Handling Missing Data In Variational Autoencoder based Item Response Theory

Karel Veldkamp

May 2023

Abstract

Recently Variational Autoencoders (VAEs) have been proposed as a method to estimate high dimensional Item Response Theory (IRT) models on large datasets. Although these improve the efficiency of estimation drastically compared to traditional methods, they have no natural way to deal with missing values. In this paper, we adapt three existing methods from the VAE literature to the IRT setting and propose one new method. We compare the performance of the different VAE-based methods to each other and to marginal maximum likelihood estimation for increasing levels of missing data in a simulation study for both three- and ten-dimensional IRT models. Additionally, we demonstrate the use of the VAE-based models on an existing algebra test dataset. Results confirm that VAE-based methods are a time-efficient alternative to marginal maximum likelihood, but that a larger number of importance-weighted samples are needed when the proportion of missing values is large.

1 Introduction

Estimating high-dimensional Item response theory (IRT) models is known to be computationally challenging due to the high dimensional integral involved in the computation of the marginal log-likelihood. Contemporary research has focused on various ways to approximate this integral efficiently, such as Markov chain Monte Carlo methods (e.g., Edwards, 2010) or stochastic approximation (e.g., Cai, 2010). Although these methods substantially improve the efficiency of estimation of multi-dimensional models, the exponential growth of complexity as the latent dimensionality increases still results in long estimation times in high-dimensional applications and on large datasets. One approach that avoids the computation of the marginal log-likelihood entirely is the use of variational autoencoders (VAE)(Curi et al., 2019; Urban & Bauer, 2021). These methods estimate a function that predicts the posterior person parameters from the observed data by maximizing a lower bound on the marginal log-likelihood. Specifically, VAEs use a feedforward neural network (FNN), which is a universal function approximator common in the field of machine learning. In addition

to avoiding the high-dimensional integral, VAEs drastically improve efficiency on large datasets by equating the parameters of this FNN across observations (see Urban & Bauer, 2021).

Despite being a promising framework for estimating high dimensional MIRT models, a drawback of the VAE approach is that dealing with missing data is not as straightforward. Where traditional methods draw strength from being full information maximum likelihood procedures in which missing data can easily be dealt with, a FNN can not handle missing input values naturally. There are various approaches to missing data in the general literature on VAEs, but only one pragmatic and potentially suboptimal method has been adapted to the multidimensional item response theory (MIRT) context (Liu et al., 2022). In this study, we adapt three missing data techniques from the VAE literature to VAE-based MIRT modeling and propose one new approach to this problem. In a simulation study in which we vary the degree of missing data, the performances of the four approaches are compared to each other and to full information marginal maximum likelihood (MML)-estimation. The rest of this paper is structured as follows. First, we briefly discuss FNNs, as these are important tools used in VAE-based MIRT. We then formally introduce MIRT and show the challenges associated with MML for high-dimensional IRT models. Next, we discuss variational inference and VAEs for MIRT models and derive different approaches to dealing with missing data in this paradigm. We compare the different missing data techniques in a simulation study and demonstrate the use of the methods on a real dataset pertaining to a large scale algebra test. We end with a general conclusion and some recommendations on how to deal with missing data in VAE based MIRT modeling.

2 Feed forward neural networks

FNNs, are a popular class of methods in the field of machine learning. They map an p dimensional vector of predictor variables $\mathbf{x}$ to an o dimensional vector of outcome variables $\mathbf{y}$. As the name suggests, a feedforward neural network consists of multiple layers, where the values of variables in the next layer depend on the values of the previous layer:

$$\mathbf{h}_{i+1} = f_i(\mathbf{W}_i\mathbf{h}_i + \mathbf{b}_i), i = 0, 1, \dots I, \tag{1}$$

where $\mathbf{W}_i$ is a matrix of weight parameters for layer i, $\mathbf{b}_i$ is a vector of bias parameters for layer i, and f_i is a nonlinear function called an activation function. $\mathbf{h_i}$ denotes the n_i dimensional representation vector in layer i, where n_i is the dimensionality of the layer. Note that for convenience $\mathbf{h}_0$ denotes the N dimensional vector of original observations $\mathbf{x}$, and $\mathbf{h}_I$ denotes the outcome predictions $\hat{\mathbf{y}}$. The intermediate representations $\mathbf{h}_i, 0 < i < I$, are generally referred to as hidden layer activations, and the individual elements in $\mathbf{h}_i$ are referred to as hidden nodes. Conceptually, a FNN consists of multiple linear transformations, each followed by a nonlinear function. Hornik et al. (1989) has shown that theoretically, FNNs with just a single hidden layer can approximate

any Borel measurable function to any degree of accuracy, given enough hidden nodes, making FNNs a form of universal function approximators.

The parameters of an FNN are generally estimated by numerical optimization of a loss function $\Lambda(\mathbf{y}, \hat{\mathbf{y}})$, such as the log-likelihood, using gradient descent. Modern implementations use Stochastic gradient descent (SGD) (Amari, 1993). The core idea of SGD is to iteratively update parameter values by partitioning the dataset into random disjoint subsets $\{\mathbf{x}_b\}_{b=1}^{B}$ (called mini-batches), computing the predicted values $\{\hat{\mathbf{y}}_b\}_{b=1}^{B}$, and updating the parameters by adding the gradients of the loss function with respect to the weights and biases multiplied by a learning rate, which is a hyperparameter that determines the size of the updates. This process is repeated until some stopping criterion is reached. The partitioning of the dataset allows the estimation process to be faster, while also making it less sensitive to local minima. The specific algorithm used in this study is the AMSGrad algorithm (Reddi et al., 2019), which is currently one of the most popular adaptations of SGD for FNNs. It makes use of an adaptive learning rate, allowing for bigger updates when gradients are small accelerating convergence, and smaller updates when gradients are big, preventing overshooting. We refer the reader to Reddi et al. (2019) for a detailed discussion of AMSGrad.

3 Estimation of MIRT models

The Multidimensional 2PL (M2PL) model is the most common MIRT model (McKinley & Reckase, 1983) due to its relation to item factor analysis (Wirth & Edwards, 2007; Takane & De Leeuw, 1987). In this model, the probability of a correct response is modeled as

$$P(X_{ij} = 1|\boldsymbol{\theta}_j) = \frac{1}{1 + \exp(-\mathbf{a}_j^T \boldsymbol{\theta}_i - b_j)}, \tag{2}$$

where X_{ij} is a random variable representing the response of participant i on item j, $\boldsymbol{\theta}_i$ is the latent variable vector for person i, $\mathbf{a}_j$ is the item slope vector for item j, and b_j is the item intercept for item j. The latent variable parameter vectors represent scores on the multidimensional latent ability constructs, whereas the item parameters represent the degree to which an item discriminates between different levels of each latent construct, as well as how likely an item is to receive a positive response generally. The marginal log-likelihood for person i is given by

$$\log P(X_i = \mathbf{x}_i|\Omega) = \log \int \cdots \int \prod_{j=1}^{J} P(X_{ij} = x_{ij}|\boldsymbol{\theta}; \Omega_j) p(\boldsymbol{\theta}) d\theta_1 d\theta_2 \cdots d\theta_D, \tag{3}$$

where X_i is a vector of random variables representing the binary item responses for subject i with elements X_{ij}, $\Omega_j = \{\mathbf{a}_j, b_j\} \in \Omega$ is the set of item parameters for item j, θ_{id} is the ability estimate for person i and dimension d, and D

is the number of latent dimensions. Some elements of the item discrimination parameters $\mathbf{a}_j, j = 1, \ldots, J$, have to be fixed to zero in order to avoid rotational indeterminacy. The complete marginal log-likelihood is simply the sum over all participants. This inegral is generally evaluated using quadrature approximation or stochochastic methods, which become increasingly more complex as the dimensionality of the latent construct increases, making it computationally intensive to estimate high dimensional MIRT models using MML.

3.1 Variational inference

As mentioned above, variational inference provides an alternative way to estimate MIRT parameters, avoiding the need for this computationally expensive integral. In variational inference, rather than optimizing the marginal likelihood directly, the posterior density $p(\boldsymbol{\theta}_i|\mathbf{x}_i)$, for which there is generally no closed expression, is approximated by another density $\hat{q}(\boldsymbol{\theta}_i|\mathbf{x}_i)$ such that the Kullback-Leibler (KL) divergence is minimized over a (parameterized) family $\mathcal{F}$ of proposal distributions:

$$\hat{q}(\boldsymbol{\theta}_i \,|\, \mathbf{x}_i) = \underset{q(\boldsymbol{\theta}|\mathbf{x}) \in \mathcal{F}}{\operatorname{argmin}} \, \mathrm{KL}[q(\boldsymbol{\theta}_i|\mathbf{x}_i)||p(\boldsymbol{\theta_i}|\mathbf{x}_i)] \tag{4}$$

(Blei et al., 2017). Since this KL divergence in this expression requires the true posterior, we can not optimize it directly. However, it can be shown that minimizing this KL divergence is equivalent to maximizing the evidence lower bound (ELBO), which is defined as

$$\mathrm{ELBO} = \mathbb{E}_{\hat{q}}[\log p(\mathbf{x}_i|\boldsymbol{\theta}_i)] - \mathrm{KL}[q(\boldsymbol{\theta}_i|\mathbf{x}_i)||p(\boldsymbol{\theta}_i)], \tag{5}$$

where $p(\boldsymbol{\theta}_i)$ is the prior density over $\boldsymbol{\theta}_i$ (i.e., the population density of the latent ability scores). The ELBO derives its name from the fact that it is a lower bound to the marginal log-likelihood. Specifically, the marginal log-likelihood can be expressed as

$$\log p(\mathbf{x}_i) = \mathrm{KL}[\hat{q}(\boldsymbol{\theta}_i|\mathbf{x}_i)||p(\boldsymbol{\theta}_i|\mathbf{x}_i)] + \mathrm{ELBO}. \tag{6}$$

This indicates that the ELBO approaches the marginal log-likelihood as the approximation distribution approaches the true posterior. Choosing a flexible family $\mathcal{F}$ for q allows this divergence to be minimized as much as possible. We will address the choice for q below.

3.2 Amortised variational inference

Where traditional variational inference optimizes the parameters of the approximate posterior directly, amortized variational inference estimates a mapping from the observed data to the posterior parameters. This means that instead of estimating a separate posterior distribution for the latent ability of each respondent, we estimate the parameters of a so-called inference model that predicts the posterior distribution of the latent ability parameters from a pattern of item

responses. This function is generally chosen to be approximated with a FNN (Curi et al., 2019; Urban & Bauer, 2021; Converse et al., 2021; Liu et al., 2022), and the parameters are shared across observations. As a result, if the posterior distribution of the latent variables is assumed to be a multivariate normal distribution, the latent variables $\boldsymbol{\theta}_i$ are modeled by:

$$\boldsymbol{\theta}_i \sim MVN(\boldsymbol{\mu}_i, \text{diag}(\boldsymbol{\sigma}_i)), \text{ with } (\boldsymbol{\mu}_i, \boldsymbol{\sigma}_i) = \text{FNN}_\phi(\mathbf{x}_i). \tag{7}$$

where ϕ are the parameters of the inference model. The full model is a VAE (Kingma & Welling, 2013) in which the parameters are estimated iteratively by taking a sample $\hat{\boldsymbol{\theta}}_i$ from the multivariate normal (cf. Equation (7)), with the FNN evaluated at the current best estimates of $\hat{\phi}$. Then, $\hat{\Omega}$ and $\hat{\phi}$ are jointly updated to maximize the ELBO with respect to these parameters. That is, the new estimates are

$$\hat{\phi}, \hat{\Omega} = \underset{\phi,\Omega}{\text{argmax}}\, \mathbb{E}[\log p(\mathbf{x}_i|\hat{\boldsymbol{\theta}}_i; \Omega)] - KL[q(\boldsymbol{\theta}_i|\mathbf{x}_i; \phi)||p(\boldsymbol{\theta}_i)], \tag{8}$$

where the approximate posterior $q(.)$ is given by Equation (7). In the simplest case, the population density $p(\boldsymbol{\theta}_i)$ is fixed to a standard multivariate normal density, but it is straightforward to accommodate correlated latent abilities in this prior. Below we discuss how this assumption of a normal approximate posterior can be relaxed.

3.3 Importance weighted amortized variational inference

As discussed above, VAEs generally assume that the posterior distribution of $\boldsymbol{\theta}_i$ is normal and highly factorized (e.g., multivariate normal with diagonal covariance matrix). These constraints on the posterior can cause the ELBO to severely underestimate the true marginal log-likelihood. Recent research has focused on various ways of tightening this lower bound (Rezende & Mohamed, 2015; Burda et al., 2015). One of these approaches, proposed by Burda et al. (2015), is the importance-weighted variational autoencoder (IWVAE). This model has the same structure as a regular VAE, but instead of taking a single sample of the approximate posterior, multiple Monte Carlo samples are taken, in order to calculate an importance weighted estimate of the log-likelihood:

$$\text{IW-ELBO} = \mathbb{E}_{1:k}\left[\log \frac{1}{K}\sum_{k=1}^{K} \frac{p(\mathbf{x}_i, \hat{\boldsymbol{\theta}}_i^{(k)})}{q(\hat{\boldsymbol{\theta}}_i^{(k)}|\mathbf{x}_i)}\right], \tag{9}$$

where K is the number of importance weights and the term inside the sum are the unnormalized importance weights for the joint distribution. When $K = 1$ the importance-weighted evidence lower bound (IW-ELBO) is equivalent to the regular ELBO in Equation (5). Using multiple importance-weighted samples from the approximate posterior gives the model more flexibility to approximate the true posterior, tightening the lower bound to the marginal log-likelihood. Burda et al. (2015) show that the IW-ELBO approaches $\log p(\mathbf{x}_i)$ as K goes to

infinity, making equation (9) a MML estimator. The IWVAE was first used in the context of MIRT by Urban & Bauer (2021) who show empirically that the IWVAE can provide accurate MIRT parameter estimates for up to 10 dimensions while being much faster than traditional expectation maximisation (EM) algorithms.

3.4 The challenge of missing data

Although the amortization step allows IWVAEs to efficiently estimate complex latent variable models, it introduces a problem regarding missing data. Since the model now includes an inference model that predicts the parameters of the latent variable distribution based on an observed response pattern (equation (7)), this function needs to be capable of dealing with missing data. A standard FNN does not meet this requirement, since missing values make it impossible to calculate gradients with respect to the network parameters. Simply neglecting the missing values in $\boldsymbol{x}_i$ will cause $\boldsymbol{\mu}_i$ and $\boldsymbol{\sigma}_i$ in Equation (7) to be on a different scale for respondents with different missing value patterns which will greatly bias the results. Furthermore, using list-wise deletion, pair-wise deletion, or mean-imputation are not recommended in general for loss of statistical efficiency. Liu et al. (2022) employ an intuitive but naïve trick to handle missing data in VAEs. However, as will be discussed below, this approach is suboptimal and potentially biases item parameter estimates. Thus, more work is needed to adapt VAE-based IRT models to missing data. Below, we derive four explicit approaches to handle missing data in IWVAE and provide a systematic comparison.

4 Handling missing data

4.1 Input drop-out

A naïve way of dealing with missing data is the so-called *input drop-out trick* (Nazabal et al., 2020; Liu et al., 2022). This trick consists of replacing the missing values in the input pattern with zeros, and only calculating the loss of the VAE over the non-missing datapoints. The new loss function for the VAE using the input drop-out trick is

$$ELBO_{ID} = \mathbb{E}_{q(\boldsymbol{\theta}_i|\tilde{\mathbf{x}}_i;\phi)}[\log p(x_{ij}, j \in O_i|\boldsymbol{\theta}_i;\Omega)] - KL[q(\boldsymbol{\theta}_i|\tilde{\mathbf{x}}_i;\phi)||p(\boldsymbol{\theta}_i)], \quad (10)$$

where $\tilde{\mathbf{x}}_i$ is the response pattern with missing values replaced by zero, and O_i is the set of indices of observed item responses. This is the method that was used in the context of MIRT by Liu et al. (2022).

A problem with the input-dropout trick is that although the reconstruction loss is only calculated over the observed data points, the inference model has no way to distinguish between missing values and responses that are truly zero, making it harder to estimate the latent ability variables. This may especially be suboptimal in MIRT applications, as zero responses are frequent in binary response data.

4.2 Conditional VAEs

To mitigate the problem above, Collier et al. (2020) introduced the conditional variational autoencoder (CVAE). In this approach, the loss function is equivalent to the input drop-out loss $ELBO_{ID}$ in equation (10), but the inference network now takes both the response pattern and the mask of missing data as input:

$$\boldsymbol{\theta}_i \sim N_d(\boldsymbol{\theta}; \boldsymbol{\mu}_i, \boldsymbol{\sigma}_i), \text{ with } (\boldsymbol{\mu}_i, \boldsymbol{\sigma}_i) = \text{FNN}_\phi(\mathbf{x}_i, \mathbf{m}_i). \tag{11}$$

Here $\mathbf{m}_i$ is a binary *mask* vector that indicates which values are missing in $\mathbf{x}_i$ and which are not—i.e., $m_{ij} = 1$ if $j \in O_i$ and $m_{ij} = 0$ if $j \notin O_j$. This additional input to the inference model allows it to differentiate between missing values and true zero responses.

This model has not been considered yet in the context of MIRT, and we consider this model as an alternative approach to handling missing data in this context.

4.3 Partial VAEs

A third approach to missing data consists of *partial variational autoencoders (PVAE)* (Ma et al., 2018). PVAEs were originally developed for the application of recommender systems, in which large fractions of missing data are common. They are based on neural networks for point net classification[1] that can deal with variable input sizes (Qi et al., 2017). Although FNNs generally require a fixed input size for each respondent, PVAEs bypass this requirement by estimating a so-called *embedding* vector for each item. An embedding vector in this context is a learned vector of parameters. By combining the embeddings from all items with an observed response using a permutation invariant aggregation operation, unobserved items can be simply ignored in the hope that the estimated embeddings will still encode essential information about $\boldsymbol{\mu}_i$ and $\boldsymbol{\sigma}_i$.

$$\boldsymbol{\theta}_i \sim N(\boldsymbol{\theta}; \boldsymbol{\mu}_i, \boldsymbol{\sigma}_i), \text{ with } (\boldsymbol{\mu}_, \boldsymbol{\sigma}_i) = \text{FNN}_\phi(g(\{\mathbf{e}_j : j \in O_i\})). \tag{12}$$

Here $\mathbf{e}_j$ denotes the embedding for item j, and g is a well-defined permutation invariant function that maps a set of points in embedding space to a single point in that same space—for example, the mean vector, or the vector of coordinate-wise maxima. This function g allows for a variable amount of embeddings and returns an output vector of a fixed length. The FNN takes the output from g and computes the variational parameters as in a regular VAE. Alternatively, g can be extended to return a vector of descriptive statistics of the embeddings, such as the mean, standard deviation, or distribution quantiles, to yield a more detailed representation of the distribution of the embedding vectors.

[1] This concerns the classification of objects based on unordered observed points in 3D space.

4.4 Imputation VAE

As a fourth approach, we propose a new missing data method, which we will refer to as the *imputation variational autoencoder (IMVAE)*. In this model, we use the same input drop-out loss function $ELBO_{ID}$ in equation (10), but rather than imputing the input with zeros, we impute the missing values with the expected values for the responses given the current IRT model parameter estimates. Specifically, we set

$$\tilde{\mathbf{x}}_i = \mathbf{x}_i \odot \mathbf{m}_i + p(\mathbf{x}_i|\hat{\boldsymbol{\theta}}_i; \hat{\Omega}) \odot (\mathbf{1} - \mathbf{m}_i), \tag{13}$$

where $\odot$ denotes the Hadamard products, $\mathbf{1}$ is a commensurate vector of ones, and $\hat{\boldsymbol{\theta}}_i$ again is the vector of latent variable estimates for subject i based on the current estimates $\hat{\phi}$ of the inference model parameters. In the first iteration, $\hat{\boldsymbol{\theta}}_i$ is set to the expected value of the prior. This adaptation still calculates the loss using only the observed data patterns but allows the inference model to make better estimates of the latent ability variables by using the current model parameters to estimate the most likely values of the missing data.

5 Simulation study

We simulated data from three and ten-dimensional MIRT models. Latent variable values were sampled from a standard normal $\boldsymbol{\theta} \sim N_d(0, \mathbf{I}_d)$. For simplicity, latent variables were uncorrelated. Item slope parameters were sampled from a uniform distribution $a_j \sim U(.5, 2), j = 1, \ldots, J$, and intercept parameters b_j were set to equally spaced values between -2 and 2. Several slope parameters were set to zero. For the three-dimensional model, we fix slope parameters to zero based on a factor configuration from the literature (da Silva et al., 2019; Curi et al., 2019). For the ten-dimensional model, we create our own configuration where each item depends on one or two latent factors. Both configurations are available on the GitHub page,[2] along with all of the code necessary to reproduce our experiments or to apply our models to new data.

Missing values were introduced at random by deleting a proportion of observed responses. The proportion of missing data was varied to ten equally spaced values between 0 and .75, inclusive. We ran 50 replications for each combination of dimensionality and missingness, resulting in $1,000$ unique datasets. We fit the input dropout variational autoencoder (IDVAE), CVAE, PVAE, and IMVAE to each dataset and estimate each model using 1, 5 and 25 importance weight samples, resulting in a 2 by 10 by 4 by 3 factorial design. We compare results to state-of-the-art MML using the R package *mirt* (Chalmers, 2012). We used the Metropolis-Hastings Robbins-Monro (MHRM) algorithm (see Cai (2010)) to estimate the three-dimensional models and we used quasi-Monte Carlo EM (QMCEM) estimation for the ten-dimensional models.

[2] `https://github.com/KarelVeldkamp/VAE-MIRT-Missing` TODO: anonymize

5.1 Implementation details

All our models are implemented in PyTorch (Paszke et al., 2019), and all code is publicly available on GitHub. Models were estimated using the AMSgrad algorithm using a learning rate of .005 and a batch size of 32. We stopped estimation when the loss did not decrease by more than 1e-7 over 10 iterations.

We used a single hidden layer of 20 nodes for the IDVAE, CVAE, and the IMVAE. In the PVAE, an embedding of length 12 was learned for each item. these embeddings were transformed to a vector of length 24 using a single layer FNN. We transform these item vectors to a fixed-length person vector using a permutation invariant operation. Specifically, we compute the mean, median, standard deviation, and the 25th and 75th quartiles of each person's item distribution, and concatenate the results, resulting in a vector of length 120 for each person. The resulting person vectors are used as input for a regular VAE encoder with a single hidden layer of 24 nodes. We have also experimented with using a single permutation invariant function, but using a concatenation has provided better results in our experience.

6 Results

Figure 1 presents the average mean square error of the parameter estimates for the different models and different numbers of importance-weighted samples as the proportion of missing data increases. Parameter estimates using a single importance weight were very poor and were left out of the figure to prevent obscuring the image. Overall we see that the accuracy of VAE-based parameter estimates decreases rapidly as the proportion of missing data increases. However, as the amount of importance samples increases the variational methods approach the accuracy of MML. This is in line with expectations, as increasing the number of samples tightens the bound to the marginal log-likelihood. As more data is missing a larger number of samples is needed to attain accurate parameter estimates. However, as becomes clear from Table 1, using VAE based methods with a large number of IW samples is still computationally much faster than MML in high dimensional applications. The complete results for the three- and ten-dimensional models are available in table 2 and 3 respectively.

Table 1: Average runtime of each model over 10 replications. IW is the number of importance weighted samples and d is the dimensionality of the model. All models were estimated on a single CPU core.

		IW=1		IW=5		IW=25	
miss	model	$d = 3$	$d = 10$	$d = 3$	$d = 10$	$d = 3$	$d = 10$
0.00	cvae	0.55	.46	.47	.41	.66	.90
	idvae	0.53	.42	.53	.52	.53	.96
	ivae	0.54	.42	.59	.65	.52	.86
	pvae	1.67	1.46	.98	3.88	4.26	3.84
	mirt	4.27	19.05	-	-	-	-

(continued)

		IW=1		IW=5		IW=25	
miss	model	$d = 3$	$d = 10$	$d = 3$	$d = 10$	$d = 3$	$d = 10$
0.25	cvae	0.45	.55	.50	.53	.61	.94
	idvae	0.51	.48	.47	.49	.55	1.00
	ivae	0.51	.49	.43	.50	.60	.95
	pvae	1.18	1.48	1.03	2.11	3.47	3.57
	mirt	4.40	20.34	-	-	-	-
0.75	cvae	0.39	.48	.45	.46	.73	.91
	idvae	0.40	.50	.54	.35	.54	1.09
	ivae	0.42	.48	.57	.42	.62	.89
	pvae	1.02	1.36	1.27	1.67	2.15	2.17
	mirt	4.49	11.54	-	-	-	-

The relative performance of variational methods is generally consistent across conditions. The CVAE parameter estimates have the lowest MSE. The CVAE and IMVAE are very close with respect to their item parameter estimates, but the extra information provided to the inference model appears to allow the CVAE to estimate ability parameters with more precision. Both the CVAE and IMVAE are consistently more accurate than the IDVAE, which is in line with expectations, as these two methods can be conceived of as improvements upon the IDVAE. The PVAE is clearly the least effective method to estimate MIRT parameters, performing substantially worse than all other methods across conditions. One surprising finding concerns the recovery of the intercept parameter in the ten-dimensional model using MML. The MSE of the intercept actually slightly decreases as the amount of missing data increases and sees a sudden jump at 75% missing values. Table 3 suggests that this initial decrease in MSE is due to a decrease in the bias of parameter estimates and that the large MSE at 75% reflects a large variance of the parameter estimates.

Table 2: Simulation study results for three-dimensional models. m denotes the proportion of missing data and p denotes the parameter that is being estimated. Note that MML is arbitrarily placed under $IW = 1$ to keep the table concise,but it does not make use of importance weighting. For the sake of brevity, the table only contains three levels of missing data. The complete table is available on GitHub.

			IW=1			IW=5			IW=25		
p	m	model	σ^2	$bias^2$	mse	σ^2	$bias^2$	mse	σ^2	$bias^2$	mse
	0.00	cvae	.0036	.0263	.0299	.0011	.0020	.0030	.0012	.0001	.0013
		idvae	.0212	.0241	.0453	.0011	.0013	.0024	.0012	.0001	.0013
		ivae	.0225	.0247	.0472	.0012	.0013	.0025	.0012	.0001	.0013
		pvae	.0968	.2769	.3738	.0011	.0016	.0027	.0012	.0003	.0016
		mirt	.0010	.0001	.0012	-	-	-	-	-	-
	0.25	cvae	.0237	.1840	.2077	.0015	.0041	.0057	.0017	.0003	.0021
		idvae	.0161	.1750	.1912	.0013	.0092	.0105	.0017	.0007	.0024
		ivae	.0269	.1182	.1452	.0014	.0048	.0062	.0017	.0004	.0021
		pvae	.1102	.6555	.7657	.0026	.0220	.0246	.0016	.0039	.0055
		mirt	.0017	.0001	.0018	-	-	-	-	-	-

(continued)

			IW=1			IW=5			IW=25		
p	m	model	σ^2	$bias^2$	mse	σ^2	$bias^2$	mse	σ^2	$bias^2$	mse
d	0.75	cvae	.0008	.9483	.9491	.0119	.1538	.1657	.0068	.0055	.0123
		idvae	.0137	.8907	.9044	.0105	.1598	.1702	.0074	.0079	.0153
		ivae	.0007	.9464	.9471	.0141	.1418	.1559	.0071	.0076	.0147
		pvae	.0007	.9472	.9479	.0079	.2357	.2436	.0067	.0229	.0296
		mirt	.0129	.0005	.0134	-	-	-	-	-	-
	0.00	cvae	.0018	.0038	.0055	.0015	.0005	.0020	.0014	.0003	.0017
		idvae	.0054	.0024	.0078	.0014	.0003	.0017	.0014	.0003	.0017
		ivae	.0054	.0027	.0081	.0014	.0002	.0016	.0015	.0003	.0018
		pvae	.0072	.0459	.0530	.0013	.0004	.0017	.0014	.0003	.0016
		mirt	.0010	.0005	.0015	-	-	-	-	-	-
	0.25	cvae	.0039	.0346	.0385	.0017	.0012	.0030	.0019	.0005	.0024
		idvae	.0042	.0388	.0429	.0017	.0040	.0057	.0019	.0007	.0026
		ivae	.0044	.0260	.0304	.0017	.0017	.0034	.0018	.0005	.0023
		pvae	.0113	.0958	.1072	.0020	.0131	.0151	.0016	.0023	.0039
		mirt	.0017	.0004	.0020	-	-	-	-	-	-
	0.75	cvae	.0024	.1318	.1342	.0050	.0353	.0403	.0054	.0035	.0088
		idvae	.0049	.1249	.1298	.0045	.0445	.0490	.0059	.0051	.0110
		ivae	.0024	.1328	.1352	.0044	.0391	.0436	.0048	.0054	.0102
		pvae	.0022	.1320	.1343	.0041	.0548	.0589	.0049	.0107	.0155
		mirt	.0079	.0008	.0087	-	-	-	-	-	-
theta	0.00	cvae	.1721	.1159	.2880	.1694	.1185	.2880	.1794	.1195	.2989
		idvae	.1888	.1277	.3165	.1894	.1057	.2950	.2042	.1132	.3173
		ivae	.1943	.1269	.3212	.1876	.1064	.2940	.2041	.1096	.3138
		pvae	.1431	.5085	.6516	.1944	.1114	.3058	.2127	.1472	.3599
		mirt	.1804	.0956	.2760	-	-	-	-	-	-
	0.25	cvae	.1416	.3923	.5339	.1922	.1711	.3633	.2044	.1797	.3841
		idvae	.1748	.3742	.5490	.2207	.2083	.4290	.2225	.2295	.4520
		ivae	.1851	.2938	.4789	.2168	.1683	.3850	.2309	.1754	.4062
		pvae	.0877	.8327	.9204	.2094	.2532	.4627	.1591	.4639	.6231
		mirt	.2019	.1388	.3408	-	-	-	-	-	-
	0.75	cvae	.0005	1.0078	1.0082	.1432	.5849	.7281	.1834	.5108	.6942
		idvae	.0139	.9789	.9927	.1249	.6723	.7973	.1625	.6183	.7808
		ivae	.0014	1.0072	1.0086	.1505	.5932	.7437	.1722	.5657	.7379
		pvae	.0004	1.0077	1.0081	.0969	.7331	.8300	.0756	.8000	.8756
		mirt	.2157	.4101	.6258	-	-	-	-	-	-

Table 3: Simulation study results for ten-dimensional model. m denotes the proportion of missing data and p denotes the parameter that is being estimated. Note that MML is arbitrarily placed under $IW = 1$ to keep the table concise, but it does not make use of importance weighting. For the sake of brevity, the table only contains three levels of missing data. The complete table is available on GitHub.

			IW=1			IW=5			IW=25		
p	m	model	σ^2	$bias^2$	mse	σ^2	$bias^2$	mse	σ^2	$bias^2$	mse
	0.00	cvae	.0218	.0241	.0459	.0009	.0074	.0082	.0004	.0020	.0024
		idvae	.0269	.0225	.0494	.0004	.0068	.0072	.0004	.0022	.0026
		ivae	.0293	.0247	.0540	.0004	.0068	.0073	.0004	.0022	.0026
		pvae	.0532	.1419	.1951	.0101	.0161	.0262	.0011	.0084	.0096
		mirt	.0003	.0016	.0019	-	-	-	-	-	-

(continued)

			IW=1			IW=5			IW=25		
p	m	model	σ^2	$bias^2$	mse	σ^2	$bias^2$	mse	σ^2	$bias^2$	mse
a	0.25	cvae	.0307	.0383	.0690	.0016	.0094	.0110	.0005	.0024	.0030
		idvae	.0380	.0481	.0861	.0020	.0130	.0150	.0005	.0046	.0051
		ivae	.0325	.0388	.0713	.0016	.0098	.0114	.0005	.0032	.0037
		pvae	.0002	.3088	.3090	.0070	.0452	.0522	.0014	.0173	.0187
		mirt	.0005	.0009	.0014	-	-	-	-	-	-
	0.75	cvae	.0003	.3094	.3097	.0054	.0231	.0285	.0021	.0055	.0076
		idvae	.0003	.3092	.3095	.0027	.0419	.0446	.0017	.0111	.0128
		ivae	.0003	.3094	.3097	.0036	.0279	.0315	.0018	.0087	.0105
		pvae	.0003	.3095	.3098	.0081	.1035	.1116	.0017	.0348	.0365
		mirt	.0085	.0002	.0088	-	-	-	-	-	-
d	0.00	cvae	.0017	.0140	.0157	.0014	.0048	.0062	.0014	.0017	.0031
		idvae	.0018	.0084	.0102	.0014	.0031	.0045	.0014	.0014	.0028
		ivae	.0022	.0091	.0114	.0014	.0031	.0045	.0013	.0015	.0028
		pvae	.0118	.0442	.0560	.0017	.0065	.0082	.0014	.0047	.0062
		mirt	.0011	.0092	.0103	-	-	-	-	-	-
	0.25	cvae	.0023	.0167	.0190	.0016	.0061	.0077	.0017	.0019	.0036
		idvae	.0033	.0285	.0318	.0016	.0105	.0121	.0017	.0039	.0056
		ivae	.0024	.0211	.0235	.0016	.0067	.0082	.0016	.0025	.0041
		pvae	.0009	.1230	.1239	.0018	.0188	.0207	.0018	.0115	.0133
		mirt	.0014	.0050	.0063	-	-	-	-	-	-
	0.75	cvae	.0023	.1230	.1252	.0042	.0149	.0191	.0048	.0048	.0096
		idvae	.0023	.1232	.1254	.0036	.0387	.0423	.0045	.0116	.0161
		ivae	.0022	.1226	.1248	.0036	.0268	.0305	.0042	.0093	.0134
		pvae	.0023	.1228	.1251	.0057	.0697	.0754	.0042	.0289	.0330
		mirt	.0498	.0013	.0511	-	-	-	-	-	-
theta	0.00	cvae	.1987	.2194	.4180	.1406	.1775	.3181	.1372	.1816	.3188
		idvae	.2238	.2264	.4502	.1622	.1705	.3327	.1695	.1772	.3467
		ivae	.2258	.2355	.4614	.1627	.1706	.3333	.1696	.1778	.3474
		pvae	.1743	.6222	.7965	.1631	.2714	.4345	.1450	.3265	.4716
		mirt	.1957	.1879	.3836	-	-	-	-	-	-
	0.25	cvae	.2270	.3044	.5314	.1666	.2192	.3858	.1634	.2242	.3876
		idvae	.2632	.3555	.6187	.1984	.2528	.4512	.1978	.2626	.4604
		ivae	.2412	.3154	.5566	.1870	.2229	.4099	.1937	.2297	.4234
		pvae	.0005	.9994	.9999	.1635	.4275	.5909	.1453	.4679	.6132
		mirt	.1984	.2080	.4063	-	-	-	-	-	-
	0.75	cvae	.0005	.9994	.9999	.1702	.4784	.6487	.1724	.4854	.6578
		idvae	.0026	.9976	1.0002	.1599	.5799	.7398	.1668	.5798	.7466
		ivae	.0032	.9992	1.0025	.1789	.4986	.6775	.1787	.5316	.7103
		pvae	.0005	.9994	1.0000	.0894	.7864	.8758	.0681	.8291	.8972
		mirt	.2043	.3840	.5883	-	-	-	-	-	-

7 Real data application

To demonstrate the practical viability of the different missing data approaches, we apply our models to a subset of the Bridge to Algebra (BTA) dataset, which was initially published in the context of the 2010 Association for Computing Machinery (ACM) Knowledge Discovery and Data-Mining (KDD) competition (Stamper & Pardos, 2016). The dataset is publicly available in the Pittsburgh Science of Learning Center Datashop (Koedinger et al., 2010) and was gathered from an online environment for learning mathematics called the cognitive tutor (Ritter et al., 2007). The complete BTA dataset consists of responses of $6,034$

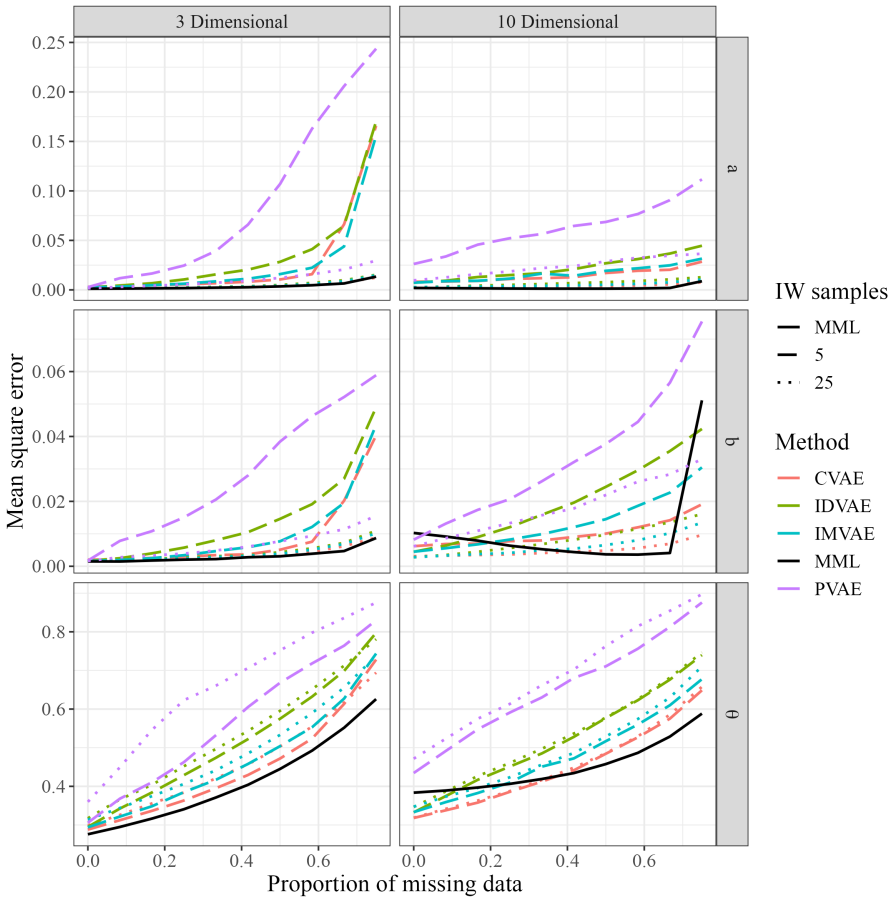

Figure 1: Mean square error of parameter estimates for increasing proportion of missing data. Note that results for models estimated using a single importance weight are omitted.

students to over 50,000 algebra problems. Importantly, each problem is divided into several steps, where each step is assumed to require one or more algebra skills. The full item set measures 60 different algebra skills. These skills consist of basic arithmetic operations, such as 'identifying number as a common factor' or 'calculating zero partial product'. In our study, we treat the individual steps as items. The skill requirements for each item make the data well suited for MIRT as the skills can be seen as latent dimensions and the dataset already describes which items require which skills, effectively describing a large Q-matrix.

To demonstrate the use of our approach, we selected 75 items which measure 9 skills in total. Table 4 contains the descriptions of the latent dimensions

as well as the number of items that load on each dimension. $1,289$ students completed this set of items. All students completed all 75 items, which allows us to introduce missing data artificially and compare results between the complete and missing data datasets. The preprocessed dataset is available on GitHub.

Table 4: Number of items per latent dimension.

Latent dim.	Number of items
List consecutive multiples of a number	16
Calculate zero partial product	16
Identify number as common factor	12
Calculate partial product – carry out	16
Identify number as common multiple	16
List factor of large number	8
Calculate partial product – carry in	13
Calculate partial product – carry in and out	12
Calculate partial product – no carry	23

As a golden standard, we fit a 9-dimensional M2PL model to the complete dataset using MML. We created the Q-Matrix based on the skill requirements in the BTA dataset. Most items load on a single latent dimension, but some load on up to 5 dimensions. Each latent dimension is measured by at least 8 items, and at most 23 items. The complete Q matrix is available on GitHub. To compare the approaches under missing data, we introduced 30% missing values at random. Parameters were estimated both using MML and using the CVAE. We used 25 IW samples. Other hyperparameters were kept equal to the simulation study.

Parameter estimation on the missing data dataset took 22 minutes using MML and just 16 seconds using the CVAE, highlighting the computational efficiency of variational methods. Figure 7 plots the parameter estimates on missing data against the parameter estimates on the entire dataset. Note that we only show the ability and discrimination parameters for a single dimension, which we think is representative of the other dimensions. The complete set of plots for all 9 dimensions is available on GitHub, and additionally, Table 5 reports the correlation coefficients between the missing data parameter estimates and the parameter estimates on the entire dataset for all dimensions. Overall, there is no systematic preference for MML over CVAE. The parameter estimates attained using the different methods are similar, and the correlations of the MML estimates with the golden standard are not systematically higher than the correlations of the CVAE estimates.

8 Discussion

We studied four different variational methods of estimating M2PL models in the presence of missing data. In a simulation study, we compared the different VAE methods to the performance of MML estimation. Results confirmed that variational methods are an efficient alternative to MML to estimate high

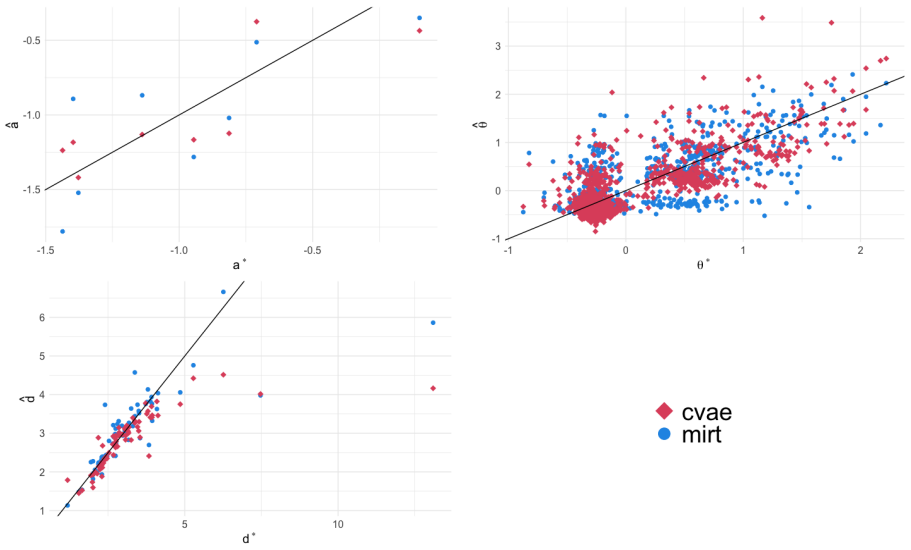

Figure 2: MIRT and CVAE parameters for 30% missing data, plotted against the mirt parameter estimates based on the complete dataset.

Table 5: Correlation of the parameter estimates with the golden standard for each dimension.

Latent dim.	1	2	3	4	5	6	7	8	9
θ_{cvae}	.65	.84	.80	.69	.86	.81	.22	.74	.83
θ_{mml}	.65	.76	.81	.74	.77	.75	.25	.67	.83
a_{cvae}	.53	.94	.84	.80	.97	.97	.29	.90	.72
a_{mml}	.64	.90	.99	.92	.93	.96	.43	.86	.87

dimensional MIRT models given that enough IW samples are used to attain performance up to par with MML. In situations where little to no data is missing, taking a small number of importance samples in the VAE-based models is already sufficient for a comparable performance to MML. However, when large parts of the data are not available, which might occur in adaptive testing settings or test equating environments for example, the ELBO appears to underestimate the marginal log-likelihood, and more IW samples are needed to accurately recover the model parameters. In terms of the different variational methods used, it is clear that the CVAE and IMVAE should be preferred to the straight-forward input dropout method, which has previously been used in the context of VAE based MIRT (Liu et al., 2022). Imputing missing values with 0 leads to suboptimal ability estimates, which in turn affects the estimation of the item parameters. This is most relevant if a substantial portion of data is missing. In terms of parameter recovery accuracy, the CVAE is the best model, as it provides the inference model with extra information regarding missing values,

enabling the model to deal with missing values correctly. The IMVAE might be used as a simpler alternative, sacrificing some estimation accuracy for a less complex model. Finally, we have shown that the CVAE can provide accurate parameter estimates on real high dimensional datasets over 80 times faster than traditional MML.

One important limitation of our study is that we did not consider correlated latent abilities. For simplicity, we assumed uncorrelated latent factors in both our simulation study and the real data application. In practice, this is unrealistic, as latent dimensions are generally correlated. For example in the BTA dataset, we might expect that students who are better at listing consecutive multiples of a number are also better at listing factors of large numbers. Converse et al. (2021) have previously adapted the VAE approach to situations with correlated latent factors by including the latent covariance matrix in the inference model. Although there is no obvious reason why our missing data approach would not generalize to these models, further research is needed to verify if the correlated factor models and our missing data models can be combined successfully.

A second limitation concerns the missing data model. In the current simulations, as well as the real data application, we assumed that all data is missing completely at random (MCAR). This simplifying assumption allowed us to make a general comparison between different approaches to handling missing data. Our model approaches full information marginal maximum likelihood as the number of samples increases, so we expect the present approach to be viable in the case of missing at random as well. However, in future work, it is important to verify whether our proposed methods generalize to situations where data is missing at random (MAR).

References

Amari, S.-i. (1993). Backpropagation and stochastic gradient descent method. *Neurocomputing*, *5*(4-5), 185–196.

Blei, D. M., Kucukelbir, A., & McAuliffe, J. D. (2017). Variational inference: A review for statisticians. *Journal of the American statistical Association*, *112*(518), 859–877.

Burda, Y., Grosse, R., & Salakhutdinov, R. (2015). Importance weighted autoencoders. *arXiv preprint arXiv:1509.00519*.

Cai, L. (2010). High-dimensional exploratory item factor analysis by a metropolis–hastings robbins–monro algorithm. *Psychometrika*, *75*, 33–57.

Chalmers, R. P. (2012). mirt: A multidimensional item response theory package for the r environment. *Journal of statistical Software*, *48*, 1–29.

Collier, M., Nazabal, A., & Williams, C. K. (2020). Vaes in the presence of missing data. *arXiv preprint arXiv:2006.05301*.

Converse, G., Curi, M., Oliveira, S., & Templin, J. (2021). Estimation of multidimensional item response theory models with correlated latent variables using variational autoencoders. *Machine learning*, *110*(6), 1463–1480.

Curi, M., Converse, G. A., Hajewski, J., & Oliveira, S. (2019). Interpretable variational autoencoders for cognitive models. In *2019 international joint conference on neural networks (ijcnn)* (pp. 1–8).

da Silva, M. A., Liu, R., Huggins-Manley, A. C., & Bazán, J. L. (2019). Incorporating the q-matrix into multidimensional item response theory models. *Educational and Psychological Measurement*, *79*(4), 665–687.

Edwards, M. C. (2010). A markov chain monte carlo approach to confirmatory item factor analysis. *Psychometrika*, *75*(3), 474–497.

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural networks*, *2*(5), 359–366.

Kingma, D. P., & Welling, M. (2013). Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Koedinger, K. R., Baker, R. S., Cunningham, K., Skogsholm, A., Leber, B., & Stamper, J. (2010). A data repository for the edm community: The pslc datashop. *Handbook of educational data mining*, *43*, 43–56.

Liu, T., Wang, C., & Xu, G. (2022). Estimating three-and four-parameter mirt models with importance-weighted sampling enhanced variational autoencoder. *Frontiers in Psychology*, *13*, 4189.

Ma, C., Gong, W., Hernández-Lobato, J. M., Koenigstein, N., Nowozin, S., & Zhang, C. (2018). Partial vae for hybrid recommender system. In *Nips workshop on bayesian deep learning* (Vol. 2018).

McKinley, R. L., & Reckase, M. D. (1983). *An extension of the two-parameter logistic model to the multidimensional latent space.* (Tech. Rep.). American Coll Testing Program Iowa City Ia Resident Programs Dept.

Nazabal, A., Olmos, P. M., Ghahramani, Z., & Valera, I. (2020). Handling incomplete heterogeneous data using vaes. *Pattern Recognition*, *107*, 107501.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., ... Chintala, S. (2019). Pytorch: An imperative style, high-performance deep learning library. In *Advances in neural information processing systems 32* (pp. 8024–8035). Curran Associates, Inc. Retrieved from http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-lear

Qi, C. R., Su, H., Mo, K., & Guibas, L. J. (2017). Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 652–660).

Reddi, S. J., Kale, S., & Kumar, S. (2019). On the convergence of adam and beyond. *arXiv preprint arXiv:1904.09237*.

Rezende, D., & Mohamed, S. (2015). Variational inference with normalizing flows. In *International conference on machine learning* (pp. 1530–1538).

Ritter, S., Anderson, J. R., Koedinger, K. R., & Corbett, A. (2007). Cognitive tutor: Applied research in mathematics education. *Psychonomic bulletin & review*, *14*, 249–255.

Stamper, J., & Pardos, Z. A. (2016). The 2010 kdd cup competition dataset: Engaging the machine learning community in predictive learning analytics. *Journal of Learning Analytics*, *3*(2), 312–316.

Takane, Y., & De Leeuw, J. (1987). On the relationship between item response theory and factor analysis of discretized variables. *Psychometrika*, *52*(3), 393–408.

Urban, C. J., & Bauer, D. J. (2021). A deep learning algorithm for high-dimensional exploratory item factor analysis. *psychometrika*, *86*(1), 1–29.

Wirth, R., & Edwards, M. C. (2007). Item factor analysis: current approaches and future directions. *Psychological methods*, *12*(1), 58.