frontiers | Frontiers in Psychology

TYPE Original Research
PUBLISHED 15 August 2022
DOI 10.3389/fpsyg.2022.935419

OPEN ACCESS

EDITED BY
Dubravka Svetina Valdivia,
Indiana University Bloomington,
United States

REVIEWED BY
Daniel Bolt,
University of Wisconsin-Madison,
United States
Hye-Jeong Choi,
University of Georgia, United States

*CORRESPONDENCE
Chun Wang
wang4066@uw.edu
Gongjun Xu
gongjun@umich.edu

SPECIALTY SECTION
This article was submitted to Quantitative Psychology and Measurement,
a section of the journal
Frontiers in Psychology

RECEIVED 03 May 2022
ACCEPTED 24 June 2022
PUBLISHED 15 August 2022

CITATION
Liu T, Wang C and Xu G (2022) Estimating three- and four-parameter MIRT models with importance-weighted sampling enhanced variational auto-encoder.
Front. Psychol. 13:935419.
doi: 10.3389/fpsyg.2022.935419

Estimating three- and four-parameter MIRT models with importance-weighted sampling enhanced variational auto-encoder

Tianci Liu[1], Chun Wang[2]* and Gongjun Xu[1]*

[1]Department of Statistics, University of Michigan, Ann Arbor, MI, United States, [2]College of Education, University of Washington, Seattle, WA, United States

Multidimensional Item Response Theory (MIRT) is widely used in educational and psychological assessment and evaluation. With the increasing size of modern assessment data, many existing estimation methods become computationally demanding and hence they are not scalable to big data, especially for the multidimensional three-parameter and four-parameter logistic models (i.e., M3PL and M4PL). To address this issue, we propose an importance-weighted sampling enhanced Variational Autoencoder (VAE) approach for the estimation of M3PL and M4PL. The key idea is to adopt a variational inference procedure in machine learning literature to approximate the intractable marginal likelihood, and further use importance-weighted samples to boost the trained VAE with a better log-likelihood approximation. Simulation studies are conducted to demonstrate the computational efficiency and scalability of the new algorithm in comparison to the popular alternative algorithms, i.e., Monte Carlo EM and Metropolis-Hastings Robbins-Monro methods. The good performance of the proposed method is also illustrated by a NAEP multistage testing data set.

KEYWORDS

Multidimensional Item Response Theory (MIRT), estimation, Monte Carlo (MC) algorithm, variational auto encoder (VAE), four parameter item response theory

1. Introduction

Item response theory (IRT) has been widely used for the evaluation and assessment of education and psychology test data. The most commonly used IRT is the *2-parameter logistic model* (2PL), which is based on a logistic model for dichotomous responses and assigns a scalar factor score for each respondent. After observing its success, flexibility beyond the 2PL model has also been pursued for decades. Notably, McDonald (1967) suggested that the lower and upper asymptote in 2PL can be freed up from fixed 0 and 1, respectively. Estimating a different lower asymptote for each item results in the so-called 3PL model, which has been quite useful for multiple-choice items where guessing is possible; but little empirical evidence was found to support that estimating upper asymptote was beneficial as well; therefore, it was widely believed that the 4PL model was only of theoretical interest and there was no compelling reason for practitioners to use it (Barton and Lord, 1981; Hambleton and Swaminathan, 1985). Until the 2000s, researchers started revisiting the 4PL model and demonstrated the rationale of introducing upper asymptote parameters after observing early signs of its importance (Reise and Waller, 2003; Loken and Rulison, 2010; Waller and Reise, 2010; Yen et al., 2012). Waller and Feuerstahler (2017) took a step further and conducted a comprehensive study of 4PL model on a variety of real and synthetic data. In their experiment, the 4PL model achieved promising accuracy on medium to large data. However, despite these existing studies and estimation methods (e.g., Ogasawara, 2002; Waller and Feuerstahler, 2017; Meng et al., 2020), difficulties of parameter estimation in 3PL and 4PL models still remain, especially when data sizes are large and the latent factors exhibit a multidimensional or even high-dimensional structure.

Multidimensional IRT (MIRT) models are a family of models where the latent trait is no longer assumed to be unidimensional. By allowing latent factors to exhibit multidimensional structures, 2PL, 3PL, and 4PL models are turned into the *multidimensional 2PL* (M2PL), 3PL (M3PL), and 4PL (M4PL) models, respectively. Compared with IRT models, MIRT models are capable to model each individual's multiple latent traits simultaneously and are usually favored by large scale and complex real data, thereof (Reckase, 2009).

In this article, we study the general MIRT models with a special focus on M3PL and M4PL models. Specifically, assume that there are N individuals who respond to J items independently with binary response Y_{ij}, for $i = 1, \dots, N$ and $j = 1, \dots, J$. The M3PL model assumes that this response from the i-th individual to the j-th item is modeled by the following *item response function* (IRF).

$$P(Y_{ij} = 1 \mid \boldsymbol{\theta}_i; \boldsymbol{a}_j, b_j, c_j) = c_j + (1 - c_j)\frac{\exp(\boldsymbol{a}_j^\top \boldsymbol{\theta}_i + b_j)}{\exp(\boldsymbol{a}_j^\top \boldsymbol{\theta}_i + b_j) + 1}, \tag{1}$$

where $\boldsymbol{a}_j$ is a K-dimensional vector of item discrimination (loading) parameters for the j-th item; b_j is referred to as the item easiness parameters. $-b_j/\|\boldsymbol{a}_j\|_2$ is sometimes termed as item difficulty (Cho et al., 2021); $c_j \geq 0$ is known as the lower asymptote of the j-th item and measures the probability of guessing j-th item correctly when $\boldsymbol{\theta}_i$ is of negative infinity. Moreover, $\boldsymbol{\theta}_i$ is a K-dimensional latent variable denoting the ability of i-th respondent, which is assumed to have a standard K-dimensional Gaussian distribution in IRT literature. Further generalizing M3PL, the M4PL model has an IRF of

$$P(Y_{ij} = 1 \mid \boldsymbol{\theta}_i; \boldsymbol{a}_j, b_j, c_j, d_j) = c_j + (d_j - c_j)\frac{\exp(\boldsymbol{a}_j^\top \boldsymbol{\theta}_i + b_j)}{\exp(\boldsymbol{a}_j^\top \boldsymbol{\theta}_i + b_j) + 1}, \tag{2}$$

where additional $d_j \leq 1$ is referred to as the upper asymptote parameter, which is the maximum probability of answering the j-th item correctly when $\boldsymbol{\theta}_i$ goes to infinity. Intuitively, $1 - d_j$ can be treated as the slipping probability that an individual who is able to answer the item correctly but miss it accidentally.

For both M3PL and M4PL models, we denote model parameters $\mathbf{A} = \{\boldsymbol{a}_j, j = 1, \dots, J\}$, $\boldsymbol{b} = \{b_j, j = 1, \dots, J\}$, $\boldsymbol{c} = \{c_j, j = 1, \dots, J\}$, $\boldsymbol{d} = \{d_j, j = 1, \dots, J\}$; and for M3PL model $d_j = 1, j = 1, \dots, J$ and $M_p = \{\mathbf{A}, \boldsymbol{b}, \boldsymbol{c}, \boldsymbol{d}\}$ is the collection of all model parameters. Under the typical local independence assumption, the marginal log-likelihood of M_p is given by

$$\begin{aligned} l(M_p; \mathbf{Y}) &= \sum_{i=1}^{N} \log P(\boldsymbol{y}_i \mid M_p) \\ &= \sum_{i=1}^{N} \log \int \prod_{j=1}^{J} P(Y_{ij} \mid \boldsymbol{\theta}_i; M_p) p(\boldsymbol{\theta}_i) \mathrm{d}\boldsymbol{\theta}_i, \end{aligned} \tag{3}$$

where $p(\boldsymbol{\theta}_i)$ is the probability density function of a standard K-dimensional Gaussian distribution.

Due to the latent variable structure, the K dimensional integrals involved in (3) makes maximization of the log-likelihood function with respect to M_p intractable. Direct numerical approximations of the integrals were proposed, including the *Gauss-Hermite quadrature* (Bock and Aitkin, 1981) and *Laplace approximation* (Tierney and Kadane, 1986; Lindstrom and Bates, 1988; Wolfinger and O'connell, 1993). However, these methods usually fail to handle complicated MIRT model, especially when the dimension K of latent factors $\boldsymbol{\theta}$ grows: Gauss-Hermite quadrature quickly becomes computationally expensive in a high-dimensional setting; the Laplace approximation, though being efficient in computation, often performs less accurately when K increases or when the likelihood function is skewed. *Monte Carlo* (MC) simulations have also been applied to obtain numerical approximations for MIRT, such as *Monte Carlo expectation-maximization* (MCEM, McCulloch, 1997), *stochastic expectation-maximization* (StEM, von Davier and Sinharay, 2010), and *Metropolis-Hastings Robbins-Monro* (MHRM, Cai, 2010a,b). Nevertheless, MC based methods need drawing samples from posterior distributions, which could be computationally demanding as well. Recently, Zhang et al. (2020) improved StEM for item factor analysis, but its stochastic E-step involves an adaptive rejection-based Gibbs sampler and may still be time consuming. All methods discussed above can be seen as variants of the *marginal maximum likelihood* (MML) estimator proposed in Bock and Aitkin (1981), where latent $\boldsymbol{\theta}$ are considered as random variables and are integrated out. Chen et al. (2019) instead studied the *constraint joint maximum likelihood estimator* (CJMLE) by treating $\boldsymbol{\theta}$ as fixed effect parameters in order to achieve higher speeds.

Unfortunately, many existing studies focusing on the M2PL model cannot be applied to M3(4)PL models easily: for MHRM, commercial software FlexMIRT (Chung and Houts, 2020) does not support M4PL, and for M3PL, MHRM is known to suffer from a lower convergence rate (Cho et al., 2021) than M2PL; for CJMLE, the authors only derived methods for M2PL and which not support M3(4)PL models. In general, computationally efficient estimation methods for M3(4)PL models are still under explored.

Variational approaches stem from the machine learning literature, which maximizes a tractable lower bound of the log-likelihood rather than maximizing the log-likelihood directly. They have been applied to fitting IRT models in recent years (Rijmen and Jeon, 2013; Natesan et al., 2016; Hui et al., 2017; Jeon et al., 2017). More recently, these variational methods also established a variety of successes on more complicated MIRT (Curi et al., 2019; Wu et al., 2020; Cho et al., 2021) and graded response models (Urban and Bauer, 2021). Notably, *variational autoencoder* (VAE), deep learning based variational method, and its variation, *importance weighted autoencoder* (IWAE), are shown to be effective in parameters estimation and achieve performances competitive to traditional techniques at much faster speeds (Curi et al., 2019; Wu et al., 2020; Urban and Bauer, 2021).

In this article, we investigate the VAE method for the more challenging M3(4)PL models with possible missing data. Extending explorations from Urban and Bauer (2021), we propose a new training strategy for VAE by enhancing it with the objective function of IWAE. As revealed in Section 2.2, although IWAE is computationally more expensive than VAE, our mixing training method inherits both the speed advantage of VAE and the better performance of IWAE. We also pay great attention to several practical issues and challenges in model training and propose corresponding methods/tricks to solve them, which allows our model to handle missing data and have better numeric robustness. Compared with the existing estimation approaches, such as MCEM and MHRM, our method succeeds in achieving comparable or better accuracy in parameter estimation and exhibits a much faster speed. Moreover, our method converges under M3(4)PL models within constant fitting times on different sizes, comparable to what Urban and Bauer (2021) found in the M2PL model, which is a key advantage of VAE based estimation over traditional methods.

The rest of this article is organized as follows. Section 2 covers our new training strategy of VAE based estimation, which is named as *Importance-Weighted sampling enhanced VAE* (IWVAE); to make the section self-contained, we also provide an overview of VAE and IWAE; important tricks for handling missing data as well as improving numerical stability are also introduced. Section 3 provides a large-scale simulation study where IWVAE shows consistently competitive performances to MHRM and MCEM methods across different sample sizes, item structures, and asymptotic regimes. Section 4 compares three methods on a real data set from a multistage testing design. We end up this article with final discussion and remarks in Section 5.

2. Methods

We start with a brief overview of variational inference and how it helps tackle maximizing likelihoods whose exact forms are unavailable. We then introduce a gradient based model from deep learning called *variational autoencoder* (VAE), along with its generalization *importance weighted autoencoder* (IWAE). Given the importance and popularity of *multilayer perceptron* (MLP) in machine learning, which provides an efficient way of parameterizing and implementing VAE and IWAE, we include a concise introduction of MLP and reveal its ability to handle missing entries which are ubiquitous in large datasets. We end up this section with our new proposed mixing training method of VAE, *importance-weighted sampling enhanced VAE* (IWVAE). IWVAE uses both VAE and IWAE's objective functions and enjoys both benefits of them.

2.1. A review of variational inference and variational autoencoder

Since the integration in Equation (3) does not admit a closed form solution, we need a tractable objective function to approximate it, and *variational inference* (VI) is a machine learning technique to achieve this (Bishop, 2006; Blei et al., 2017). There are two equivalent ways to setup the VI objective. The first one aims to find the best approximation of the posterior of latent variable $\boldsymbol{\theta}_i$ given $\boldsymbol{y}_i$ and M_p, which results in a lower bound of Equation (3). Additionally, the second one directly derives the bound using Jensen Inequality.

We start with the first derivation as it better clarifies the connection between VI and the *expectation maximization* (EM) algorithm (Dempster et al., 1977). The second derivation is revisited in Section 2.2 when we introduce a new tighter lower bound. Let $\boldsymbol{\Theta} = (\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_N)$ denote the collection of all latent variables. The *best* approximation of posterior $p(\boldsymbol{\Theta} \mid \mathbf{Y}; M_p)$, which we refer to as $q(\boldsymbol{\Theta})$, is obtained by finding a candidate from some simple and tractable variational distribution families such that the *Kullback-Leibler* (KL) divergence $D_{\mathrm{KL}}[q(\boldsymbol{\Theta}) \| p(\boldsymbol{\Theta} \mid \mathbf{Y}; M_p)]$ is minimized. One common variational family is the factorized distribution $q(\boldsymbol{\Theta}) = \prod_{i=1}^{N} \prod_{k=1}^{K} q_{ik}(\theta_{ik})$, where the subscript *ik* in $q_{ik}(\theta_{ik})$ is to emphasize that different dimensions can follow different distributions, or follow the same distribution but have different parameters. For instance, we can choose the popular Gaussian distribution for each $q_{ik}(\theta_{ik})$, equivalently, we have $q(\boldsymbol{\theta}_i)$ to follow a K-dimensional diagonal Gaussian distribution. If one intends to characterize the dependence structure among different dimensions of $\boldsymbol{\theta}_i$, we may choose the factorized distribution family $q(\boldsymbol{\Theta}) = \prod_{i=1}^{N} q_i(\boldsymbol{\theta}_i)$, with $q_i(\boldsymbol{\theta}_i)$ following a K-dimensional Gaussian distribution.

Under this setting, the optimal variational approximation $q^*(\boldsymbol{\Theta})$ is given by (Blei et al., 2017).

$$\begin{aligned} q^*(\boldsymbol{\Theta}) &\triangleq \underset{q(\boldsymbol{\Theta})}{\operatorname{argmin}} \, D_{\mathrm{KL}}[q(\boldsymbol{\Theta}) \| p(\boldsymbol{\Theta} \mid \mathbf{Y}; M_p)] \\ &= \underset{q(\boldsymbol{\Theta})}{\operatorname{argmin}} \int q(\boldsymbol{\Theta}) \log q(\boldsymbol{\Theta}) \mathrm{d}\boldsymbol{\Theta} \\ &\quad - \int q(\boldsymbol{\Theta}) \log \frac{p(\mathbf{Y} \mid \boldsymbol{\Theta}; M_p) p(\boldsymbol{\Theta})}{p(\mathbf{Y} \mid M_p)} \mathrm{d}\boldsymbol{\Theta}. \end{aligned} \tag{4}$$

Note that $\log p(\mathbf{Y} \mid M_p)$ is independent of $\boldsymbol{\Theta}$, it is easy to obtain the optimization objective

$$q^*(\boldsymbol{\Theta}) = \underset{q(\boldsymbol{\Theta})}{\operatorname{argmin}} \, D_{\mathrm{KL}}[q(\boldsymbol{\Theta}) \| p(\boldsymbol{\Theta})] - \mathbb{E}_{q(\boldsymbol{\Theta})}[\log p(\mathbf{Y} | \boldsymbol{\Theta}; M_p)], \tag{5}$$

and following decomposition

$$\begin{aligned} \log p(\mathbf{Y} \mid M_p) &= \mathbb{E}_{q(\boldsymbol{\Theta})}[\log p(\mathbf{Y} | \boldsymbol{\Theta}; M_p)] - D_{\mathrm{KL}}[q(\boldsymbol{\Theta}) \| p(\boldsymbol{\Theta})] \\ &\quad + D_{\mathrm{KL}}[q(\boldsymbol{\Theta}) \| p(\boldsymbol{\Theta} \mid \mathbf{Y}; M_p)]. \end{aligned} \tag{6}$$

Since $D_{\mathrm{KL}}[q(\boldsymbol{\Theta}) \| p(\boldsymbol{\Theta} \mid \mathbf{Y}; M_p)]$ is non-negative, the decomposition reveals the fact that minimizing Equation (5) is equivalent to maximizing a lower bound of the marginal log-likelihood, which is known as *evidence lower bound* (ELBO).

Remark 1. *The derivation of VI above has a close connection to the EM algorithm. Using the decomposition from Bishop (2006), we have*

$$\begin{aligned} \log p(\mathbf{Y} \mid M_p) &= \int q(\boldsymbol{\Theta}) \log p(\mathbf{Y} \mid M_p) d\boldsymbol{\Theta} \\ &= \int q(\boldsymbol{\Theta}) \log \frac{p(\mathbf{Y}, \boldsymbol{\Theta} \mid M_p)}{q(\boldsymbol{\Theta})} d\boldsymbol{\Theta} \\ &\quad + \int q(\boldsymbol{\Theta}) \log \frac{q(\boldsymbol{\Theta})}{p(\boldsymbol{\Theta} \mid \mathbf{Y}, M_p)} d\boldsymbol{\Theta} \\ &= \mathcal{L}(q(\boldsymbol{\Theta}), M_p) + D_{\mathrm{KL}}[q(\boldsymbol{\Theta}) \| p(\boldsymbol{\Theta} \mid \mathbf{Y}, M_p)], \end{aligned} \tag{7}$$

where $q(\boldsymbol{\Theta})$ is an arbitrary distribution that includes the variational distribution families. And the first term $\mathcal{L}(q(\boldsymbol{\Theta}), M_p)$ is precisely the ELBO. In the EM algorithm, $\mathcal{L}(q(\boldsymbol{\Theta}), M_p)$ is maximized with respect to $q(\boldsymbol{\Theta})$ and M_p in an iterative way. In the E-step, the maximization is over $q(\boldsymbol{\Theta})$, which requires a closed-form solution: the true posterior of $\boldsymbol{\Theta}$ given $\mathbf{Y}$ and fixed M_p^{old}. By doing so the second KL divergence disappears and $\mathcal{L}(q(\boldsymbol{\Theta}), M_p^{old}) = \log p(\mathbf{Y} \mid M_p^{old})$. Since the right hand side does not depend on $q(\boldsymbol{\Theta})$, the ELBO takes equality thereof. In M-step, the M_p is optimized to maximize the $\mathcal{L}(q(\boldsymbol{\Theta}), M_p)$ by fixing $q(\boldsymbol{\Theta})$. By repeating two steps the EM algorithm is guaranteed to converge to a local optimum of $\log p(\mathbf{Y} \mid M_p)$.

The main difference between VI on IRT and EM algorithm is that because $p(\boldsymbol{\Theta} \mid \mathbf{Y}; M_p)$ is intractable, we cannot obtain the analytic update of $q(\boldsymbol{\Theta})$ in each step, as a result, plain EM algorithm does not scale up well to the high-dimensional MIRT model. VI, on the other hand, finds a tractable approximation in its "E-step" and consequently, it always optimizes a strict lower bound. In general, another philosophical difference between VI and EM is that unknown parameters in VI are usually treated as latent variables as well, refer to Bishop (2006) for more clarifications. In our setup, we distinguish model parameters M_p and latent variable $\boldsymbol{\Theta}$, but this is not necessary, refer to Wu et al. (2020) where M_p was also treated as latent variables as well and modeled together with $\boldsymbol{\Theta}$.

Evidence lower bound derived in Equation (5) is a global lower bound of the marginal log-likelihood of all observations. Given the local independence assumption, we can obtain a tighter lower bound by constructing each individual a corresponding local lower bound. Deriving local lower bounds indicate finding $q_i(\boldsymbol{\theta}_i)$ such that $D_{\mathrm{KL}}[q_i(\boldsymbol{\theta}_i) \| p(\boldsymbol{\theta}_i \mid \boldsymbol{y}_i; M_p)]$ is minimized. This is called *local variational methods*, we recommend Chapter 10.4 of Bishop (2006) for more detailed explanations, and Cho et al. (2021) for its successful implementations on M2PL and M3PL models.

However, despite the success of Cho et al. (2021), in general, the local variational method is computationally expensive on large scale data. One alternate to handle this challenge is the *amortized variational inference* (AVI). To characterize $q_i(\boldsymbol{\theta}_i) = \mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\sigma}_i^2)$, where $\boldsymbol{\sigma}_i^2$ denotes the diagonal of the covariance matrix, AVI assumes that $\boldsymbol{\mu}_i, \boldsymbol{\sigma}_i^2$ depend on $\boldsymbol{y}_i$ through a function $F(\cdot)$ parameterized by $\boldsymbol{\phi}$, formally

$$(\boldsymbol{\mu}_i, \log \boldsymbol{\sigma}_i^2) = F_{\boldsymbol{\phi}}(\boldsymbol{y}_i), \quad q_i(\boldsymbol{\theta}_i) = \mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\sigma}_i^2). \tag{8}$$

Henceforth, we denote $q_i(\boldsymbol{\theta}_i)$ as $q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_i \mid \boldsymbol{y}_i)$. In practice, $F_{\boldsymbol{\phi}}$ can be flexible and expressed by a deep neural network. One of its most famous applications on AVI is the *variational autoencoder* (VAE) proposed by Kingma and Welling (2014). VAE uses two neural networks together to maximize the ELBO bound: $F_{\boldsymbol{\phi}}$ is termed as *inference* or *encoder* network (please refer to Section 2.3.1 for the specification of $F_{\boldsymbol{\phi}}$); the other *generative* or *decoder* network learns the generative process of $\boldsymbol{y}_i$ given $\boldsymbol{\theta}_i$, where this process in MIRT is essentially estimating model parameters M_p.

In VAE, $\boldsymbol{\phi}$ and M_p are learned through stochastic gradient descents. Following Kingma and Welling (2014) and Urban and Bauer (2021), we give a brief review here. Note the ELBO for the i-th individual is given by

$$\mathrm{ELBO}_i = \mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_i | \boldsymbol{y}_i)}[\log p(\boldsymbol{y}_i \mid \boldsymbol{\theta}_i; M_p)] - D_{\mathrm{KL}}[q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_i \mid \boldsymbol{y}_i) \| p(\boldsymbol{\theta}_i)] \tag{9}$$

The gradient $\nabla_{M_p}\text{ELBO}_i$ can be estimated readily with S Monte Carlo samples $\boldsymbol{\theta}_i^s \sim q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_i \mid \boldsymbol{y}_i)$ for $s = 1, \dots, S$ as $\nabla_{M_p}\text{ELBO}_i \approx \frac{1}{S}\sum_{s=1}^{S} \nabla_{M_p} \log p(\boldsymbol{y}_i \mid \boldsymbol{\theta}_i^s; M_p)]$. However, gradient $\nabla_{\boldsymbol{\phi}}\text{ELBO}_i$ cannot be obtained in the same way, as in general $\nabla_{\boldsymbol{\phi}}$ and $\mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_i|\boldsymbol{y}_i)}$ cannot be switched. To solve this problem, Kingma and Welling (2014) reparameterized $\boldsymbol{\theta}_i \sim \mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\sigma}_i^2)$ as follows

$$\boldsymbol{e}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \quad \boldsymbol{\theta}_i = \boldsymbol{e}_i \odot \boldsymbol{\sigma}_i + \boldsymbol{\mu}_i, \tag{10}$$

where $\odot$ means the element-wise multiplications. By transforming the integration over $q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_i \mid \boldsymbol{y}_i)$ to $p(\boldsymbol{e}_i)$, we have

$$\begin{aligned} \nabla_{\boldsymbol{\phi}}\text{ELBO}_i &= \nabla_{\boldsymbol{\phi}}\mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_i|\boldsymbol{y}_i)}[\log p(\boldsymbol{y}_i \mid \boldsymbol{\theta}_i; M_p)] \\ &\quad - \nabla_{\boldsymbol{\phi}} D_{\text{KL}}[q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_i \mid \boldsymbol{y}_i) \| p(\boldsymbol{\theta}_i)]. \end{aligned}$$

Then, the first term can be estimated with Monte Carlo samples $\frac{1}{S}\sum_{s=1}^{S} \nabla_{\boldsymbol{\phi}} \log p(\boldsymbol{y}_i \mid \boldsymbol{e}_i^s \odot \boldsymbol{\sigma}_i + \boldsymbol{\mu}_i; M_p)$, and the second term can be computed effectively by observing that the KL divergence between $q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_i \mid \boldsymbol{y}_i)$ and $p(\boldsymbol{\theta}_i)$ has an analytic form (Kingma and Welling, 2014).

$$D_{\text{KL}}[q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_i \mid \boldsymbol{y}_i) \| p(\boldsymbol{\theta}_i)] = \frac{1}{2}\sum_{k=1}^{K}(\mu_{ik} + \sigma_{ik}^2 - 1 - \log \sigma_{ik}^2). \tag{11}$$

The gradient can be computed readily through the chain rule thereof. For more details, please refer to Kingma and Welling (2014) and Urban and Bauer (2021).

2.2. Importance weighted variational inference

Since the ELBO is a lower bound of the marginal likelihood that we want to maximize, a tighter ELBO is appealing as the true likelihood can be approximated more accurately. It is known that the tightness of the ELBO is coupled with the expressiveness of the variational family and limited expressivity can negatively affect the learned models, and there have been many works on reducing the gap between ELBO and marginal log-likelihood (Burda et al., 2016; Kingma et al., 2016; Kingma and Welling, 2019). Some studies aimed to extend the capacity of the variational family, and techniques including normalizing flows have been applied (Kingma et al., 2016; Papamakarios et al., 2021).

Burda et al. (2016) introduced a new *importance-weighted ELBO* (IW-ELBO) which alleviated the coupling without changing the variational families. To better illustrate the connection between IW-ELBO bound and ELBO, we start with the second derivation of ELBO *via* Jensen Inequality.

$$\begin{aligned} \log p(\boldsymbol{y}_i \mid M_p) &= \log \mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_i|\boldsymbol{y}_i)}\left[\frac{p(\boldsymbol{y}_i, \boldsymbol{\theta}_i \mid M_p)}{q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_i \mid \boldsymbol{y}_i)}\right] \\ &\geq \mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_i|\boldsymbol{y}_i)}\left[\log \frac{p(\boldsymbol{y}_i, \boldsymbol{\theta}_i \mid M_p)}{q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_i \mid \boldsymbol{y}_i)}\right] = \text{ELBO}_i. \end{aligned} \tag{12}$$

The above derivation can be generalized as follows

$$\begin{aligned} \log p(\boldsymbol{y}_i \mid M_p) &= \log \mathbb{E}_{\boldsymbol{\theta}_i^1, \dots, \boldsymbol{\theta}_i^R \sim q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_i|\boldsymbol{y}_i)}\left[\frac{1}{R}\sum_{r=1}^{R}\frac{p(\boldsymbol{y}_i, \boldsymbol{\theta}_i^r \mid M_p)}{q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_i^r \mid \boldsymbol{y}_i)}\right] \\ &\geq \mathbb{E}_{\boldsymbol{\theta}_i^{1:R} \sim q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_i|\boldsymbol{y}_i)}\left[\log \frac{1}{R}\sum_{r=1}^{R} w_i^r\right]. \end{aligned} \tag{13}$$

Equation (13) is known as IW-ELBO where $w_i^r \triangleq p(\boldsymbol{y}_i, \boldsymbol{\theta}_i^r \mid M_p)/q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_i^r \mid \boldsymbol{y}_i)$. When $q_{\boldsymbol{\phi}}$ is reparameterizable, Monte Carlo estimates of IW-ELBO and its gradient are given by

$$\begin{aligned} &\mathbb{E}_{\boldsymbol{\theta}_i^{1:R} \sim q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_i|\boldsymbol{y}_i)}\left[\log \frac{1}{R}\sum_{r=1}^{R} w_i^r\right] \\ &\approx \frac{1}{S}\sum_{s=1}^{S}\log \frac{1}{R}\sum_{r=1}^{R} w_i^{rs}, \end{aligned} \tag{14}$$

$$\begin{aligned} &\nabla_{\boldsymbol{\phi}, M_p}\mathbb{E}_{\boldsymbol{\theta}_i^{1:R} \sim q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_i|\boldsymbol{y}_i)}\left[\log \frac{1}{R}\sum_{r=1}^{R} w_i^r\right] \\ &= \mathbb{E}_{\boldsymbol{e}_i^{1:R}}[\nabla_{\boldsymbol{\phi}, M_p}\log \frac{1}{R}\sum_{r=1}^{R} w_i^r] \\ &= \mathbb{E}_{\boldsymbol{e}_i^{1:R}}\left[\frac{w_i^r}{\sum_{r=1}^{R} w_i^r}\nabla_{\boldsymbol{\phi}, M_p}\log w_i^r\right] \\ &\approx \frac{1}{S}\sum_{s=1}^{S}\sum_{r=1}^{R}\frac{w_i^{rs}}{\sum_{r=1}^{R} w_i^{rs}}\nabla_{\boldsymbol{\phi}, M_p}\log w_i^{rs}, \end{aligned} \tag{15}$$

where S and R are corresponding numbers of Monte Carlo samples and importance-weighted samples. Replacing ELBO with IW-ELBO in VAE leads to IWAE, which is a generalization of the VAE, as indicated by observing that IW-ELBO will reduce to ELBO for $R = 1$. Notably, IW-ELBO increases in R and converges to $\log p(\boldsymbol{y}_i \mid M_p)$ as $R \to \infty$ under mild conditions (Burda et al., 2016).

However, Rainforth et al. (2018) showed that using more important samples is not always helpful. The authors introduced the *signal-to-noise ratio* (SNR) of an estimator δ as the ratio between the absolute value of its expectation and its SD, i.e., $\text{SNR}(\delta) \triangleq |\mathbb{E}(\delta)|/\sigma(\delta)$. Then they show the below orders (rewritten with our notations)

$$\text{SNR}(M_p) = \mathcal{O}(\sqrt{RS}), \quad \text{SNR}(\boldsymbol{\phi}) = \mathcal{O}(\sqrt{S/R}).$$

In words, for any given S, increasing R makes gradient estimates of parameters $\boldsymbol{\phi}$ in the inference network noisier.

TABLE 1 Mean and SE of RMSE of M_p estimate on M4PL models under single regime setting, best results are in bold.

N, J	Item structure	Model	rot(A)	b	c	d	Success rates
500 100	Between	MCEM	9.400 ± 0.181	11.477 ± 0.424	0.175 ± 0.010	0.183 ± 0.010	1.00
		MHRM	/	/	/	/	/
		IWVAE	**0.674 ± 0.02**	**0.384 ± 0.027**	**0.081 ± 0.008**	**0.087 ± 0.008**	1.00
	Within	MCEM	10.406 ± 0.240	11.500 ± 0.481	0.163 ± 0.010	0.146 ± 0.008	1.00
		MHRM	/	/	/	/	/
		IWVAE	**0.744 ± 0.022**	**0.402 ± 0.034**	**0.073 ± 0.008**	**0.088 ± 0.008**	1.00
1000 100	Between	MCEM	8.230 ± 0.170	11.785 ± 0.450	0.189 ± 0.012	0.178 ± 0.011	1.00
		MHRM	/	/	/	/	/
		IWVAE	**0.498 ± 0.019**	**0.341 ± 0.028**	**0.079 ± 0.008**	**0.080 ± 0.008**	1.00
	Within	MCEM	7.799 ± 0.230	8.999 ± 0.464	0.132 ± 0.010	0.161 ± 0.011	1.00
		MHRM	/	/	/	/	/
		IWVAE	**0.609 ± 0.022**	**0.386 ± 0.029**	**0.069 ± 0.007**	**0.078 ± 0.008**	1.00
5,000 100	Between	MCEM	3.240 ± 0.156	4.351 ± 0.276	0.189 ± 0.011	0.155 ± 0.011	1.00
		MHRM	/	/	/	/	/
		IWVAE	**0.369 ± 0.027**	**0.378 ± 0.043**	**0.091 ± 0.011**	**0.082 ± 0.009**	1.00
	Within	MCEM	3.235 ± 0.221	2.939 ± 0.254	0.139 ± 0.012	0.133 ± 0.010	1.00
		MHRM	/	/	/	/	/
		IWVAE	**0.535 ± 0.029**	**0.383 ± 0.035**	**0.075 ± 0.008**	**0.086 ± 0.010**	1.00
10,000 100	Between	MCEM	1.988 ± 0.099	2.690 ± 0.184	0.174 ± 0.010	0.186 ± 0.011	1.00
		MHRM	/	/	/	/	/
		IWVAE	**0.379 ± 0.028**	**0.399 ± 0.042**	**0.084 ± 0.008**	**0.079 ± 0.008**	1.00
	Within	MCEM	1.823 ± 0.145	1.674 ± 0.151	0.136 ± 0.009	0.125 ± 0.007	1.00
		MHRM	/	/	/	/	/
		IWVAE	**0.516 ± 0.030**	**0.343 ± 0.038**	**0.084 ± 0.010**	**0.079 ± 0.008**	1.00

Factors are diagonal. **Between** item structure: each item depends on 1 factor. **Within** item structure: each item depends on 2 factors.

Despite the fact that the estimates of M_p along may benefit from a tighter likelihood bound, the final result can be deteriorated due to the worse inference network as shown in Rainforth et al. (2018).

To mitigate this problem, one simple solution is to increase S of the same order, but such modification takes more computational costs and slows down the training. We apply the *doubly reparameterized gradient estimator* (DReG) from Tucker et al. (2018), a recently developed method that gets rid of a similar issue. Specifically, we use the below estimator to update the inference network

$$\begin{aligned}&\nabla_{\boldsymbol{\phi}}\mathbb{E}_{\boldsymbol{\theta}_i^{1:R}\sim q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_i|\boldsymbol{y}_i)}\left[\log\frac{1}{R}\sum_{r=1}^{R}w_i^r\right]\\ &=\mathbb{E}_{\boldsymbol{e}_i^{1:R}}\left[\left(\frac{w_i^r}{\sum_{r=1}^{R}w_i^r}\right)^2\frac{\partial\log w_i^r}{\partial\boldsymbol{\theta}_i^r}\frac{\partial\boldsymbol{\theta}_i^r}{\partial\boldsymbol{\phi}}\right]\\ &\approx\frac{1}{S}\sum_{s=1}^{S}\sum_{r=1}^{R}\left(\frac{w_i^{rs}}{\sum_{r=1}^{R}w_i^{rs}}\right)^2\frac{\partial\log w_i^{rs}}{\partial\boldsymbol{\theta}_i^{rs}}\frac{\partial\boldsymbol{\theta}_i^{rs}}{\partial\boldsymbol{\phi}}.\end{aligned}\qquad(16)$$

Empirically, computing IW-ELBO and its gradient estimates can be numerically unstable due to exponential operations involved in $p(\boldsymbol{\theta}_i^{rs})$ and $q_\phi(\boldsymbol{\theta}_i^{rs}\mid\boldsymbol{y}_i)$. To solve this problem, we compute $v_i^{rs}=\log w_i^{rs}=\log p(\boldsymbol{y}_i\mid\boldsymbol{\theta}_i^{rs};M_p)-\log q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_i^{rs}\mid\boldsymbol{y}_i)+\log p(\boldsymbol{\theta}_i^{rs})$, and apply the well-known *log-sum-exp* trick (Zhang et al., 2021) to $\log\frac{1}{R}\sum w_i^{rs}$ in Equation (14) and $w_i^{rs}/\sum w_i^{rs}$ in Equations (15) and (16) as follows:

$$\begin{aligned}\log\frac{1}{R}\sum_{r=1}^{R}w_i^{rs}&=\max_r v_i^{rs}+\log\sum_{r=1}^{R}\exp\left(v_i^{rs}-\max_r v_i^{rs}\right)-\log R,\\ \frac{w_i^{rs}}{\sum_{r=1}^{R}w_i^{rs}}&=\frac{\exp\left(v_i^{rs}-\max_r v_i^{rs}\right)}{\sum_{r=1}^{R}\exp\left(v_i^{rs}-\max_r v_i^{rs}\right)}.\end{aligned}$$

2.3. Implementation details

2.3.1. MLP and optimization

We provide a basic overview of *multilayer perceptron* (MLP) applied in this study, which is used to model the variational distribution q as in Equation (8). For more details about MLP and DNN, we recommend readers to Goodfellow et al. (2016).

Multilayer perceptron, also known as *feedforward neural networks* (FNN), is one of the most popular architectures of neural networks because of its simple form and flexibility. To

TABLE 2 Mean and SE of RMSE of M_p estimate on M4PL models under double regime setting, best results are in bold.

N, J	Item structure	Model	rot(A)	b	c	d	Success rates
500 100	Between	MCEM	9.400 ± 0.181	11.477 ± 0.424	0.175 ± 0.010	0.183 ± 0.010	1.00
		MHRM	/	/	/	/	/
		IWVAE	**0.674 ± 0.021**	**0.384 ± 0.027**	**0.081 ± 0.008**	**0.087 ± 0.008**	1.00
	Within	MCEM	10.406 ± 0.240	11.500 ± 0.481	0.163 ± 0.010	0.146 ± 0.008	1.00
		MHRM	/	/	/	/	/
		IWVAE	**0.744 ± 0.022**	**0.402 ± 0.034**	**0.073 ± 0.008**	**0.088 ± 0.008**	1.00
1000 200	Between	MCEM	5.397 ± 0.069	8.312 ± 0.188	0.180 ± 0.007	0.178 ± 0.007	1.00
		MHRM	/	/	/	/	/
		IWVAE	**0.500 ± 0.016**	**0.377 ± 0.025**	**0.080 ± 0.006**	**0.088 ± 0.007**	1.00
	Within	MCEM	8.242 ± 0.174	9.254 ± 0.354	0.151 ± 0.007	0.158 ± 0.007	1.00
		MHRM	/	/	/	/	/
		IWVAE	**0.600 ± 0.017**	**0.413 ± 0.024**	**0.080 ± 0.005**	**0.088 ± 0.007**	1.00
5,000 300	Between	MCEM	1.624 ± 0.058	1.519 ± 0.068	0.163 ± 0.006	0.156 ± 0.005	1.00
		MHRM	/	/	/	/	/
		IWVAE	**0.429 ± 0.014**	**0.338 ± 0.019**	**0.084 ± 0.005**	**0.081 ± 0.005**	1.00
	Within	MCEM	1.388 ± 0.079	0.876 ± 0.076	0.092 ± 0.005	0.086 ± 0.004	1.00
		MHRM	/	/	/	/	/
		IWVAE	**0.564 ± 0.015**	**0.319 ± 0.019**	**0.083 ± 0.005**	**0.080 ± 0.005**	1.00
10,000 500	Between	MCEM	1.022 ± 0.023	1.152 ± 0.036	0.150 ± 0.004	0.153 ± 0.004	1.00
		MHRM	/	/	/	/	/
		IWVAE	**0.432 ± 0.012**	**0.338 ± 0.015**	**0.086 ± 0.004**	**0.086 ± 0.004**	1.00
	Within	MCEM	0.930 ± 0.018	0.993 ± 0.031	0.119 ± 0.004	0.119 ± 0.003	1.00
		MHRM	/	/	/	/	/
		IWVAE	**0.564 ± 0.013**	**0.346 ± 0.015**	**0.085 ± 0.004**	**0.087 ± 0.004**	1.00

Factors are diagonal. **Between** item structure: each item depends on 1 factor. **Within** item structure: each item depends on 2 factors.

approximate an unknown function f^* such that $\boldsymbol{u} = f^*(\boldsymbol{v})$ where $\boldsymbol{v} \in \mathbb{R}^P, \boldsymbol{u} \in \mathbb{R}^Q$, MLP takes the recursive form $\boldsymbol{h}_l = f_l\left(\mathbf{W}_l\boldsymbol{h}_{l-1} + \boldsymbol{b}_l\right), l = 1, \ldots, L$, and $\boldsymbol{h}_0 = \boldsymbol{v}, \boldsymbol{h}_L = \boldsymbol{u}$. Here, $f_1, \ldots, f_L$ are scalar functions which are almost everywhere differentiable and are applied elementwisely when inputs are vectors. These functions are typically termed as *activation* functions. When $f_1, \ldots, f_L$ are set to identity function $g(z) = z$, MLP will reduce to linear regression; using non-linear activation functions, we get a flexible function $u = f(v)$. MLP has been shown an universal approximator under a variety of activation functions (Cybenko, 1989; Hornik, 1991; Sonoda and Murata, 2017), including *sigmoid function* $g(x) = 1/(e^{-x} + 1)$, *rectified linear unit function* (ReLU, Nair and Hinton, 2010) $g(x) = \max(0, x)$, and *hyperbolic tangent function* (Tanh) $g(x) = (e^x - e^{-x})/(e^x + e^{-x})$; refer to Goodfellow et al. (2016) for other choices of activation functions. In this article, we use Tanh activation for $f_1 \ldots, f_{L-1}$.

The f_L at the last layer is chosen depending on the data form. To see this, note that the last layer of MLP $\boldsymbol{u} = f_L(\mathbf{W}_L\boldsymbol{h}_{L-1} + \boldsymbol{b}_L)$ can be seen as a generalized linear model with independent variable $\boldsymbol{h}_{L-1}$. When $\boldsymbol{u}$ is continuous, f_L can be set to the identity function and we get the last layer a linear regression. When $\boldsymbol{u}$ is binary (categorical), f_L can be set to the sigmoid (softmax) function and we get a logistic (multinomial logistic) regression, respectively.

In this article, we use the following encoder network

$$\boldsymbol{h}_i = \mathrm{Tanh}(\boldsymbol{b}_L + \mathbf{W}_L\mathrm{Tanh}(\boldsymbol{b}_{L-1} + \ldots \mathrm{Tanh}(\boldsymbol{b}_1 + \mathbf{W}_1\boldsymbol{y}_i)) \ldots),$$
$$\boldsymbol{\mu}_i = \mathbf{W}_{\mu}\boldsymbol{h}_i + \boldsymbol{b}_{\mu},$$
$$\boldsymbol{\sigma}_i^2 = \exp(\mathbf{W}_{\sigma^2}\boldsymbol{h}_i + \boldsymbol{b}_{\sigma^2}).$$

Here, $\boldsymbol{h}_i$ denotes the intermediate output of the encoder given input the i-th individual data $\boldsymbol{y}_i$, and we have $\boldsymbol{\phi} = \{\mathbf{W}_1, \boldsymbol{b}_1, \ldots, \mathbf{W}_L, \boldsymbol{b}_L, \mathbf{W}_{\mu}, \boldsymbol{b}_{\mu}, \mathbf{W}_{\sigma^2}, \boldsymbol{b}_{\sigma^2}\}$. In the decoder, to effectively utilize the gradient based method, following Kucukelbir et al. (2017), we map $\boldsymbol{c}$ and $\boldsymbol{d}$ from constrained ranges $[0, 1]^J$ to unconstrained space $\mathbb{R}^J$ through the differentiable $\mathrm{logit}(x) = \log x/(1 - x)$ transformation and conduct gradient ascent in the unconstrained space. To avoid cluttering, we still use original notations $\boldsymbol{c}$ and $\boldsymbol{d}$ in the following.

TABLE 3 Mean and SE of RMSE of M_p estimate on M3PL models under single regime setting, best results are in bold.

N, J	Item structure	Model	rot(A)	*b*	*c*	Success rates
500 100	Between	MCEM	10.020 ± 0.318	13.638 ± 0.752	0.213 ± 0.013	1.00
		MHRM	**0.217 ± 0.026**	0.567 ± 0.080	0.099 ± 0.007	0.35
		IWVAE	0.641 ± 0.022	**0.391 ± 0.031**	**0.081 ± 0.008**	1.00
	Within	MCEM	8.133 ± 0.459	8.687 ± 0.727	0.194 ± 0.014	1.00
		MHRM	**0.417 ± 0.034**	**0.345 ± 0.049**	0.078 ± 0.005	0.40
		IWVAE	0.708 ± 0.021	0.461 ± 0.039	**0.073 ± 0.008**	1.00
1000 100	Between	MCEM	5.338 ± 0.287	7.799 ± 0.544	0.237 ± 0.016	1.00
		MHRM	**0.159 ± 0.017**	**0.280 ± 0.025**	0.089 ± 0.007	0.30
		IWVAE	0.492 ± 0.019	0.320 ± 0.026	**0.079 ± 0.008**	1.00
	Within	MCEM	2.564 ± 0.235	3.023 ± 0.423	0.120 ± 0.011	1.00
		MHRM	**0.438 ± 0.038**	**0.313 ± 0.040**	0.075 ± 0.005	0.65
		IWVAE	0.590 ± 0.020	0.325 ± 0.024	**0.069 ± 0.007**	1.00
5,000 100	Between	MCEM	1.031 ± 0.110	1.190 ± 0.204	0.144 ± 0.013	1.00
		MHRM	**0.151 ± 0.024**	0.264 ± 0.028	**0.090 ± 0.008**	0.30
		IWVAE	0.403 ± 0.024	**0.259 ± 0.028**	0.091 ± 0.011	1.00
	Within	MCEM	0.881 ± 0.063	0.575 ± 0.077	0.097 ± 0.009	1.00
		MHRM	**0.292 ± 0.035**	**0.123 ± 0.009**	**0.033 ± 0.004**	0.90
		IWVAE	0.562 ± 0.026	0.279 ± 0.032	0.075 ± 0.008	1.00
10,000 100	Between	MCEM	0.810 ± 0.078	1.008 ± 0.169	0.112 ± 0.011	1.00
		MHRM	**0.106 ± 0.019**	0.381 ± 0.131	**0.055 ± 0.006**	0.70
		IWVAE	0.393 ± 0.027	**0.318 ± 0.036**	0.084 ± 0.008	1.00
	Within	MCEM	0.754 ± 0.045	0.662 ± 0.129	0.076 ± 0.007	1.00
		MHRM	**0.343 ± 0.035**	**0.154 ± 0.012**	**0.040 ± 0.004**	0.75
		IWVAE	0.535 ± 0.027	0.283 ± 0.039	0.084 ± 0.010	1.00

Factors are diagonal. **Between** item structure: each item depends on 1 factor. **Within** item structure: each item depends on 2 factors.

2.3.2. Handling missing data

When $\boldsymbol{y}_i$ given latent factors $\boldsymbol{\theta}_i$ are conditionally independent, exactly as the MIRT models assume, MLP based VAE and IWAE can handle incomplete data containing entries *missing at random* (MAR) readily (Nazabal et al., 2020). Here, we provide a brief summary of the *input drop-out* trick.

First, we replace missing entries in $\boldsymbol{y}_i$ with zeros and denote the resultant vector as $\tilde{\boldsymbol{y}}_i$; we further use indicator vector $\mathbf{1}_i$ to record which entries are observed, specifically, $\mathbf{1}_{ij} \triangleq \mathbf{1}(y_{ij} \text{ is observed})$[1]. Next, we replace w_i^r with $\tilde{w}_i^r = \exp \tilde{v}_i^r$ where $\tilde{v}_i^r$ is defined as

$$\tilde{v}_i^r = \sum_{j=1}^{J}\left[\mathbf{1}_{ij}\log p(\tilde{y}_{ij} \mid \tilde{\boldsymbol{\theta}}_i^r; M_p)\right] - \log q_{\boldsymbol{\phi}}(\tilde{\boldsymbol{\theta}}_i^r \mid \tilde{\boldsymbol{y}}_i) + \log p(\tilde{\boldsymbol{\theta}}_i^r).$$

For now, we use $\tilde{\boldsymbol{\theta}}_i$ to emphasize that the inference network takes $\tilde{\boldsymbol{y}}_i$ as input. Next we show the imputing missing entries with 0 does not influence the training. For M_p, based on Equation (14), its gradient estimate is determined by $\nabla_{M_p}\tilde{v}_i^{rs}$ and does not depend on imputed entries because of the multiplication of $\mathbf{1}_{ij}$, therefore M_p is also independent of them.

Additionally, if neither $q_{\boldsymbol{\phi}}(\tilde{\boldsymbol{\theta}}_i \mid \tilde{\boldsymbol{y}}_i)$ nor $\tilde{\boldsymbol{\theta}}_i$ is affected by imputed entries, then such imputation will not influence the model training as v_i^r (and w_i^r) does not rely on these entries. To this end, we rely on the MLP architecture. The output of each neuron in MLP is a non-linear transformation of a linear combination of its inputs. This property ensures that all intermediate states and output of the inference network, which determines $\boldsymbol{\mu}_i$ and $\boldsymbol{\sigma}_i$ for variational distribution $q_{\phi}(\tilde{\boldsymbol{\theta}}_i \mid \tilde{\boldsymbol{y}}_i)$, does not depend on zero entries in its inputs.

These observations together guarantee the condition for v_i^r (and w_i^r) being independent of imputed entries, as in both ELBO and IW-ELBO, gradient estimates of all parameters are determined by collections of these terms.

2.3.3. Training strategy and hyperparameter choices

We propose a three-stage training strategy for VAE by enhancing it with IW-ELBO. We first train a standard VAE through maximizing its own objective function ELBO. After

[1] Here is a bit of abuse of notation: we use $\mathbf{1}$ to denote both the indicator function $\mathbf{1}(\cdot)$ and its output.

TABLE 4 Mean and SE of RMSE of M_p estimate on M3PL models under double regime setting, best results are in bold.

N, J	Item structure	Model	rot(A)	*b*	*c*	Success rates
500, 100	Between	MCEM	10.020 ± 0.318	13.638 ± 0.752	0.213 ± 0.013	1.00
		MHRM	**0.217 ± 0.026**	0.567 ± 0.080	0.099 ± 0.007	0.35
		IWVAE	0.641 ± 0.022	**0.391 ± 0.031**	**0.081 ± 0.008**	1.00
	Within	MCEM	8.133 ± 0.459	8.687 ± 0.727	0.194 ± 0.014	1.00
		MHRM	**0.417 ± 0.034**	**0.345 ± 0.049**	0.078 ± 0.005	0.40
		IWVAE	0.708 ± 0.021	0.461 ± 0.039	**0.073 ± 0.008**	1.00
1000, 200	Between	MCEM	5.224 ± 0.192	8.544 ± 0.407	0.243 ± 0.011	1.00
		MHRM	/	/	/	0.00
		IWVAE	**0.506 ± 0.015**	**0.348 ± 0.024**	**0.080 ± 0.006**	1.00
	Within	MCEM	2.976 ± 0.193	3.441 ± 0.304	0.157 ± 0.008	1.00
		MHRM	/	/	/	0.00
		IWVAE	**0.638 ± 0.015**	**0.345 ± 0.020**	**0.080 ± 0.005**	1.00
5,000, 300	Between	MCEM	0.612 ± 0.019	0.508 ± 0.048	0.114 ± 0.007	1.00
		MHRM	/	/	/	0.00
		IWVAE	**0.459 ± 0.012**	**0.261 ± 0.016**	**0.084 ± 0.005**	1.00
	Within	MCEM	0.693 ± 0.015	0.306 ± 0.020	**0.075 ± 0.004**	1.00
		MHRM	/	/	/	0.00
		IWVAE	**0.595 ± 0.013**	**0.271 ± 0.017**	0.082 ± 0.005	1.00
10,000, 500	Between	MCEM	0.561 ± 0.010	0.572 ± 0.032	0.097 ± 0.004	1.00
		MHRM	/	/	/	0.00
		IWVAE	**0.465 ± 0.011**	**0.258 ± 0.013**	**0.086 ± 0.004**	1.00
	Within	MCEM	0.678 ± 0.010	0.581 ± 0.047	**0.058 ± 0.003**	1.00
		MHRM	/	/	/	0.00
		IWVAE	**0.592 ± 0.011**	**0.271 ± 0.014**	0.085 ± 0.004	1.00

Factors are diagonal. **Between** item structure: each item depends on 1 factor. **Within** item structure: each item depends on 2 factors.

reaching a local optimum, we train it to maximize the tighter IW-ELBO until it converges again. Since the computation cost of IW-ELBO is more expensive than ELBO, our strategy is cheaper than training an IWAE from scratch. We refer to our model as *importance-weighted sampling enhanced VAE*(IWVAE).

To be more specific, in the first 1% of total iterations, we apply the *KL annealing* technique, i.e., at step *t*, we multiply the KL divergence term $D_{\mathrm{KL}}[q_{\phi}(\boldsymbol{\theta}_i \mid \boldsymbol{y}_i)\|p(\boldsymbol{\theta}_i)]$ by a factor $\frac{t}{T_{\mathrm{anl}}}$, where $T_{\mathrm{anl}} = \lceil 0.01 T_{\max} \rceil$ and $T_{\max} = 2{,}00{,}000$ is a pre-specified maximum number of iterations to avoid the algorithm running forever due to convergence issues. In this stage, the weight of the KL term increases from 0 to 1 linearly. KL annealing has shown great improvement in deep generative models (Gulrajani et al., 2016; Sønderby et al., 2016). The rationale behind this technique is that the KL divergence term can over-regularize the model by forcing the approximate posterior $q_{\phi}(\boldsymbol{\theta}_i)$ close to the prior $p(\boldsymbol{\theta}_i)$ and leading the model to converge early to unsatisfactory local minimums. To mitigate this issue, at the beginning of training, we simply reduce the effect of the KL term. During the annealing stage, we fix $\boldsymbol{c}$ and $\boldsymbol{d}$ and only update $\phi, \mathbf{A}, \boldsymbol{b}$.

After the annealing stage, we train IWVAE until its estimated ELBO converges such that the averaged ELBO value in every 100 steps stops increasing for $L = 50$ times. We refer to this stage as *ELBO converging*. Finally, we use importance-weighted samples to train IWVAE until it converges again in terms of IW-ELBO with this same rule. This stage is referred to *IW-ELBO converging*. After this stage, we end up training.

Algorithm 1 demonstrates our training method in a simplified version where at each step only 1 sample is drawn randomly from the data to estimate gradients. In practice, people can instead collect multiple samples (known as a *mini batch*) at each step and take the average for better gradients estimators. In practice, we used a mini batch size of 16 for each iteration step throughout all stages, $S = 1$ Monte Carlo sample in all three stages, and $R = 5$ importance samples in the last IW-ELBO converging stage following (Urban and Bauer, 2021). In terms of parameter updates, we use stochastic gradient ascent with fixed step size to maximize the ELBO or IW-ELBO. We assign a smaller step size (0.001) for parameters $\boldsymbol{c}$ and $\boldsymbol{d}$ as their ranges are smaller, and all other parameters are optimized with step size (0.01). No

Algorithm 1 Stochastic gradient ascent of IWVAE.

```
Input: data $\mathbf{Y}$; latent factor's dimension $K$; Monte Carlo and importance sample sizes $S, R$; maximum number of iterations $T_{\max}$.
Initialize $\hat{\phi}, \hat{M}_p$ using random samples

(KL annealing stage)
while iteration number $t$ not reaching $T_{\text{anl}} = \lceil 0.01 T_{\max} \rceil$ do
    randomly draw $\boldsymbol{y}_i$ from $Y$;
    draw $S$ samples $\boldsymbol{\theta}_i^s \sim \mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\sigma}_i^2)$ with Equation (10) where $(\boldsymbol{\mu}_i, \boldsymbol{\sigma}_i^2) = F_\phi(\boldsymbol{y}_i)$;
    compute $\log p(\boldsymbol{y}_i \mid \boldsymbol{\theta}_i^s; M_p)$ with Equation (1) or Equation (2), $D_{\text{KL}}[q_\phi(\boldsymbol{\theta}_i \mid \boldsymbol{y}_i) \| p(\boldsymbol{\theta}_i)]$ with Equation (11). Take 1 gradient ascent step on
        $\hat{\phi}, \hat{M}_p = \underset{\phi, M_p}{\text{argmax}} \frac{1}{S} \sum_{s=1}^{S} \log p(\boldsymbol{y}_i \mid \boldsymbol{\theta}_i^s; M_p) - \frac{t}{T_{\text{anl}}} D_{\text{KL}}[q_\phi(\boldsymbol{\theta}_i \mid \boldsymbol{y}_i) \| p(\boldsymbol{\theta}_i)]$
end while

(ELBO converging stage)
while iteration number $t$ not reaching $T_{\max}$ and ELBO not converging do
    randomly draw $\boldsymbol{y}_i$ from $Y$;
    draw $S$ samples $\boldsymbol{\theta}_i^s \sim \mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\sigma}_i^2)$ with Equation (10) where $(\boldsymbol{\mu}_i, \boldsymbol{\sigma}_i^2) = F_\phi(\boldsymbol{y}_i)$;
    compute $\log p(\boldsymbol{y}_i \mid \boldsymbol{\theta}_i^s; M_p)$ with Equation (1) or Equation (2), $D_{\text{KL}}[q_\phi(\boldsymbol{\theta}_i \mid \boldsymbol{y}_i) \| p(\boldsymbol{\theta}_i)]$ with Equation (11). Take 1 gradient ascent step on
        $\hat{\phi}, \hat{M}_p = \underset{\phi, M_p}{\text{argmax}} \frac{1}{S} \sum_{s=1}^{S} \log p(\boldsymbol{y}_i \mid \boldsymbol{\theta}_i^s; M_p) - D_{\text{KL}}[q_\phi(\boldsymbol{\theta}_i \mid \boldsymbol{y}_i) \| p(\boldsymbol{\theta}_i)]$
end while

(IW-ELBO converging stage)
while iteration number $t$ not reaching $T_{\max}$ and IW-ELBO not converging do
    randomly draw $\boldsymbol{y}_i$ from $Y$;
    draw $SR$ samples $\boldsymbol{\theta}_i^{rs} \sim \mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\sigma}_i^2)$ with Equation (10) where $(\boldsymbol{\mu}_i, \boldsymbol{\sigma}_i^2) = F_\phi(\boldsymbol{y}_i)$;
    compute $\log p(\boldsymbol{y}_i \mid \boldsymbol{\theta}_i^s; M_p)$ with Equation (1) or Equation (2), $w_i^{rs} = \exp\left[\log p(\boldsymbol{y}_i \mid \boldsymbol{\theta}_i^{rs}; M_p) - \log q_\phi(\boldsymbol{\theta}_i^{rs} \mid \boldsymbol{y}_i) + \log p(\boldsymbol{\theta}_i^{rs})\right]$. Take 1 gradient ascent step on
        $\hat{\phi}, \hat{M}_p = \underset{\phi, M_p}{\text{argmax}} \frac{1}{S} \sum_{s=1}^{S} \left[\log \frac{1}{R} \sum_{r=1}^{R} w_i^{rs}\right]$
end while
Output: parameter estimates $\hat{\phi}, \hat{M}_p$
```

further tweaks such as gradient clippings (Pascanu et al., 2013) are used.

3. Simulation study

3.1. Data generation

To evaluate the performances of applying IWVAE to M3PL and M4PL models, we conducted a thorough simulation study. We considered both within item and between item multidimensionality. In particular, for the within item multidimensionality, each item was loaded on two factors; and for the between item multidimensionality, each item was loaded on one factor. Under both settings, items dependency were distributed to different factors evenly in an indirect way through a sparse $J \times K$ loading matrix $\mathbf{A}$. Specifically, we first generated a blocked diagonal submatrix $\mathbf{A}'$. Next, we repeated two steps iteratively: (1) flipped $\mathbf{A}'$ horizontally, and (2) concatenated to previous results, until we have the full s-shaped matrix $\mathbf{A}$. When J is not a multiple of row numbers of $\mathbf{A}'$, we truncated the resultant matrix at the bottom. To make the design more realistic and challenging, we considered a missing data design. For datasets with large J, it is impractical to have all items observed from every single respondent in realistic scenarios. To reflect this concern, we randomly masked a large portion (80% in our experiments) of responses from each respondent, assuming each respondent only answer 20% of the items.

Parameters M_p and latent factors Θ were generated as follows. For latent factor $\boldsymbol{\theta}_i$, under the *independent* factors setting, it was sampled from the standard multivariate Gaussian $\mathcal{N}(\mathbf{0}, \mathbf{I})$. Under the *correlated* factors setting, a covariance matrix Σ was first generated and shared by all $\boldsymbol{\theta}_i$. Specifically, the diagonal entries were set to 1 so that each factor has unit variance; and off-diagonal (specifically, upper diagonal) entries were sampled independently from $\mathcal{U}(0, 1)$. This Σ was accepted if it was positive semi-definitive, otherwise, another matrix was regenerated. For free parameters in the discrimination matrix $\alpha_{ij} \in \mathbf{A}'$, we sampled it from $\mathcal{U}(0.5, 1.5)$. For J pairs of guessing and upper asymptote parameters (c_j, d_j), we sampled them from $c_j \sim Beta(1, 9)$, $d_j \sim Beta(9, 1)$ in parallel and kept them if all $c_j < d_j$.

Our experiments were conducted as follows. First, we chose latent factors $\boldsymbol{\theta}_i$ for $i = 1, \ldots, N$ to be uncorrelated and studied two asymptotic regimes. Specifically, in the **single** asymptotic regime, the dimensions of items J and factors K were fixed to 100 and 5 respectively, and sample size N was increased from 500 to 10,000. In the **double** asymptotic regime, only K was fixed to 5 and J was increased from 100 to 500 as N grew. Under both settings, we chose $N \in \{500, 1{,}000, 5{,}000, 10{,}000\}$ and in the double asymptotic settings we further chose $J \in \{100, 200, 300, 500\}$. Under each combination of N, J, K, we evaluated performances of IWVAE, MCEM, and MHRM on the M3(4)PL model by checking item parameters M_p estimation. Finally, we duplicated this series of experiments to correlated factors settings.

We implemented IWVAE in PyTorch (Paszke et al., 2019) and MCEM in the mirt R package (Chalmers, 2012). All experiments were run on the same *high performance computing cluster* (HPCC) with 4 CPUs and 4 GB memory, and no GPU

TABLE 5 Mean and SE of RMSE of M_p estimate on M4PL models under single regime setting, best results are in bold.

N, J	Item structure	Model	rot(A)	$\boldsymbol{b}$	$\boldsymbol{c}$	$\boldsymbol{d}$	Success rates
500 100	Between	MCEM	11.248 ± 0.217	13.315 ± 0.491	0.172 ± 0.011	0.178 ± 0.010	1.00
		MHRM	/	/	/	/	/
		IWVAE	**0.654 ± 0.023**	**0.363 ± 0.024**	**0.081 ± 0.008**	**0.087 ± 0.008**	1.00
	Within	MCEM	12.038 ± 0.286	12.611 ± 0.665	0.148 ± 0.010	0.138 ± 0.008	1.00
		MHRM	/	/	/	/	/
		IWVAE	**0.736 ± 0.022**	**0.416 ± 0.032**	**0.073 ± 0.008**	**0.088 ± 0.008**	1.00
1000 100	Between	MCEM	8.231 ± 0.159	11.774 ± 0.417	0.180 ± 0.011	0.189 ± 0.012	1.00
		MHRM	/	/	/	/	/
		IWVAE	**0.486 ± 0.019**	**0.334 ± 0.028**	**0.079 ± 0.008**	**0.080 ± 0.008**	1.00
	Within	MCEM	7.181 ± 0.302	7.160 ± 0.527	0.108 ± 0.009	0.134 ± 0.010	1.00
		MHRM	/	/	/	/	/
		IWVAE	**0.623 ± 0.020**	**0.408 ± 0.033**	**0.069 ± 0.007**	**0.078 ± 0.008**	1.00
5,000 100	Between	MCEM	3.315 ± 0.167	4.790 ± 0.354	0.182 ± 0.014	0.159 ± 0.012	1.00
		MHRM	/	/	/	/	/
		IWVAE	**0.379 ± 0.026**	**0.363 ± 0.043**	**0.091 ± 0.011**	**0.082 ± 0.009**	1.00
	Within	MCEM	1.886 ± 0.152	1.468 ± 0.186	0.094 ± 0.009	0.087 ± 0.007	1.00
		MHRM	/	/	/	/	/
		IWVAE	**0.529 ± 0.031**	**0.397 ± 0.037**	**0.075 ± 0.008**	**0.086 ± 0.010**	1.00
10,000 100	Between	MCEM	1.918 ± 0.114	2.555 ± 0.213	0.157 ± 0.010	0.176 ± 0.011	1.00
		MHRM	/	/	/	/	/
		IWVAE	**0.380 ± 0.028**	**0.402 ± 0.040**	**0.084 ± 0.008**	**0.079 ± 0.008**	1.00
	Within	MCEM	1.127 ± 0.075	0.943 ± 0.079	0.095 ± 0.007	0.087 ± 0.006	1.00
		MHRM	/	/	/	/	/
		IWVAE	**0.520 ± 0.033**	**0.360 ± 0.039**	**0.085 ± 0.010**	**0.079 ± 0.008**	1.00

Factors are correlated. **Between** item structure: each item depends on 1 factor. **Within** item structure: each item depends on 2 factors.

was used. MHRM was implemented with FlexMIRT (Chung and Houts, 2020) and all experiments were fitted on a laptop with Intel Intel(R) Core(TM) i7-10750H CPU and 16 GB memory[2]. Because of the platform difference, we ran $B = 100$ independent replications for IWVAE and MCEM on each simulated dataset, and $B = 20$ replications for MHRM.

To evaluate the performances of MCEM, MHRM, and IWVAE, we followed Cho et al. (2021) and Urban and Bauer (2021) and reported *rooted mean squared error* (RMSE) across B independent experiment replications. Specifically, for each scalar parameter ξ (one of $\alpha_{jk}, b_j, c_j, d_j$ for $j = 1, \ldots, J, k = 1, \ldots, K$), RMSE for each parameter was computed by

$$\mathrm{RMSE}(\hat{\xi}) = \sqrt{\frac{1}{B}\sum_{b=1}^{B}(\hat{\xi}_b - \xi)^2}, \tag{17}$$

where $\hat{\xi}_b$ is the estimated value from the b-th replication. The final reported RMSEs were averages of corresponding entries in matrix $\mathbf{A}$ or vectors $\boldsymbol{b}, \boldsymbol{c}, \boldsymbol{d}$, and standard error were shown after each value in the parenthesis.

Note that the matrix $\mathbf{A}$ in MIRT (IRT) models can be only identified up to a rotation if no further prior constraint is imposed, and we conducted *post-hoc* processing on $\hat{\mathbf{A}}$ following other literature. Our transformation consisted of three steps. First, we applied the *promax* (Hendrickson and White, 1964) rotation to the estimated $\hat{\mathbf{A}}$, which allowed different factors to be correlated; we denoted this intermediate result with $\hat{\mathbf{A}}^r$. Next, for each column in $\hat{\mathbf{A}}^r$ that had a negative sum, we flipped its sign and the corresponding factor (refer to, e.g., Urban and Bauer, 2021), we marked the resultant matrix in this step as $\hat{\mathbf{A}}^{rf}$. Finally, we searched over the best permutation of columns of $\hat{\mathbf{A}}^{rf}$ such that RMSE was minimized, and the corresponding RMSEs were reported in tables.

We also utilized the CF-Quartimax rotation as in Cho et al. (2022) to evaluate the sparsity structure estimation of different methods. However, since sparsity estimation is not the main focus of this article, we defer presenting these results to the appendix.

Finally, Considering that M3PL is notoriously hard for MHRM to fit (Cho et al., 2021), and M4PL is expected to be more difficult, we reported the *success rate* of each method, which refers to the percentage of successful replications. The exact

[2] HPCC cannot be used due to the license issue of FlexMIRT.

TABLE 6 Mean and SE of RMSE of M_p estimate on M4PL models under double regime setting, best results are in bold.

N, J	Item structure	Model	rot(A)	*b*	*c*	*d*	Success rates
500 100	Between	MCEM	11.248 ± 0.217	13.315 ± 0.491	0.172 ± 0.011	0.178 ± 0.010	1.00
		MHRM	/	/	/	/	/
		IWVAE	**0.654 ± 0.023**	**0.363 ± 0.024**	**0.081 ± 0.008**	**0.087 ± 0.008**	1.00
	Within	MCEM	12.038 ± 0.286	12.611 ± 0.665	0.148 ± 0.010	0.138 ± 0.008	1.00
		MHRM	/	/	/	/	/
		IWVAE	**0.736 ± 0.022**	**0.416 ± 0.032**	**0.073 ± 0.008**	**0.088 ± 0.008**	1.00
1000 200	Between	MCEM	8.314 ± 0.097	11.742 ± 0.298	0.178 ± 0.008	0.197 ± 0.008	1.00
		MHRM	/	/	/	/	/
		IWVAE	**0.503 ± 0.015**	**0.362 ± 0.024**	**0.080 ± 0.006**	**0.088 ± 0.007**	1.00
	Within	MCEM	7.323 ± 0.213	7.628 ± 0.362	0.125 ± 0.006	0.131 ± 0.007	1.00
		MHRM	/	/	/	/	/
		IWVAE	**0.609 ± 0.015**	**0.407 ± 0.024**	**0.080 ± 0.005**	**0.088 ± 0.007**	1.00
5,000 300	Between	MCEM	2.041 ± 0.090	2.292 ± 0.154	0.143 ± 0.006	0.139 ± 0.006	1.00
		MHRM	/	/	/	/	/
		IWVAE	**0.426 ± 0.013**	**0.340 ± 0.020**	**0.084 ± 0.005**	**0.081 ± 0.005**	1.00
	Within	MCEM	1.049 ± 0.053	0.525 ± 0.065	**0.066 ± 0.005**	**0.059 ± 0.004**	1.00
		MHRM	/	/	/	/	/
		IWVAE	**0.582 ± 0.014**	**0.339 ± 0.019**	0.083 ± 0.005	0.080 ± 0.005	1.00
10,000 500	Between	MCEM	1.062 ± 0.036	1.163 ± 0.060	0.129 ± 0.004	0.131 ± 0.004	1.00
		MHRM	/	/	/	/	/
		IWVAE	**0.426 ± 0.011**	**0.332 ± 0.015**	**0.086 ± 0.004**	**0.086 ± 0.004**	1.00
	Within	MCEM	0.884 ± 0.015	0.944 ± 0.031	0.098 ± 0.003	0.099 ± 0.003	1.00
		MHRM	/	/	/	/	/
		IWVAE	**0.562 ± 0.012**	**0.367 ± 0.016**	**0.085 ± 0.004**	**0.087 ± 0.004**	1.00

Factors are correlated. **Between** item structure: each item depends on 1 factor. **Within** item structure: each item depends on 2 factors.

definition of *success* for different methods differs. For MCEM, it refers to the case where the MCEM algorithm terminates and provides estimates successfully, regardless of convergence[3]. For IWVAE, it also refers to successful termination without reaching the maximum iteration number, which implies proper convergence. The difference in *success*, as we shall see later, is influential: MHRM usually performed the best if it succeeds. MCEM, on the contrary, had much worse performances while *succeeding* in all experiments.

3.2. Numeric results

In this section, we show detailed numeric results on M_p estimations, which are summarized in Tables 1–8. In a nutshell, IWVAE achieved competitive or better performances compared to the two other statistical methods. IWVAE achieved much lower RMSE on nearly all item parameters in almost all experiments than MCEM; and unlike MHRM, IWVAE succeed in all experiments from small- to large-scale datasets. Additionally, IWVAE required much more scalable training times on all experiments, while MCEM and MHRM had time costs growing faster as sample size increased.

Tables 1–4 show RMSE of M_p estimation in M4PL and M3PL models under single and double asymptotic regimes, where different entries in each latent factor $\boldsymbol{\theta}$ were generated independently. Two item structures were reported together in the same table. First, we observed that MCEM and IWVAE are more robust, as they succeed in all experiments, while MHRM achieved a success rate of 50% on few experiments in the M3PL model. Next, IWVAE reached much lower RMSE than MCEM, especially on small to medium sized data. In addition, IWVAE showed similar tendencies as MCEM and MHRM did: as N grew, its RMSE showed remarkable decreases, and on more challenging within-item structure scenarios, IWVAE also had slightly higher RMSEs.

For experiments where each latent factor $\boldsymbol{\theta}$ has correlated components, we organized results in the same way as before. Tables 5–8 show RMSE of M_p estimation in M4PL and M3PL

[3] Unlike the mirt package, FlexMIRT only provides convergent estimates.

TABLE 7 Mean and SE of RMSE of M_p estimate on M3PL models under single regime setting, best results are in bold.

N, J	Item structure	Model	rot(A)	b	c	Success rates
500 100	Between	MCEM	9.137 ± 0.396	12.034 ± 0.799	0.192 ± 0.012	1.00
		MHRM	**0.331 ± 0.040**	0.441 ± 0.062	**0.077 ± 0.006**	0.05
		IWVAE	0.659 ± 0.020	**0.411 ± 0.029**	0.081 ± 0.008	1.00
	Within	MCEM	7.644 ± 0.465	7.291 ± 0.612	0.153 ± 0.011	1.00
		MHRM	**0.492 ± 0.048**	**0.364 ± 0.054**	**0.064 ± 0.005**	0.45
		IWVAE	0.733 ± 0.020	0.440 ± 0.038	0.073 ± 0.008	1.00
1000 100	Between	MCEM	5.231 ± 0.279	8.080 ± 0.573	0.219 ± 0.014	1.00
		MHRM	**0.284 ± 0.027**	0.350 ± 0.041	0.090 ± 0.007	0.25
		IWVAE	0.483 ± 0.019	**0.312 ± 0.024**	**0.079 ± 0.008**	1.00
	Within	MCEM	2.855 ± 0.299	2.912 ± 0.457	0.118 ± 0.010	1.00
		MHRM	**0.428 ± 0.034**	0.336 ± 0.020	**0.045 ± 0.004**	0.80
		IWVAE	0.601 ± 0.024	**0.299 ± 0.026**	0.069 ± 0.007	1.00
5,000 100	Between	MCEM	0.762 ± 0.063	0.638 ± 0.117	0.130 ± 0.015	1.00
		MHRM	**0.242 ± 0.019**	**0.120 ± 0.011**	**0.051 ± 0.005**	0.75
		IWVAE	0.415 ± 0.023	0.256 ± 0.031	0.091 ± 0.011	1.00
	Within	MCEM	0.986 ± 0.075	0.421 ± 0.054	0.066 ± 0.006	1.00
		MHRM	**0.357 ± 0.034**	0.460 ± 0.013	**0.030 ± 0.002**	0.80
		IWVAE	0.578 ± 0.029	**0.308 ± 0.034**	0.075 ± 0.008	1.00
10,000 100	Between	MCEM	0.917 ± 0.107	1.073 ± 0.170	0.110 ± 0.012	1.00
		MHRM	**0.155 ± 0.013**	**0.114 ± 0.014**	**0.037 ± 0.006**	0.65
		IWVAE	0.397 ± 0.027	0.297 ± 0.035	0.084 ± 0.008	1.00
	Within	MCEM	0.898 ± 0.056	0.751 ± 0.165	0.057 ± 0.006	1.00
		MHRM	**0.378 ± 0.036**	0.461 ± 0.027	**0.017 ± 0.002**	0.40
		IWVAE	0.547 ± 0.032	**0.302 ± 0.040**	0.084 ± 0.010	1.00

Factors are correlated. **Between** item structure: each item depends on 1 factor. **Within** item structure: each item depends on 2 factors.

models under single and double asymptotic regimes. Again, we observed similar results from IWVAE to MCEM and MHRM in terms of success times and RMSE, indicating the advantage of the proposed IWVAE method.

We finally analyzed the fitting time of IWVAE and MCEM and reported averaged time with stand error (in shadow) in Figure 1 (M3PL) and Figure 2 (M4PL). We combined different factor settings (*independent* and *correlated*), and item structures (*between* and *within*) for every pair of sample size N and item size J. Each point contains 80 trials. As MHRM could not fit M4PL and its convergence results under M3PL were not stable, here, we do not report their results. From Figures 1, 2, compared to MCEM, IWVAE required significantly lower fitting time. Unlike MCEM, IWVAE had a much more stable fitting time across different data sizes, which was also observed in Urban and Bauer (2021) for estimating M2PL. As in Urban and Bauer (2021), we also note that the computational time of IWVAE appeared not to increase with N and J, which may be due to that VAE-based models are more difficult to train on small data sets. Similarly, in some cases, the computational time of MCEM also dropped when N increased to $10,000$, which may also be because of the easier convergence of the algorithm for the larger datasets. Moreover, we observed that fitting time of MCEM under M4PL depended more on the choices of initialization, revealed by the width of empirical intervals in Figure 2.

4. Real data analysis

In this section, we evaluated the performance of IWVAE, MCEM, and MHRM on the multistage testing (MST) dataset from the National Assessment of Education Progress (NAEP). The data is from the 2011 grade 8 math assessment study. The NAEP MST design takes a two-stage form: in the routing stage, a block of items with medium difficulty is administered. Then in the second stage, there are three targeted blocks with varying difficulty—blocks of easy, medium, and hard items. Based on a person's performance in the routing block, one of the three targeted blocks is assigned in the second-stage accordingly. Because the assignment in stage II depends on

TABLE 8 Mean and SE of RMSE of M_p estimate on M3PL models under double regime setting, best results are in bold.

N, J	Item structure	Model	rot(A)	*b*	*c*	Success rates
500 100	Between	MCEM	9.137 ± 0.396	12.034 ± 0.799	0.192 ± 0.012	1.00
		MHRM	**0.331 ± 0.040**	0.441 ± 0.062	**0.077 ± 0.006**	0.05
		IWVAE	0.659 ± 0.020	**0.411 ± 0.029**	0.081 ± 0.008	1.00
	Within	MCEM	7.644 ± 0.465	7.291 ± 0.612	0.153 ± 0.011	1.00
		MHRM	**0.492 ± 0.048**	**0.364 ± 0.054**	**0.064 ± 0.005**	0.45
		IWVAE	0.733 ± 0.020	0.440 ± 0.038	0.073 ± 0.008	1.00
1000 200	Between	MCEM	4.662 ± 0.200	7.364 ± 0.410	0.220 ± 0.010	1.00
		MHRM	/	/	/	0.00
		IWVAE	**0.512 ± 0.014**	**0.329 ± 0.022**	**0.080 ± 0.006**	1.00
	Within	MCEM	2.233 ± 0.171	2.294 ± 0.266	0.104 ± 0.006	1.00
		MHRM	/	/	/	0.00
		IWVAE	**0.670 ± 0.014**	**0.318 ± 0.020**	**0.080 ± 0.005**	1.00
5,000 300	Between	MCEM	0.576 ± 0.017	0.450 ± 0.048	0.101 ± 0.006	1.00
		MHRM	/	/	/	0.00
		IWVAE	**0.450 ± 0.012**	**0.263 ± 0.016**	**0.084 ± 0.005**	1.00
	Within	MCEM	0.728 ± 0.013	**0.215 ± 0.017**	**0.050 ± 0.003**	1.00
		MHRM	/	/	/	0.00
		IWVAE	**0.624 ± 0.013**	0.290 ± 0.018	0.082 ± 0.005	1.00
10,000 500	Between	MCEM	0.571 ± 0.012	0.720 ± 0.049	0.089 ± 0.004	1.00
		MHRM	/	/	/	0.00
		IWVAE	**0.451 ± 0.011**	**0.246 ± 0.013**	**0.086 ± 0.004**	1.00
	Within	MCEM	0.858 ± 0.018	1.411 ± 0.087	**0.065 ± 0.002**	1.00
		MHRM	/	/	/	0.00
		IWVAE	**0.587 ± 0.011**	**0.267 ± 0.014**	0.085 ± 0.004	1.00

Factors are correlated. **Between** item structure: each item depends on 1 factor. **Within** item structure: each item depends on 2 factors.

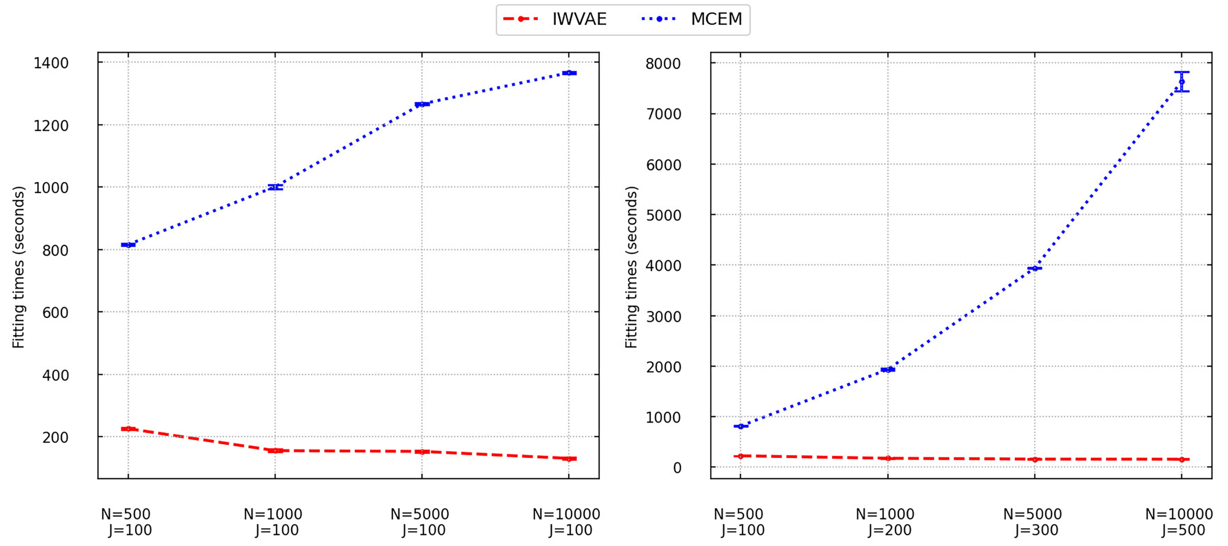

FIGURE 1
Fitting times for IWVAE and MCEM with M3PL model under the single (different sample sizes *N* and fixed item dimension *J*) and double [different (*N*, *J*)] asymptotic regimes. Vertical bar areas mark empirical 95% intervals.

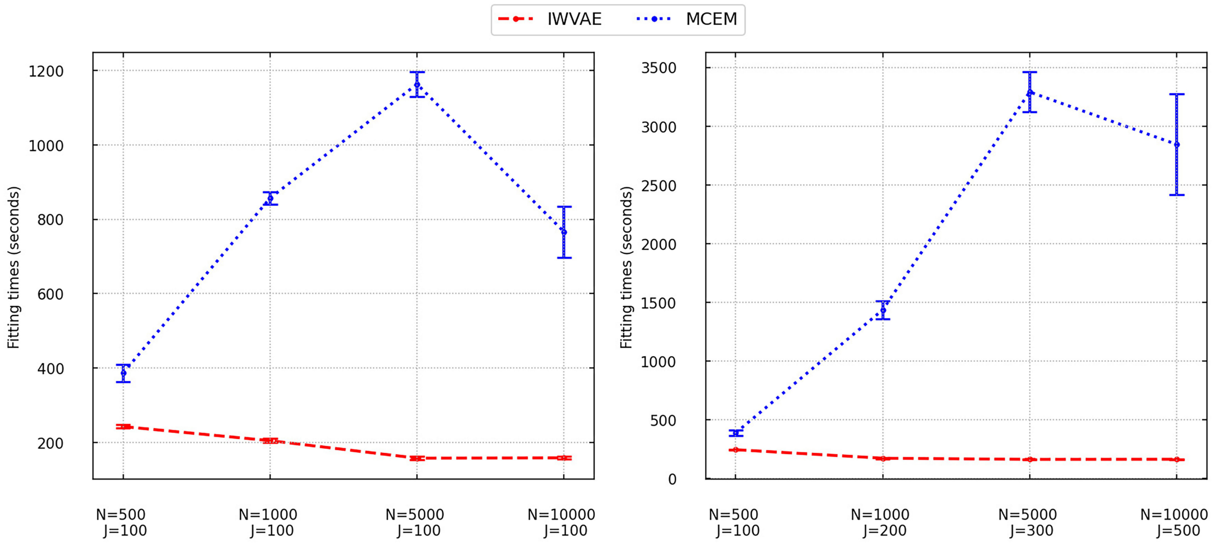

FIGURE 2
Fitting times for IWVAE and MCEM with M4PL model under the single (different sample sizes *N* and fixed item dimension *J*) and double [different (N, J)] asymptotic regimes. Vertical bar areas mark empirical 95% intervals.

TABLE 9 Comparison of estimated R_θ from different models on MST dataset.

Model	IWVAE	MCEM	MHRM
M4PL	$\begin{bmatrix} 1. & & & & \\ 0.624 & 1. & & & \\ 0.571 & 0.51 & 1. & & \\ 0.695 & 0.625 & 0.551 & 1. & \\ 0.53 & 0.521 & 0.457 & 0.546 & 1. \end{bmatrix}$	$\begin{bmatrix} 1. & & & & \\ 0.628 & 1. & & & \\ 0.616 & 0.604 & 1. & & \\ 0.595 & 0.599 & 0.543 & 1. & \\ 0.655 & 0.655 & 0.621 & 0.626 & 1. \end{bmatrix}$	/
M3PL	$\begin{bmatrix} 1. & & & & \\ 0.585 & 1. & & & \\ 0.557 & 0.699 & 1. & & \\ 0.531 & 0.671 & 0.653 & 1. & \\ 0.47 & 0.52 & 0.521 & 0.496 & 1. \end{bmatrix}$	$\begin{bmatrix} 1. & & & & \\ 0.465 & 1. & & & \\ 0.669 & 0.363 & 1. & & \\ 0.712 & 0.449 & 0.662 & 1. & \\ 0.668 & 0.43 & 0.603 & 0.665 & 1. \end{bmatrix}$	$\begin{bmatrix} 1. & & & & \\ 0.581 & 1. & & & \\ 0.52 & 0.589 & 1. & & \\ 0.565 & 0.689 & 0.582 & 1. & \\ 0.48 & 0.586 & 0.494 & 0.61 & 1. \end{bmatrix}$
M2PL	$\begin{bmatrix} 1. & & & & \\ 0.74 & 1. & & & \\ 0.622 & 0.628 & 1. & & \\ 0.58 & 0.529 & 0.515 & 1. & \\ 0.594 & 0.593 & 0.507 & 0.451 & 1. \end{bmatrix}$	$\begin{bmatrix} 1. & & & & \\ 0.264 & 1. & & & \\ 0.422 & 0.469 & 1. & & \\ 0.479 & 0.564 & 0.709 & 1. & \\ 0.435 & 0.479 & 0.625 & 0.69 & 1. \end{bmatrix}$	$\begin{bmatrix} 1. & & & & \\ 0.518 & 1. & & & \\ 0.593 & 0.584 & 1. & & \\ 0.548 & 0.548 & 0.618 & 1. & \\ 0.551 & 0.601 & 0.624 & 0.638 & 1. \end{bmatrix}$

the observed student performance in stage I, the MST design essentially generates a unique missing-at-random pattern. Due to the prevalence of MST design in large scale assessments, it would interesting to evaluate how the different estimation methods fare with such a design.

The data set contains $N = 3,344$ respondents and 74 items in total. The routing block contains two parallel forms with 17 items in each form. The three blocks in stage II contain 14, 13, and 13 items, respectively. Each person responded to 31 or 30 items out of 74. The items cover 5 different content domains, i.e., number properties and operations, measurement, geometry, data analysis statistics and probability, and algebra. The break down of items from each content domain in each form is presented in Table 8 by Wang et al. (2020). The content coverage is pretty balanced, which suggests a five-dimensional model to be appropriate. Hence, five-dimensional exploratory M2PL,

TABLE 10 Mean and SE of train and held-out accuracy/log-likelihood on MST dataset (over 5 replications).

Method	Model	Train accuracy	Held-out accuracy	Train log-likelihood	Held-out log-likelihood
	M4PL	0.707 ± 0.001	0.704 ± 0.002	-0.531 ± 0.001	-0.539 ± 0.001
IWVAE	M3PL	0.707 ± 0.000	0.706 ± 0.002	-0.530 ± 0.000	-0.537 ± 0.000
	M2PL	0.706 ± 0.001	0.703 ± 0.001	-0.531 ± 0.001	-0.539 ± 0.001
	M4PL	0.764 ± 0.001	0.693 ± 0.002	-0.481 ± 0.001	-0.603 ± 0.001
MCEM	M3PL	0.761 ± 0.000	0.697 ± 0.001	-0.482 ± 0.000	-0.599 ± 0.000
	M2PL	0.759 ± 0.001	0.697 ± 0.001	-0.485 ± 0.001	-0.589 ± 0.001
	M4PL	/	/	/	/
MHRM	M3PL	0.612 ± 0.003	0.613 ± 0.002	-0.682 ± 0.003	-0.683 ± 0.003
	M2PL	0.622 ± 0.003	0.623 ± 0.002	-0.662 ± 0.003	-0.664 ± 0.003

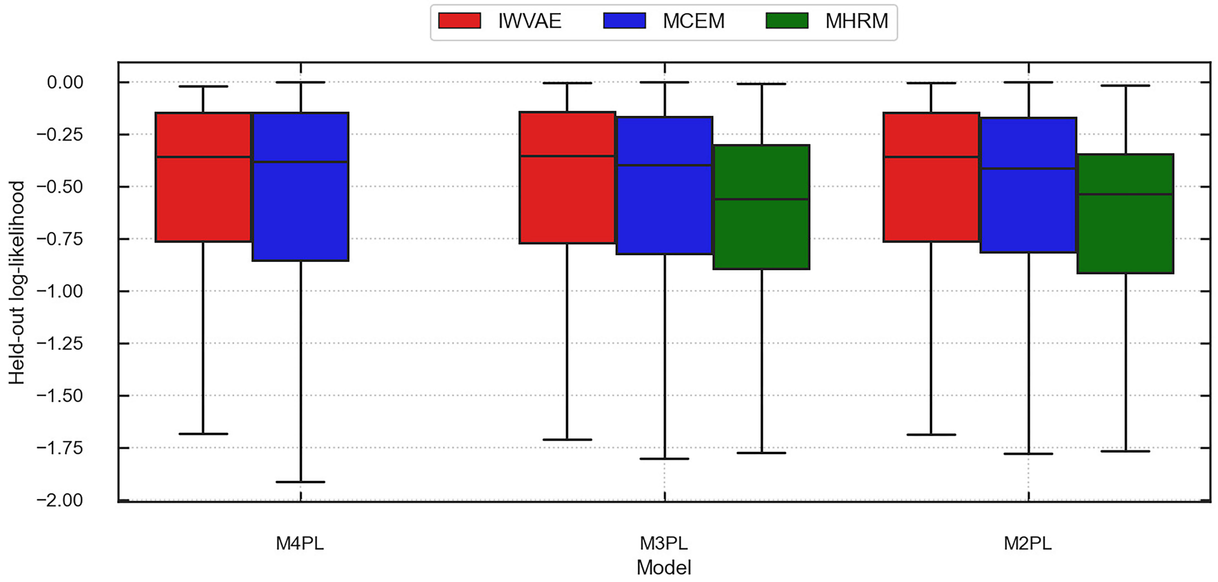

FIGURE 3
Predicted log-likelihood on held-out items using different methods (IWVAE, MHRM, MCEM) to fit different MIRT models on MST data from a randomly selected trial. Outlier predictions are removed.

M3PL, and M4PL models were fitted to the MST data using IWVAE, MHRM, and MCEM. We used the same algorithm, architecture, hyper-parameters, and stop criteria on IWVAE as in the simulation study except that we use a larger learning rate of 0.1 for $\phi, \mathbf{A}, \boldsymbol{b}$ and 0.01 for $\boldsymbol{c}, \boldsymbol{d}$.

First, we studied the estimates of the covariance matrix Σ_x and the comparison between the three methods. Due to the identifiability issue, in all models we assumed that covariance matrix of latent factors is $\Sigma = \mathbf{I}$ and conducted the promax rotation to estimate the correlation matrix $\hat{\mathbf{R}}$. After rotation, we adjusted the sign of the correlation depending on the sign of *post-hoc* transformed $\hat{\mathbf{A}}^{rf}$ as in the previous section. In particular, we flipped the sign of each column in $\hat{\mathbf{A}}^{r}$ if its sum was negative, and did the same to the corresponding columns and rows in the $\hat{\mathbf{R}}$. Table 9 shows estimated matrices under M4PL, M3PL, and M2PL, respectively. Under all settings, the correlation matrix recovered by IWVAE was similar to those from MCEM and MHRM, and a bit even closer to MHRM than MCEM did on M3PL and M2PL models.

Next, given that the true parameters are unknown, we evaluated the predictive performances of the three methods using a held-out validation. That is, we randomly marked 20% of items as *missing*, which played the role of held-out data, and used the remaining data to estimate our person and item parameters. That is, we used the estimated parameters to produce model-based predicted responses, compared them with the observed responses, and computed their consistency as a measure of accuracy. We computed such accuracy on both training data and held-out data. Higher accuracy indicates more alignment between model prediction and observed data. Meanwhile, we also used the estimated model parameters to compute log-likelihood, with a higher likelihood implying the

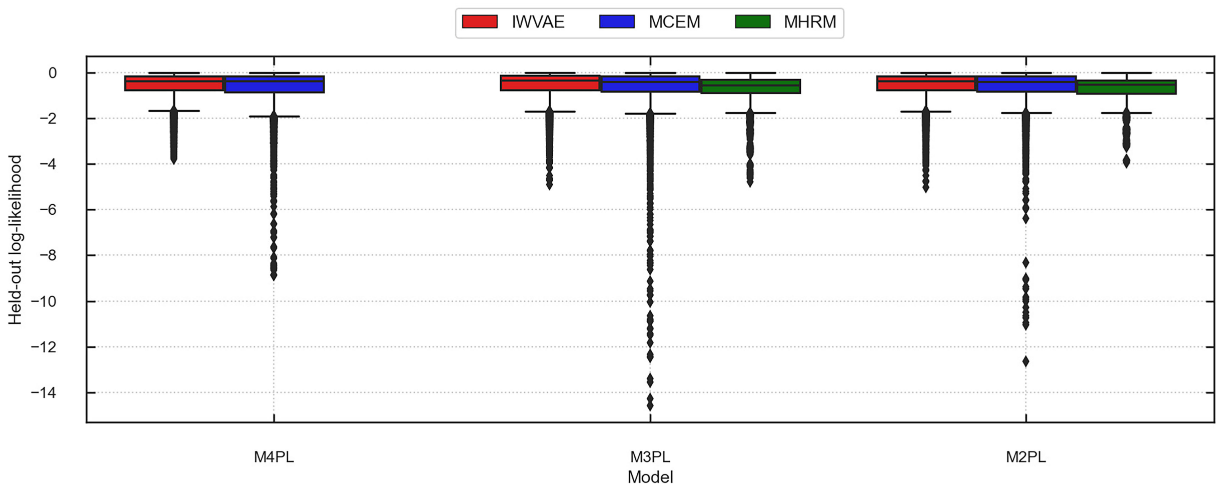

FIGURE 4
Predicted log-likelihood on held-out items using different methods (IWVAE, MHRM, MCEM) to fit different MIRT models on MST data from a randomly selected trial. Outlier predictions are kept.

estimated parameters may be better reflective of unknown truth. We reported accuracy and log-likelihood predicted by different methods on the training and held-out data in Table 10. To eliminate potential randomness in generating observed responses, 5 replications were done for each model, and we generated a different train and held-out data in each replication.

Table 10 summarizes the averaged accuracy and log-likelihood (of each item) on the train and held-out sets, where values in parentheses are stand errors across 5 replications. In this experiment, IWVAE achieved the highest held-out accuracy and log-likelihood. Figures 3, 4 further showed the corresponding log-likelihood values of each item. First, we observed that IWVAE had much fewer outliers than MCEM; after removing outliers, IWVAE achieved the highest log-likelihood on three MIRT models. Moreover, among the three models, the held-out accuracy, training data log-likelihood, and held-out log-likelihood from IWVAE were the best for M3PL. This is expected in that the operational model for NAEP analysis is indeed 3PL.

5. Discussions

In this article, we extend a variational autoencoder estimation method (Urban and Bauer, 2021) for the parameter estimation of the M3PL and M4PL models. By approximating the intractable log-likelihood with variational techniques, it provides a computationally efficient and scalable method for the estimation of large-scale assessment data. Simulation studies demonstrate that the proposed method outperforms the widely used MHRM and MCEM methods in terms of parameter recovery and computation time in both M3PL and M4PL. The proposed method is also more robust with many fewer issues of convergence. That said, we do want to caution readers that a robust algorithm cannot compensate for a lack of data. For M3PL and M4PL to be estimated well, there needs to be enough data at the two extreme ends of the latent trait scales to help estimate the lower and upper asymptote adequately.

Although this study focuses on the exploratory item factor analysis, the proposed algorithm can be easily applied to the confirmatory item factor analysis, where certain entries of the loading matrix are set to be 0 by users. Such structural restrictions can be naturally incorporated into the estimation. In addition, it would be also of interest to further estimate the sparsity loading structure from the responses. This can be achieved by adding a lasso-type regularization term into the loss function (the marginal log-likelihood function), which would induce sparse estimation results from the regularized algorithms.

Finally, a few interesting problems are left for future investigations. Very recent works suggested that some aspects of our training strategy can be improved; for instance, Collier et al. (2021) revealed that the missing data can be handled better than zero-imputation; and Wang et al. (2021) indicated a possible direction of understanding and solving the posterior collapse, which was solved by a KL annealing stage in our proposed method. Moreover, this work does not directly study the estimation uncertainty of the VAE estimation procedure. It is interesting to further develop valid statistical procedures to make inferences for the corresponding estimation results. Such an important problem, however, still remains unaddressed for VAE and related deep learning methods in the machine learning and statistics literature.

Data availability statement

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

Author contributions

TL contributed to implementing the studies and writing the initial draft. GX contributed to mentoring, conceptualizing ideas, acquiring funding, and manuscript revision. CW contributed to conceptualizing ideas, acquiring funding, and manuscript revision. All authors contributed to the article and approved the submitted version.

Funding

This study was partially supported by IES grant R305D200015, NSF grants SES-1846747 and NSF SES-2150601.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Supplementary material

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fpsyg.2022.935419/full#supplementary-material

References

Barton, M. A., and Lord, F. M. (1981). An upper asymptote for the three-parameter logistic item-response model. *ETS Res. Rep. Series* 1981, i-8. doi: 10.1002/j.2333-8504.1981.tb01255.x

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Berlin; Heidelberg: Springer-Verlag.

Blei, D. M., Kucukelbir, A., and McAuliffe, J. D. (2017). Variational Inference: a Review for Statisticians. *J. Am. Stat. Assoc.* 112, 859–877. doi: 10.1080/01621459.2017.1285773

Bock, R. D., and Aitkin, M. (1981). Marginal maximum likelihood estimation of item parameters: application of an EM algorithm. *Psychometrika* 46, 443–459. doi: 10.1007/BF02293801

Burda, Y., Grosse, R. B., and Salakhutdinov, R. (2016). "Importance weighted autoencoders," in *ICLR* (San Juan).

Cai, L. (2010a). High-dimensional exploratory item factor analysis by a metropolis "hastings robbins" monro algorithm. *Psychometrika* 75, 33–57. doi: 10.1007/s11336-009-9136-x

Cai, L. (2010b). Metropolis-hastings robbins-monro algorithm for confirmatory item factor analysis. *J. Educ. Behav. Stat.*35, 307–335. doi: 10.3102/1076998609353115

Chalmers, R. P. (2012). mirt: a multidimensional item response theory package for the R environment. *J. Stat. Softw.* 48, 1–29. doi: 10.18637/jss.v048.i06

Chen, Y., Li, X., and Zhang, S. (2019). Joint maximum likelihood estimation for high-dimensional exploratory item response analysis. *Psychometrika* 84, 124–146. doi: 10.1007/s11336-018-9646-5

Cho, A. E., Wang, C., Zhang, X., and Xu, G. (2021). Gaussian variational estimation for multidimensional item response theory. *Br. J. Math. Stat. Psychol.* 74, 52–85. doi: 10.1111/bmsp.12219

Cho, A. E., Xiao, J., Wang, C., and Xu, G. (2022). Regularized variational estimation for exploratory item factor analysis. *Psychometrika*. doi: 10.1007/s11336-022-09874-6

Chung, S., and Houts, C. (2020). flexMIRT: a flexible modeling package for multidimensional item response models. *Measurement Interdisc. Res. Perspect.* 18, 40–54. doi: 10.1080/15366367.2019.1693825

Collier, M., Nazabal, A., and Williams, C. K. I. (2021). VAEs in the presence of missing data. *arXiv preprint arXiv:2006.05301*.

Curi, M., Converse, G. A., Hajewski, J., and Oliveira, S. (2019). "Interpretable variational autoencoders for cognitive models," in *2019 International Joint Conference on Neural Networks (IJCNN)* (Budapest: IEEE), 1–8.

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Math. Control Signals Syst.* 2, 303–314. doi: 10.1007/BF02551274

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *J. R. Stat. Soc. B* 39, 1–38.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. The MIT Press.

Gulrajani, I., Kumar, K., Ahmed, F., Taiga, A. A., Visin, F., Vazquez, D., et al. (2016). PixelVAE: a latent variable model for natural images. *arXiv:1611.05013 [cs.LG]*. doi: 10.48550/arXiv.1611.05013

Hambleton, R. K., and Swaminathan, H. (1985). *Item Response Theory: Principles and Applications*. Boston, MA: Kluwer Academic.

Hendrickson, A. E., and White, P. O. (1964). Promax: a quick method for rotation to oblique simple structure. *Br. J. Stat. Psychol.* 17, 65–70. doi: 10.1111/j.2044-8317.1964.tb00244.x

Hornik, K. (1991). Approximation capabilities of multilayer feedforward networks. *Neural Networks* 4, 251–257. doi: 10.1016/0893-6080(91)90009-T

Hui, F. K., Warton, D. I., Ormerod, J. T., Haapaniemi, V., and Taskinen, S. (2017). Variational approximations for generalized linear latent variable models. *J. Comput. Graph. Stat.* 26, 35–43. doi: 10.1080/10618600.2016.1164708

Jeon, M., Rijmen, F., and Rabe-Hesketh, S. (2017). A variational maximization-maximization algorithm for generalized linear mixed models with crossed random effects. *Psychometrika* 82, 693–716. doi: 10.1007/s11336-017-9555-z

Kingma, D. P., Salimans, T., Jozefowicz, R., Chen, X., Sutskever, I., and Welling, M. (2016). "Improving variational inference with inverse autoregressive flow," *Neural Information Processing Systems* (Barcelona), 29.

Kingma, D. P., and Welling, M. (2014). Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*. doi: 10.48550/arXiv.1312.6114

Kingma, D. P., and Welling, M. (2019). An introduction to variational autoencoders. *Foundat. Trends Mach. Learn*. 12, 307–392. doi: 10.1561/2200000056

Kucukelbir, A., Tran, D., Ranganath, R., Gelman, A., and Blei, D. M. (2017). Automatic differentiation variational inference. *J. Mach. Learn. Res*. 18, 1–45. doi: 10.48550/arXiv.1603.00788

Lindstrom, M. J., and Bates, D. M. (1988). Newton-raphson and EM algorithms for linear mixed-effects models for repeated-measures data. *J. Am. Stat. Assoc*. 83, 1014–1022.

Loken, E., and Rulison, K. L. (2010). Estimation of a four-parameter item response theory model. *Br. J. Math. Stat. Psychol*. 63, 509–525. doi: 10.1348/000711009X474502

McCulloch, C. (1997). Maximum likelihood algorithms for generalized linear mixed models. *J. Am. Stat. Assoc*. 92, 162–170. doi: 10.1080/01621459.1997.10473613

McDonald, R. P. (1967). Nonlinear factor analysis. *Psychometric. Monogr*. 15, 167–167.

Meng, X., Xu, G., Zhang, J., and Tao, J. (2020). Marginalized maximum a posteriori estimation for the four-parameter logistic model under a mixture modelling framework. *Br. J. Math. Stat. Psychol*. 73, 51–82. doi: 10.1111/bmsp.12185

Nair, V., and Hinton, G. E. (2010). "Rectified linear units improve restricted boltzmann machines," in *International Conference on Machine Learning* (Haifa).

Natesan, P., Nandakumar, R., Minka, T., and Rubright, J. D. (2016). Bayesian prior choice in IRT estimation using MCMC and variational bayes. *Front. Psychol*. 7, 1422. doi: 10.3389/fpsyg.2016.01422

Nazabal, A., Olmos, P. M., Ghahramani, Z., and Valera, I. (2020). Handling incomplete heterogeneous data using VAEs. *Pattern Recognit*. 107, 107501. doi: 10.1016/j.patcog.2020.107501

Ogasawara, H. (2002). Stable response functions with unstable item parameter estimates. *Appl. Psychol. Meas*. 26, 239–254. doi: 10.1177/0146621602026003001

Papamakarios, G., Nalisnick, E., Rezende, D. J., Mohamed, S., and Lakshminarayanan, B. (2021). Normalizing flows for probabilistic modeling and inference. *J. Mach. Learn. Res*. 22, 1–64. doi: 10.48550/arXiv.1912.02762

Pascanu, R., Mikolov, T., and Bengio, Y. (2013). "On the difficulty of training recurrent neural networks," in *International Conference on Machine Learning* (PMLR), 1310–1318.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., et al. (2019). "Pytorch: an imperative style, high-performance deep learning library," in *Neural Information Processing Systems Vol. 32* (Vancouver, CA).

Rainforth, T., Kosiorek, A., Le, T. A., Maddison, C., Igl, M., Wood, F., et al. (2018). "Tighter variational bounds are not necessarily better," in *International Conference on Machine Learning* (Stockholm), 4277–4285.

Reckase, M. D. (2009). "Multidimensional item response theory models," in *Multidimensional Item Response Theory* (Springer), 79–112.

Reise, S. P., and Waller, N. G. (2003). How many IRT parameters does it take to model psychopathology items? *Psychol. Methods* 8, 164. doi: 10.1037/1082-989x.8.2.164

Rijmen, F., and Jeon, M. (2013). Fitting an item response theory model with random item effects across groups by a variational approximation method. *Ann. Operat. Res*. 206, 647–662. doi: 10.1007/s10479-012-1181-7

Sønderby, C. K., Raiko, T., Maaløe, L., Sønderby, S. K., and Winther, O. (2016). "Ladder variational autoencoders," in *Neural Information Processing Systems, Vol. 29* (Barcelona).

Sonoda, S., and Murata, N. (2017). Neural network with unbounded activation functions is universal approximator. *Appl. Comput. Harmon Anal*. 43, 233–268. doi: 10.1016/j.acha.2015.12.005

Tierney, L., and Kadane, J. B. (1986). Accurate approximations for posterior moments and marginal densities. *J. Am. Stat. Assoc*. 81, 82–86. doi: 10.1080/01621459.1986.10478240

Tucker, G., Lawson, D., Gu, S., and Maddison, C. J. (2018). "Doubly reparameterized gradient estimators for monte carlo objectives," in *International Conference on Learning Representations* (Vancouver, CA).

Urban, C. J., and Bauer, D. J. (2021). A deep learning algorithm for high-dimensional exploratory item factor analysis. *Psychometrika* 86, 1–29. doi: 10.1007/s11336-021-09748-3

von Davier, M., and Sinharay, S. (2010). Stochastic approximation methods for latent regression item response models. *J. Educ. Behav. Stat*. 35, 174–193. doi: 10.3102/1076998609346970

Waller, N. G., and Feuerstahler, L. (2017). Bayesian modal estimation of the four-parameter item response model in real, realistic, and idealized data sets. *Multivariate Behav Res*. 52, 350–370. doi: 10.1080/00273171.2017.1292893

Waller, N. G., and Reise, S. (2010). "Measuring psychopathology with non-standard IRT models: fitting the four-parameter model to the MMPI," in *Measuring Psychological Constructs With Model-Based Approaches* (American Psychological Association), 147–173.

Wang, C., Chen, P., and Jiang, S. (2020). Item calibration methods with multiple subscale multistage testing. *J. Educ. Meas*. 57, 3–28. doi: 10.1111/jedm.12241

Wang, Y., Blei, D., and Cunningham, J. P. (2021). "Posterior collapse and latent variable non-identifiability," in *Neural Information Processing Systems, Vol. 34*.

Wolfinger, R., and O'connell, M. (1993). Generalized linear mixed models a pseudo-likelihood approach. *J. Stat. Comput. Simul*. 48, 233–243.

Wu, M., Davis, R. L., Domingue, B. W., Piech, C., and Goodman, N. (2020). Variational item response theory: fast, accurate, and expressive. *arXiv preprint arXiv:2002.00276*. doi: 10.48550/arXiv.2002.00276

Yen, Y.-C., Ho, R.-G., Laio, W.-W., Chen, L.-J., and Kuo, C.-C. (2012). An empirical evaluation of the slip correction in the four parameter logistic models with computerized adaptive testing. *Appl. Psychol. Meas*. 36, 75–87. doi: 10.1177/0146621611432862

Zhang, A., Lipton, Z. C., Li, M., and Smola, A. J. (2021). Dive into deep learning. *arXiv preprint arXiv:2106.11342*.

Zhang, S., Chen, Y., and Liu, Y. (2020). An improved stochastic EM Algorithm for large-scale full-information item factor analysis. *Br. J. Math. Stat. Psychol*. 73, 44–71. doi: 10.1111/bmsp.12153