Sequential Gibbs Sampling Algorithm for Cognitive Diagnosis Models with Many Attributes

Juntao Wang\textsuperscript{a}, Ningzhong Shi\textsuperscript{a}, Xue Zhang\textsuperscript{b}, and Gongjun Xu\textsuperscript{c}

\textsuperscript{a}Department of Statistics, KLAS, School of Mathematics and Statistics, Northeast Normal University, Changchun, China; \textsuperscript{b}China Institute of Rural Education Development, Northeast Normal University, Changchun, China; \textsuperscript{c}Department of Statistics, University of Michigan, Ann Arbor, Michigan, USA

\textbf{ABSTRACT}

Cognitive diagnosis models (CDMs) are useful statistical tools to provide rich information relevant for intervention and learning. As a popular approach to estimate and make inference of CDMs, the Markov chain Monte Carlo (MCMC) algorithm is widely used in practice. However, when the number of attributes, $K$, is large, the existing MCMC algorithm may become time-consuming, due to the fact that $O(2^K)$ calculations are usually needed in the process of MCMC sampling to get the conditional distribution for each attribute profile. To overcome this computational issue, motivated by Culpepper and Hudson's earlier work in 2018, we propose a computationally efficient sequential Gibbs sampling method, which needs $O(K)$ calculations to sample each attribute profile. We use simulation and real data examples to show the good finite-sample performance of the proposed sequential Gibbs sampling, and its advantage over existing methods.

\textbf{KEYWORDS}

Cognitive diagnosis model; Markov chain Monte Carlo; sequential Gibbs sampling

\section*{Introduction}

In recent years, cognitive diagnosis models (CDMs) have gained great achievements in educational and psychological assessments, where latent binary random vectors are often assumed to represent the presence or absence of multiple fine-grained skills or attributes. The CDMs can be viewed as a family of restricted latent class models, with the goal of achieving personalized diagnostic classification. Compared with the Item Response Theory (IRT) models, the CDMs can provide more informative feedbacks on attribute profiles and allow for the design of more effective intervention strategies (Rupp et al., 2010).

Many CDMs have been proposed in the literature. An incomplete list contains the Deterministic Input, Noisy “And” gate and Noisy Inputs, Deterministic “And” gate models (DINA and NIDA; Haertel, 1989; Junker & Sijtsma, 2001), the reduced version of the Reparameterized Unified Model (rRUM; Hartz, 2002; Rupp et al., 2010), the Deterministic Input, Noisy “Or” gate and Noisy Inputs, Deterministic “Or” gate models (DINO and NIDO; Templin & Henson, 2006), the general diagnostic model (GDM; von Davier, 2005), the log-linear cognitive diagnosis model (LCDM; Henson et al., 2009), and the generalized DINA model (GDINA; de la Torre, 2011).

To estimate the CDM parameters and perform classification of examinees, the Bayesian MCMC method is one popular approach, as it will not only provide the point estimation but also the whole posterior distributional information for statistical inferences. In the Bayesian framework, the MCMC algorithm is used to generate the unique stationary distribution that weakly converges to the true target distribution of parameters of interest. The MCMC algorithm provides a useful tool to solve many complicated problems in statistics and psychometrics. In the CDM literature, the Bayesian MCMC estimation of CDMs has also been studied. For instance, under the confirmatory setting with the Q-matrix prespecified, Culpepper (2015) proposed an efficient Gibbs sampling for the DINA model, in which all parameters were sampled from their full conditional distributions. Chung (2014, 2019) estimated the DINA and rRUM models in the Bayesian framework using a Gibbs sampling algorithm.

Culpepper and Hudson (2018) further proposed a Bayesian sequential Gibbs sampler of the rRUM which samples each latent attribute sequentially from the
corresponding conditional Bernoulli distribution. In addition, the software "JAGS" has also been used to fit many common CDMs (e.g. Zhan et al., 2019). In the exploratory CDM setting, Bayesian method has also been used to estimate the model parameters and the Q-matrix jointly under identifiability conditions. For instance, Chen et al. (2018) proposed an easily implemented MCMC algorithm through a data augmentation strategy and item parameter reparameterizations. Following Culpepper and Hudson (2018), Culpepper and Chen (2019) proposed a similar sequential sampler to estimate the Q-matrix under the exploratory rRUM.

In modern psychological, educational and medical applications of CDMs, large-scale data, with large numbers of manifest attributes of interest (denoted by K), are often collected. In many applications, the number of the corresponding latent classes 2^K could become comparable or even larger than the number of examinees N. Examples with large number of latent classes can be found in educational assessment (Lee et al., 2011) and the medical diagnosis (Wu et al., 2016). The increasing dimension of attributes and items often causes high computational cost and therefore introduces new challenges for the estimation and inference of the CDMs.

In this paper, we focus on improving the MCMC with the Gibbs sampling in the setting of many latent attributes under the confirmatory CDMs with the Q-matrix prespecified. Existing MCMC algorithms often directly sample from the posterior distribution of each latent attribute profile (see Zhan et al., 2019), with the whole attribute profile treated as one random sample from a categorical distribution with 2^K different categories. Therefore, in order to sample one attribute profile, it is needed to evaluate 2^K posterior probabilities of each possible profile candidate. The corresponding computational overhead for sampling each individual’s attribute profile is of the order O(2^K). For a large K, this would lead to a significant computational burden and also affect the convergence of the MCMC algorithm. Alternatively, Zhan et al. (2015) proposed to model the attributes as independent variables by introducing an independent Bernoulli prior for each attribute, the corresponding sampling method is named as the independent Gibbs sampling in this article. Without modeling the dependence among the attributes, the computation cost of sampling each attribute profile in the independent Gibbs Sampling is then reduced to O(K), however, the independent assumption of attributes is often too strong to satisfy in practice.

Since the computational difficulty for large K mainly arises from the sampling of the attribute profiles, we follow the novel idea of sequential Gibbs sampler proposed in Culpepper and Hudson (2018) to develop an efficient sequential Gibbs sampling method. This work extends Culpepper and Hudson (2018), which focuses on the rRUM model, to the more general GDINA model under the high-dimensional setting with many attributes; such a high-dimensional setting arises in many applications but the related estimation challenge has not been addressed. Following Culpepper and Hudson (2018), the sequential sampler samples each attribute separately instead of sampling the attribute profile as a whole, and consequently, the computational overheads of sampling attribute profiles is greatly reduced from O(2^K) to O(K). For a large K, the improvement is especially significant as shown in the simulation studies.

The rest of the paper is organized as follows. In Section “Bayesian GDINA model,” we give an overview on the CDMs and a Bayesian formulation for the estimation. Section “The sequential Gibbs sampling” introduces the proposed sequential Gibbs sampling, with a focus on the estimation of the GDINA model as a general version of CDMs. The simulations and real data analyses are shown in Section “Simulation studies” and Section “Real data analysis,” respectively. A discussion is given in Section “Discussion.” The supplementary materials include more details for the proposed algorithm. Source code of the proposed method will be made publicly available upon the acceptance of this work.

**Bayesian GDINA model**

This section focus on the GDINA model as the general framework for CDMs, which include many CDMs as special cases such as DINA, DINO, and Reduced RUM (Hartz, 2002; Junker & Sijtsma, 2001; Rupp et al., 2010). We present the formulation for the Bayesian GDINA model, which contains the model setup, item parameter priors and population parameter prior.

**The DINA and GDINA models**

In CDMs, the examinee’s responses depend on his/her latent attribute profile which is denoted by a K-dimensional vector \( \mathbf{a} = (a_1, a_2, ..., a_K)' \), where the superscript ‘ \( ' \) denotes transpose. Each attribute \( a_k \) indicates the mastery of attribute \( k = 1, ..., K \), and there are in total \( C = 2^K \) latent classes \( \mathbf{a} \in \{0, 1\}^K \). Let the binary vector \( \mathbf{Y} = (Y_1, ..., Y_j) \) represent an examinee’s responses to \( j \) items. Both \( \mathbf{a} \) and \( \mathbf{Y} \) are examinee-specific; a particular examinee \( i \)'s attribute profile and
responses are denoted by \( \mathbf{z} \), and \( \mathbf{Y} \) for \( i = 1, \ldots, N \). The \( N \) examinees’ attribute profiles are random samples from a population distribution with the probability: \( \pi_{\mathbf{z}} = P(\mathbf{z} = \mathbf{x}) \), where \( \sum_{\mathbf{x}} \pi_{\mathbf{z}} = 1, 0 \leq \pi_{\mathbf{z}} < 1 \). Thus, the population distribution of attribute profiles is characterized by the vector \( \pi = (\pi_{\mathbf{z}}, \mathbf{z} \in \{0,1\}^K) \).

For notational convenience, for \( \mathbf{z} = (x_1, x_2, \ldots, x_K)^\prime \), we will also write \( \pi_{\mathbf{z}} \) as \( \pi_c \) with \( c = 1 + \sum_{k=1}^{K} 2^{k-1} \). Note that the two representations \( \pi = (\pi_{\mathbf{z}}, \mathbf{z} \in \{0,1\}^K) \) and \( \pi = (\pi_c, c = 1, \ldots, C) \) are equivalent.

The binary Q-matrix (K. K. Tatsuoka, 1983) is a key component for CDMs. For each pair of \( j \) and \( k \), \( q_{jk} = 1 \) indicates attribute \( k \) is required by item \( j \), otherwise \( q_{jk} = 0 \), for \( j = 1, \ldots, J \). Particularly, the \( j \)th row vector \( \mathbf{q} \) of the Q-matrix corresponds to the attributes required by item \( j \).

The DINA model (Haertel, 1989; Junker & Sijtsma, 2001) is one of, if not the simplest, consequently most restrictive, interpretable CDMs available for dichotomously scored tests. For a specific examinee with an attribute profile \( \mathbf{z} \), we can define the ideal response \( \eta(\mathbf{z}, \mathbf{q}) \) to item \( j \) relying on \( \mathbf{z} \) and \( \mathbf{q} \) as:

\[
\eta(\mathbf{z}, \mathbf{q}) = \prod_{k=1}^{K} z_{jk}^{q_{jk}}.
\]

For brevity, given examinee \( i \)’s attribute profile \( \mathbf{z}_i \), the ideal response \( \eta(\mathbf{z}_i, \mathbf{q}_i) \) can also be written as \( \eta_{ij} \) if the context permits. The \( \eta_{ij} \) is an indicator of whether examinee \( i \) masters all the required attributes for item \( j \), which indicates that each item partitions all examinees into two latent groups. Let \( g_j = P(Y_{ij} = 1|\eta_{ij} = 0) \) and \( s_j = P(Y_{ij} = 0|\eta_{ij} = 1) \) be the guessing and slipping parameters, respectively. For examinee \( i \) and item \( j \), the positive response probability, denoted by \( \theta_{j,\mathbf{z}_i} = P(Y_{ij} = 1|\mathbf{z}_i) \), takes the form:

\[
\theta_{j,\mathbf{z}_i} = g_j^{1-\eta_{ij}}(1-s_j)^{\eta_{ij}}.
\]

de la Torre (2011) proposed a general framework for CDMs based on the DINA model, called the GDINA model, which characterized more complex relationships between attribute profiles and response data. In the GDINA model, the positive response probability can be decomposed into the sum of the effects due the presence of required attributes and their interactions. We let \( K_j^* = \sum_{k=1}^{K} q_{jk} \) be the number of required attributes by item \( j \), which is determined by the \( j \)th row vector \( \mathbf{q}_j \) in the Q-matrix. For a specific examinee with \( \mathbf{z} \) and item \( j \), we rearrange the structure of attribute profiles, so that the first \( K_j^* \) attributes are the attributes required by item \( j \). The reduced attribute profile for item \( j \) consists of the first \( K_j^* \) required attributes denoted by \( \mathbf{z}_j^* = (z_{j1}^*, \ldots, z_{jK_j^*}^*)^\prime, j = 1, \ldots, J \). Similarly to \( \mathbf{z} \) and \( \mathbf{Y} \), there also exists the examinee-specific reduced attribute profile, denoted by \( \mathbf{z}_i^* = (z_{i1}^*, \ldots, z_{iK_j^*}^*)^\prime \). Given \( \mathbf{z}_j^* \), the item response probability of item \( j \) is modeled as:

\[
h(\theta_{j,\mathbf{z}_i}^*) = \lambda_{j0} + \sum_{k=1}^{K_j^*} \lambda_{jk} z_{jk}^* + \sum_{k=1}^{K_j^*-1} \sum_{k+k=1}^{K_j^*-1} \lambda_{jk0} z_{jk}^* z_{jk'}^* + \cdots
\]

\[
+ \lambda_{j12-K_j^*} \prod_{k=1}^{K_j^*} z_{jk}^*,
\]

where \( h(\cdot) = h(\cdot) \) represents the positive response probability of the examinees with the reduced \( \mathbf{z}_j^* \) to item \( j \), \( h(\cdot) \) is the link function where usually probit, identity, log and logit links can be employed, \( \lambda_{j0} \) is the intercept, \( \lambda_{jk} \) is the main effect corresponding to \( z_{jk}^* \), \( \lambda_{jk0} \) is the two-way interaction corresponding to \( z_{jk}^* \) and \( z_{jk'}^* \), \( \lambda_{j12-K_j^*} \) is the \( K_j^* \)-way interaction corresponding to all required attributes. We let \( \lambda = (\lambda_0, \lambda_1, \ldots, \lambda_{K_j^*-1}, \lambda_{j12-K_j^*})^\prime \) represent the item parameters for item \( j \) and \( \lambda = (\lambda_1, \ldots, \lambda_I) \) represent the item parameters for all items. The number of item parameters is determined by the structure of the Q-matrix. Specifically, for item \( j \), the number of item parameters is \( 2^{K_j^*} \). In this paper, we shall focus on the probit link function, whereas the proposed method can be applied to other link functions as well.

Under the GDINA model, each item \( j \) can divide examinees into \( 2^{K_j^*} \) latent groups. Because \( \mathbf{z}_j^* \) is a sub-vector of \( \mathbf{z} \), we should notice that \( \theta_{j,\mathbf{z}_i} = \theta_{j,\mathbf{z}_i^*} \). For Equation (3), we can use the vector-notation to rewrite the positive response probability as follows:

\[
\theta_{j,\mathbf{z}_i} = h(\theta_{j,\mathbf{z}_i}^*) = \mathbf{X}_j^* \lambda_j,
\]

where \( \mathbf{X}_j^* = (1, z_{j1}^*, \ldots, z_{jK_j^*}^*)^\prime, \mathbf{X}_j = (z_{j1}^*, z_{j2}^*, \ldots, z_{jK_j^*}^*)^\prime \) denotes a \( 2^{K_j^*} \)-dimensional vector relying on \( \mathbf{z}_j^* \). For a particular examinee \( i = 1, \ldots, N \), the examinee-specific \( \mathbf{X}_j^* \) is denoted by \( \mathbf{X}_j^* = (1, z_{j1}^*, \ldots, z_{jK_j^*}^*)^\prime \). The GDINA model degenerates to the DINA model by setting all item parameters, except \( \lambda_{j0} \) and \( \lambda_{j12-K_j^*} \), to zero. Then, we can obtain \( g_j = h^{-1}(\lambda_{j0}) \) and \( 1-s_j = h^{-1}(\lambda_{j0} + \lambda_{j12-K_j^*}) \), where \( h^{-1} \) is the inverse function of \( h \).

The collection of positive response probabilities is denoted by a \( J \times C \) matrix \( \Theta = (\theta_{j,\mathbf{z}_i}) \), which may depend on different forms of item parameters in different CDMs. Given the response data and all
attribute profiles, the conditional likelihood function takes the following form:

\[ p(Y|x, \Theta) = \prod_{i=1}^{N} \prod_{j=1}^{J} \theta_{j,x_i}^{Y_{ij}}(1 - \theta_{j,x_i})^{1-Y_{ij}}. \]  

(4)

After integrating attribute profiles, the marginal likelihood function takes the following form:

\[ p(Y|\pi, \Theta) = \prod_{i=1}^{N} \sum_{c_1=1}^{C} \prod_{j=1}^{J} \theta_{j,x_i}^{Y_{ij}}(1 - \theta_{j,x_i})^{1-Y_{ij}}. \]  

(5)

### Priors of measurement models’ parameters

The population proportion parameter \( \pi \) includes the saturated information about the attribute profile distribution in CDMs. The Dirichlet distribution is commonly used as a conjugated prior for \( \pi \), such as Culpepper (2015), Culpepper and Hudson (2018) and Zhan et al. (2019). The specific form of the Dirichlet prior for \( \pi = (\pi_1, \ldots, \pi_C) \) is

\[ \pi \sim \text{Dirichlet}(\delta). \]

where \( \delta = (\delta_1, \ldots, \delta_C)' \) represent a \( C \)-dimensional hyper-parameter vector for \( \pi \).

In different CDMs, item parameters will be presented in the different forms, such as \( g_i \) and \( s_i \) in the DINA model, and \( \lambda_j \) in the GDINA model. For the DINA model, independent Beta distributions, \( \text{Beta}(a_i, b_i) \) and \( \text{Beta}(a_j, b_j) \), are often used as the priors for guessing and slipping parameters, respectively. We may also constraint \( 0 < g_i < 1 - s_i < 1 \), to ensure the model identifiability (Chen et al., 2018; Gu & Xu, 2019; Junker & Sijtsma, 2001; Xu & Zhang, 2016). For the GDINA model, the normal distributions are often taken as priors for the item parameters \( \lambda \) (e.g., Zhan et al., 2019). Specifically, two types of priors are often chosen: one is a multivariate normal distribution, \( \lambda_j \sim N(\mu_k, \Sigma_k) \), as a general choice; the other is a truncated multivariate normal distribution, \( \lambda_j \sim N(\mu_k, \Sigma_k)[I(\lambda_j \in T)] \), which is used to ensure certain monotonicity assumption of the item response function. Here \( I(\cdot) \) denotes the indicator function, \( T = \{T_1, \ldots, T_m, \ldots, T_Y\} \), and each \( T_m \) represents some pre-specified constraint of the \( m \)-th element of \( \lambda_j \). For instance, we may restrict the main effect terms in \( \lambda_j \) to be positive to ensure the monotonicity assumption.

As discussed in the introduction, for a large \( K \), the existing MCMC algorithms with Dirichlet prior for \( \pi \) often suffer from the increasing computational cost of sampling each latent attribute profile \( x_i \) from its conditional distribution, which is a categorical distribution with \( 2^K \) different categories. Therefore, it needs to evaluate \( 2^K \) posterior probabilities of each possible profile candidate to sample each \( x_i \), and the corresponding computational overhead for sampling \( x_i \) is \( O(2^K) \). For a large \( K \), this would lead to a significant computational burden and also affect the convergence of the MCMC algorithm.

### The sequential Gibbs sampling

In this section, we introduce the sequential Gibbs sampling method, which samples each attribute separately and is computationally efficient for large \( K \). The sequential Gibbs sampling algorithm will be derived for the GDINA model. It’s natural to apply the sequential Gibbs sampling to other CDMs, and an example of the DINA model is given in Appendix A.

#### Motivation

With the commonly used Dirichlet prior for \( \pi \), many existing Gibbs sampling methods suggest that the full conditional distribution for \( \pi \) is

\[ p(x|\pi) \propto p(Y|x, \lambda)p(x|\pi), \]  

(6)

where the “\( * \)” represented all the other parameters and responses. To infer a specific examinee’s attribute profile, we need to calculate the posterior probability \( p(x|\pi) \) for \( c = 1, \ldots, C \) to obtain the posterior distribution. For large \( K \), the computation is challenging. Since in this Gibbs sampling method, the whole \( x \) should be sampled simultaneously, hereafter this sampling method is referred to as the *simultaneous Gibbs sampling*.

Following Culpepper and Hudson (2018), we first describe the sequential sampling method for the attributes. Let \( x_{\setminus k} \) denote the sub-vector of \( x \) excluding the \( k \)-th attribute. Based on the fact that knowing \( x_{\setminus k} \) and \( x_k \) is equivalent to knowing the attribute profile \( x \), it’s obvious that \( p(x|\pi) = p(x_{\setminus k}, x_k|\pi) \). According to Bayes’ theorem, given the \( x_{\setminus k} \) and \( \pi \), the conditional probability of \( x_k \) is

\[ p(x_k|x_{\setminus k}, \pi) = \frac{p(x_{\setminus k}, x_k|\pi)}{p(x_{\setminus k}|\pi)} \]

(7)

Considering Equation (7), the full conditional distribution for \( x_k \) is calculated from

\[ p(x_k|*, x_{\setminus k}) \propto p(Y|x, \lambda)p(x_{\setminus k}|x_k, \pi). \]

(8)

In Equation (8), \( p(Y|x, \lambda) \) is the conditional likelihood function.
For the second term on RHS of Equation (8), noticing the binary nature of \( z_k \in \{0, 1\} \), we know that conditional on \( \mathbf{z}_{\setminus k} \) and \( \mathbf{p} \),

\[
\alpha_k(\mathbf{z}_{\setminus k}, \mathbf{p}) \sim \text{Bernoulli}(p_k|\mathbf{z}_{\setminus k}, \mathbf{p})
\]

with

\[
p_k|\mathbf{z}_{\setminus k}, \mathbf{p} = \frac{\sum_{c=1}^{C} \pi_c I(\mathbf{z}_c = \mathbf{z}_{\setminus k})}{\sum_{c=1}^{C} \pi_c I(\mathbf{z}_c = \mathbf{z}_{\setminus k})},
\]

where we use the notation \( \mathbf{z}_{\setminus k} \) to represent the \( \mathbf{z} \) vector corresponding to a general latent class \( c \) excluding the \( k \)-th attribute. The above indicator function \( I(\mathbf{z}_c = \mathbf{z}_{\setminus k}) \) only selects one component from \( \mathbf{p} \), and the indicator function \( I(\mathbf{z}_c = \mathbf{z}_{\setminus k}) \) can select two components from \( \mathbf{p} \). According to the two indicator functions, we can construct a ratio as \( p_k|\mathbf{z}_{\setminus k}, \mathbf{p} \). For different CDMs, the Bernoulli distribution form to describe \( z_k \) remains unchanged. Here we can interpret \( p_k|\mathbf{z}_{\setminus k}, \mathbf{p} \) as the “prior” conditional probability before incorporating any information of the responses. The examinee-specific \( p_k|\mathbf{z}_{\setminus k}, \mathbf{p} \) depends on the population parameter \( \mathbf{p} \) and examinee-specific \( \mathbf{z}_{\setminus k} \); that is, for a specific examinee \( i \) with attribute profile \( \mathbf{z}_i, p_k|\mathbf{z}_i, \mathbf{p} \) depends on \( \mathbf{p} \) and \( \mathbf{z}_i, i = 1, \ldots, N \). When there is no ambiguity, we will write \( p_k|\mathbf{z}_{\setminus k}, \mathbf{p} \) and \( p_k|\mathbf{z}_{\setminus k}, \mathbf{p} \) as \( p_k \) and \( p_k \) in the following.

Equations (8) and (9) imply a sampling method that can sample the latent attributes sequentially one by one. Without loss of generality, the attributes are sampled in an increasing order (i.e., \( z_1, \ldots, z_K \) are sampled in turns). In Table 1, an example with three attributes is presented to show how Equation (9) works. An 8-dimensional vector \( \mathbf{p} \) (i.e., \( K = 3 \)) is used to represent the saturated population information and \( z_1, z_2 \) and \( z_3 \) are generated in turns. The first two rows show a one-to-one mapping between \( \mathbf{p} \) and \( \mathbf{z} \). Similar to Gibbs sampling, an initial value of the attribute profile is needed as the starting point. Without loss of generality, let the initial value of \( \mathbf{p} \) equal to (000). When to sample the first attribute \( z_1, z_2 = z_3 = 0 \) is used in Equation (9), then the first attribute \( z_1 \) can be drawn from a Bernoulli distribution with \( p_1 = \frac{\pi_1}{\pi_1 + \pi_2} \), which is the prior conditional probability of \( z_1 = 1 \) given \( z_2 = z_3 = 0 \). Assuming the realization of the first attribute \( z_1 = 1 \), then we can sample the second attribute \( z_2 \), conditional on \( z_1 = 1 \) and \( z_3 = 0 \), from a Bernoulli distribution with \( p_2 = \frac{\pi_3}{\pi_3 + \pi_4} \) in Table 1. When to sample \( z_3 \), conditional on \( z_1 = 1 \) and \( z_2 = 0 \), from a Bernoulli distribution with \( p_3 = \frac{\pi_5}{\pi_5 + \pi_6} \) in Table 1.

In both of the sequential and simultaneous sampling methods, the sampling of attribute profiles depends on \( \mathbf{p} \) and \( \pi \). However, in the simultaneous Gibbs sampling, each attribute profile is treated as a basic unit, and the joint information \( p(\mathbf{z}|\pi) \) is used to sample \( \mathbf{z} \). This method is very slow when \( K \) is large.

In the sequential Gibbs sampling, each element of the attribute profile is sampled separately from the conditional Bernoulli distribution of \( z_k|\mathbf{z}_{\setminus k}, \mathbf{z} \), which would reduce the computational cost significantly.

### Table 1. A sample with three attributes for the conditional Bernoulli distribution.

| \( \alpha \) | 000 | 100 | 010 | 110 | 001 | 101 | 011 | 111 | \( p_k \) | Prob |
|----------|-----|-----|-----|-----|-----|-----|-----|-----|------|-----|
| \( z_1 = 1 \) | \( z_2 = 0, z_3 = 0, \pi_1 \) | \( \pi_2 \) | \( \pi_3 \) | \( \pi_4 \) | \( \pi_5 \) | \( \pi_6 \) | \( \pi_7 \) | \( \pi_8 \) | \( \pi_9 \) | \( \pi_9 \) |
| \( z_2 = 0 \) | \( z_1 = 1, z_3 = 0, \pi_1 \) | \( \pi_2 \) | \( \pi_3 \) | \( \pi_4 \) | \( \pi_5 \) | \( \pi_6 \) | \( \pi_7 \) | \( \pi_8 \) | \( \pi_9 \) | \( \pi_9 \) |
| \( z_3 = 1 \) | \( z_1 = 1, z_2 = 0, \pi_1 \) | \( \pi_2 \) | \( \pi_3 \) | \( \pi_4 \) | \( \pi_5 \) | \( \pi_6 \) | \( \pi_7 \) | \( \pi_8 \) | \( \pi_9 \) | \( \pi_9 \) |

Notes. The column “\( p_k \)” represents the conditional probability of \( z_k = 1 \). The column “Prob” is the probability of realization \( z_k \) shown in the first column (in this tble, the realizations are \( z_1 = 1, z_2 = 0 \) and \( z_3 = 1 \)).

#### Sequential Gibbs sampling schedules

With the above introduced sequential sampling method for attributes, in this section we derive the Gibbs sampling updates for other model parameters. To illustrate our method, we shall focus on the GDINA model with a probit link function and use the prior settings introduced in Section “Priors of measurement models’ parameters.”

We will use a data augmentation strategy to derive a closed-form Gibbs sampling method for the item parameters. Please note that similar sampling methods have been proposed in the CDM literature (Chen et al., 2020; Culpepper, 2019a, 2019b). Specifically, we introduce the data augmentation process for the examinee with \( \mathbf{z} \) to item \( j \) as follows

\[
Z_j = X_j^T \lambda_j + \epsilon_j,
\]

where \( \epsilon_j \) follows a standard normal distribution and \( Z_j \) is a latent auxiliary variable. The \( Z_j \) is the examinee-specific, and the augmented data of item \( j \) for examinee \( i \) is denoted by \( Z_{ij} \), for \( i = 1, \ldots, N \). For examinee \( i \), the augmented data \( Z_{ij} \) is distributed \( N(X_i^T \lambda_j, 1) \), and the item response \( Y_{ij} \) is defined as \( Y_{ij} = 1 \) if \( Z_j \) is positive, and \( Y_{ij} = 0 \) otherwise.

With the introduced augmented data \( Z \), the Gibbs sampling needs to sample from the four full
conditional distributions: \( p(Z|Y, x, \pi), p(\hat{a}|Y, Z, x, \pi), p(x_{ik}|Y, Z, x_{-ik}, \pi), \) and \( p(\pi|Y, Z, x, \lambda) \).

**Sample augmented data**

For examinee \( i \) and item \( j \), the augmented data is \( Z_{ij} \). Conditional on \( x \), the distribution of \( Z_{ij} \) is independent of the parameter \( \pi \), which means the distributions \( p(Z|Y, x, \pi) \) and \( p(Z|Y, Z, x, \lambda) \) are equivalent.

According to the \( j \)th row vector \( q_j \) in the Q-matrix, we can get the reduced vector \( \mathbf{x}_j^* \). Based on \( \mathbf{x}_j^*, \hat{a}_j \) and \( Y_{ij} \), the augmented data \( Z_{ij} \) is normally distributed with the mean \( \mu_{ij} = X_{ij}^* \hat{a}_j \) and the variance one. The range of \( Z_{ij} \) is determined by \( Y_{ij} \); if \( Y_{ij} = 1, Z_{ij} > 0 \), otherwise \( Z_{ij} \leq 0 \). The augmented data is generated by the formula

\[
Z_{ij}|Y_{ij}, \mathbf{x}_j^*, \hat{a}_j \sim \begin{cases} N(\mu_{ij}, 1) & Y_{ij} = 1 \\ N(\mu_{ij}, 1) & Y_{ij} = 0 \end{cases}.
\] (10)

**Sample item parameters**

In the GDINA model, two considered types of item parameter priors are the multivariate normal distribution and the truncated multivariate normal distribution, which will induce two sampling methods to sample item parameters. The truncated prior is suitable for the case where we have some known constraints on item parameters. The multivariate normal distribution is suitable for the case where we don’t have additional information about item parameters.

For the item parameters, the conditional independence implies \( p(\hat{a}_j|Y, Z, x, \pi) \) and \( p(\hat{a}_j|Y, Z, x) \) are equivalent. To sample the item parameters for item \( j \), the information of all examinees for this item need to be considered. We arrange all examinees’ augmented data about item \( j \) in a vector \( \mathbf{Z}_j = (Z_{ij1}, Z_{ij2}, ..., Z_{ijn})' \). Furthermore, let \( \mathbf{X}_j = (\mathbf{X}_{ij1}', \mathbf{X}_{ij2}', ..., \mathbf{X}_{ijn}')' \) denote an \( N \times 2^K \) matrix relying on \( \mathbf{x}_j^* \) for \( i = 1, ..., N \). Given \( \mathbf{Z}_j \) and \( \mathbf{X}_j \), a linear regression model is obtained as follows:

\[
\mathbf{Z}_j = \mathbf{X}_j \hat{a}_j + \mathbf{e}_j,
\]

where \( \mathbf{e}_j = (e_{ij1}, e_{ij2}, ..., e_{ijn})' \) is a standard random vector from a standard normal distribution. If there are no constraints on the item parameter \( \hat{a}_j \), which follows a the prior \( N(\mu_{a_j}, \Sigma_{a_j}) \), then we can obtain the full conditional distribution (Minka, 2000) whose form is shown as

\[
\hat{a}_j|x, \mathbf{Z}_j \sim N(\hat{\mu}_{a_j}, \hat{\Sigma}_{a_j}),
\] (11)

where \( \hat{\Sigma}_{a_j}^{-1} = \Sigma_{a_j}^{-1} + \mathbf{X}_j'\mathbf{X}_j \) and \( \hat{\mu}_{a_j} = \Sigma_{a_j}^{-1} \mu_{a_j} \). The sampling method using Equation (11) is called the sampling without truncation. The specifics of the derivation can be found in the Appendix B. If the prior of \( \hat{a}_j \) is the truncated distribution \( N(\mu_{a_j}, \Sigma_{a_j})I(k_1 \in T_j) \), we can obtain the closed form for \( \hat{a}_j \)’s full conditional distribution:

\[
\hat{a}_j|x, \mathbf{Z}_j \sim N(\mu_{a_j}, \Sigma_{a_j})I(k_1 \in T_j).
\] (12)

The sampling method using Equation (12) is called the sampling with truncation. The details about how to sample from the truncated multivariate normal distribution will be discussed in the Appendix C.

**Sample attribute profiles**

In the sequential Gibbs sampling, attributes are sampled one by one, instead of the whole attribute profile. For examinee \( i \), if the \( k \)-th attribute \( x_{ik} \) isn’t required by an item, the value of \( x_{ik} \) won’t affect the item’s likelihood. So when sampling attribute \( x_{ik} \), the only we need to pay attention to the items requiring the \( k \)-th attribute. Hence, we define a set \( \Omega_k = \{ j|q_{ik} = 1. k = 1, ..., j \} \), which represents the items which require attribute \( x_k \), and the complementary set of \( \Omega_k \) is defined as \( \Omega_k^c = \{ j|q_{ik} = 0, k = 1, ..., j \} \). Only the items from \( \Omega_k^c \) will affect the inference about \( x_k \).

Assuming item \( j \) belongs to \( \Omega_k \) and giving the reduced attribute profile \( \mathbf{x}_{ij}^* \), the positive response probability \( \theta_{ij} = \Phi(\mathbf{x}_{ij}^*, \hat{a}_j) \). For the specific examinee \( i \), the likelihood function for the \( k \)-th attribute \( x_{ik} \) is

\[
p(Y_{ij}|\mathbf{x}_{ij}^*, \hat{a}_j) = \Phi(\mathbf{x}_{ij}^*, \hat{a}_j)^{Y_{ij}}(1 - \Phi(\mathbf{x}_{ij}^*, \hat{a}_j))^{1-Y_{ij}}
= \Phi(T_{0j}^{ij} + x_{ik} T_{1j}^{ij})^{Y_{ij}}(1 - \Phi(T_{0j}^{ij} + x_{ik} T_{1j}^{ij}))^{1-Y_{ij}}
= \Phi(T_{0j}^{ij} + x_{ik} T_{1j}^{ij})^{Y_{ij}}(1 - \Phi(T_{0j}^{ij} + x_{ik} T_{1j}^{ij}))^{1-x_{ik}}
= \left[ (1 - \Phi(T_{0j}^{ij} + x_{ik} T_{1j}^{ij}))^{1-x_{ik}} \right]^{1-Y_{ij}},
\] (13)

where \( T_{0j}^{ij} \) and \( T_{1j}^{ij} \) are the sums of the terms which don’t contain \( x_{ik} \) and \( T_{1j}^{ij} \) is the sum of the terms related to \( x_{ik} \). If examinee \( i \) masters attribute \( x_k \), the positive response probability is \( \Phi(T_{0j}^{ij} + T_{1j}^{ij}) \), otherwise, the positive response probability is \( \Phi(T_{0j}^{ij}) \). Therefore, the positive response probability is \( \Phi(T_{0j}^{ij} + T_{1j}^{ij}) \). We can obtain a similar expression, \( (1 - \Phi(T_{0j}^{ij} + T_{1j}^{ij}))^{1-x_{ik}} \), can be obtained for the negative response. For example, assume that the vector \( \mathbf{q}_i = (110) \), the third attribute doesn’t affect the positive response probability and the likelihood function. In other words, from the responses on this item we can’t get any information about the third attribute. We show how to calculate

\[
\]
To investigate the first attribute $x_1$, the positive response probability is that

$$
\Phi(\lambda_0 + \lambda_1 x_1 + \lambda_2 x_2 + \lambda_{12} x_1 x_2)
$$

then notations $T_0^0 = \lambda_0 + \lambda_2 x_2$ and $T_0^1 = \lambda_1 + \lambda_{12} x_2$.

According to Equation (13), it’s obvious that only the items in $\Omega_k$ will affect the full conditional distribution for $x_{ik}$. The parameters $\pi$ and $\alpha_{i,k}$ are used to calculate the prior conditional probability for $x_{ik}$, with $p_{ik}$ calculated as in Section “The sequential Gibbs sampling.” Then the full conditional distribution for $x_{ik}$ is calculated by

$$
\sum_{j \in \Omega_k} p(x_{ik} | \text{other variables, } \pi) \propto \prod_{j \in \Omega_k} \Phi(T_0^j + T_1^j)^{x_{ik} j} (1 - \Phi(T_0^j + T_1^j))^{1 - x_{ik} j} \prod_{j \in \Omega_k} \Phi(T_0^j + T_1^j)^{x_{ik} j} (1 - \Phi(T_0^j + T_1^j))^{1 - x_{ik} j}
$$

hence, the full conditional distribution for $x_{ik}$ is Bernoulli(\(\bar{p}_{ik}\)), where the value of $\bar{p}_{ik}$ is given by

$$
\frac{\prod_{j \in \Omega_k} \Phi(T_0^j + T_1^j)^{x_{ik} j} (1 - \Phi(T_0^j + T_1^j))^{1 - x_{ik} j}}{\prod_{j \in \Omega_k} \Phi(T_0^j + T_1^j)^{x_{ik} j} (1 - \Phi(T_0^j + T_1^j))^{1 - x_{ik} j} + \prod_{j \in \Omega_k} \Phi(T_0^j + T_1^j)^{x_{ik} j} (1 - \Phi(T_0^j + T_1^j))^{1 - x_{ik} j} + \prod_{j \in \Omega_k} \Phi(T_0^j + T_1^j)^{x_{ik} j} (1 - \Phi(T_0^j + T_1^j))^{1 - x_{ik} j} + \prod_{j \in \Omega_k} \Phi(T_0^j + T_1^j)^{x_{ik} j} (1 - \Phi(T_0^j + T_1^j))^{1 - x_{ik} j}}
$$

Sample the population parameter

The population parameter $\pi$ is a C-dimensional vector, whose prior is Dirichlet(\(\delta\)). Given $\alpha$, we can calculate the number of examinees within the latent class $c$, $N_c = \sum_{i=1}^N I[x_c, x_c]$, and the vector $N = (N_1, N_2, ..., N_C)$. From the conditional independence, we know $p(\pi | Y, Z, x, \lambda, \delta)$ and $p(\pi | x, \delta)$ are equivalent. And we can write the posterior of $\pi$ as

$$
\pi | x, \delta \sim \text{Dirichlet}(\delta + N).
$$

We summarize the sequential Gibbs sampling for the GDINA model in Algorithm 1. The sequential sampling method can be straightforwardly applied to other CDMs as well, and the DINA example is illustrated in the Appendix.

**Algorithm 1**: Sequential Gibbs Sampling for GDINA models

**Input**: Initialize $\lambda^{(0)}$, $x^{(0)}$, $\pi^{(0)}$, $Y$, $m = 0$, $M$ and specify priors.

**Output**: Markov chains of $\lambda$, $x$, $\pi$.

In this section, the simultaneous Gibbs sampling, independent Gibbs sampling (Zhan et al., 2015), and sequential Gibbs sampling are used to estimate parameters in the DINA and GDINA models. The simulation studies intend to implement on different settings of $K$. However, for large $K$, the simultaneous Gibbs sampling methods don’t work due to the high computational cost, so only the results of independent and sequential Gibbs sampling are shown. To show that the difference among these methods is purely caused by the difference among the sampling techniques rather than the software, we code and compile all these three methods by ourselves. The computation of the simulation study is implemented by Dell XPS with 3.0 GHz Intel Core i7-9700, 24 GB RAM.

The statistical software JAGS (Just Another Gibbs Sampling; Plummer, 2003), as the off-the-shelf sampling method, is also used to implement the simultaneous Gibbs sampling method for the DINA and GDINA models. The JAGS is similar to WinBUGS (Lunn et al., 2000) and OpenBUGS (Fouley & Jaffrčicz, 2010). Zhan et al. (2019) showed how to implement the DINA and linear logistic models (LLM, see Maris, 1999) by JAGS. We use JAGS to analyze the DINA and GDINA models. When using JAGS, the initial values of all parameters are generated by the default way within JAGS. Under the DINA model, another simultaneous sampler, the R package “dina” (Culpepper, 2015), is also used to estimate the model. Our simulation results indicate that “dina” is faster than JAGS and the parameter estimates of “dina” are similar to JAGS. As the R package “dina” can’t handle the GDINA model, the detailed results of “dina” are not shown.

1For the independent Gibbs sampling, we use the independent Bernoulli prior that $x_{ik} \sim \text{Bernoulli}(\bar{p}_{ik})$ and $\bar{p}_{ik} \sim \text{Beta}(1, 1)$, where $i$ and $k$ indicate examinee and attribute, respectively.

2Particularly, using “dina” to run the chains with 2000 iterations for the DINA settings in our simulation with $\{N = 2000, K = 3\}$, $\{N = 2000, K = 5\}$, and $\{N = 2000, K = 7\}$ needs about 20, 82, and 344 seconds, which are faster than JAGS but slower than the proposed method.
**Simulation design**

The attribute profiles are generated from the two following structures.

**Uniform structure**

The uniform structure assumes that all latent classes share the same probability.

**Correlated structure**

Chiu et al. (2009) proposed a correlated structure for attribute profiles, which can be viewed as a special case of the higher-order attribute structure. For each examinee, the $K$-dimensional vector $\mathbf{\theta} = (\theta_1, ..., \theta_K)$ follows a multivariate normal distribution $N(0, \Sigma)$, where the covariance matrix $\Sigma$ has a common correlation $\rho$ as follows

$$
\begin{pmatrix}
1 & \rho \\
\rho & 1
\end{pmatrix},
$$

then attributes are determined by

$$x_k = \begin{cases} 
1 & \text{if } \theta_k > 0, \\
0 & \text{otherwise.}
\end{cases} \quad (18)
$$

Chen et al. (2015) also called this situation as “Dependent Attributes”.

For the DINA and GDINA models, the generation methods of item parameters need to be introduced separately. For the DINA model, we set the guessing and slipping parameters to 0.2. For the GDINA model, another equivalent notation is introduced to make the description of item parameters clear. For item $j$, let $j_{0j}^{(b)}$ and $j_{wj}^{(w)}$ denote the intercept parameter and $w$-way interaction parameter, respectively. We generate $j_{k}$ from a multivariate normal distribution with a diagonal covariance matrix. In particular, the distribution to generate $j_{k}$ is specified as:

$$j_{k}^{(w)} \sim \begin{cases} 
N(-1.2, 0.4^2), & w = 0, \\
N(0.9, 0.2^2)/w^2, & w = 1, ..., K^w. 
\end{cases} \quad (19)
$$

This generation method of item parameters indicates the same-way interactions share similar properties (i.e., the same distribution).

Based on the model identifiability and generic identifiability restrictions (Chen et al., 2020; Gu & Xu, 2020; Xu, 2017; Xu & Zhang, 2016), the $Q$-matrix has this form

$$Q' = \left( I_K, I_K, \tilde{Q}_2, \tilde{Q}_3, \tilde{Q}' \right),$$

where each item in $\tilde{Q}_2$ and $\tilde{Q}_3$ requires two and three attributes, respectively. In addition, we randomly sample non-zero $q$-vectors which require three or fewer attributes to fill $\tilde{Q}$. Given the attribute profiles, the item parameters, the $Q$-matrix, and the response data can be generated. For different $K$, the $Q$-matrices are fixed and shown in Appendix D.

The sampling methods are compared from three aspects: speed, parameter estimation accuracy and classification accuracy. The running times of the sampling methods are used to reflect the speed, and the bias, root mean squared error (RMSE) and mean squared error (MSE) are used to evaluate the accuracy. The average bias, RMSE and MSE, denoted by $\text{Bias}$, $\text{RMSE}$ and $\text{MSE}$, are computed for each type parameter, according to

$$\text{Bias}_\phi = {1 \over H} \sum_{h=1}^{H} {1 \over R} \sum_{r=1}^{R} \left( \hat{\phi}_h - \phi_h \right),$$

$$\text{RMSE}_\phi = {1 \over H} \sum_{h=1}^{H} {1 \over R} \sum_{r=1}^{R} \left( \hat{\phi}_h - \phi_h \right)^2,$$

$$\text{MSE}_\phi = {1 \over H} \sum_{h=1}^{H} {1 \over R} \sum_{r=1}^{R} \left( \hat{\phi}_h - \phi_h \right)^2,$$

where $\hat{\phi}_r$ denotes the estimation from $r$-th replication of a parameter, $\phi_h$ denotes the true value, and $R$ denotes the number of replications (i.e., for guessing and slipping parameters, $H=J$; for the item parameters in the GDINA model, $H=K$). The total number of item parameters denoted by $\#(\lambda_i)$; for population parameters, $H=C$). The subscript $\phi$ of the indices is used to discriminate the types of parameters. On the other hand, two widely used indices, attribute-wise agreement rate (AAR) and pattern-wise agreement rate (PAR), can be used to examine the classification accuracy. The AAR = $\sum_{r=1}^{R} \sum_{i=1}^{N} \sum_{k=1}^{K} I \{ \hat{q}_{ik} = q_{ik} \}/RN$ can be used for arbitrary $K$. The PAR = $\sum_{r=1}^{R} \sum_{i=1}^{N} I \{ \sum_{k=1}^{K} | \hat{q}_{ik} - q_{ik} | = 0 \}/RN$, however, is usually too small to provide valid information about the classification accuracy when $K$ is large. Hence, we define a more practical measure, PARn = $\sum_{r=1}^{R} \sum_{i=1}^{N} I \{ \sum_{k=1}^{K} | \hat{q}_{ik} - q_{ik} | \leq n \}/RN$, which denotes that, for one pattern, up to $n$ misestimated attributes can be tolerated. When $n$ is a positive integer, PARn is a relaxation of PAR (i.e., PAR0).

| Method       | Sim | Seq | Ind | S  | M  | L  | Ind |
|--------------|-----|-----|-----|----|----|----|-----|
| Number of items | $J = 30$ |
| Attribute structure | uniform or correlations $\rho = 0.3$ and $0.7$ |
| Replications   | $R = 25$ |
| Examinee sample sizes | $N = 1000$ and $2000$ |
| Chain length   | 2000 |
| Burn-in        | 1000 |

Note. The column “Sim”, “Seq” and “Ind” represent the simultaneous, sequential and independent Gibbs samplings, respectively.

Table 2. The settings for simulation studies.
Table 3. Parameter recovery for low dimension with the uniform population.

For any estimation of the parameter $\pi$, $\text{Bias}_{\pi} = 0$ always holds. So the maximum norm is used to replace the bias to evaluate the performance of population parameter estimations. When the true value and estimation of population parameter are $\pi$ and $\hat{\pi}^{(r)}$, the maximum norm of difference is $||\pi - \hat{\pi}^{(r)}||_\infty = \max_j |\pi_j - \hat{\pi}_j|$. The maximum norm measures the maximum of absolute deviance. If the estimation repeats $R$ times, the average maximum norm (MN) is

$$\overline{\text{MN}_e} = \frac{1}{R} \sum_{r=1}^{R} ||\pi - \hat{\pi}^{(r)}||_\infty.$$  

Table 2 summaries the simulation study basic settings: sample sizes $N=1000$ and 2000; the number of the items $J=30$; attribute structures uniform, correlated structures with two correlation levels $\rho = 0.3$ and 0.7. We call the cases $K=3$ and 5 as the low dimension cases, where the simultaneous and sequential Gibbs sampling are conducted. The cases $K=7$ and 15 are named as the high dimension cases, only the sequential Gibbs sampling is performed. For each particular case, 25 independent response datasets are generated.

For the low dimension ($K=3, 5$), the Dirichlet prior’s hyper-parameter $\delta$ is the C-dimensional vector 1, leading to a non-informative prior. For the high dimension ($K=7, 15$), three $\delta$’s are used: $\delta = 0.01, 0.1$ and 1, which are indicated as “S”, “M” and “L” in Table 2. For the GDINA model, the priors for the item parameter $\lambda_i$ are shown as follows:

$$\lambda_i^{(w)} \sim \begin{cases} N(-1.2, 0.4^2), & w = 0, \\ N(0.9, 0.3^2)/w, & w = 1, \ldots, K_j^* \end{cases}$$  

which have a similar support set and a large variance compared with the generation regime for $w > 0$. For the DINA model, non-informative Beta priors are taken for the slipping and guessing parameters. For fair comparison, all methods use the same set of priors.

Besides the priors, we need to specify initial values for $\theta, \pi$ and $\lambda$. The initial value of $\pi$ is that each attribute is randomly sampled from an independent Bernoulli (0.5). The initial value of population parameter $\pi$ is the C-dimensional vector 1/C. The initial value of $\lambda$ is a random sample from the $\lambda$’s prior.

Calpepper (2015) showed that, for the DINA model, simultaneous Gibbs sampling only needed about 750 iterations to reach convergence. Consequently, in this paper, a 2000 iterations Markov chain is run and we discard the first 1000 iterations as burn-in which are adequate to reach convergence. For
### Table 4. Parameter recovery for low dimension with the correlation $p = 0.3$

| Sample size | Method | Attribute number | Sim | Seq | Ind |
|-------------|--------|------------------|-----|-----|-----|
| 1000        | DNA    | 0.004            | 0.005 | 0.003 | 0.004 |
|             | RMSE   | 0.012            | 0.011 | 0.012 | 0.011 |
|             | MSE    | 0.001            | 0.001 | 0.002 | 0.001 |
| 2000        | DNA    | 0.001            | 0.001 | 0.002 | 0.001 |
|             | RMSE   | 0.000            | 0.000 | 0.001 | 0.000 |
|             | MSE    | 0.000            | 0.000 | 0.001 | 0.000 |

### Table 5. Parameter recovery for low dimension with the correlation $p = 0.7$

| Sample size | Method | Attribute number | Sim | Seq | Ind |
|-------------|--------|------------------|-----|-----|-----|
| 1000        | DNA    | 0.004            | 0.005 | 0.003 | 0.004 |
|             | RMSE   | 0.012            | 0.011 | 0.012 | 0.011 |
|             | MSE    | 0.001            | 0.001 | 0.002 | 0.001 |
| 2000        | DNA    | 0.001            | 0.001 | 0.002 | 0.001 |
|             | RMSE   | 0.000            | 0.000 | 0.001 | 0.000 |
|             | MSE    | 0.000            | 0.000 | 0.001 | 0.000 |

Note. The "Sim", "Seq" and "Ind" represent the simultaneous (implemented by JAGS), sequential, independent and self-compiled simultaneous Gibbs sampling methods, respectively.

Note. The "Sim", "Seq" and "Ind" represent the simultaneous (implemented by JAGS), sequential, independent and self-compiled simultaneous Gibbs sampling methods, respectively.
For the simulation results, potential scale reduction factor $R^{1/2}$ (Brooks & Gelman, 1998; Gelman & Rubin, 1992) is used for convergence diagnosis. Brooks and Gelman (1998) suggested that $R^{1/2} < 1.2$ for all model parameters indicates that convergence has been reached. To make the conclusion more reliable, the condition $R^{1/2} < 1.1$ can be used. In Figure 1, the plots of the maximum of all potential scale reduction factors are shown for six representative conditions. The six conditions are from high-dimensional GDINA models. For each condition, we generate 5 independent response data. For each response data, 5 parallel chains are run to calculate the potential scale reduction factors for all item and population parameters. The plots are based upon the first 2000 chain lengths (from 50 to 2000 by increments of 50). The results in Figure 1 (and other diagnosis results not presented here) support that all chains can reach convergence after 2000 iterations under the GDINA model.

### Simulation results

There are 7 tables to show the results for different settings of the DINA and GDINA models. For the GDINA model with a probit link function, the generation of item parameters has a great probability that the intercept is negative and the interaction is positive. Consequently the results of the sampling with or without truncation are very similar, and here we only show the results of sampling with truncation. The digits of bias, RMSE, MSE, AAR and PAR are rounded off to four decimal places and the digits of time are rounded off to five significant figures.

Tables 3–5 show the results of the uniform population, correlated structure with $\rho = 0.3$ and 0.7 for the low dimension cases, respectively. Hereafter, the “Sim”, “Seq”, “Ind”, and “sSim” represent the simultaneous (implemented by JAGS), sequential, independent, and self-compiled simultaneous Gibbs sampling methods, respectively. For the DINA model, “Sim”, “Seq” and “sSim” always obtain similar estimation results for item parameters, population parameters, and attribute patterns, which indicates the estimation consistency between the sequential and simultaneous Gibbs sampling methods. When the population is uniform, the “Ind” method also performs similarly to the other methods. However, as the correlation $\rho$ among the attributes increases, the “Ind” method performs more poorly in $\text{M}_N$. For the GDINA model, we conduct a Markov chain with length 3000 and burn-in length 2000.

### Table 6. Parameter recovery for high dimension with the uniform population.

| Attribute number | Sample size | Method | $K=15$ | $K=2000$ |
|------------------|-------------|--------|---------|-----------|
|                  |             |        |         | 0.01 | 1     |
|                  |             |        |         | 0.01 | 1     |
|                  |             |        |         | 0.01 | 1     |
|                  |             |        |         | 0.01 | 1     |

Note: In the row “Method”, the bold number represents a vector with same components. The bold number represents a vector with same components. The bold number represents a vector with same components.
| Sample size | 1000 | 1 | 1 | 1 | 1 | 1 |
|-------------|------|---|---|---|---|---|
| Method      | DNA  | Bias | MSE | RMSE | MSE | AAR |
|             |      | 0.012 | 0.019 | 0.024 | 0.028 | 0.930 |
|             |      | 0.004 | 0.006 | 0.008 | 0.010 | 0.930 |
|             |      | 0.002 | 0.003 | 0.004 | 0.005 | 0.930 |
|             |      | 0.001 | 0.001 | 0.001 | 0.001 | 0.930 |
|             |      | 0.000 | 0.000 | 0.000 | 0.000 | 0.930 |
|             |      | 0.000 | 0.000 | 0.000 | 0.000 | 0.930 |

Note: In the row "Method", there are three different levels of \( d \): the sequential Gibbs samplings with \( d = 3 \), \( d = 5 \), and \( d = 7 \). The bold number represents a vector with same components.
GDINA model, inferences between the sequential and simultaneous Gibbs sampling methods are also consistent. As the correlation $\rho$ increases, the “Ind” method will underperform the other methods, due to the violation of its assumption on the independence of the attributes.

For both models, the accuracy of estimations becomes better with the large sample size. The classification accuracy of the “Sim” method, as a baseline, is comparable to that of the other methods.

Tables 6–8 show the results of the uniform population and correlated structures with $\rho = 0.3$ and 0.7 for the high dimension cases. If the population distribution of attribute profiles is uniform, the “Ind” method is comparable or slightly outperforms the sequential Gibbs sampling method. When the attributes become more correlated, the sequential sampling method outperforms the “Ind” method. For instance, in the high-dimensional condition with $\rho = 0.7$ and $K=15$, the sequential sampling with a smaller $\delta$ is superior to the other methods. We also find some common phenomena that when $K=15$, the $\text{RMSE}_\pi$ is approximate to 0.0001 (i.e., the $\text{MSE}_\pi$ is approximate to 0), which is due to the large number of latent classes.

In Table 9, the average computational time is reported. Since the dependent structure of the attribute profiles has almost negligible influence on the computational time, we only report the calculation time of different methods under the uniform population. The results show that the simultaneous Gibbs sampling implemented by JAGS has the slowest speed among the four methods. In order to show the computational superiority of the sequential Gibbs than that of the simultaneous Gibbs sampling, we also add the $K=7$ condition for the self-compiled simultaneous Gibbs sampling. Comparing the time of “Seq” and “sSim”, when $K=7$, the superiority of the sequential Gibbs sampling method appears in the GDINA model.

However, the superiority of the sequential Gibbs sampling method has been reflected in the DINA model with $K=5$. In addition, “Ind” is also computationally efficient, with comparable computational time to the sequential Gibbs method.

In the simulation study, using JAGS to estimate the GDINA model needs more memory space than the DINA model. Particularly, for the GDINA model, we find that the estimation process in JAGS is often executed or shut down due to “run out of memory”. Therefore, the presented computational time for the GDINA model is only the average of those well-converged replications in JAGS, which leads to a counter-intuitive observation that the simultaneous Gibbs sampling for the GDINA model is “faster” than the simultaneous Gibbs sampling of the DINA model. Due to the high computational cost, the simultaneous Gibbs sampling time for some high dimension cases is not reported.

Through simulation studies, we can find when $K$ is small, the sequential Gibbs can use less time to obtain the results with similar accuracy as JAGS. When $K$ is large, the sequential Gibbs sampling algorithm still works well, but the simultaneous Gibbs sampling algorithm doesn’t due to the high computational cost. The speed advantage of the sequential Gibbs sampling become more apparent as $K$ increases. In addition, when the number of attributes $K$ goes large, the moderately small hyperparameter $\delta$ is preferred. Comparing the independent and sequential Gibbs sampling methods, for the uniform structure with independent attributes, the independent Gibbs sampling method is comparable or slightly outperforms the sequential Gibbs sampling method. However, for
correlated structures, the sequential Gibbs sampling method provides better performance (see the Bias in Tables 5 and 8).

Real data analysis

In this analysis, the DINA and GDINA models are used to deal with the Tatsuoka’s fraction-subtraction data (C. Tatsuoka, 2002). The fraction-subtraction data has been widely analyzed. For the data set, the Q-matrix (de la Torre & Douglas, 2004) and contents are shown in Table 10. This data set contains responses of 536 middle school students (i.e., \( N = 536 \)) to 20 items (i.e., \( J = 20 \)). There are 8 attributes and \( 2^8 = 256 \) latent classes.

We fit both the DINA and GDINA models. Based on the results of simulation studies, we set the hyper-parameter of the Dirichlet prior \( \delta = 0.1 \). When applying the GDINA model, we assume the item parameter prior for all items as \( \lambda^{(w)} \sim N(0,1) \) for both \( w = 0 \) and \( w > 0 \). The prior hyper-parameters and MCMC chain lengths and burn-in are listed in Table 11. The computation of analyses is performed by a 2018 MacBook Pro with 2.2 GHz Intel Core i7, 16 GB 2400 MHz DDR4 and Radeon Pro 555/4096 MB; Intel UHD Graphics 630 1536 MB. The only feasible case for JAGS is running the 8-attribute DINA model.

Table 11. Summary of analyzing conditions for the fraction-subtraction data.

| Method | DINA | GDINA |
|--------|------|-------|
| GDINA \( \lambda \) prior | \( - \) | \( N(0,1) \) |
| DINA \( s, g \) prior | Beta(1,1) | \( - \) |
| \( \delta \) | 0.1 | 0.1 |
| Chain Length | 2000 | 4000 |
| Burn-in | 1000 | 2000 |

When to analyze the fraction-subtraction data with the DINA model, the independent, sequential and simultaneous Gibbs sampling methods spend 5.46, 4.91 and 942 seconds, respectively. The off-the-shelf software JAGS is treated as a benchmark. We find that the simultaneous Gibbs sampling using JAGS is time-consuming. Figure 2(a,b) shows the posterior means and 95% confidence interval of the guessing and slipping parameters, respectively. When to compare estimation accuracy of item parameters, no matter from posterior means or 95% confidence interval, the simultaneous Gibbs sampling gives very similar results to the sequential Gibbs sampling, while the latter has obvious speed advantage. Compared with the MCMC estimates obtained by JAGS, the independent Gibbs sampling trends to overestimate the guessing parameters and underestimate the slipping parameters, which may be due to the correlation of the attributes.

In the GDINA model, the item parameters are intercept and interaction parameters rather than guessing and slipping parameters. The time costs of the independent and sequential Gibbs sampling algorithms are 19.25 and 17.65 seconds, respectively. The notation \( \lambda^{(w)} \) represents the \( w \)-way interaction. Since the items in the fraction-subtraction data need up to 5 attributes, there exist up to the 5-way interaction parameters. Figure 2(c) presents the box-plot of \( w \) versus \( \lambda^{(w)} \) for the estimated item parameters. The common property shared by the same-way

Figure 2. The estimations of item parameters for the fraction-subtraction data. The “Ind”, “Seq” and “Sim” represent the results from the independent, sequential and simultaneous (by implemented JAGS) Gibbs sampling, respectively. (a) DINA Guessing parameters. (b) DINA Slipping parameters. (c) GDINA parameters.
interactions can be obtained by the box-plot. The item parameter estimations show that the means of intercept parameters and 4-way interaction parameters are negative and the others are positive. The conclusion that the intercept term is negative is consistent with our intuition, because a subject without any required attributes is usually expected to have a low positive response probability.

**Discussion**

In practice, one computational challenge is that when the number of attributes is large, the existing MCMC for the CDMs may become slow. In this paper, a computationally efficient algorithm, named as the sequential Gibbs sampling, was proposed for a general CDM, i.e., the GDINA model. In the situation with small $K$, compared to the existing method (e.g., JAGS), the proposed method can also yield similar results. The proposed method still works well and fast for the case with large $K$. When $K = 15, N = 2000, J = 40$ and the model is the GDINA model, running 3,000 iterations only needs less 110 seconds. Especially, for a large $K$, the computational advantage becomes more significant compared with the simultaneous Gibbs sampling method. The proposed method can be easily applied to other CDMs. In the appendix, we show the algorithm for the DINA model for an illustration.

In this paper, we only focus on the computational challenge for large $K$, given the Q-matrix is correctly specified. Most references about identification theory pointed out that the Q-matrix need to contain an identity matrix at least for strict identifiability (Gu & Xu, 2020; Xu & Shang, 2018). In practice, however, the Q-matrix may be misspecified, and it would be needed to estimate the Q-matrix together with the model parameters and latent attributes. The estimation of the Q-matrix is known to be a challenging issue, especially when $K$ is large, and the proposed algorithm may be extended to such applications to help reduce the computational cost of the conventional MCMC approaches. Another interesting extension is to use the idea in this paper to solve other latent variable modeling problems with many latent attributes. Not only for the discrete but also for continuous abilities, this idea may be helpful.

**Article information**

Conflict of interest disclosures: Each author signed a form for disclosure of potential conflicts of interest. No authors reported any financial or other conflicts of interest in relation to the work described.

**Ethical principles:** The authors affirm having followed professional ethical guidelines in preparing this work. These guidelines include obtaining informed consent from human participants, maintaining ethical treatment and respect for the rights of human or animal participants, and ensuring the privacy of participants and their data, such as ensuring that individual participants cannot be identified in reported results or from publicly available original or archival data.

**Funding:** This work was partially supported by Grant 12001092 from the National Natural Science Foundation of China, Grant 2412020QD004 by the Fundamental Research Funds for the Central Universities and the Key Laboratory of Applied Statistics of MOE, Grant 201806620039 from the China Scholarship Council, and NSF Grants SES-1659328 and SES-1846747.

**Role of the funders/sponsors:** None of the funders or sponsors of this research had any role in the design and conduct of the study; collection, management, analysis, and interpretation of data; preparation, review, or approval of the manuscript; or decision to submit the manuscript for publication.

**Acknowledgments:** The authors would like to thank the editor, an associate editor, and two reviewers for their comments on prior versions of this manuscript. The ideas and opinions expressed herein are those of the authors alone, and endorsement by the author’s institutions or the NSF is not intended and should not be inferred.

**ORCID**

Xue Zhang http://orcid.org/0000-0002-3900-0118

**References**

Anderson, T. W. (1958). *An introduction to multivariate statistical analysis*. Wiley.

Brooks, S. P., & Gelman, A. (1998). General methods for monitoring convergence of iterative simulations. *Journal of Computational and Graphical Statistics, 7*(4), 434–455. https://doi.org/10.1080/10618600.1998.10474787

Chen, Y., Culpepper, S. A., Chen, Y., & Douglas, J. (2018). Bayesian estimation of the DINA Q matrix. *Psychometrika, 83*(1), 89–108. https://doi.org/10.1007/s11336-017-9579-4

Chen, Y., Culpepper, S., & Liang, F. (2020). A sparse latent class model for cognitive diagnosis. *Psychometrika, 85*(1), 121–153. https://doi.org/10.1007/s11336-019-09693-2
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. https://doi.org/10.1080/01621459.2014.934827

Chiu, C.-Y., Douglas, J. A., & Li, X. (2009). Cluster analysis for cognitive diagnosis: Theory and applications. *Psychometrika, 74*(4), 633–665. https://doi.org/10.1007/s11336-009-9125-0

Chung, M. (2014). Estimating the Q-matrix for cognitive diagnosis models in a Bayesian framework [Unpublished doctoral dissertation], Columbia University.

Chung, M. (2019). A Gibbs sampling algorithm that estimates the Q-matrix for the DINA model. *Journal of Mathematical Psychology, 93*, 102275. https://doi.org/10.1016/j.jmp.2019.07.002

Culpepper, S. A. (2015). Bayesian estimation of the DINA model with Gibbs sampling. *Journal of Educational and Behavioral Statistics, 40*(5), 454–476. https://doi.org/10.3102/1076998615595403

Culpepper, S. A. (2019). Estimating the cognitive diagnosis Q matrix with expert knowledge: Application to the fraction-subtraction dataset. *Psychometrika, 84*(2), 333–357. https://doi.org/10.1007/s11336-018-9643-8

Culpepper, S. A. (2019b). An exploratory diagnostic model for ordinal responses with binary attributes: Identifiability and estimation. *Psychometrika, 84*(4), 921–940. https://doi.org/10.1007/s11336-019-09683-4

Culpepper, S. A., & Chen, Y. (2019). Development and application of an exploratory reduced reparameterized unified model. *Journal of Educational and Behavioral Statistics, 44*(1), 3–24. https://doi.org/10.3102/1076998618791306

Culpepper, S. A., & Hudson, A. (2018). An improved strategy for Bayesian estimation of the reduced reparameterized unified model. *Applied Psychological Measurement, 42*(2), 99–115. https://doi.org/10.1177/0146621617707511
de la Torre, J., & Douglas, J. A. (2004). Higher-order latent trait models for cognitive diagnosis. *Psychometrika, 69*(3), 333–353. https://doi.org/10.1007/BF02295640
de la Torre, J. (2011). The generalized DINA model framework. *Psychometrika, 76*(2), 179–199. https://doi.org/10.1007/s11336-011-9207-7

Foulley, J.-L., & Jaffrézic, F. (2010). Modelling and estimating heterogeneous variances in threshold models for ordinal discrete data via WinBUGs/Openbugs. *Computer Methods and Programs in Biomedicine, 97*(1), 19–27. https://doi.org/10.1016/j.cmpb.2009.05.004

Gelman, A., & Rubin, D. B. (1992). Inference from iterative simulation using multiple sequences. *Statistical Science, 7*(4), 457–472. https://doi.org/10.1214/ss/1177011136

Gu, Y., & Xu, G. (2019). The sufficient and necessary condition for the identifiability and estimability of the DINA model. *Psychometrika, 84*(2), 468–483. https://doi.org/10.1007/s11336-018-9619-8

Gu, Y., & Xu, G. (2020). Partial identifiability of restricted latent class models. *Annals of Statistics, 48*(4), 2082–2107. https://doi.org/10.1214/19-AOS1878

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. https://doi.org/10.1111/j.1745-3984.1989.tb00336.x

Hartz, S. M. (2002). A Bayesian framework for the unified model for assessing cognitive abilities: Blending theory with practicality [Unpublished doctoral dissertation]. ProQuest Information & Learning.

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. https://doi.org/10.1007/s11336-008-9089-5

Junker, B. W., & Sijtsma, K. (2001). Cognitive assessment models with few assumptions, and connections with non-parametric item response theory. *Applied Psychological Measurement, 25*(3), 258–272. https://doi.org/10.1177/1046621012203026

Lee, Y.-S., Park, Y. S., & Taylan, D. (2011). A cognitive diagnostic modeling of attribute mastery in Massachusetts, Minnesota, and the US national sample using the TIMSS 2007. *International Journal of Testing, 11*(2), 144–177. https://doi.org/10.1080/15305058.2010.534571

Lunn, D. J., Thomas, A., Best, N., & Spiegelhalter, D. (2000). *WinBUGS—A Bayesian modelling framework:* Concepts, structure, and extensibility. *Statistics and Computing, 10*(4), 325–337. https://doi.org/10.1023/A:1008929526011

Maris, E. (1999). Estimating multiple classification latent class models. *Psychometrika, 64*(2), 187–212. https://doi.org/10.1007/BF02294535

Minka, T. (2000). *Bayesian linear regression* (Tech. Rep.). MIT.

Park, Y. S., & Lee, Y.-S. (2014). An extension of the DINA model using covariates: Examining factors affecting response probability and latent classification. *Applied Psychological Measurement, 38*(5), 376–390. https://doi.org/10.1177/0146621614523830

Park, Y. S., Xing, K., & Lee, Y.-S. (2018). Explanatory cognitive diagnostic models: Incorporating latent and observed predictors. *Applied Psychological Measurement, 42*(5), 376–392. https://doi.org/10.1177/0146621617738012

Plummer, M. (2003). *JAGS: A program for analysis of Bayesian graphical models using Gibbs sampling.* In *Proceedings of the 3rd International Workshop on Distributed Statistical Computing* (Vol. 124, pp. 1–10). https://www.r-project.org/conferences/DSC-2003/Proceedings/Plummer.pdf

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

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. https://doi.org/10.1111/1467-9876.00272

Tatsuoka, K. K. (1983). Rule space: An approach for dealing with misconceptions based on item response theory. *Journal of Educational Measurement, 20*(4), 345–354. https://doi.org/10.1111/j.1745-3984.1983.tb00212.x

Templin, J. L., & Henson, R. A. (2006). Measurement of psychological disorders using cognitive diagnosis models. *Psychological Methods, 11*(3), 287–305. https://doi.org/10.1037/1082-989X.11.3.287

von Davier, M. (2005). A general diagnostic model applied to language testing data. *ETS Research Report Series, 2005*(2), i–35. https://doi.org/10.1348/000711007X193957

Wu, Z., Deloria-Knoll, M., & Zeger, S. L. (2016). Nested partially latent class models for dependent binary data;
estimating disease etiology. *Biostatistics, 18*(2), 200–213. https://doi.org/10.1093/biostatistics/kxw037

Xu, G. (2017). Identifiability of restricted latent class models with binary responses. *The Annals of Statistics, 45*(2), 675–707. https://doi.org/10.1214/16-AOS1464

Xu, G., & Shang, Z. (2018). Identifying latent structures in restricted latent class models. *Journal of the American Statistical Association, 113*(523), 1284–1295. https://doi.org/10.1080/01621459.2017.1340889

Xu, G., & Zhang, S. (2016). Identifiability of diagnostic classification models. *Psychometrika, 81*(3), 625–649. https://doi.org/10.1007/s11336-015-9471-z

Zhan, P., Jiao, H., Man, K., & Wang, L. (2019). Using JAGS for Bayesian cognitive diagnosis modeling: A tutorial. *Journal of Educational and Behavioral Statistics, 44*(4), 473–503. https://doi.org/10.3102/1076998619826040

Zhan, P., Li, X., Wang, W.-C., Bian, Y., & Wang, L. (2015). The multidimensional testlet-effect cognitive diagnostic models. *Acta Psychologica Sinica, 47*(5), 689–701. https://doi.org/10.3724/SP.J.1041.2015.00689

### Appendix A: Sequential Gibbs sampling for the DINA model

Under the DINA model, the item parameters are guessing parameter $g$ and slipping parameter $s$. The priors of guessing and slipping parameters follow Beta($a_q, b_q$) and Beta($a_s, b_s$) respectively. The rest of the measurement model remains similarly to the GDINA model. The sampling methods for $g, s$, and $\pi$ can be founded in Culpepper (2015). We only focus on sampling attribute profiles: for fixed $\mathbf{Y}, \mathbf{z}$, $g$, $s$, and $\pi$, the full conditional distribution for $x_{ik}$ is proportional to

$$p(x_{ik} | \mathbf{Y}, \mathbf{z}, g, s, \pi) \propto \prod_{j=1}^{I} \left[ P(x_{ij})^Y_i (1 - P(x_{ij}))^{1 - Y_i} \right] p(x_{ik} | p_{ik})$$

(A.1)

$$\propto \prod_{j=1}^{I} \left[ (g_j^{1 - Y_i}(1 - s_j)^{Y_i} ((1 - g)^{1 - Y_i})^{s_j} \right] p_{ik}^{x_{ik}} (1 - p_{ik})^{1 - x_{ik}}$$

(A.2)

$$\propto \prod_{j=1}^{I} \left[ (s_j^{1 - Y_i}(1 - s_j)^{1 - Y_i}) (g_j^{1 - Y_i} (1 - g_j)^{1 - Y_i}) \right] p_{ik}^{x_{ik}} (1 - p_{ik})^{1 - x_{ik}}$$

(A.3)

$$\propto \prod_{j=1}^{I} \left[ (s_j^{1 - Y_i}(1 - s_j)^{1 - Y_i}) \prod_{k \neq j} p_{ik}^{x_{ik}} (1 - p_{ik})^{1 - x_{ik}} \right]$$

(A.4)

where $p_{ik}$ is the prior conditional probability as in Equation (9). For simplify, we define a set $\Omega_{ik} = \{j \in \mathbb{N} | 1 \leq j \leq I\}$, and the complementary set is $\Omega_{ik}^c = \{j \notin \Omega_{ik}, \forall j = 1, \ldots, I\}$. We know that when item $j$ belongs to $\Omega_{ik}$, $\eta_{ij} = x_{ik}$. The items in the set $\Omega_{ik}$ satisfy at least one of two conditions, $\prod_{k \neq j} x_{ik} = 0$ and $\eta_{ik} = 0$, which indicate that the value of $x_{ik}$ doesn’t affect the value of $\eta_{ik}$. In other words, the items in the set $\Omega_{ik}$ don’t affect the full conditional distribution. We can rewrite the full conditional distribution as follows:

$$p(x_{ik} | \mathbf{Y}, \mathbf{z}, g, s, p) \propto \prod_{j=1}^{I} \left[ (s_j^{1 - Y_i}(1 - s_j)^{Y_i}) p_{ik}^{x_{ik}} (1 - p_{ik})^{1 - x_{ik}} \right]$$

(A.5)

Obviously, the posterior distribution of $x_{ik}$ is a Bernoulli distribution Bernoulli($p_{ik}$) with the parameter

$$\hat{p}_{ik} = \frac{\prod_{j \in \Omega_{ik}} (s_j^{1 - Y_i}(1 - s_j)^{Y_i}) p_{ik} + \prod_{j \in \Omega_{ik}^c} (g_j^{1 - Y_i} (1 - g_j)^{1 - Y_i}) (1 - p_{ik})}{(1 - p_{ik})}$$

(A.6)

We next introduce the initial values for the DINA model. The initial values of item parameters $g_j^{(0)}, s_j^{(0)}$ are randomly sampled from the uniform distribution $U[0, 0.4]$. The initial population parameter $\pi$ is the C-dimensional 1/C; the hyper-parameters are $a_q = b_q = a_s = b_s = 1$. The other initial values are same as those in section “Simulation design.” The sequential Gibbs sampling for the DINA model is presented in Algorithm 2.

### Algorithm 2: Sequential Gibbs Sampling for DINA

**Input:** Initialize $g_j^{(0)}, s_j^{(0)}, \mathbf{x}^{(0)}, \pi^{(0)}, Y, m = 0, M$ and specify priors.

**Output:** Markov chains of $g, s, \mathbf{x}, \pi$.

**while** $m < M$ do

Sample attribute profiles from Equation (A.5).

Sample the other parameters according to the reference (Culpepper, 2015).

Set $m = m + 1$.

end

### Appendix B: Full conditional distribution for $l_j$

Focusing on the $j$th item, we can get a classical linear regression as follows

$$\mathbf{Z} = \mathbf{X}_j \lambda_j + \mathbf{e}_j,$$

where $\lambda_j$ is the item parameter and the residual $\mathbf{e}_j = (e_1, e_2, \ldots, e_N)'$ is a random sample from standard normal distribution. The kernel of likelihood function is given as

$$p(\mathbf{Z} | \mathbf{X}_j, \lambda_j) \propto \exp \left( -\frac{1}{2}(\mathbf{Z} - \mathbf{X}_j \lambda_j)' \mathbf{I}^{-1} (\mathbf{Z} - \mathbf{X}_j \lambda_j) \right),$$

(B.1)

where the $\mathbf{I}$ represents an identity matrix. Assuming the joint prior for $\lambda_j$ is $N(\mu_{\lambda_j}, \Sigma_{\lambda_j})$, the specific form of parameter’s prior is

$$p(\lambda_j) \propto \exp \left( -\frac{1}{2} (\lambda_j - \mu_{\lambda_j})' \Sigma_{\lambda_j}^{-1} (\lambda_j - \mu_{\lambda_j}) \right).$$

(B.2)

According to the Bayesian linear regression, the kernel of the posterior is

$$p(\lambda_j | \mathbf{Z}, \mathbf{X}_j, \lambda_j)$$

$$\propto \exp \left( -\frac{1}{2} (\mathbf{Z} - \mathbf{X}_j \lambda_j)' \mathbf{I}^{-1} (\mathbf{Z} - \mathbf{X}_j \lambda_j) + (\lambda_j - \mu_{\lambda_j})' \Sigma_{\lambda_j}^{-1} (\lambda_j - \mu_{\lambda_j}) \right),$$

$$\propto \exp \left( -\frac{1}{2} \left( \lambda_j' (\mathbf{X}_j' \mathbf{X}_j + \Sigma_{\lambda_j})^{-1} \lambda_j - \lambda_j' (\mathbf{X}_j' \mathbf{Z} + \Sigma_{\lambda_j} \mu_{\lambda_j}) - (\mathbf{X}_j' \mathbf{Z} + \Sigma_{\lambda_j} \mu_{\lambda_j})' \lambda_j \right) \right).$$

(B.3)
Table D1. The transposed Q-matrices for different K.

| K  | A1   | A2   | A3   | A4   | A5   | A6   | A7   | A8   | A9   | A10  | A11  | A12  | A13  | A14  | A15  |
|----|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| 3  | 1 0 0 | 0 1 0 | 1 0 0 | 0 1 0 | 1 0 0 | 0 1 0 | 1 0 0 | 0 1 0 | 1 0 0 | 0 1 0 | 1 0 0 | 0 1 0 | 1 0 0 | 0 1 0 |
| 5  | 1 0 0 | 0 1 0 | 1 0 0 | 0 1 0 | 1 0 0 | 0 1 0 | 1 0 0 | 0 1 0 | 1 0 0 | 0 1 0 | 1 0 0 | 0 1 0 | 1 0 0 | 0 1 0 |
| 7  | 1 0 0 | 0 1 0 | 1 0 0 | 0 1 0 | 1 0 0 | 0 1 0 | 1 0 0 | 0 1 0 | 1 0 0 | 0 1 0 | 1 0 0 | 0 1 0 | 1 0 0 | 0 1 0 |
| 15 | 1 0 0 | 0 1 0 | 1 0 0 | 0 1 0 | 1 0 0 | 0 1 0 | 1 0 0 | 0 1 0 | 1 0 0 | 0 1 0 | 1 0 0 | 0 1 0 | 1 0 0 | 0 1 0 |

Let $\Sigma^{-1} = X X' + \Sigma^{-1}$, using the undetermined coefficient method to solve $\mu_j = \Sigma(X Z + \Sigma^{-1}\mu_j)$, so we can get the full conditional distribution for jth item’s parameters easily. The full conditional distribution is $N(\mu_j, \Sigma_j)$.

**Appendix C: Sample from the truncated multivariate normal distribution**

Because the posterior is a normal distribution, only the multivariate normal distribution with truncation needs to be derived. Assuming that a random vector $X \sim N_p(\mu, \Sigma)$ and $X = (X_1, X_2, ..., X_p)'$ is a p-dimensional vector, where

$$
\mu = (\mu_1, ..., \mu_p)',
$$

$$
\Sigma = 
\begin{pmatrix}
\sigma_{11} & \sigma_{12} & \cdots & \sigma_{1p} \\
\sigma_{21} & \sigma_{22} & \cdots & \sigma_{2p} \\
\vdots & \vdots & \ddots & \vdots \\
\sigma_{p1} & \sigma_{p2} & \cdots & \sigma_{pp}
\end{pmatrix}
$$

When we ignore the truncation, the marginal distribution of $X_1$ is a normal distribution which can be used to generate $x_1$, and then the conditional distributions of the following parameters:

$$
X_2 | X_1 = x_1,
$$

$$
X_p | X_{1:p-1} = x_{1:p-1},
$$

are also normal distributions (Anderson, 1958) which are used to generate $x_2, ..., x_p$. As a result, we can get a realization $(x_1, x_2, ..., x_p)'$ following $N_p(\mu, \Sigma)$.

In this paper, the first component $X_1$ is negative, the else components are positive. Imposing the restrictions to $(x_1, x_2, ..., x_p)'$, we sample $x_1$ from the interval $(-\infty, 0)$ part of the marginal distribution of $X_1$ and sample $x_2, ..., x_p$ from the $(0, \infty)$ of remaining conditional distributions.

Furthermore, through this method, more complex restrictions can be easy to impose. For any $X_p$, the left censoring, right censoring and interval censoring can be employed. Generating censoring data from a unidimensional normal distribution is easy, so this method is rather flexible and simple.

**Appendix D: The Q-matrices for different K**

The Q-matrices are used for different K in the simulation studies, please see Table D1.

**Appendix E: Analyses of TIMSS 2007**

The Trends in Mathematics and Science Study (TIMSS), a quadrennial assessment, assessed the mathematics and science abilities of fourth and eighth students since 1995. TIMSS 2007 (Grade 4) dataset with 25 mathematics (dichotomized) items used in Lee et al. (2011), Park and Lee (2014) and Park et al. (2018). The dataset includes a sample of 698 Austrian students. The chosen data contain $f = 25$ items (i.e., booklets 4 and 5), which consist of two parts: 11 items released for booklet 4 were new items developed for TIMSS 2007 and the remaining 14 items from booklet 5 were previously administered during TIMSS 2003.

The Q-matrix used by Lee et al. (2011) is listed in Table E1. There are 15 attributes (i.e., $K=15$) belonging to three content domains: Number, Geometric Shapes and Measures and Data Display. For this data analyses, the prior hyper-
parameters and MCMC chain lengths and burn-in are listed in Table E2. Due to the large number of attributes, the independent and sequential Gibbs sampling methods are the only two methods used here.

Analyzing the TIMSS 2007 data with the DINA model, the time costs of the independent and sequential Gibbs sampling are 11.51 and 13.36 seconds, respectively. The estimations and 95% HDP regions of guessing and slipping parameters are shown in Figure E1(a,b), respectively. When the DINA model is used to analyze, large guessing and slipping parameters often indicate the model may not fit the data well. In the results of sequential Gibbs sampling, the guessing parameters for items 15, 18, 22 and 25 are greater than 0.5, meanwhile the slipping parameters for items 2, 3, 4, 10, 11, 17, 21 and 24 are greater than 0.5.

When analyzing the TIMSS 2007 using the GDINA model, the time costs of the independent and sequential Gibbs sampling are 35.63 and 42.75 seconds, respectively. Since the items in TIMSS 2007 need up to 6 attributes, there exist the 6-way interaction parameters. The box-plot of estimated item parameters is given in Figure E1(c). We can see that the average effect of intercept is negative and the others are positive.