\in (0, 1)^{J \times K}$ that catalogs the binary attributes of items that are required for individuals to master. The Q matrix consists of binary elements $(q_1, ..., q_J)^T$, where $(q_{j1}, ..., q_{jK})$ is the j-th row of Q.

The GDCM can be formulated as

$$f\left(\mathrm{Pr}_{j,\alpha}\right) = \beta_{j0} + \sum_{k=1}^{K_j} \beta_{jk} q_{jk} \alpha_k + \sum_{k=1}^{K_j} \sum_{k'>k} \beta_{j(kk')} q_{jk} q_{jk'} \alpha_k \alpha_{k'} + \cdots + \beta_{j(12...K_j)} \prod_{k=1}^{K_j} q_{jk} \alpha_k . \quad (1)$$

Here, the β_{j0} are intercept parameters, the β_{jk} represent the K attributes' main effects, the $\beta_{j(kk')}$ represent two-way interactions between attributes, the remaining terms represent higher-order

interactions, $f(.)$ is a link function, and $\text{Pr}_{j,\alpha}$ is a user-defined item response probability for item j and latent attributes α. The GDCM subsumes, for instance, the LCDM by specifying $f(.)$ to be the logit function. The GDCM can subsume various DCMs as a special cases, for example the DINA and DINO models (Henson et al., 2009).

DINA Model

The DINA model (Haertel, 1989; Junker & Sijtsma, 2001; Macready & Dayton, 1977) specifies a conjunctive relationship between latent attributes α_i and observed binary responses y_{ij}, which means that the probability of a correct response is high if all the required attributes have been mastered by individual i. Specifically, the probability of a correct response is

$$\text{Pr}_{j,\alpha} = \text{Pr}\left(Y_{ij} = 1 \mid \alpha_{ik}, s_j, g_j\right) = \left(1 - s_j\right)^{\eta_{ij}} g_j^{\left(1 - \eta_{ij}\right)}, \tag{2}$$

where s_j is a slipping parameter and g_j is a guessing parameter, and η_{ij} is a binary variable characterized by the conjunctive relationship

$$\eta_{ij} = \prod_{k=1}^{K} \alpha_{ik}^{q_{jk}}. \tag{3}$$

The conjunctive relationship classifies examinees into two groups— one with examinees who have all the required attributes to answer item j correctly with high probability (i.e., $1 - s_j$), and another containing examinees who do not have all the required attributes and can only answer item j correctly with low probability (i.e., g_j). This conjunctive relationship can be compactly expressed as $\alpha \succcurlyeq q_j$ which means that $\alpha_k \geq q_{jk}$ for all k. Equation (3) also possesses the non-compensatory property that mastering part of the required attributes cannot compensate for other non-mastered attributes in probability. Junker and Sijtsma (2001) suggested to impose the constraint $0 \leq g_j < 1 - s_j \leq 1$ to ensure monotonicity of the item response function.

The DINA model can be reparameterized as a GDCM by only retaining the intercept and highest order interaction parameters, and letting $f(.)$ be the identity function:

$$\text{Pr}_{j,\alpha} = \begin{cases} \kappa_{j0} = \beta_{j0} & \text{if } \alpha \nsucceq q_j \\ \kappa_{j(12\ldots K_j)} = \beta_{j0} + \beta_{j(12\ldots K_j)} & \text{if } \alpha \succcurlyeq q_j \end{cases} \tag{4}$$

where $\kappa_{j0} = g_j$ is the guessing parameter, $\kappa_{j(12\ldots Kj)} = 1 - s_j$ is one minus the slipping parameter and $\beta_{j(12\ldots K_j)} > 0$ to ensure monotonicity.

In conventional applications of DCMs, the components of the Q matrix are assumed to be correctly specified by content experts and Q is thus fixed in parameter estimation. Allowing the elements of the Q matrix to be estimated is possible, but the required constraints on Q are usually not imposed.

Identifiability of the Q Matrix

Chen et al. (2015) obtained necessary conditions for an identified Q matrix for the DINA model, given unknown item parameters, structural parameters and Q matrix components. Chen et al. (2017) summarized the identification conditions as follows:

C1. There exists a $J \times J$ permutation matrix $\mathbf{P}$, so the Q can be transformed as

$$\mathbf{P}Q = \begin{bmatrix} \mathbf{I}_K \\ \mathbf{I}_K \\ Q^* \end{bmatrix}, \tag{5}$$

where $\mathbf{I}_K$ is a $K \times K$ identity matrix, Q^* is a $(J - 2K) \times K$ matrix, and $J - 2K \geq 2$.

C2. Each attribute is measured by at least three items; namely we define the column margin such that $Q_k^{\mathrm{T}} \mathbf{1}_j \geq 3$, where Q_k^{T} is the transposed k-th column of Q.

C3. Each item loads onto at least one attribute; namely we define the row margin such that $q_j \mathbf{1}_j \geq 1$, where q_j is the j-th row of Q.

Xu and Shang (2017) proved more general identifiability conditions for the LCDM which subsumes the DINA, r-RUM, and DINO models. They showed that C1 and the following condition are sufficient for identifying the Q matrix:

C4. There exists at least one item in Q^* such that the probability of a positive response to the item is different for individuals with attribute profiles α and α', where $\alpha \neq \alpha'$ and $\alpha \succcurlyeq \alpha'$, where $\alpha \succcurlyeq \alpha'$ means that $\alpha_k \geq \alpha'_k$ for all k.

Note that C3 should be respected in parameter estimation, because each item measures at least one attribute in practice. Under conditions C1, C3, and C4, the Q matrix and model parameters are identifiable. A condition that ensures this is using three identity submatrices in the Q matrix (Xu & Shang, 2017). If this is not easy to achieve in practice, Xu and Shang

(2017) alternatively suggest a more lenient condition with at least two 'complete' matrices (i.e., C1), where each attribute is measured by at least three items (i.e., C2), and each item is linked to at least one attribute (i.e., C3). This condition is identical to the condition proposed by Chen et al. (2015). In Xu and Shang's (2017) simulation studies the $\boldsymbol{Q}$ matrix of the DINA model and the LCDM were well recovered under the more lenient condition when (1) sample sizes were 500, 1000, and 2000 for $K = 3$, and (2) sample sizes were 1000, and 2000 for $K = 4$ and 5. Thus, we adhered to the lenient condition that satisfies C1, C2, and C3 for the DINA model in the subsequent simulation studies.

Constrained Metropolis-Hastings Robbins-Monro (cMHRM) Algorithm

We now formulate the likelihood function we intend to maximize and discuss some numerical problems that may be encountered. Given the observed responses $\mathbf{y}$ and a known $\boldsymbol{Q}$, the marginal log-likelihood function of the unknown parameters of DCMs is in proportional form given by

$$l\left(\xi,\pi;\mathbf{y},\boldsymbol{Q}\right) \propto \sum_{i=1}^{N}\log\left[\sum_{c=1}^{2^K}\pi_c\prod_{j=1}^{J}\Pr(y_{ij}\mid\alpha_c,\xi_j,\boldsymbol{Q}_j)\right], \tag{6}$$

where ξ is the collection of all item parameters and where the structural parameter $\pi_c \equiv \Pr(\alpha_i = \alpha_c\mid\pi)$ represents the prior probability of latent attribute configuration c with $\sum_{c=1}^{C}\pi_c = 1$, $0 \leq \pi_c \leq 1$. Letting $\boldsymbol{Q}$ be treated as an unknown latent variable, we note that $\boldsymbol{Q}$ resides in a binary parameter space, whereas ξ and π are in truncated continuous parameter spaces. Instead of resorting to mixed discrete-continuous optimization, we propose to marginalize over both α and $\boldsymbol{Q}$. Thus, the marginal log-likelihood function is defined as

$$l\left(\xi,\pi;\mathbf{y}\right) \propto \sum_{i=1}^{N}\log\left[\sum_{c=1}^{2^K}\pi_c\sum_{b=1}^{2^{JK}}\Pr\left(\boldsymbol{Q}_{(b)}\right)\prod_{j=1}^{J}\Pr(y_{ij}\mid\alpha_c,\xi_j,\boldsymbol{Q}_{j(b)})\right], \tag{7}$$

where $\Pr(\boldsymbol{Q}_{(b)})$ is the prior information for $\boldsymbol{Q}_{(b)}$, where the identified elements have uniform mass whereas the nonidentified elements have zero mass. We have not yet imposed identification constraints on the parameter space of $\boldsymbol{Q}$, but will address this issue when we discuss how to estimate $\boldsymbol{Q}$ matrices.

The marginal log-likelihood function will be maximized with respect to ξ and π to

obtain maximum likelihood estimates. However, the summation over α and $\boldsymbol{Q}$ is inside the logarithm function, which may lead to very slow computation in maximizing the marginal log-likelihood function (Bock & Lieberman, 1970). To circumvent the difficulty of directly maximizing $l(\xi, \pi; \mathbf{y})$, we use the EM algorithm (Dempster, Laird, & Rubin, 1977) to maximize a surrogate function:

$$Q^*\left(\zeta, \zeta^{\text{old}}\right) = \sum_{i=1}^{N}\sum_{c=1}^{2^K}\sum_{b=1}^{2^{JK}} \Pr\left(\alpha_c, \boldsymbol{Q}_{(b)} \mid \mathbf{y}_i, \zeta^{\text{old}}\right)\left\{\log \pi_c + \sum_{j=1}^{J}\log \Pr(y_{ij} \mid \alpha_c, \xi_j, \boldsymbol{Q}_{j(b)})\right\}. \quad (8)$$

This is proportional to a lower bound on $l(\xi, \pi; \mathbf{y})$, where $\zeta^{\text{old}} \in (\xi^{\text{old}}, \pi^{\text{old}})$ is the collection of parameter estimates from the previous iteration in the EM. The posterior distribution of α_c and $\boldsymbol{Q}_{(b)}$ is

$$\Pr\left(\alpha_c, \boldsymbol{Q}_{(b)} \mid \mathbf{y}_i, \zeta^{\text{old}}\right) = \frac{\pi_c^{\text{old}} \prod_{j=1}^{J} \Pr(y_{ij} \mid \alpha_c, \xi_j^{\text{old}}, \boldsymbol{Q}_{j(b)})}{\sum_{c=1}^{2^K}\sum_{b=1}^{2^{JK}} \pi_c^{\text{old}} \prod_{j=1}^{J} \Pr(y_{ij} \mid \alpha_c, \xi_j^{\text{old}}, \boldsymbol{Q}_{j(b)})}. \quad (9)$$

The obvious difficulty in maximizing the Q^* function is that computation becomes extremely expensive, even when J and K are small. For instance, for a test with $J = 20$ and $K = 5$, $2^5 \times 2^{20 \times 5}$ calculations are required in the denominator of equation (9) per iteration. To address this challenge, the Q^* function can be considered as a regular Monte Carlo integration problem (Wei & Tanner, 1990) and approximated by

$$Q^*\left(\zeta, \zeta^{\text{old}}\right) = \sum_{i=1}^{N} E_{\alpha, Q \mid \mathbf{y}_i, \zeta^{\text{old}}}\left[\log \pi^{I(\alpha_i = \alpha_c)} + \sum_{j=1}^{J}\log \Pr(y_{ij} \mid \alpha_i, \xi_j, \boldsymbol{Q}_j)\right]$$
$$\cong \sum_{i=1}^{N} M^{-1}\sum_{m=1}^{M}\left[\log \pi^{I(\alpha_i^{(m)} = \alpha_c)} + \sum_{j=1}^{J}\log \Pr(y_{ij} \mid \alpha_i^{(m)}, \xi_j, \boldsymbol{Q}_{j(b)}^{(m)})\right]. \quad (10)$$

Here M is a large number of Monte Carlo samples drawn from the posterior distribution of α and $\boldsymbol{Q}$, $I(\alpha_i^{(m)} = \alpha_c)$ is an indicator function equal to 1 if $\alpha_i^{(m)}$ is equal to the attribute pattern in class c and otherwise 0. Cai (2010) proposed using the Metropolis-Hastings (MH) algorithm for item response models with multidimensional continuous latent variables.

Bayesian MCMC is promising for efficient sampling α and $\boldsymbol{Q}$. However, the sample space of α and $\boldsymbol{Q}$ is binary and the identification constraints on $\boldsymbol{Q}$ have to be imposed when sampling the elements of $\boldsymbol{Q}$. Therefore, we need a binary sampler for α that is unconditional

(i.e., having no sampling constraints) and a binary sampler for Q that respects identification constraints.

In this study, we propose to use the modified MH sampler (Madigan, York, & Allard, 1995) for sampling α. This sampler has been shown to be more efficient than the Gibbs sampler (Liu, 1996) and has been used in Bayesian variable selection problems (George & McCulloch, 1997). Moreover, the modified MH sampler is straightforward to implement in practice. For sampling Q, we propose a constrained modified MH sampler which is closely related to the constrained Gibbs sampler proposed by Chen et al. (2017), as both respect the parameter constraints for Q. The basic idea of our cMHRM is similar to the original MHRM, except that we propose to use binary samplers for α and Q. Since it is based on the RM algorithm, we expect the convergence of cMHRM to be similar to the original MHRM (cf. Cai, 2010).

The iterative scheme is outlined below. First of all, randomly initialize $\alpha^{(0)}$, $Q^{(0)}$, $\xi^{(0)}$, and $\pi^{(0)}$. At iteration t, update the parameters by sampling as follows:

(a). Update $\alpha^{(t+1)} \sim \Pr(\alpha | \mathbf{y}, \xi^{(t)}, \pi^{(t)}, Q^{(t)})$.

(b). Update $Q^{(t+1)} \sim \Pr(Q | \mathbf{y}, \alpha^{(t+1)}, \xi^{(t)}, \pi^{(t)})$.

(c). $\pi^{(t+1)} = \underset{\pi}{\arg\max}\, Q^*\left(\zeta, \zeta^{(t)}\right)$ using the RM algorithm.

(d). $\xi^{(t+1)} = \underset{\xi}{\arg\max}\, Q^*\left(\zeta, \zeta^{(t)}\right)$ using the RM algorithm.

The following two steps contain the details of the cMHRM.

Step 1: Stochastic Imputation

Sampling Candidates of Latent Attributes

The modified MH sampler uses the following proposal distribution to sample a candidate $\alpha^{(\text{new})}$:

$$p\left[\alpha^{(\text{new})} | f\left(\alpha^{(t)}\right)\right] = f\left(\alpha^{(t)}\right)^{\alpha^{(\text{new})}} \left[1 - f\left(\alpha^{(t)}\right)\right]^{1-\alpha^{(\text{new})}}, \qquad (11)$$

where the parameter of the Bernoulli distribution is defined as

$$f\left(\alpha^{(t)}\right) \equiv 1 - \alpha^{(t)}, \qquad (12)$$

which means that the current state at each iteration for each attribute is always proposed to

change, $\alpha^{(new)} = 1$ if $\alpha^{(t)} = 0$ or $\alpha^{(new)} = 0$ if $\alpha^{(t)} = 1$. The generic MH algorithm constructs an acceptance probability function to determine whether a proposal value of $\alpha^{(new)}$ is accepted or not. Specifically,

$$
\begin{aligned}
a &= \min\left\{ \frac{\Pr\left(\alpha^{(new)} \mid \mathbf{y}, \boldsymbol{\xi}^{(t)}, \boldsymbol{\pi}^{(t)}, \boldsymbol{Q}^{(t)}\right) p\left[\alpha^{(t)} \mid f\left(\alpha^{(new)}\right)\right]}{\Pr\left(\alpha^{(t)} \mid \mathbf{y}, \boldsymbol{\xi}^{(t)}, \boldsymbol{\pi}^{(t)}, \boldsymbol{Q}^{(t)}\right) p\left[\alpha^{(new)} \mid f\left(\alpha^{(t)}\right)\right]}, 1 \right\} \\
&= \min\left\{ \frac{\Pr\left(\alpha^{(new)} \mid \mathbf{y}, \boldsymbol{\xi}^{(t)}, \boldsymbol{\pi}^{(t)}, \boldsymbol{Q}^{(t)}\right)}{\Pr\left(\alpha^{(t)} \mid \mathbf{y}, \boldsymbol{\xi}^{(t)}, \boldsymbol{\pi}^{(t)}, \boldsymbol{Q}^{(t)}\right)}, 1 \right\}
\end{aligned}
\tag{13}
$$

where the two proposal distributions have cancelled out due to their symmetry. The $\alpha^{(new)}$ is accepted (i.e., $\alpha^{(t+1)} = \alpha^{(new)}$) when $a \geq U \sim Uniform(0, 1)$, otherwise it is rejected (i.e., $\alpha^{(t+1)} = \alpha^{(t)}$). For examinee i, the corresponding posterior distribution of $\boldsymbol{\alpha}_i$ is given by

$$
\Pr\left(\boldsymbol{\alpha}_i \mid \mathbf{y}, \boldsymbol{\xi}^{(t)}, \boldsymbol{\pi}^{(t)}, \boldsymbol{Q}^{(t)}\right) \propto \pi^{I(\boldsymbol{\alpha}_i = \boldsymbol{\alpha}_c)} \prod_{j=1}^{J} \Pr(y_{ij} \mid \boldsymbol{\alpha}_i, \boldsymbol{\xi}^{(t)}, \boldsymbol{Q}_j^{(t)}), \quad \forall i = 1, ..., N.
\tag{14}
$$

In the implementation, each element of an attribute vector is updated sequentially for each examinee. To accelerate the calculation, the kth element of the attributes can be updated simultaneously for all examinees by means of the assumption of local independence between examinees. Programming languages such as R and MATLAB are very efficient at vectorizing the calculations.

Sampling Candidates of Q

Chen et al. (2017) proposed a constrained Gibbs sampler to ensure the identification of the Q matrix and ensure that the Markov chain is irreducible. Their simulation studies showed that the constrained Gibbs sampler had better performance in recovering Q than other samplers in most of their simulation conditions. Here, we propose a new constrained version of the modified MH sampler which is similar to the constrained Gibbs sampler in that we adopt the same constraint step to ensure the identification of Q (see Step a) below). The elements of Q are sequentially updated. For the current iteration at time t, the procedure consists of two steps for $j = 1, ..., J, k = 1, ..., K$ through the MH sampling:

Step a): $q_{jk}^{(t)}$ is a single element of Q for item j and attribute k that will not be updated if it is in one of the following three positions (Chen et al., 2017): (1) a 1 from the row

vector $\boldsymbol{e}_{jk}$, where $\boldsymbol{e}_{jk}$ only has a 1 at k and others are 0; (2) a 1 from the column which has a column sum equal to 3; (3) a 0 from a row vector $\boldsymbol{e}_{jk}$ when there exist only two such vectors in the current $\boldsymbol{Q}$ (e.g., $\boldsymbol{e}_{jk}$ and $\boldsymbol{e}_{j*k}$ for $j \neq j^*$), so that the constraint that at least two $\mathbf{I}_K$ are in $\boldsymbol{Q}$ can be retained.

Step b): Otherwise, $q_{jk}^{(t)}$ can be updated to 0 or 1 according to the following procedure: a new candidate (i.e., 0 or 1) is sampled from a Bernoulli distribution with probability parameter equal to

$$f\left(q_{jk}^{(t)}\right) \equiv 1 - q_{jk}^{(t)}, \tag{15}$$

and the acceptance probability function is expressed as

$$a = \min\left\{ \frac{\Pr\left(q_{jk}^{(\text{new})} \mid \mathbf{y}, \boldsymbol{Q}_{-jk}^{(t)}, \boldsymbol{Q}_{-jk}^{(t+1)}, \boldsymbol{\xi}^{(t)}, \boldsymbol{\pi}^{(t)}, \boldsymbol{\alpha}^{(t+1)}\right)}{\Pr\left(q_{jk}^{(t)} \mid \mathbf{y}, \boldsymbol{Q}_{-jk}^{(t)}, \boldsymbol{Q}_{-jk}^{(t+1)}, \boldsymbol{\xi}^{(t)}, \boldsymbol{\pi}^{(t)}, \boldsymbol{\alpha}^{(t+1)}\right)}, 1 \right\}, \tag{16}$$

where

$$\begin{aligned} &\Pr\left(q_{jk}^{(t)} \mid \mathbf{y}, \boldsymbol{Q}_{-jk}^{(t)}, \boldsymbol{Q}_{-jk}^{(t+1)}, \boldsymbol{\xi}^{(t)}, \boldsymbol{\pi}^{(t)}, \boldsymbol{\alpha}^{(t+1)}\right) \\ &\propto \Pr\left(\mathbf{y} \mid q_{jk}^{(t)}, \boldsymbol{Q}_{-jk}^{(t)}, \boldsymbol{Q}_{-jk}^{(t+1)}, \boldsymbol{\xi}^{(t)}, \boldsymbol{\pi}^{(t)}, \boldsymbol{\alpha}^{(t+1)}\right) \end{aligned} \tag{17}$$

is proportional to the likelihood function, and the identified elements of $\boldsymbol{Q}$ have uniform prior distributions and nonidentified parameters have zero mass. $\boldsymbol{Q}_{-jk}^{(t)}$ denotes the elements of $\boldsymbol{Q}$ that have not been updated and $\boldsymbol{Q}_{-jk}^{(t+1)}$ the elements that have been updated, excluding $q_{jk}^{(t)}$. Then, $q_{jk}^{(t+1)}$ is set equal to the new candidate if $a \geq U \sim Uniform(0, 1)$ and retained otherwise.

Both the constrained MH sampler and the constrained Gibbs sampler are valid samplers. The only difference is that the former uses the modified MH sampler in Step b) (Madigan et al., 1995) which has been shown to be more efficient in exploring the binary sampling space than the Gibbs sampler (Liu, 1996). Both samplers use the same Step a) in order to respect the constraints for identifying $\boldsymbol{Q}$. Using either the Gibbs sampler or the MH sampler will respect the constraints of $\boldsymbol{Q}$ as shown in Step a). Therefore, identification of $\boldsymbol{Q}$ (including C1, C2, and C3) holds and the Markov chain is irreducible (cf. Chen et al., 2017) for the constrained modified MH sampler as well as the Gibbs sampler.

Step 2: Stochastic Approximation

Given the samples of $\boldsymbol{\alpha}^{(t+1)}$ and $\boldsymbol{Q}^{(t+1)}$ from Step 1, maximizing the empirical average $Q*(\zeta, \zeta^{\text{old}})$ with respect to ξ and π uses standard maximum likelihood estimation. The gradient vector and Hessian matrix are obtained by taking the first and second-order derivatives of $Q*(\zeta, \zeta^{\text{old}})$ with respect to ξ and π, which are used to find stationary points in iteration t. However, the gradient and Hessian matrix are error-corrupted due to the Monte Carlo error from Step 1 and the RM algorithm is therefore applied to average out the errors (Cai, 2010; Gu & Kong, 1998).

As implied by Equation (10), ξ and π can be estimated in turn. First, the structural parameter vector π can be estimated by the RM algorithm and the update is given by

$$\boldsymbol{\pi}^{(t+1)} = \boldsymbol{\pi}^{(t)} + g_t \left(M_t^{-1} \sum_{m=1}^{M_t} \frac{x_c^{(t+1,m)}}{N} - \boldsymbol{\pi}^{(t)} \right), \tag{18}$$

where $x_c^{(t+1,m)}$ denotes the number of individuals of attribute pattern $\boldsymbol{\alpha}_c$ in class c for the mth sample at iteration $t+1$, $x_c^{(t+1,m)}/N$ is equal to the maximum likelihood estimates of the multinomial distribution of $x_c^{(t+1,m)}$, and g_t is the gain constant defined as

$$g_t \in (0, 1], \quad \sum_{t=1}^{\infty} g_t = \infty, \quad \text{and} \quad \sum_{t=1}^{\infty} g_t^2 < \infty . \tag{19}$$

This approach has been applied to estimating latent regression models (von Davier & Sinharay, 2010).

Next, the estimation of item parameters (i.e., ξ) follows previous studies (Cai, 2010; Gu & Kong, 1998) where the gradients are approximated by the average of the M sets of gradients of the MH samples

$$\tilde{\mathbf{s}}_{t+1} = M_t^{-1} \sum_{m=1}^{M_t} s\left(\boldsymbol{\xi}^{(t)} \mid \mathbf{y} \right), \tag{20}$$

where

$$s\left(\boldsymbol{\xi}^{(t)} \mid \mathbf{y} \right) = \frac{\partial}{\partial \boldsymbol{\xi}} \log \Pr(\mathbf{y} \mid \boldsymbol{\alpha}^{(t+1,m)}, \boldsymbol{\xi}^{(t)}, \boldsymbol{Q}^{(t+1,m)}) . \tag{21}$$

The Hessian matrix is recursively approximated by

$$\boldsymbol{\Gamma}_{t+1} = \boldsymbol{\Gamma}_t + g_t \left\{ M_t^{-1} \sum_{m=1}^{M_t} H\left(\boldsymbol{\xi}^{(t)} \mid \mathbf{y} \right) - \boldsymbol{\Gamma}_t \right\}, \tag{22}$$

where

$$H\left(\xi^{(t)} \mid \mathbf{y}\right) = \frac{-\partial^2}{\partial \xi \partial \xi'} \log \Pr(\mathbf{y} \mid \alpha^{(t+1,m)}, \xi^{(t)}, \mathbf{Q}^{(t+1,m)}), \tag{23}$$

and Γ_0 for $t = 0$ is an initialized positive-definite matrix. Finally, the parameters, $\xi^{(t)}$, are updated by the RM algorithm:

$$\xi^{(t+1)} = \xi^{(t)} + g_t \left(\Gamma_{t+1}^{-1}\right) \tilde{\mathbf{s}}_{t+1}. \tag{24}$$

Steps 1 and 2 constitute a cycle of cMHRM. Note that the estimation order of ξ and π may be interchanged, but the order does not affect the results according to our experiments.

Configuration of the cMHRM

In the early stage of iterations, the g_t should not decrease too fast since the starting values are usually far from the optimal solution. Typically, a three-stage procedure is implemented as suggested by Yang and Cai (2014). First, g_t is set equal to 1 to move the current estimates quickly to the vicinity of stationary points (Cai, 2010; Diebolt & Ip, 1996). Second, the g_t remains equal to 1 and several iterations are run to obtain an average of the estimates. Finally, averaged estimates are used as starting values and the g_t gradually decreases. The lengths of stage 1 and 2 were set at 50 and 100, respectively, which was shown to be sufficient in our subsequent Monte Carlo studies. In this study, $g_t = 1/t$ is used, the burn-in period of the MH samplers is set at 5, and the sample size of the MH samplers was set at 5 (i.e., $M = 5$) which appeared to be sufficient. Iteration of the cMHRM is terminated if the difference in all parameter estimates across three successive iterations is less than .0001 (Cai, 2010).

A Note on Some Computational Issues

First, calculating the number of examinees in each attribute pattern (i.e., x_c) efficiently is desirable. The bijection method was used (Chen et al., 2017; von Davier, 2014) where the basic idea is to first transform the α matrix into an index vector that records the index of the attribute patterns. The calculation is $\alpha\nu$, where $\nu = [2^{K-1}, 2^{K-2}, \ldots, 1]$ is a mapping vector. Next, the index vector is tabulated to obtain the number of examinees in each attribute pattern.

Second, zero cell counts of $x_c^{(t,m)}$ for the mth sample at iteration t may occur when the

number of the MH samples is small (e.g., $M = 1$). When this occurs, the future estimates are likely to be stuck in "absorbing states" where the class probability estimate is equal to zero and the corresponding $log(\pi_c = 0) = -\infty$ (DeCarlo, 2011). Increasing M does not necessarily remedy this problem. A possible solution is to employ a prior Dirichlet distribution for π_c to smooth the cell counts. It follows that the posterior distribution of π_c is also a Dirichlet,

$$\Pr\left(\pi_c \mid x_c, a_c\right) \propto \prod_{c=1}^{C} \pi_c^{x_c + a_c - 1}, \tag{25}$$

where a_c is a concentration parameter for class c. For simplicity, we used the expected a posteriori (EAP) estimator, $\hat{\pi}_c^{(EAP)} = (x_c + a_c)/\sum_{c=1}^{C}(x_c + a_c)$, if there are any zero cell counts (Agresti & Hitchcock, 2005), otherwise $\hat{\pi}_c^{(MLE)}$ is used. Note that the above procedure is used only for Stage 1 and 2 of cMHRM. For the final stage, only the maximum likelihood estimates are used because the RM algorithm is applied and the estimates gradually approach a stationary point. We set $a_c = 1/C$ (Perks, 1947), which was found effective in the subsequent Monte Carlo studies.

Standard Error Estimation

There are two approaches to obtain the observed information matrix of the marginal log-likelihood function. *Recursively approximated standard errors* are obtained by approximating the information matrix by Louis' (1982) formula:

$$
\begin{aligned}
-\frac{\partial^2 l\left(\hat{\zeta} \mid \mathbf{y}\right)}{\partial \zeta \partial \zeta'} = &-E_{\boldsymbol{\alpha}, Q|\mathbf{y}, \hat{\zeta}}\left[H\left(\hat{\zeta} \mid \mathbf{y}\right)\right] \\
&-E_{\boldsymbol{\alpha}, Q|\mathbf{y}, \hat{\zeta}}\left\{s\left(\hat{\zeta} \mid \mathbf{y}\right)\left[s\left(\hat{\zeta} \mid \mathbf{y}\right)\right]^{\mathrm{T}}\right\} \\
&+E_{\boldsymbol{\alpha}, Q|\mathbf{y}, \hat{\zeta}}\left[s\left(\hat{\zeta} \mid \mathbf{y}\right)\right] E_{\boldsymbol{\alpha}, Q|\mathbf{y}, \hat{\zeta}}\left[s\left(\hat{\zeta} \mid \mathbf{y}\right)^{\mathrm{T}}\right]
\end{aligned}
\tag{26}
$$

where $\zeta \in (\xi, \pi)$. The gradient vector and Hessian matrix are by-products of cMHRM. The Monte Carlo version of Louis' formula sometimes produces a non-positive definite information matrix due to an unsmooth likelihood surface. *Post-convergence approximated standard errors* (Yang and Cai, 2014) utilize Louis' formula directly. A set of Monte Carlo samples are drawn after convergence and then these samples are used to approximate the gradients and Hessian matrix. This approach was shown to be more stable for complex

models than the first approach (Yang & Cai, 2014) and was therefore adopted here.

Latent Attribute Estimation

When cMHRM has converged, the examinees' latent attribute estimates can be obtained by expected a posteriori estimation. A set of Monte Carlo samples of α are drawn and the proportions of each α are calculated. α is set to 1 if the corresponding proportion is larger than 0.5, otherwise to 0 (George, Robitzsch, Kiefer, Groß, & Ünlü, 2016).

Q Matrix Estimation

The mode of the samples of Q is taken as the maximum a posteriori estimator to avoid the problem of column permutations (Chen et al., 2017). First, a set of Monte Carlo samples of Q is drawn after convergence. Then, the bijection method is used to efficiently find the mode. Define the mapping vector $\varpi^T = [2^{J-1}, 2^{J-2}, \ldots, 1]$ for a column vector of Q. With the mapping vector, an index vector for all column vectors of Q can be obtained by calculating $\varpi^T Q$. Next, to eliminate the problem of column permutations, the elements of each index vector are arranged in decreasing order. Finally, the occurrences of each unique vector in the Q samples are counted and the vector appearing most often (corresponding to the mode of Q) is taken as the estimate of the Q matrix.

Monte Carlo Simulations

The primary goal of the simulations is to investigate the recovery of item parameters and the Q matrix in plausible situations. The performance of cMHRM is compared with the constrained Gibbs sampler (cGibbs) of Chen et al. (2017), because cGibbs is also a data-driven approach that can simultaneously estimate the high-dimensional attributes and Q, and appeared to perform best in most of their simulation conditions. Moreover, an implementation of cGibbs in C++ is available from the supplementary material of Chen et al. (2017).

Design

For comparison purposes, our simulation design resembled that of Chen et al. (2017) who indicated that the performance of Bayesian estimators (including cGibbs) was sensitive to the sample size and the number of attributes. Thus, sample sizes were set to 500 and 4000, and the number of attributes were 3, 4, and 5. Two Q matrices with $K = 3$ and $K = 4$ from

Chen et al. (2017) and one Q matrix with $K = 5$ from de la Torre and Douglas (2004) were used (see Appendix A for details). Note that all three Q matrices satisfy the aforementioned identification constraints.

The DINA model was used to generate data with $g_j = s_j = 0.2$ for each item j (Culpepper, 2015). The correlation between attributes, ρ, was set to 0 and .25, respectively. For $\rho = 0$, a uniform distribution with equal class probabilities was used. For $\rho = .25$, the α were characterized by the underlying continuous random vector θ with a multivariate normal distribution with zero means and equal correlations. The entries in α were generated by

$$\alpha_k = \begin{cases} 1 & \text{if } \theta_k \geq 0 \\ 0 & \text{otherwise} \end{cases}, \quad k = 1, ..., K . \tag{27}$$

The posterior distribution tends to be highly multi-modal due to the large Q and attribute matrices. If there is no prior information regarding Q, using multiple starting values for Q is suggested (Chen et al., 2015; Sun et al., 2016). Then, starting values for attributes and item parameters can be generated via observed scores based on the starting values of Q (see Rupp et al., 2010, p. 252). We ran cMHRM with several sets of starting values, and selected the set that led to the maximum log-likelihood value. Twenty sets of starting values were randomly generated, which appeared to be sufficient in these simulations. For these starting values, M was changed to 1 and thus the estimates were very likely perturbed by Monte Carlo errors. We found that the perturbation tends to push cMHRM away from local maxima. In practice, it is efficient to run a short warm up period first (e.g., 50 iterations) for each set of starting values and then proceed with the set that produces a maximum log-likelihood value, calculated as the average of the last 20 iterations. M was then changed back to 5 for the selected set of starting values to stabilize the final estimation. This approach was found effective in that cMHRM usually approaches the neighborhood of a stationary point quickly within a few iterations (cf. Cai, 2010).

Regarding cGibbs, Chen et al. (2017) did not explicitly state how many sets of starting values they used. Our simulation results (see Results section) suggest that they ran cGibbs with a single set of starting values (denoted as cGibbs1). The burn-in length of cGibbs was

set to 15,000, followed by 30,000 samples for cGibbs1 (Chen et al., 2017). We further

considered multiple starting values, using twenty sets of starting values (denoted as

cGibbs20). To make computation feasible for cGibbs20, we used a 5,000 burn-in length,

followed by drawing 5,000 samples, for each set of starting values. Because MCMC is a

sampling method rather than a pointwise estimation method, we selected the set that

produced the lowest Watanabe-Akaike information criterion (WAIC; Gelman, Hwang, &

Vehtari, 2014; Watanabe, 2010). Next, we used the obtained parameter estimates as starting

values to run cGibbs again to draw 10,000 samples. Finally, the drawn samples were

averaged to obtain parameter estimates.

The number of replications was 100 and the simulated data for each replication were

analyzed by cMHRM and cGibbs. The set-up for cMHRM was the same as described

previously. For the Q matrix estimator, the full recovery rate (the percentage of replications

where all elements of Q were recovered) and the entry-wise average rate were calculated. The

criteria to assess parameter recovery were the bias ($R^{-1} \sum_{r=1}^{R} (\hat{\zeta}_r - \zeta)$) and the root-mean-

squared error (RMSE; $\sqrt{R^{-1} \sum_{r=1}^{R} (\hat{\zeta}_r - \zeta)^2}$), where $\hat{\zeta}_r$ is the estimator, ζ is the true

parameter, and R is the number of replications. Note that the 'true' parameter π given $\rho > 0$

was not fixed in the 100 replications. We simulated 10,000 sets of attributes by Equation (27)

and recorded the value of π. Finally, the average of all the simulated π was regarded as our

estimate of the 'true' value. Also note that Chen et al. (2017) only reported the mean squared

errors (i.e., squared RMSE) of item parameter estimators (see their Figure 1 and 2) and did

not consider the performance of structural parameter estimators and $N = 4000$.

Results

Table 1 reports the full recovery rate and entry-wise average rate for cGibbs1, cGibbs20,

and cMHRM. For $N = 500$, the full recovery rates were similar for cMHRM and cGibbs20

and were better than cGibbs1 in some cases. cMHRM and cGibbs20 performed better than or

equal to cGibbs1 regarding entry-wise average rates. Notably, cMHRM and cGibbs20

performed equally well, suggesting that multiple starting values are beneficial for both

methods. All algorithms tended to perform worse as K increased for $N = 500$, which might be

because the item response information became insufficient when the number of estimated elements of Q increased. Correlated attributes did not appear to affect the Q estimators. As the sample size increased to $N = 4000$, cGibbs1 became inaccurate in most cases. For $I = 18$, cGibbs1 performed worse as K increased from 3 to 4, which is consistent with the results in Chen et al. (2017). In contrast, cMHRM and cGibbs20 were 100% accurate in all cases. It is evident that cMHRM and cGibbs20 were accurate for both large and small sample sizes for our simulation conditions.

<Table 1 here>

Regarding item parameter recovery, we present results for three conditions for illustration purposes. For the first condition ($K = 3$ and $N = 500$, and $\rho = .25$) the three methods performed equally well and were 100% accurate in recovering Q; the second condition ($K = 5$ and $N = 500$, and $\rho = 0$) yielded the lowest accuracy (94%) for cMHRM and cGibbs20 among all conditions; for the third condition ($K = 4$ and $N = 4000$, and $\rho = 0$) cGibbs1 had the lowest accuracy (57%). Figure 1 shows that the three methods have similar bias and RMSE for guessing and slipping estimators for the first condition. In addition, cMHRM and cGibbs20 yielded similar bias and RMSE for the class probability estimator, whereas cGibbs1 had slightly larger RMSE.

<Figure 1 here>

Figure 2 shows that cMHRM and cGibbs20 produced similar results for the guessing parameter and class probability for the second condition. However, cGibbs1 tended to yield upward bias for the slipping parameter as the number of required attributes increased. In contrast, cMHRM and cGibbs20 produced bias much closer to zero for most of the items.

<Figure 2 here>

Figure 3 shows that cGibbs1 was biased for most of the parameters for the third condition, likely a result of the inaccurate recovery of the Q matrix. cMHRM and cGibbs20, on the other hand, yielded biases close to zero and considerably lower RMSEs than cGibbs1. For the other simulation conditions the bias and RMSE are given in Appendix B for completeness. In general, we see that cMHRM and cGibbs20 performed better than cGibbs1.

<Figure 3 here>

The absolute computational efficiency for cMHRM (coded in R) and cGibbs (coded in C++) was compared based on the elapsed CPU time on a computer with 8 GB RAM, 3.40 GHz Intel Core processor, and 64-bit OS. The results are shown in Table 2 where we note that correlated attributes did not impact the CPU time.

<Table 2 here>

We see that cMHRM is somewhat faster than cGibbs1 for most conditions and considerably faster than cGibbs20 for all conditions examined.

Relative computational efficiency can be assessed by the ratio of the elapsed CPU time when increasing N or K. For all comparisons regarding an increase in the sample size N from 500 to 4000, cMHRM had superior efficiency. For example, the computational burden increased 8.06 times for cGibbs20 when N changed from 500 to 4000 for $K = 5$, whereas it only increased about 5.42 times for cMHRM. With respect to increasing the number of attributes, cGibbs20 was more efficient for an increase of K from 3 to 4 but cMHRM was more efficient for an increase of K from 4 to 5. For example, cGibbs20 and cMHRM increased the computational burden 2.23 times and 1.71 times, respectively, when K increased from 4 to 5 for $N = 500$.

In conclusion, cMHRM seems to be considerably more computationally efficient than cGibbs20.

Other Simulation Conditions

Further simulation conditions were investigated to address two issues that were not considered above— (a) using only one set of starting values for Q and (b) using a large K.

The first condition investigated the effect on Q matrix recovery when only one set of starting values for Q is used for cMHRM. We considered randomly generated starting values when no prior knowledge regarding Q is available and investigated this issue under the previous condition with $K = 5$, $N = 500$, $\rho = 0$, and $I = 30$. The full recovery rate and entry-wise average recovery rate for cMHRM were 75 and 94, respectively, which were inferior to recovery rates of 94 and 99.96, respectively, when multiple starting values were used. Note that cMHRM is a pointwise algorithm that can only achieve local maxima and cMHRM with one set of random starting values is therefore not recommended in practice.

The second condition is concerned with the computational efficiency of cMHRM and cGibbs20 when the dimension of the binary sampling space becomes large. We considered the Q matrix from TIMSS 2011 Mathematics for 4th Grade Austrian Students for the purpose of this simulation (see data.timss11.G4.AUT\$q.matrix1, George et al., 2016). The original Q matrix has 174 items and 9 attributes, and each item solely measures one of the 9 attributes. Because cGibbs20 was very slow for $K = 9$, we chose 27 items out of the 174 items to form a reduced Q where each attribute was measured by three items. We simulated data based on the DINA model with $g_j = s_j = 0.2$ for each item, $\rho = 0$, and $N = 1,000$. As the binary space is very large for $K = 9$, we considered two situations— 1) where starting values for Q were randomly generated, and 2) where approximately 2/3 of the entries of the starting values for Q were correctly specified.

The average CPU time (in minutes) for one replication was 1524 and 67 for cGibbs20 and cMHRM, respectively, showing that cGibbs20 required much more computation time than cMHRM when K was as large as 9. Regarding the full recovery rate and entry-wise average recovery rate when starting values for Q were randomly drawn, cGibbs20 yielded 9 and 89, whereas cMHRM yielded 11 and 88. In contrast, when using informative starting values for Q, both algorithms produced 100% accuracy for the full and entry-wise average recovery rates. The results suggest that randomly drawn starting values for Q did not perform well, likely because of the high-dimensional binary space of Q.

Examples

Fraction-Subtraction Data

We consider Tatsuoka's fraction-subtraction data (Tatsuoka, 2002; Tatsuoka, 1984) and apply cMHRM to estimate the Q matrix and item and structural parameters of the DINA model. This dataset has been widely used for demonstration purposes (e.g., Chen et al., 2017; Chen et al., 2015; de la Torre, 2008; de la Torre & Douglas, 2004; DeCarlo, 2011) and we have for this reason used it to compare cMHRM with cGibbs1. The dataset comprises 536 middle school students responding to 20 items (coded as *correct* = 1 and *incorrect* = 0). The full dataset and Q matrix are available in the **CDM** package in R (see fraction.subtraction.data, George et al., 2016).

The original expert-derived Q matrix consists of eight skills ($K = 8$) that students have to master: (1) convert a whole number to a fraction, (2) separate a whole number from a fraction, (3) simplify before subtraction, (4) find a common denominator, (5) borrow from the whole number part, (6) column borrow to subtract the second numerator from the first, (7) subtract numerators, and (8) reduce answers to the simplest form. Chen et al. (2017) indicated that eight attributes might be too many for 20 items, because several attributes are required at the same time (e.g., Attribute 7) that could lead to lack of identification. Therefore, $K = 3$ and $K = 4$ were considered here and the Q matrix was assumed unknown and estimated from the data (Chen et al., 2017; Chen et al., 2015). After convergence of cMHRM, 100,000 samples of α and Q were drawn given fixed item and structural parameters and the posterior mode of samples of Q was used as the estimate of Q. Concurrently, 100,000 samples were drawn to calculate the log-likelihood and standard error estimates of item and structural parameters.

Table 3 shows the estimated Q matrix, guessing and slipping parameters, and standard errors of guessing and slipping estimators for $K = 3$ and $K = 4$. The results for cGibbs20 and cMHRM were identical. For $K = 3$, none of the Geweke (1992) statistics for continuous parameters were significant at the 5% level for cGibbs20, suggesting that there were no convergence problems. Compared to Table 3 of Chen et al. (2017) when $K = 3$ (see their 'MH' column with results identical to cGibbs1) the entries of the estimated Q were the same except for the third attribute of Item 7 (denoted by an asterisk). The interpretation of the estimated Q given by Chen et al. (2017) was: (1) finding a common denominator, (2) borrowing from the integer part, and (3) applying subtraction for integer and fraction parts separately. In our estimated Q, Item 7, ($3 - 2\frac{1}{5}$), seemed to only require the first attribute for the examinees. The estimates of guessing and slipping parameters were very close to each other. The standard error estimates were generally lower than 0.04. We may compare our estimated Q with that of cGibbs1 in terms of model-data fit. First, we fixed the two Q matrices and ran cMHRM. Next, the model with the higher log-likelihood was considered the better model, because the two models had the same number of parameters. Our model yielded –4519.2 and their model –4519.4, which is a small difference. In the context of DCMs,

George & Robitzsch (2015) suggested to assess absolute model fit by using a standardized root mean square residual (SRMSR; Maydeu-Olivares, 2013) smaller than 0.05 to indicate a well-fitting model. The SRMSR was .086 and .091 for cGibbs1 and cMHRM, respectively, indicating that neither model has reasonable absolute model fit.

<Table 3 here>

Our estimated Q matrix was identical to that from cGibbs20 for $K = 4$. Again, none of the Geweke statistics were significant for cGibbs20, suggesting that there were no convergence problems for this algorithm. Comparing our estimated Q to cGibbs1 (see the 'MH' column of Table 4 in Chen et al., 2017), the first two attributes were identical for all items. However, there were two items that were different for the third attribute and three items that were different for the fourth attribute. In addition, the interpretation of the estimated attributes requires post hoc reasoning, which seems to be demanding in this case. Domain experts may interpret the results together with test constructors to confirm the elements of the estimated Q. Our item parameter estimates were close to those of Chen et al. (2017). By fixing the estimated Q of Table 2 and the estimated Q of Chen et al. (2017), we calculated the log-likelihood. Our model yielded –4414.7 and their model –4474.9 (cGibbs1), suggesting that the fit of our model was superior to theirs. cMHRM yielded a SRMSR of .077 and cGibbs1 a SRMSR of .081, suggesting that the DINA model does not have a good absolute fit to these data.

In addition to model fit (e.g., SRMSR), we assessed item fit by using the so-called item-fit root mean square error of approximation (item-fit RMSEA; Kunina-Habenicht, Rupp, & Wilhelm, 2009; von Davier, 2005) and the S-X2 statistic (Orlando & Thissen, 2000). For item-fit RMSEA, < .05 represents good fit, < .10 moderate fit, and > .10 poor fit, according to Kunina-Habenicht, Rupp, and Wilhelm (2009). For K = 3, eight items had at least moderate fit using the item-fit RMSEA, whereas the other items did not fit well. The S-X2 statistics suggested that nine of the items had reasonable fit (p-value > .05). Three of the items fit well according to both criteria. For K = 4, five items had at least moderate fit according to the item-fit RMSEA, whereas the S-X2 statistics suggest that 13 of the items fit well (p-value > .05). Four items fit reasonably well by both criteria. In general, the DINA model did not fit

the data well.

Examination for the Certificate of Proficiency in English Data

The Examination for the Certificate of Proficiency in English (ECPE) data comprises 2,922 examinees and 28 items. This dataset has been analyzed with DCMs including the DINA model in several studies (e.g., Buck & Tatsuoka, 1998; George & Robitzsch, 2015; Templin & Bradshaw, 2014; Templin & Hoffman, 2013). Three attributes purported to underlie the ECPE were (1) morphosyntactic rules, (2) cohesive rules, and (3) lexical rules (Buck & Tatsuoka, 1998). George and Robitzsch (2015) provided partial evidence showing that the DINA model could yield at least satisfactory model fit with the three attributes. Although a linear hierarchy of attributes has been studied for the ECPE data (Templin & Bradshaw, 2014), we focus on Q matrix estimation with the DINA model here. Previous studies assumed a correctly specified Q and did not address estimation of Q. In our analysis, interest lies in whether cMHRM and cGibbs would yield different Q matrix estimates, item and structural estimates, and model-data fit, when 40 sets of randomly generated starting values are used for both methods (denoted as cMHRM and cGibbs40). In addition to $K = 3$, previous studies had also considered the $K = 2$ condition by merging α_1 with α_3 or α_2 with α_3 (George & Robitzsch, 2015). We also considered higher K; that is, we estimated Q for $K = 2$, 3, 4, and 5. The EM algorithm was also used for the ECPE data with an expert-specified Q ($K = 3$) for comparison purposes.

To compare the model fit for EM, cGibbs40, and cMHRM, we first estimated the Q matrix with cGibbs40 and cMHRM and then derived the log-likelihood value (LL), Bayesian information criterion (BIC; Schwarz, 1978), and SRMSR given the estimated Q.

Table 4 shows the LL, BIC, and SRMSR, for the EM (with expert-specified Q), cGibbs40, and cMHRM, with $K = 2$, 3, 4, and 5. The results indicate that $K = 3$ yielded the lowest BIC for both cGibbs40 and cMHRM, compared with the expert-specified Q. The solutions were the same for cGibbs40 and cMHRM for all K except $K = 5$. None of the Geweke statistics were significant for cGibbs40, suggesting that there were no convergence problems. Regarding the log-likelihood values for $K = 3$, cMHRM had -42770 which is larger than the -42843 of EM, suggesting that estimating Q yields a better fit. Based on the criterion

that a SRMSR smaller than 0.05 indicates a well-fitting model, the DINA model with $K = 3$ yields good model-data fit.

<Table 4 here>

Regarding item fit for $K = 3$, all the item-fit RMSEA values were below .05 (average = .022) and the S-X2 statistics also suggested that most items fit reasonably well, apart from Item 2 (p-value = .025), Item 12 (p-value = .044), and Item 26 (p-value = .008). Overall, the DINA model with $K = 3$ fitted the data quite well.

A comparison between parameter estimates for EM and cMHRM is shown in Table 5. In general, the guessing and slipping estimates for the two methods were close, but the Q matrix estimates were different for many of the items. This analysis provides partial evidence that the expert-specified Q might need to be re-examined, in collaboration with domain experts and test developers.

<Table 5 here>

Table 6 shows the structural parameter estimates ($\hat{\pi}$) for the ECPE dataset using EM (with expert-specified Q) and cMHRM. The two methods yielded somewhat similar patterns. It is clear that most of the students possess either Pr([0,0,0]) = .30 or Pr([1,1,1]) = .46 for cMHRM, and EM yielded a similar result. Students having skill α_1 seem to also have α_3 in that Pr([1,1,1]) + Pr([1,0,1]) = .53, compared to students having skill α_1 but not skill α_3 because Pr([1,1,0]) + Pr([1,0,0]) = .12. Similarly, students having skill α_2 also seem to have α_3 in that Pr([1,1,1]) + Pr([0,1,1]) = .50, compared to students having skill α_2 but not skill α_3 because Pr([1,1,0]) + Pr([0,1,0]) = .13. The results imply that α_3 (lexical rules) could be a prerequisite for either α_1 (morphosyntactic rules) or α_2 (cohesive rules) (George & Robitzsch, 2015; Templin & Bradshaw, 2014).

<Table 6 here>

Table 7 displays the cross-classification count for the attribute estimates (MAP) for EM and cMHRM. The matching classification (i.e., percentage of person estimates for which the two methods gave identical estimates) was 95%. The imperfect matching may be attributed to different Q matrices.

<Table 7 here>

Concluding Remarks

Instead of relying on prior knowledge regarding the Q matrix, cMHRM treats Q and the latent attributes as missing data imputed via a Bayesian MCMC sampler with identification constraints on Q. This is followed by maximum likelihood estimation and the application of the RM algorithm to obtain estimates of the item and structural parameters. Thus, cMHRM is a data-driven approach that is similar in spirit to exploratory factor analysis where the loading parameters are completely unknown. The cost of enjoying data-driven flexibility is that we have to provide a meaningful interpretation of the estimated Q, which is generally not an easy task in practice. Therefore, we emphasize that the estimated Q should only serve as suggestive regarding the relation between attributes and items which domain experts should verify or modify before formal testing.

For cMHRM we have utilized a specific identifiability condition proposed by Xu and Shang (2017). For restricted latent class models, such as the DINA model considered here, Gu and Xu (2018) classified the identifiability condition into three categories— strict, partial, and generic. The identifiability condition used in our study is an instance of the strict case. The other types of identifiability conditions could in principle also be used for cMHRM. However, a challenge would be how to sample Q under these conditions and a new MCMC sampler would have to be developed and investigated (e.g., whether the Markov chain is irreducible). If such MCMC samplers can be developed, it would be straightforward to use them in cMHRM.

Our simulation results suggest that cMHRM usually outperformed the cGibbs1 method of Chen et al. (2017) in small and large sample sizes with regard to recovery of the Q matrix and item parameters. For a small sample size (i.e., 500), cMHRM performed very well for K = 3. Although the performance slightly decreased as K increased, cMHRM still produced 94% accuracy in Q matrix recovery and a 99.96% entry-wise average rate for $K = 5$. For a large sample size (i.e., 4000), cMHRM had 100% accuracy for all cases in our simulations. In contrast, cGibbs1 performed acceptably for the small sample size but not for the large sample size, which is consistent with the results of Chen et al. (2017). For both approaches, correlations between attributes did not affect the results. When using multiple sets of starting

values for $\boldsymbol{Q}$, cGibbs20 performed identically to cMHRM in terms of $\boldsymbol{Q}$ matrix recovery. In summary, cMHRM and cGibbs20 are preferred over cGibbs1 for both small and large sample sizes.

We may also compare the results of the regularization approach of Chen et al. (2015) with our proposed approach. Regularization performed similar to cMHRM in terms of full recovery rate for $N = 4000$; however, for $N = 500$, regularization yielded 98% for $K = 3$ but just 30% for $K = 4$ (see Table 4 in Chen et al. (2015)), whereas cMHRM reached 100% and 98%, respectively. Regarding the entry-wise average, cMHRM still outperformed regularization (i.e., 100% > 99.9% and 99.96% > 97.6%, respectively), a conclusion that also applies to conditions with correlated attributes (i.e., $\rho = .25$). This suggests that cMHRM is preferred over the regularization approach for small sample sizes.

Estimation of $\boldsymbol{Q}$ matrices is challenging when K is very large. In our simulation for $K = 9$, the sampling space of $\boldsymbol{Q}$ becomes huge which meant that cMHRM and cGibbs20 did not perform well with randomly-generated starting values for $\boldsymbol{Q}$. In such situations, we showed that using starting values based on partial knowledge of $\boldsymbol{Q}$ could mitigate this issue and enhance $\boldsymbol{Q}$ matrix recovery.

Prior information on $\boldsymbol{Q}$ matrices can be embedded into cMHRM, although our focus has been on a data-driven cMHRM. Equation (17) can be modified as

$$
\begin{aligned}
&\Pr\left(q_{jk}^{(t)} \mid \mathbf{y}, \boldsymbol{Q}_{-jk}^{(t)}, \boldsymbol{Q}_{-jk}^{(t+1)}, \boldsymbol{\xi}^{(t)}, \boldsymbol{\pi}^{(t)}, \boldsymbol{\alpha}^{(t+1)}\right) \\
&\propto \Pr\left(\mathbf{y} \mid q_{jk}^{(t)}, \boldsymbol{Q}_{-jk}^{(t)}, \boldsymbol{Q}_{-jk}^{(t+1)}, \boldsymbol{\xi}^{(t)}, \boldsymbol{\pi}^{(t)}, \boldsymbol{\alpha}^{(t+1)}\right) \Pr\left(q_{jk}^{(t)}\right)
\end{aligned},
\tag{28}
$$

where $\Pr\left(q_{jk}^{(t)}\right)$ represents the prior distribution. In practice, domain experts may just be uncertain about some $\boldsymbol{q}$'s in $\boldsymbol{Q}$. $\Pr(q) = 1$ means the q is assumed known based on prior knowledge. The $\Pr\left(q_{jk}^{(t)}\right)$ could be determined by the votes ($q = 1$ or $q = 0$) of all the domain experts and the proportions of the votes can be used as prior information. Prior information tends to smooth the log-likelihood surface and improves the prospects of obtaining a global solution.

Although our attention has been mainly devoted to the DINA model, cMHRM is applicable to the GDCM because the item response probability function can be user-

specified. However, the GDCM usually needs complex inequality item constraints to maintain monotonicity of the item response probability (Henson et al., 2009). In addition, sufficient and necessary conditions for the identifiability of the GDCM should be investigated and included in the cMHRM (Gu & Xu, 2018). To achieve such a goal, however, new MCMC samplers for Q that respect different types of identification conditions should be developed and incorporated (Gu & Xu, 2018; Zhang, Chen, & Liu, 2018). When applying cMHRM to the GDCM we note that an additional issue arises. Considering, for instance, the GDCM with main effects (i.e., only containing the first and second terms in Equation 1), the numerator of the acceptance probability function (Equation 16) cannot be computed when $q_{jk}^{(t)} = 0$ and $q_{jk}^{(\text{new})} = 1$, because the value of β_{jk} is undetermined for $q_{jk}^{(\text{new})} = 1$. One potential way to address this issue is by sampling a random value of β_{jk} from its prior distribution, which can be a distribution that it is convenient to sample from (e.g., a uniform or normal distribution), so that the numerator of the acceptance probability function can be computed. The other parts of the cMHRM algorithm remain unchanged. However, exploring such a large binary sample space could be slow if the prior does not resemble the posterior distribution.

There are several notable advantages of our cMHRM algorithm compared to previous approaches. First, cMHRM alleviates the difficulty of high-dimensional summations in the expectation step of the EM algorithm, having a computational burden that is linearly proportional to K and J instead of exponentially proportional. Second, cMHRM automatically respects constraints on the Q matrix by using the constrained MH sampler. Such identification constraints will ensure an identified solution that corresponds to a peak of the likelihood surface (for more discussion, see Chen et al., 2017). Thus, cMHRM can be used as a data-driven technique for DCMs when experts want to validate their derived Q matrix or when Q is completely unknown. Third, cMHRM does not depend on the hierarchical structure of DCMs in contrast to Gibbs sampling which is only applicable when the posterior distribution of each parameter can be derived (Chen et al., 2017; Culpepper, 2015, 2019). Fourth, standard errors can be readily estimated by the post-convergence approximated standard errors. In contrast, the computation of standard errors is somewhat convoluted in the

EM algorithm (Philipp, Strobl, de la Torre, & Zeileis, 2017). Lastly, we note that the cMHRM algorithm is computationally superior to the cGibbs algorithm, having a substantially faster convergence when estimating the DCM model parameters.

References

Agresti, A., & Hitchcock, D. B. (2005). Bayesian inference for categorical data analysis. *Statistical Methods & Applications, 14*(3), 297-330.

Bock, R. D., & Lieberman, M. (1970). Fitting a response model forn dichotomously scored items. *Psychometrika, 35*(2), 179-197.

Buck, G., & Tatsuoka, K. (1998). Application of the rule-space procedure to language testing: Examining attributes of a free response listening test. *Language Testing, 15*(2), 119-157.

Cai, L. (2010). High-dimensional exploratory item factor analysis by a Metropolis–Hastings Robbins–Monro algorithm. *Psychometrika, 75*(1), 33-57.

Chen, Y., Culpepper, S. A., Chen, Y., & Douglas, J. (2017). Bayesian estimation of the DINA Q matrix. *Psychometrika 83*(1), 89-108.

Chen, Y., Liu, J., Xu, G., & Ying, Z. (2015). Statistical analysis of Q-matrix based diagnostic classification models. *Journal of the American Statistical Association, 110*(510), 850-866.

Culpepper, S. A. (2015). Bayesian estimation of the DINA model with Gibbs sampling. *Journal of Educational and Behavioral Statistics, 40*(5), 454-476.

da Silva, M. A., de Oliveira, E. S. B., von Davier, A. A., & Bazán, J. L. (2018). Estimating the DINA model parameters using the No-U-Turn Sampler. *Biometrical Journal*, 60(2), 352-368.

de la Torre, J. (2008). An empirically based method of Q-matrix validation for the DINA model: Development and applications. *Journal of Educational Measurement, 45*(4), 343-362.

de la Torre, J. (2011). The generalized DINA model framework. *Psychometrika, 76*(2), 179-199.

de la Torre, J., & Chiu, C.-Y. (2016). A general method of empirical Q-matrix validation. *Psychometrika, 81*(2), 253-273.

de la Torre, J., & Douglas, J. A. (2004). Higher-order latent trait models for cognitive diagnosis. *Psychometrika, 69*(3), 333-353.

DeCarlo, L. T. (2011). On the analysis of fraction subtraction data: The DINA model, classification, latent class sizes, and the Q-matrix. *Applied Psychological Measurement, 35*(1), 8-26.

DeCarlo, L. T. (2012). Recognizing uncertainty in the Q-matrix via a Bayesian extension of the DINA model. *Applied Psychological Measurement, 36*(6), 447-468.

Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological), 39*(1), 1-38.

Diebolt, J., & Ip, E. H. S. (1996). Stochastic EM: Method and application. In W. R. Gilks, S. Richardson & D. J. Spiegelhalter (Eds.), *Markov chain Monte Carlo in practice* (pp. 259–273). London: Chapman and Hall.

Gelman, A., Hwang, J., & Vehtari, A. (2014). Understanding predictive information criteria for Bayesian models. *Statistics and Computing, 24*(6), 997-1016.

George, A. C., & Robitzsch, A. (2015). Cognitive diagnosis models in R: A didactic. *The Quantitative Methods for Psychology, 11*(3), 189-205.

George, A. C., Robitzsch, A., Kiefer, T., Groß, J., & Ünlü, A. (2016). The R package CDM for cognitive diagnosis models. *Journal of Statistical Software, 74*(2), 1-24.

George, E. I., & McCulloch, R. E. (1997). Approaches for Bayesian variable selection. *Statistica Sinica, 7*(2), 339-373.

Gu, M. G., & Kong, F. H. (1998). A stochastic approximation algorithm with Markov chain Monte-Carlo method for incomplete data estimation problems. *Proceedings of the National Academy of Sciences, 95*(13), 7270-7274.

Haertel, E. H. (1989). Using restricted latent class models to map the skill structure of achievement items. *Journal of Educational Measurement, 26*(4), 301-321.

Hartz, S. M. (2002). *A Bayesian framework for the unified model for assessing cognitive abilities: Blending theory with practicality.* Unpublished doctoral dissertation, University of Illinois at Urbana-Champaign.

Henson, R. A., Templin, J. L., & Willse, J. T. (2009). Defining a family of cognitive diagnosis models using log-linear models with latent variables. *Psychometrika, 74*(2), 191-210.

Junker, B. W., & Sijtsma, K. (2001). Cognitive assessment models with few assumptions, and connections with nonparametric item response theory. *Applied Psychological Measurement, 25*(3), 258-272.

Kunina-Habenicht, O., Rupp, A. A., & Wilhelm, O. (2009). A practical illustration of multidimensional diagnostic skills profiling: Comparing results from confirmatory factor analysis and diagnostic classification models. *Studies in Educational Evaluation*, 35, 64–70.

Liu, J. (2017). On the consistency of Q-matrix estimation: A commentary. *Psychometrika, 82*(2), 523-527.

Liu, J., Xu, G., & Ying, Z. (2012). Data-driven learning of Q-matrix. *Applied Psychological Measurement, 36*(7), 548-564.

Liu, J., Xu, G., & Ying, Z. (2013). Theory of the self-learning Q-matrix. *Bernoulli, 19*(5A), 1790.

Liu, J. S. (1996). Peskun's theorem and a modified discrete-state Gibbs sampler. *Biometrika,*

83(3), 681-682.

Louis, T. A. (1982). Finding the observed information matrix when using the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological), 44*(2), 226-233.

Macready, G. B., & Dayton, C. M. (1977). The use of probabilistic models in the assessment of mastery. *Journal of Educational Statistics, 2*(2), 99-120.

Madigan, D., York, J., & Allard, D. (1995). Bayesian graphical models for discrete data. *International Statistical Review, 63*(2), 215-232.

Maydeu-Olivares, A. (2013). Goodness-of-fit assessment of item response theory models. *Measurement: Interdisciplinary Research and Perspectives, 11*(3), 71-101.

Orlando, M., & Thissen, D. (2000). Likelihood-based item-fit indices for dichotomous item response theory models. *Applied Psychological Measurement*, 24(1), 50-64.

Perks, W. (1947). Some observations on inverse probability including a new indifference rule. *Journal of the Institute of Actuaries (1886-1994), 73*(2), 285-334.

Philipp, M., Strobl, C., de la Torre, J., & Zeileis, A. (2017). On the estimation of standard errors in cognitive diagnosis models. *Journal of Educational and Behavioral Statistics, 43*(1), 88-115.

Robbins, H., & Monro, S. (1951). A stochastic approximation method. *The Annals of Mathematical Statistics, 22*(3), 400-407.

Rupp, A. A., & Templin, J. L. (2008). The effects of Q-matrix misspecification on parameter estimates and classification accuracy in the DINA model. *Educational and Psychological Measurement, 68*(1), 78-96.

Rupp, A. A., Templin, J. L., & Henson, R. A. (2010). *Diagnostic measurement: Theory, methods, and applications*. New York: Guilford Press.

Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics, 6*(2), 461-464.

Sun, Y., Ye, S., Su, G., & Sun, Y. (2016, 11-13 Nov. 2016). *Q-matrix learning and DINA model parameter estimation.* Paper presented at the 2016 International Conference on Behavioral, Economic and Socio-cultural Computing (BESC), Durham, NC, USA

Tatsuoka, C. (2002). Data analytic methods for latent partially ordered classification models. *Journal of the Royal Statistical Society: Series C (Applied Statistics), 51*(3), 337-350.

Tatsuoka, K. K. (1984). *Analysis of errors in fraction addition and subtraction problems* (Final Report for NIE-G-81-0002). University of Illinois, Urbana-Champaign.

Templin, J. L, & Bradshaw, L. (2014). Hierarchical diagnostic classification models: A family of models for estimating and testing attribute hierarchies. *Psychometrika, 79*(2), 317-339.

Templin, J. L, & Hoffman, L. (2013). Obtaining diagnostic classification model estimates using Mplus. *Educational Measurement: Issues and Practice, 32*(2), 37-50.

Templin, J. L., & Henson, R. A. (2006). Measurement of psychological disorders using

cognitive diagnosis models. *Psychological methods, 11*(3), 287.

von Davier, M. (2008). A general diagnostic model applied to language testing data. *British Journal of Mathematical and Statistical Psychology*, 61(2), 287-307.

von Davier, M. (2014). The DINA model as a constrained general diagnostic model: Two variants of a model equivalency. *British Journal of Mathematical and Statistical Psychology, 67*(1), 49-71.

von Davier, M., & Sinharay, S. (2010). Stochastic approximation methods for latent regression item response models. *Journal of Educational and Behavioral Statistics, 35*(2), 174-193.

Wang, S. (2018). Two-stage maximum likelihood estimation in the misspecified restricted latent class model. *British Journal of Mathematical and Statistical Psychology*, 71(2), 300-333.

Wang, W., Song, L., Ding, S., Meng, Y., Cao, C., & Jie, Y. (2018). An EM-based method for Q-matrix validation.. *Applied Psychological Measurement*, 42(6), 446-459.

Watanabe, S. (2010). Asymptotic equivalence of Bayes cross validation and widely applicable information criterion in singular learning theory. *Journal of Machine Learning Research, 11*(Dec), 3571-3594.

Wei, G. C., & Tanner, M. A. (1990). A Monte Carlo implementation of the EM algorithm and the poor man's data augmentation algorithms. *Journal of the American Statistical Association, 85*(411), 699-704.

Xu, G. (2017). Identifiability of restricted latent class models with binary responses. *The Annals of Statistics, 45*(2), 675-707.

Xu, G., & Shang, Z. (2017). Identifying latent structures in restricted latent class models. *Journal of the American Statistical Association*, 113(423), 1284-1295.

Yang, J. S., & Cai, L. (2014). Estimation of contextual effects through nonlinear multilevel latent variable modeling with a Metropolis–Hastings Robbins–Monro algorithm. *Journal of Educational and Behavioral Statistics, 39*(6), 550-582.

Table 1. Full recovery rate and entry-wise average recovery rate for cMHRM (with 20 sets of random starting values of Q) versus cGibbs1 (with one set of random starting values of Q) versus cGibbs20 (with 20 sets of random starting values of Q) by number of attributes (K), sample size (N), correlations (ρ), and test length (I).

K	N	ρ	I	Full recovery rate			Entry-wise average rate		
				cGibbs1	cGibbs20	cMHRM	cGibbs1	cGibbs20	cMHRM
3	500	0	18	99	100	100	99.60	100	100
3	500	.25	18	100	100	100	100	100	100
4	500	0	18	96	98	98	99.40	99.96	99.96
4	500	.25	18	98	99	99	99.70	99.99	99.99
5	500	0	30	94	94	94	99.70	99.96	99.96
5	500	.25	30	92	94	94	99.60	99.96	99.96
3	4000	0	18	87	100	100	95.80	100	100
3	4000	.25	18	90	100	100	96.50	100	100
4	4000	0	18	57	100	100	86.00	100	100
4	4000	.25	18	75	100	100	92.80	100	100
5	4000	0	30	82	100	100	95.10	100	100
5	4000	.25	30	88	100	100	96.90	100	100

Table 2. Program run time comparisons of absolute computational efficiency for cMHRM (with 20 sets of random starting value of Q), cGibbs1 (with one set of random starting value of Q, 15,000 burn-in, and 30,000 samples) and cGibbs20 (with 20 sets of random starting value of Q, 5,000 burn-in, 5,000 samples, and finally 10,000 samples), based on average elapsed CPU time (in minutes) for $\rho = 0$ (correlation) and $I = 18$ (test length).

K	N	cGibbs1	cGibbs20	cMHRM
3	500	3.03	9.67	2.80
4	500	5.05	15.90	6.57
5	500	10.04	35.49	11.24
3	4000	18.07	72.76	13.11
4	4000	28.91	117.86	26.76
5	4000	102.3	286.27	60.95

Table 3. Estimated $\mathbf{Q}$, slipping (s), and guessing (g) parameters for Tatsuoka's fraction-subtraction data using cMHRM.

		cMHRM												
		$K=3$						$K=4$						
Item	Content	$\widehat{\mathbf{Q}}$			$\hat{s}$	SE	$\hat{g}$	SE	$\widehat{\mathbf{Q}}$				$\hat{s}$	SE	$\hat{g}$	SE
1	$\frac{5}{3}-\frac{3}{4}$	1	0	0	0.14	0.02	0.03	0.01	1	0	0	0	0.11	0.02	0.03	0.02
2	$\frac{3}{4}-\frac{3}{8}$	1	0	0	0.07	0.01	0.05	0.02	1	0	0	0	0.04	0.02	0.04	0.01
3	$\frac{5}{6}-\frac{1}{9}$	1	0	0	0.14	0.02	0.00	0.01	1	0	0	0	0.12	0.02	0.00	0.04
4	$3\frac{1}{2}-2\frac{3}{2}$	0	1	0	0.12	0.02	0.20	0.02	0	1	0	0	0.13	0.02	0.21	0.02
5	$4\frac{3}{5}-3\frac{4}{10}$	1	0	1	0.21	0.02	0.31	0.03	1	0	1	0	0.20	0.02	0.31	0.03
6	$\frac{6}{7}-\frac{4}{7}$	0	0	1	0.04	0.01	0.30	0.04	0	0	1	0	0.05	0.01	0.27	0.04
7	$3-2\frac{1}{5}$	1	0	0*	0.35	0.03	0.03	0.02	0	0	0	1	0.21	0.03	0.00	0.02
8	$\frac{2}{3}-\frac{2}{3}$	1	1	0	0.05	0.01	0.58	0.03	0	0	0*	1*	0.07	0.02	0.57	0.03
9	$3\frac{7}{8}-2$	0	0	1	0.25	0.02	0.34	0.04	0	0	1	0	0.25	0.02	0.32	0.04
10	$4\frac{4}{12}-2\frac{7}{12}$	0	1	0	0.23	0.03	0.02	0.01	0	1	0	0	0.23	0.03	0.02	0.01
11	$4\frac{1}{3}-2\frac{4}{3}$	0	1	1	0.07	0.02	0.07	0.02	0	1	1	0	0.08	0.02	0.07	0.02
12	$1\frac{1}{8}-\frac{1}{8}$	0	0	1	0.09	0.02	0.19	0.04	0	0	1	0	0.10	0.02	0.17	0.04
13	$3\frac{3}{8}-2\frac{5}{6}$	1	1	1	0.35	0.03	0.02	0.01	1	1	1	0*	0.33	0.03	0.02	0.01
14	$3\frac{4}{5}-3\frac{2}{5}$	0	0	1	0.06	0.01	0.09	0.03	0	0	1	0	0.07	0.01	0.05	0.03
15	$2-\frac{1}{3}$	1	0	1	0.25	0.03	0.05	0.02	0	0	1*	1	0.12	0.02	0.03	0.01
16	$4\frac{5}{7}-1\frac{4}{7}$	0	0	1	0.11	0.02	0.11	0.03	0	0	1	0	0.12	0.02	0.10	0.03
17	$7\frac{3}{5}-2\frac{4}{5}$	0	1	1	0.13	0.02	0.05	0.01	0	1	1	0	0.14	0.02	0.04	0.01
18	$4\frac{1}{10}-2\frac{8}{10}$	0	1	1	0.15	0.02	0.13	0.02	0	1	1	0*	0.15	0.02	0.13	0.02
19	$4-1\frac{4}{3}$	1	1	1	0.33	0.03	0.02	0.01	0	1	1	1	0.27	0.03	0.02	0.01
20	$4\frac{1}{3}-1\frac{5}{3}$	0	1	1	0.18	0.02	0.01	0.01	0	1	1	0	0.19	0.03	0.01	0.01

Note. * indicates that outcome is opposite to Table 4 of Chen, Culpepper, Chen, and Douglas (2017) (see column 'MH', whose results were identical to 'cGibbs'); 'SE' refers to standard error.

Table 4. Model-data fit with different number of attributes (K = 2, 3, 4, and 5) for Examination for Certificate of Proficiency in English (ECPE) dataset using EM (with expert-specified $\boldsymbol{Q}$), cGibbs40, and cMHRM.

	Log-likelihood			BIC			SRMSR		
K	EM	cGibbs40	cMHRM	EM	cGibbs40	cMHRM	EM	cGibbs40	cMHRM
2		-42833	-42833		86138	86138		.034	.034
3	-42843	-42770	-42770	86190	86042	86042	.033	.032	.032
4		-42748	-42748		86062	86062		.031	.031
5		-42738	-42733		86170	86160		.032	.031

Note. BIC: Bayesian information criterion; SRMSR: standardized root mean square residual; 40 sets of random starting values for $\boldsymbol{Q}$ were used for cGibbs40 and cMHRM.

Table 5. Estimated $\boldsymbol{Q}$, slipping (s), and guessing (g) parameters for Certificate of Proficiency in English (ECPE) dataset using EM (with expert-specified $\boldsymbol{Q}$) and cMHRM.

						$K = 3$									
	EM							cMHRM							
Item	$\boldsymbol{Q}$			$\hat{s}$	SE	$\hat{g}$	SE	$\widehat{\boldsymbol{Q}}$			$\hat{s}$	SE	$\hat{g}$	SE
1	1	1	0	0.09	0.01	0.71	0.01	1	1	1	0.09	0.01	0.71	0.01
2	0	1	0	0.10	0.01	0.72	0.02	0	0	1	0.11	0.01	0.72	0.02
3	1	0	1	0.27	0.01	0.44	0.01	1	1	1	0.26	0.01	0.44	0.01
4	0	0	1	0.16	0.01	0.48	0.02	0	1	1	0.15	0.01	0.51	0.02
5	0	0	1	0.04	0.01	0.76	0.01	0	0	1	0.04	0.01	0.75	0.02
6	0	0	1	0.07	0.01	0.72	0.01	0	0	1	0.07	0.01	0.71	0.02
7	1	0	1	0.09	0.01	0.54	0.01	1	0	0	0.09	0.01	0.48	0.02
8	0	1	0	0.04	0.01	0.80	0.01	0	1	1	0.05	0.01	0.82	0.02
9	0	0	1	0.20	0.01	0.53	0.02	1	1	0	0.17	0.01	0.57	0.02
10	1	0	0	0.16	0.01	0.48	0.01	1	1	1	0.16	0.01	0.50	0.02
11	1	0	1	0.10	0.01	0.56	0.01	1	1	1	0.09	0.01	0.56	0.01
12	1	0	1	0.31	0.01	0.20	0.01	1	1	1	0.29	0.01	0.20	0.01
13	1	0	0	0.12	0.01	0.63	0.01	1	0	0	0.13	0.01	0.60	0.02
14	1	0	0	0.21	0.01	0.52	0.01	1	1	1	0.20	0.01	0.53	0.01
15	0	0	1	0.04	0.01	0.75	0.01	0	0	1	0.04	0.01	0.74	0.02
16	1	0	1	0.13	0.01	0.55	0.01	1	0	0	0.13	0.01	0.49	0.02
17	0	1	1	0.06	0.01	0.82	0.01	1	0	0	0.06	0.01	0.81	0.01
18	0	0	1	0.09	0.01	0.73	0.01	0	0	1	0.08	0.01	0.72	0.02
19	0	0	1	0.15	0.01	0.47	0.02	0	1	0	0.14	0.01	0.46	0.03
20	1	0	1	0.30	0.01	0.24	0.01	1	1	1	0.28	0.01	0.24	0.01
21	1	0	1	0.10	0.01	0.62	0.01	0	1	0	0.12	0.01	0.54	0.02
22	0	0	1	0.19	0.01	0.32	0.02	0	0	1	0.19	0.01	0.30	0.02
23	0	1	0	0.08	0.01	0.64	0.02	0	1	1	0.08	0.01	0.67	0.01
24	0	1	0	0.32	0.01	0.31	0.02	0	1	1	0.33	0.01	0.35	0.02
25	1	0	0	0.27	0.01	0.51	0.01	1	1	1	0.27	0.01	0.52	0.01
26	0	0	1	0.21	0.01	0.56	0.02	0	0	1	0.21	0.01	0.54	0.02
27	1	0	0	0.37	0.01	0.27	0.01	1	1	0	0.36	0.02	0.25	0.01
28	0	0	1	0.09	0.01	0.66	0.02	0	1	0	0.08	0.01	0.65	0.02

Table 6. Structural parameter estimates ($\widehat{\pi}$) for Certificate of Proficiency in English (ECPE) dataset using EM (with expert-specified Q) and cMHRM.

Attribute Pattern				K = 3	
α_1	α_2	α_3		EM	cMHRM
0	0	0	$\widehat{\pi}_1$	0.31	0.30
0	0	1	$\widehat{\pi}_2$	0.05	0.00
0	1	0	$\widehat{\pi}_3$	0.04	0.01
0	1	1	$\widehat{\pi}_4$	0.10	0.04
1	0	0	$\widehat{\pi}_5$	0.01	0.00
1	0	1	$\widehat{\pi}_6$	0.03	0.07
1	1	0	$\widehat{\pi}_7$	0.01	0.12
1	1	1	$\widehat{\pi}_8$	0.45	0.46

Table 7. Cross-classification count for attribute estimates for Certificate of Proficiency in English (ECPE) dataset using EM (with expert-specified Q) and cMHRM.

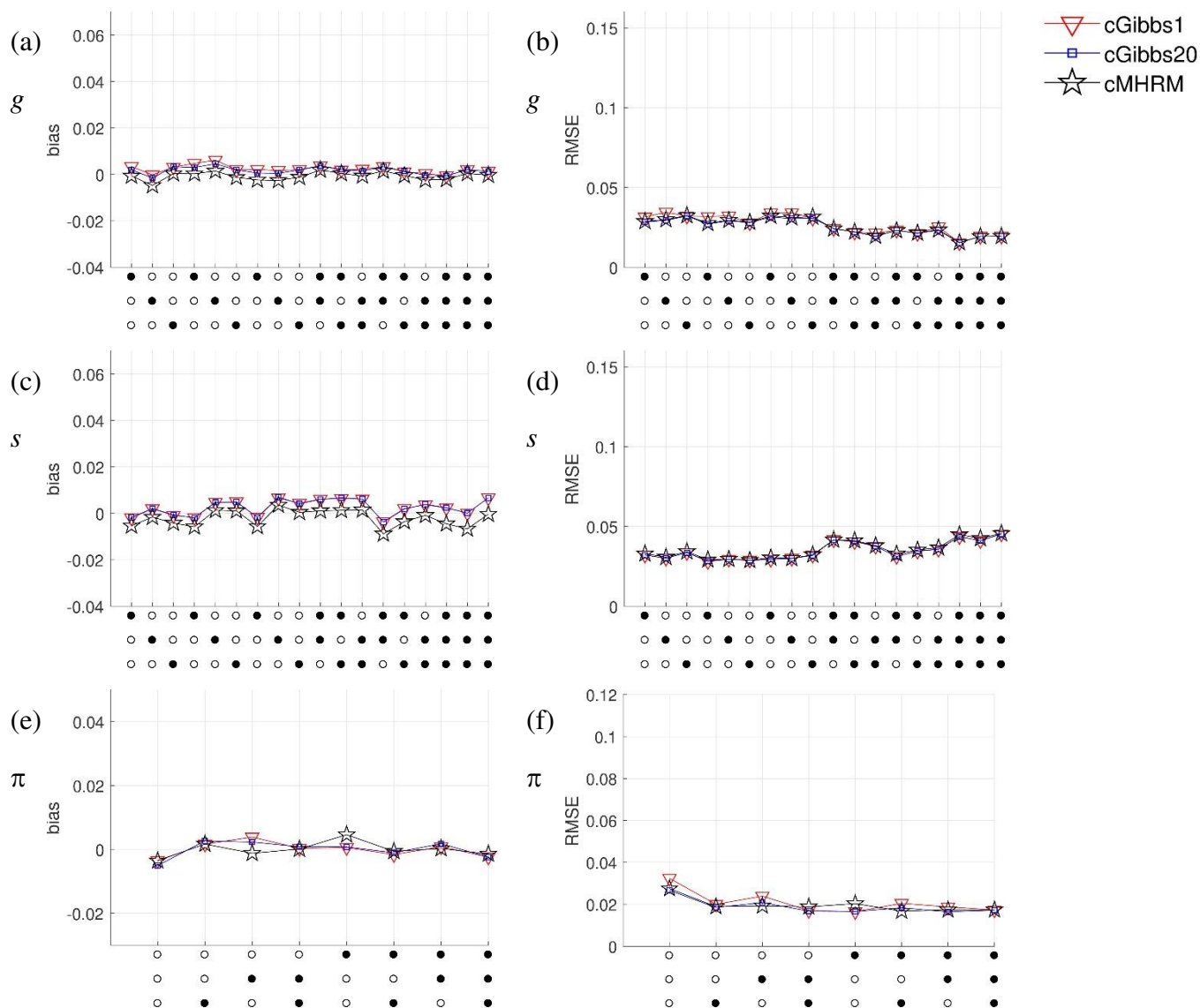| EM	cMHRM							
	α_{000}	α_{001}	α_{010}	α_{011}	α_{100}	α_{101}	α_{110}	α_{111}
α_{000}	878	6	22	7	19	1	5	0
α_{001}	0	0	0	0	0	0	0	0
α_{010}	0	0	0	0	0	0	0	0
α_{011}	0	3	3	75	0	1	0	5
α_{100}	0	0	0	0	0	0	0	0
α_{101}	0	0	0	0	5	143	0	3
α_{110}	0	0	10	0	9	0	290	21
α_{111}	0	0	0	2	2	8	9	1395

Figure 1. Bias and root-mean-square error (RMSE) for item parameter estimators (*g*: guessing parameter, *s*: slip parameter), and bias and root-mean-square square (RMSE) for class probability parameter (π), comparing cMHRM (using 20 sets of random starting values for Q), cGibbs1 (using one set of random starting values for Q), and cGibbs20 (using 20 sets of random starting values for Q). Q-matrix with $K = 3$ and 18 items with 500 examinees generated for $\rho = .25$ under DINA model.
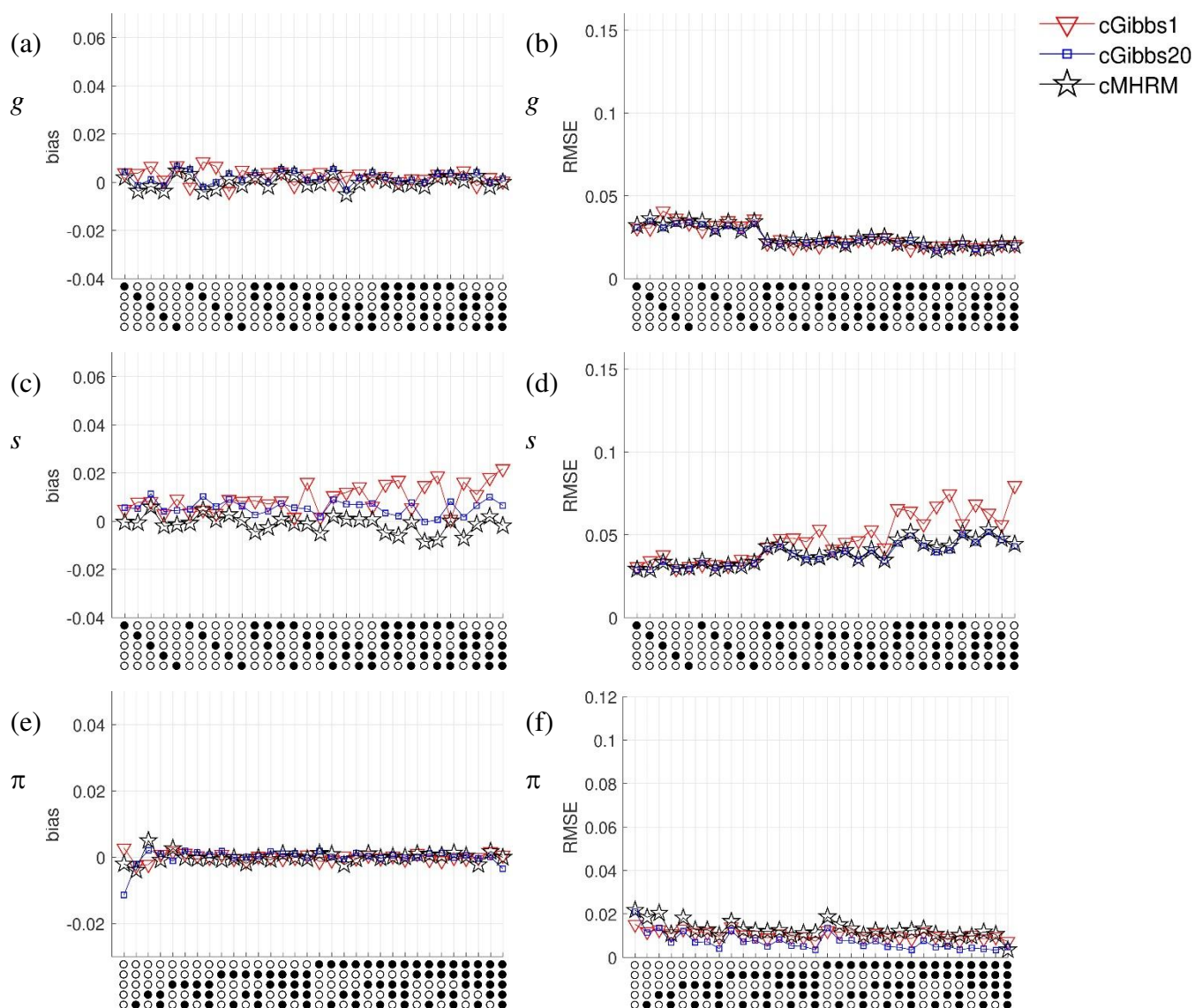
Figure 2. Bias and root-mean-square error (RMSE) for item parameter estimators (*g*: guessing parameter, *s*: slip parameter) and class probability parameter (π), comparing cMHRM (using 20 sets of random starting values for Q) with cGibbs1 (using one set of random starting values for Q) and cGibbs20 (using 20 sets of random starting values for Q). Q-matrix with $K = 5$ and 30 items with 500 examinees generated for $\rho = 0$ (uniform distribution) under DINA model.
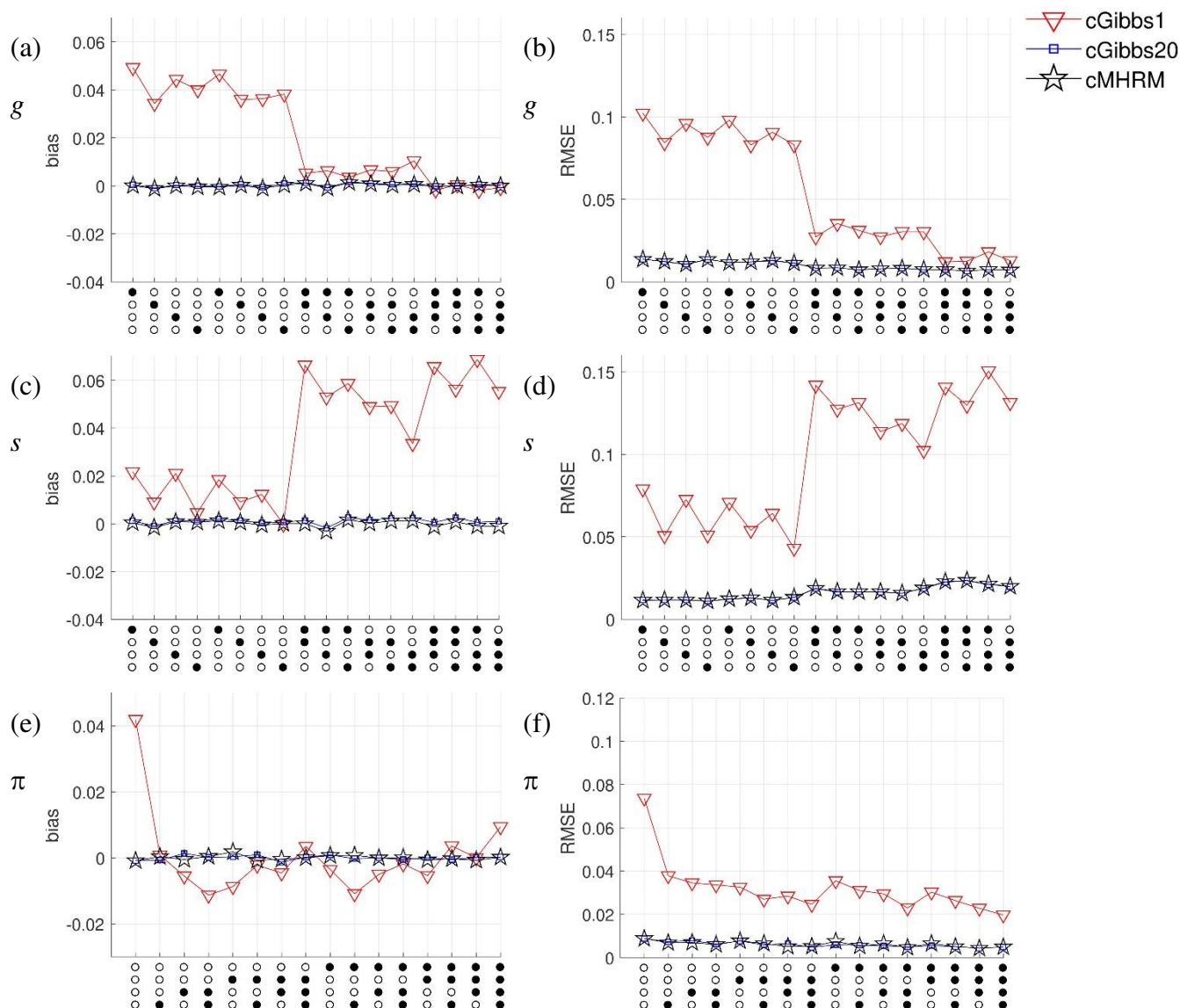
Figure 3. Bias and root-mean-square error (RMSE) for item parameter estimators (g: guessing parameter, s: slip parameter) and class probability parameter (π), comparing cMHRM (using 20 sets of random starting values for $\boldsymbol{Q}$) with cGibbs1 (using one set of random starting values for $\boldsymbol{Q}$) and cGibbs20 (using 20 sets of random starting values for $\boldsymbol{Q}$). Q-matrix with $K = 4$ and 18 items with 4000 examinees generated for $\rho = 0$ (uniform distribution) under DINA model.

Click here to access/download
Supplementary Material (must not contain author information)
Appendix4.docx