# VarFA: A Variational Factor Analysis Framework For Efficient Bayesian Learning Analytics

Zichao Wang[1], Yi Gu[2], Andrew S. Lan[3], Richard G. Baraniuk[1,4]
[1]Rice University, [2]Northwestern University, [3]University of Massachusetts Amherst, [4]OpenStax
jzwang@rice.edu, Yi.Gu@u.northwestern.edu, andrewlan@cs.umass.edu, richb@rice.edu



## ABSTRACT

We propose VarFA, a variational inference factor analysis framework that extends existing factor analysis models for educational data mining to efficiently output uncertainty estimation in the model's estimated factors. Such uncertainty information is useful, for example, for an adaptive testing scenario, where additional tests can be administered if the model is not quite certain about a students' skill level estimation. Traditional Bayesian inference methods that produce such uncertainty information are computationally expensive and do not scale to large data sets. VarFA utilizes variational inference which makes it possible to efficiently perform Bayesian inference even on very large data sets. We use the sparse factor analysis model as a case study and demonstrate the efficacy of VarFA on both synthetic and real data sets. VarFA is also very general and can be applied to a wide array of factor analysis models. Code and instructions to reproduce results in this paper are available from https://tinyurl.com/tvm4332.

## Keywords

factor analysis, variational inference, learning analytics, educational data mining

## 1. INTRODUCTION

A core task for many practical educational systems is *student modeling*, i.e., estimating students' mastery level on a set of skills or knowledge components (KC) [67, 11]. Such estimates allow in-depth understanding of students' learning status and form the foundation for automatic, intelligent learning interventions. For example, intelligent tutoring systems (ITSs) [57, 21, 3, 47, 25, 52] rely on knowing the students' skill levels in order to effectively recommend individualized learning curriculum and improve students' learning outcomes. In the big data era, student modeling is usually formulated as an educational data mining (EDM) problem [54, 4, 41] where an underlying machine learning (ML) model estimates students' skill mastery levels from students' learning records, i.e., their answers to assessment questions.

Many student modeling methods have been proposed in prior literature. A fruitful line of research for student modeling follows the *factor analysis (FA)* approach. FA models usually assume that an unknown, potentially multi-dimensional student parameter, in which each dimension is associated with a certain skill, explains how a student answers questions and is to be estimated. Popular and successful FA models include item response theory (IRT) [66], multi-dimensional IRT [2], learning factor analysis (LFA) [6], performance factor analysis (PFA) [50], DASH (short for difficulty, ability, and student history) [35, 45], DAS3H (short for difficulty, ability, skill and student skill history) [10], knowledge tracing machines (KTM) [68], and so on. Recently, more complex student models based on deep neural networks (DNN) have also been proposed [51, 72, 43]. Nevertheless, thanks to their simplicity, effectiveness and robustness, FA models remain widely adopted and investigated for practical EDM tasks. Moreover, there is evidence that simple FA models could even outperform DNN models for student modeling in terms of predicting students' answers [69]. Because of FA models' competitive performance and its elegant mathematical form, we focus on FA-based student models in this paper.

Most of the aforementioned FA models compute a single point estimate of skill levels for each student. Often, however, it is not enough to obtain mere point estimates of students' skill levels; knowing the model's uncertainty in its estimation is crucial because it potentially helps improve the model's performance and improve both students' and instructors' experience with educational systems. For example, in ITS, an underlying model can use the uncertainty information in its estimation of students' skill level to automatically decide that its recommendations based on highly uncertain estimations are unreliable and instead notify a human instructor to evaluate the students' performance. This enables collaboration between ITS and instructors to create a more effective learning environment. In adaptive testing systems [7, 71], knowing the uncertainty in model's estimation could help the model intelligently pick the next test items to most effectively reduce its uncertainty about estimated students' skill levels. This will help to potentially reduce the number of items needed to have a confident, accurate estimation of the students' skill mastery level, saving time for both students to take the test and instructors to have a good assessment of the student's skills.

All the above applications require the model to "know what it does not know," i.e., to quantify the uncertainty of its estimation. Achieving this does not necessary changes the model. rather, we need a different inference algorithm for inferring not only a point estimate of student's skill level from observed data (i.e., students' answer records to questions) but also uncertainty in the estimations.

Fortunately, there exist methods that both compute point

estimates of students' skill levels and quantifies the uncertainty of those estimates. These methods usually follow the Bayesian inference paradigm, where each student's unknown skill levels are treated as random variables drawn from a *posterior distribution*. Thus, one can use the *credible interval* to quantify the model's uncertainty on the student's estimated skill level. A classical method for Bayesian inference is Monte Carlo Markov Chain (MCMC) sampling [17] which has been widely used in many other disciplines [19] other than EDM. In the context of student modeling with FA models, existing works such as [33, 15, 48, 42] have applied MCMC methods to obtain credible intervals.

Unfortunately, classic Bayesian inference methods suffer from extensive computational complexity. For example, each computation step in MCMC involves a time-consuming evaluation of the posterior distribution. Making matters worse, MCMC typically takes many more steps to converge than non-Bayesian inference methods. As a concrete illustration, in [33], the Bayesian inference method (10 minutes) is about 100 times slower than the other non-Bayesian inference method based on stochastic gradient descent (6 seconds). The high computational cost prevents Bayesian inference methods from mass-deployment in large-scale, real-time educational systems where timely feedback is critical for learning [14] and tens of thousands of data points need to be processed in seconds or less instead of minutes or hours. It is thus highly desirable to accelerate Bayesian inference so that one can quantify model's uncertainty in its estimation as efficiently as non-Bayesian inference methods.

*Contributions.* In this work, we propose VarFA, a novel framework based on *variational inference* (VI) to perform efficient, scalable Bayesian inference for FA models. The key idea is to approximate the true posterior distribution, whose costly computation slows down Bayesian inference, with a *variational distribution*. We will see in Section 3 that, with this approximation, we turn Bayesian inference into an optimization problem where we can use the same efficient inference algorithms as in non-Bayesian inference methods. Moreover, the variational distribution is very flexible and we have full control specifying it, allowing us to freely use the latest development in machine learning, e.g., deep neural networks (DNNs), to design the variational distribution that closely approximates the true posterior. Thus, we also regard our work as a first step in applying DNNs to FA models for student modeling, achieving efficient Bayesian inference (enabled by DNNs) without losing interpretability (brough by FA models). We demonstrate the efficacy of our framework on both synthetic and real data sets, showcasing that VarFA substantially accelerates classic Bayesian inference for FA models with no compromise on performance.

The remainder of this paper is organized as follows. Section 2 introduces the problem setup in FA and reviews the important FA models and their inference methods, in particular Bayesian inference. Section 3 explains our VarFA framework in detail. Section 4 presents extensive experimental results that substantiate the claimed advantages of our framework. Section 5 concludes this paper and discusses possible extensions to VarFA.

# 2. BACKGROUND AND RELATED WORK

We first set up the problem and review related work. Assume we have a data set $\boldsymbol{Y} \in \mathbb{R}^{N\times Q}$ organized in matrix format where $N$ is the total number of students and $Q$ is the number of questions. This is a binary students' answer record matrix where each entry $y_{ij}$ represents whether student $i$ correctly answered question $j$. Usually, not all students answer all questions. Thus, $\boldsymbol{Y}$ contains missing values. We use $\{i,j\} \in \Omega_{\text{obs}}$ to denote entries in $\boldsymbol{Y}$, i.e., the $i$-th student's answer record to the $j$-th question, that are observed.

We are interested in models capable of inferring each $i$-th student's skill mastery level that can accurately predict the student's answers given the above data. These models are often evaluated on the prediction accuracy and whether the inferred student skill mastery levels are easily interpretable and educationally meaningful. We now review factor analysis models (FA), one of the most widely adopted and successful methodologies for the student modeling task.

## 2.1 Factor Analysis For Student Modeling

One of the earliest FA model for student modeling is based on the item response theory (IRT) [66]. It usually has the following form

$$\mathbb{P}(y_{ij}=1) = \sigma(c_i + \mu_j)\,, \tag{1}$$

which assumes that each student's answer $y_{ij}$ is independently Bernoulli distributed. The above formula says that the students' answers can be explained by an unknown scalar student skill level factor $c_i$ for each student $i$ and an unknown scalar question difficulty level factor $\mu_j$ for each question $j$. $\sigma(\cdot)$ is a Sigmoid activation function, i.e., $\sigma(x) = 1/(1+\exp(-x))$. The multi-dimensional IRT model (MIRT) [2] extends IRT by using a multi-dimensional vector to represent the student skill levels. More recently, [33] proposed sparse factor analysis model (SPARFA) which extends MIRT by imposing additional assumptions, resulting in improved interpretations of the inferred factors.

Other FA models seek to improve student modeling performance by cleverly incorporate additional auxiliary information. For example, the additive factor model (AFM) [6] incorporates students' accumulative correct answers for a question and the skill tags associated with each question:

$$\mathbb{P}(y_{ij}=1) = \sigma\left(c_i + \sum_{k=1}^{\kappa}\left(q_{kj}\beta_k + q_{kj}\rho_k b_{ik}\right)\right)\,, \tag{2}$$

where $\kappa$ is the total number of skills in the data, $b_{ik}$ is the $i$-th student's total number of correct answers for skill $k$ and $q_{kj} \in \{0,1\}$ indicates whether the skill $k$ is associated with question $j$. In particular, $q_{kj}$'s form matrix $\boldsymbol{Q} \in \mathbb{R}^{K\times Q}$ which is commonly known as the Q-matrix in literature [63]. The unknown, to-be-inferred factors are $\beta_k$, a scalar difficulty factor for each skill $k$, and $\rho_k$, a scalar learning rate of skill $k$. The performance factor model (PFM) [50] builds upon AFM that additionally incorporate the total number of a student's incorrect answers to questions associated with a skill. The instructor factor model (IFM) [9] further builds on PFM to incorporate the prior knowledge of whether a student has already mastered a skill. More recently, [68] introduces knowledge tracing machines (KTM) that could

flexibly incorporate a number of auxiliary information mentioned above, thus generalizing AFM, PFM and IFM.

Another way to improve student modeling performance is to utilize the so-called *memory*, i.e., using students' historic action data over time. For example, [35, 45] proposed DASH (short for difficulty, ability, and student history) that incorporates a student's total number of correct answers and total number of attempts for a question in a given time window

$$\mathbb{P}(y_{ij}=1)=\sigma\Bigg(c_i-\mu_j+\sum_{w=0}^{W-1}\Big(\theta_{2w+1}\log(1+\rho_{ijw}) - \theta_{2w+2}\log(1+\gamma_{ijw})\Big)\Bigg) \tag{3}$$

where $W$ is the length of the time windows (e.g., number of days), $\rho_{ijw}$ and $\gamma_{ijw}$ are the $i$-th student's total number of correct answers and total number of attempts for question $j$ at time $w$, respectively. $\theta$'s are parameters that captures the effect of correct answers and total attempts. More recently, DAS3H [10] extends DASH to further include skill tags associated with each question which essentially amounts to adding, within the last summation term in Eq. 3, another summation over the number of skills.

### 2.1.1 A general formulation for FA models

Although the above FA models differ in their formulae, modeling assumptions and the available auxiliary data used, we argue that the aforementioned FA models can be unified into a canonical formulation below

$$\mathbb{P}(y_{ij}=1)=\sigma(\mathbf{c}_i^\top\mathbf{m}_j+\mu_j)\,, \tag{4}$$

where $\mathbf{c}_i\in\mathbb{R}^K$, $\mathbf{m}_j\in\mathbb{R}^K$ and $\mu_j\in\mathbb{R}$ are factors whose dimension, interpretations and subscript indices depend on the specific instantiations of the FA model. To illustrate that Eq. 4 subsumes the FA models mentioned above, we demonstrate, as examples, how to turn SPARFA and AFM into the form in Eq. 4 and how to reinterpret the parameters in the reformulation. The equivalence between Eq. 4 and the original SPARFA formula is immediate and we can interpret the parameters in Eq. 4 as follows to recover SPARFA: $K$ is the number of *latent skills* that group the actual skill tags in the data set into meaningful coarse clusters; $\mathbf{c}_i$ represents the $i$-th student's skill level on the latent skills; $\mathbf{m}_j$ represents the strength of association of the $j$-th question with the latent skills which is assumed to be nonnegative and sparse for improved interpretation; and $\mu_j$ represents the difficulty of question $j$. All three factors in SPARFA are assumed to be unknown.

Similarly, for AFM, we can perform the following change of variables and indices to obtain Eq. 4. We first change the indices $\mathbf{m}_j$ to $\mathbf{m}_{ij}$ and $\mu_j$ to $\mu_i$, and remove the index in $\mathbf{c}_i$ to $\mathbf{c}$. Then, we simply set $\mathbf{c}=[\beta_1,...,\beta_\kappa,\rho_1,...,\rho_\kappa]^\top$, $\mathbf{m}_{ij}=[q_{1j},...,q_{\kappa j},q_{1j}b_{ik},...,q_{\kappa j}b_{i\kappa}]^\top$ and $\mu_i=\alpha_i$ to obtain Eq. 4. We can interpret the parameters in the reformulation as follows to recover AFM: $\kappa=K/2$ is the total number of skill tags in the data; $\mathbf{c}$ represents the unknown skill information including skill difficulty and learning rate; $\mathbf{m}_{ij}$ summarizes known auxiliary information for the $i$-th student and $j$-th question; $\mu_i$ represents the unknown $i$-th student's skill mastery level. Tthe other FA models can be cast into Eq. 4 in a similar fashion. Note that, if the observed data $\boldsymbol{Y}$ is not binary but rather continuous or categorical, we can simply use a Gaussian or categorical distribution for the observed data and change the Sigmoid activation $\sigma(\cdot)$ to some other activation functions accordingly. In this way, we still retain the general FA model in Eq. 4. We will use this canonical form to facilitate discussions in the rest of this paper.

## 2.2 Inference Methods for FA Models

We first introduce the inference objective and briefly review maximum log likelihood estimation method. Then, we review existing works that uses Bayesian inference for FA and highlight their high computational complexity, paving the way to VarFA, our proposed efficient Bayesian inference framework based on variational inference.

The factors in Eq. 4 may contain known factors that need no inference. Therefore, for convenience of notation, let $[\nu,\theta,\psi]$ be a partition of the factors $\mathbf{c}_i$, $\mathbf{m}_j$ and $\mu_j$ where $\nu$ contains the unknown students' skill level factors to be estimated, $\theta$ contains the remaining unknown factors and $\psi$ contains the known factors. For example, for AFM, $\nu=\{\mu_1,...,\mu_N\}$, $\theta=\mathbf{c}$ and $\psi=\{\mathbf{m}_{11},...,\mathbf{m}_{NQ}\}$. For SPARFA, $\nu=\{\mathbf{c}_1,...,\mathbf{c}_N\}$, $\theta=\{\mathbf{m}_1,...,\mathbf{m}_Q,\mu_1,...,\mu_Q\}$ and $\psi=\emptyset$. Further, let the subscripts for these partitions indicate the corresponding latent factors in FA; i.e., for SPARFA, $\theta_i=[\mathbf{m}_i,\mu_i]$ and $\nu_i=\mathbf{c}_i$. Since $\psi$ summarizes known factors, we will omit it from the mathematical expositions in the remainder of the paper.

The inference objective in FA is then to obtain a good estimate of the unknown factors, represented by $\theta$ and $\nu$, with respect to some loss function $\mathcal{L}$, usually the marginal data log likelihood. There are two ways to infer the unknown factors.

### 2.2.1 Maximum Likelihood Estimation

The first way is through the maximum likelihood estimate (MLE) which obtains a point estimate of the unknown factors

$$\widehat{\theta},\widehat{\nu}=\underset{\theta,\nu}{\operatorname{argmin}}\left(-\sum_{i,j\in\Omega_{\text{obs}}}\log p(y_{ij};\theta,\nu)\right)+\lambda\mathcal{R}(\theta,\nu)\,. \tag{5}$$

Recall that $\Omega_{\text{obs}}$ indicates which student $i$ has provided an answer to question $j$. $\mathcal{R}(\theta,\nu)$ is a regularization term, e.g., $\ell 2$ regularization on the factors and $\lambda$ is a hyper-parameter that controls the strength of the regularization. Most of the works in FA use this method of inference [6, 50, 9, 68, 10, 35, 45]. The advantage of MLE is that the above optimization objective allows the use of fast inference algorithms, in particular stochastic gradient descent (SGD) and its many variants, making the inference highly efficient.

### 2.2.2 Maximum A Posteriori Estimation

The second way is through maximum a posteriori (MAP) estimation through Bayesian inference. This is the focus of our paper. Recall that we wish to obtain not only a point estimate but also credible interval, which MAP allows while MLE does not.

Bayesian inference methods treat the unknown parameters $\nu$ and $\theta$ as random variables drawn from some distribution.

We are thus interested in inferring the *posterior* distribution of $\nu$ and $\theta$. Using the Bayes rule, we have that

$$p(\nu|\boldsymbol{Y},\theta) = \frac{p(\boldsymbol{Y},\nu,\theta)}{p(\boldsymbol{Y})} \tag{6}$$

$$= \frac{p(\boldsymbol{Y}|\nu,\theta)\,p(\nu)\,p(\theta)}{\iint p(\boldsymbol{Y}|\nu,\theta)\,p(\nu)\,p(\theta)d\nu d\theta}. \tag{7}$$

We can similarly derive the posterior for the other unknown factor $\theta$, although in this paper we focus on Bayesian inference for the unknown student skill level parameter $\nu$. In Eq. 6 to Eq. 7 we have applied standard Bayes rule. Note that, because $\psi$ contains known factors and thus does not enter the Bayes rule equation, we have omitted it from the above equations. Once we estimate the posterior distribution for $\nu$ from data, we can obtain both point estimates and credible intervals by respectively taking the mean and the standard deviation of the posterior distributions. A number of prior works have proposed to use Bayesian inference for factor analysis, mostly developed for the IRT model [16, 44]. More recently, Bayesian methods are developed for more complex models. For example, [33] proposed SPARFA-B, a specialized MCMC algorithm that accounts for SPARFA's sparsity and nonnegativity assumptions.

However, Bayesian inference suffers from high computational cost despite its desired capability to quantify uncertainty. The challenge stems from the difficulty to evaluate the denominator in the posterior distribution in Eq. 6 and 7 which involves an integration over a potentially multi-dimensional variable. This term usually cannot be computed in close form, thus we usually cannot get an analytical formula for the posterior distribution. Monte Carlo Markov Chain (MCMC) method gets around this difficulty by using analytically tractable proposal distributions to approximate the true posterior and sequentially updating the proposal distribution in a manner reminiscent of gradient descent. MCMC methods also enjoy the theoretical advantage that, when left running for long enough, the proposal distribution is guaranteed to converge to the true posterior distribution [17]. However, in practice, it might take very long time for MCMC to converge to the point that running MCMC is no longer practical when data set is very large. The high computational complexity of MCMC is apparently impractical for educational applications where timely feedback is of critical importance [14].

# 3. VarFA: A VARIATIONAL INFERENCE FACTOR ANALYSIS FRAMEWORK

We are ready to introduce VarFA, our variational inference (VI) factor analysis framework for efficient Bayesian learning analytics. The core idea follows the variational principle, i.e., we use a parametric variational distribution to approximate the true posterior distribution. VarFA is highly flexible and efficient, making it suitable for large scale Bayesian inference for FA models in the context of educational data mining.

In this current work, we focus on obtaining credible interval for the student skill mastery factor $\nu$ as a first step of VarFA because, recall from Section 1, this factor is often of more practical interest than the other unknown factors. Therefore, currently VarFA is a hybrid MAP and MLE inference method: we perform VI on the unknown student factor $\nu$ and MLE on all other unknown factors $\theta$. We consider this hybrid nature of VarFA in its current formulation as a novel feature because classic MLE or MAP estimation are not capable of performing Bayesian inference on a subset of the unknown factors. Extension to VarFA to full Bayesian inference for all unknown factors is part of an ongoing research; see 5 for more discussions.

## 3.1 VarFA Details

Now, we explain in detail how to apply variational inference for FA models for efficient Bayesian inference. Because the posterior distribution is intractable to compute (recall Eq. 6 and 7 and the related discussion), we approximate the true posterior distribution for $\nu$ with a parametric variational distribution

$$p(\nu|\boldsymbol{Y},\theta) \approx q_\phi(\nu|\boldsymbol{Y}) = \prod_{i=1}^{N} q_\phi(\nu_i|\boldsymbol{y}_i)\,, \tag{8}$$

where $\phi$ is a collection of learnable parameters that parametrize the variational distribution and $\boldsymbol{y}_i$ is all the answer records by student $i$. Notably, we have removed the dependency of the variational distribution on $\psi$ and $\theta$ so that the variational distribution is solely controlled by the variational parameter $\phi$. Thus, the design of the variational distribution is highly flexible. All we need to do is to specify a class of distributions and design a function parametrized by $\phi$ to output the parameters of $q_\phi$. Common in prior literature is to use a Gaussian with diagonal covariance for $q_\phi$:

$$q_\phi(\nu|\boldsymbol{y}_i) = \mathcal{N}(\boldsymbol{u}_j, \mathrm{diag}(\boldsymbol{v}_j))\,, \tag{9}$$

where its mean and variance $[\boldsymbol{u}_j^\top, \boldsymbol{v}_j^\top]^\top = f_\phi(\boldsymbol{y}_i)$. We can use arbitrarily complex functions such as a deep neural network for $f_\phi$ as long as they are differentiable; more details in Section 3.2. With the above approximation, Bayesian inference turns into an optimization problem under the variational principle, where we now optimize a lower bound, known as the evidence lower bound (ELBO) [5], of the marginal data log likelihood. We derive a novel ELBO objective for variational inference applied to FA models in the EDM setting, summarized in the proposition below.

**Proposition 1.** *In VarFA, the ELBO objective for an FA model in the form of Eq. 4 is*

$$\mathcal{L}_{\mathrm{ELBO}}(\phi,\theta) = \sum_{i,j\in\Omega_{\mathrm{obs}}} \Big( - D_{\mathrm{KL}}[q_\phi(\nu_i|\boldsymbol{y}_i)\|p(\nu_i)] + \mathbb{E}_{\nu_i\sim q_\phi(\nu_i|\boldsymbol{y}_i)}[\log p_{\theta_j}(y_{ij}|\nu_i)]\Big) \tag{10}$$

*where $D_{\mathrm{KL}}$ is the KullbackâĂŞLeibler (KL) divergence [39] between two distributions.*

*Proof.* We start with the marginal data log likelihood which is the objective we want to maximize and introduce the ran-

dom variables $\nu_i$'s:

$$\begin{aligned}
\log p(\boldsymbol{Y}) &= \sum_{i,j\in\Omega_{\text{obs}}} \log p_{\theta_j}(y_{ij}) \\
&= \sum_{i,j\in\Omega_{\text{obs}}} \log \int p_{\theta_j}(y_{ij},\nu_j)\, d\nu_i \\
&= \sum_{i,j\in\Omega_{\text{obs}}} \log \int q_\phi(\nu_i|\boldsymbol{y}_i) \frac{p_{\theta_j}(y_{ij},\nu_j)}{q_\phi(\nu_i|\boldsymbol{y}_i)}\, d\nu_i \\
&\overset{(i)}{\geq} \sum_{i,j\in\Omega_{\text{obs}}} \mathbb{E}_{\nu_i\sim q_\phi(\nu_i|\boldsymbol{y}_i)} \left[\log \frac{p_{\theta_j}(y_{ij},\nu_j)}{q_\phi(\nu_i|\boldsymbol{y}_i)}\right] \\
&\geq \sum_{i,j\in\Omega_{\text{obs}}} -\mathbb{E}_{\nu_i}\left[\log \frac{q_\phi(\nu_i|\boldsymbol{y}_i)}{p(\nu_j)}\right] + \mathbb{E}_{\nu_i}\left[\log p_{\theta_j}(y_{ij}|\nu_j)\right] \\
&= \mathcal{L}_{\text{ELBO}}(\phi,\theta)
\end{aligned}$$

where step (i) follows from Jensen's inequality [28]. □ □

The ELBO objective differs from those used in existing literature in that, in our case, because the data matrix is only partially observed, the summation is only over $\{i,j\} \in \Omega_{\text{obs}}$, i.e., the observed entries in the data matrix. In contrast, in prior literature, the summation in the ELBO objective is over all all entries in the data matrix.

We form the following optimization objective to estimate $\phi$ and $\theta$:

$$\widehat{\theta},\widehat{\phi} = \underset{\theta,\phi}{\text{argmin}} \; -\mathcal{L}_{\text{ELBO}}(\phi,\theta) + \lambda\mathcal{R}(\theta)\,, \tag{11}$$

where $\mathcal{R}(\theta)$ is a regularization term. That is, we perform VI on the student factor $\nu$ and MLE inference on the remaining factors $\theta$. More explicitly, to perform VI on $\nu$, we simply compute the mean and standard deviation of $q_\phi$ using $f_\phi$ with the learnt variational parameters $\phi$.

## 3.2 Why is VarFA efficient?

*VarFA supports efficient stochastic optimization.* By formulating Bayesian inference as an optimization problem via the ELBO objective, VarFA allows efficient optimization algorithms such as SGD, with a few additional tricks. In particular, we require all terms in $\mathcal{L}(\theta,\phi)$ to be differentiable with respect to $\theta$ and $\phi$ which enable gradient computation necessary for SGD. This requirement is easily satisfied with a few moderate assumptions. Specifically, we let the variational distribution $q_\phi$ and the prior distribution $p(\nu_i)$ for each $i$ to be Gaussian (See Eq. 9; we use a standard Gaussian for the prior distribution $p(\nu_i)$, i.e., $p(\nu_i) = \mathcal{N}(0,\boldsymbol{I})$) following existing literature [31, 53]. These two assumptions are not particularly limiting especially given the vast number of successful applications of VI that rely on the same assumptions [75, 40, 58, 12, 8, 26, 29]. Thanks to the Gaussian assumption, for the first term in $\mathcal{L}(\theta,\phi)$, we can easily compute the KL divergence term analytically in closed form. For the second term in $\mathcal{L}(\theta,\phi)$, we can use the so-called reparametrization trick, i.e., $\nu_i = \boldsymbol{u}_i + \boldsymbol{v}_i \odot \boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0},\boldsymbol{I})$, which allows a low variance estimation of the gradient of the second term in $\mathcal{L}(\theta,\phi)$ with respect to $\phi$. See Section 2.3 in [31], Section 2.3 in [32] and Section 3 in [53] for more details on stochastic gradient computation in VI. As a result, our framework can be efficiently implemented using a number of open source, automatic differentiation packages such as Tensorflow [1] and PyTorch [49].

*VarFA supports amortized inference.* Classic Bayesian inference methods such as MCMC infer each parameter $\nu_i$ for each student. As the number of students increases, the number of parameters that MCMC needs to infer also increases, which may not be scalable in large-scale data settings. In contrast, unlike classic Bayesian inference methods, VI is *amortized*. It does *not* infer each parameter $\nu_i$. Rather, it estimates a *single* set of variational parameter $\phi$ responsible for inferring *all* $\nu_i$'s. In this way, once we have trained FA models using VarFA and obtain the variational parameter $\phi$, we can easily infer $\nu_i$ by simply computing $q_\phi$, even for new students, without invoking any additional optimization or inference procedures.

## 3.3 Dealing with missing entries

Note from Eq. 10 that the variational distribution $q_\phi$ for each $\nu_i$ is conditioned on $\boldsymbol{y}_i \in \mathbb{R}^Q$, i.e., an entire row in $\boldsymbol{Y}$. However, in practice, the data matrix $\boldsymbol{Y}$ is often only partially observed, i.e., some students only answer a subset of questions. Then, $\boldsymbol{y}_i$'s will likely contain missing values which computation cannot be performed on. To work around this issue, we use "zero imputation", a simple strategy that transforms the missing values to 0, following prior work on applying VI in the context of recommender systems [34, 55, 56, 46] that demonstrated the effectiveness of this strategy despite its simplicity. We note that there exist more elaborate ways to deal with missing entries, such as designing specialized function $f_\phi$ for the variational distribution [37, 20, 36]. We leave the investigation of more effectively dealing with missing entries to future work.

## 3.4 Remarks

*Applicability of VarFA.* The VarFA framework is general and flexible and can be applied to a wide array of FA models. The recipe for applying VarFA to an FA model of choice is as follows: 1) formulate the FA model into the canonical formulation as in Eq. 4; 2) partition the factors into student factor $\nu$, the remaining unknown factors $\theta$ and known factors $\psi$; 3) perform VI on $\nu$ and MLE estimation on $\theta$, following Section. 3.1.

*Relation to variational auto-encoders.* Our proposed framework can be regarded as a standard variational auto-encoder (VAE) but with the decoder implemented not as a neural network but by the FA model. Because the decoder is constraint to a FA model, it is more interpretable than a neural network.

*Relation to other efficient Bayesian factor analysis methods.* We acknowledge that VI applied to general FA models have been proposed in existing literature [18, 74]. However, to our knowledge, little prior work have applied VI to educational FA models nor investigated its effectiveness. One concurrent work applied VI to IRT [70]. Our VarFA framework applies generally to a number of other FA models by following the recipe in the preceding paragraph, including IRT. Thus, our work complements existing literature in providing promising results in applying VI for FA models in the context of educational data mining.

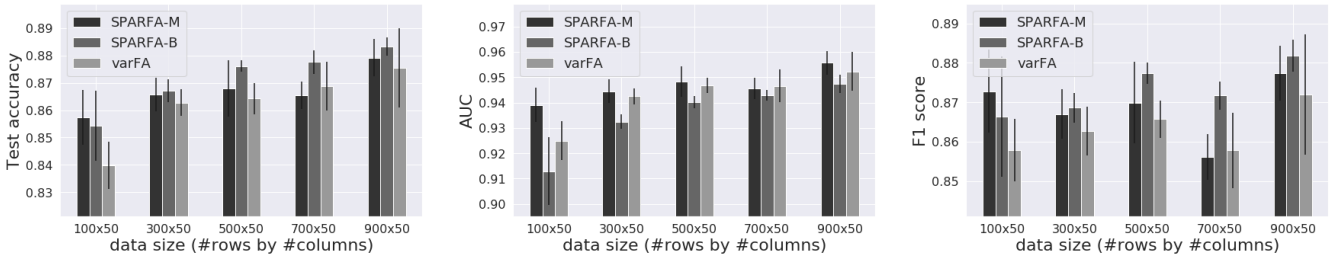



Figure 1: Performance of the SPARFA-M, SPARFA-B, and VarFA algorithms on the synthetic data set with different data sizes. Plots from left to right show comparison on accuracy (ACC), area under curve (AUC) and F1 metrics, respectively. Higher is better for all metrics. VarFA performs similarly to SPARFA-M and SPARFA-B.

# 4. EXPERIMENTS

We demonstrate the efficacy of VarFA variational inference framework using the sparse factor analysis model (SPARFA) as the underlying FA model. This choice of SPARFA as the FA model to investigate is motivated by its mathematical generality: it assumes no auxiliary information is available and all three factors $\mathbf{c}_i$'s, $\mathbf{m}_j$'s and $\mu_j$'s need to be estimated, which makes the inference problem more challenging. [33] provides two inference algorithms including SPARFA-M (based on MLE) and SPARFA-B (based on MAP).

We conduct experiments on both synthetic and real data sets. Using synthetic data sets, we compare VarFA to both SPARFA-M and SPARFA-B. We demonstrate that 1) VarFA predicts students' answers as accurately as SPARFA-M and SPARFA-B; 2) VarFA is almost 100× faster than SPARFA-B. Using real data sets, we compare VarFA to SPARFA-M. We demonstrate that 1) VarFA predicts students' answers more accurately than SPARFA-M; 2) VarFA can output the same insights as SPARFA-M, including point estimate of students' skill levels and questions' associations with skill tags; 3) VarFA can additionally output meaningful uncertainty quantification for student skill levels, which SPARFA-M is incapable of, without sacrifice to computational efficiency. Note that SPARFA-B can also compute uncertainty for small data sets but fails for large data sets due to scalability issues and thus we do not compare to SPARFA-B for real data sets.

Specifically for SPARFA, using the notation convention in Section 2.2, the question-skill association factors $\mathbf{m}_j$'s and the question difficulty factors $\mu_j$ are collected in $\theta = \{\mathbf{m}_1, ..., \mathbf{m}_Q, \mu_1, ..., \mu_Q\}$. The student skill level factors $\mathbf{c}_j$'s are collected in $\nu = \{\mathbf{c}_1, ..., \mathbf{c}_N\}$. Because the factors $\mathbf{m}_j$'s are unknown, the "skills" in SPARFA are *latent* (referred to as "latent skills" in subsequent discussions) and are not attached to any specific interpretation. However, as we will see in Section 4.2.4, assuming the skill tags are available as auxiliary information, we can associate the estimated latent skills with the provided skills tags using the same approach proposed in [33]. For the regularization term in Eq. 11, because the factors $\mathbf{m}_j$'s are assumed to be sparse and nonnegative, we use $\ell 1$ regularization for $\mathbf{m}_j$'s. When performing MLE for $\mathbf{m}_j$'s, we use the proximal gradient algorithm for optimization problem with nonnegative and sparsity requirements; see Section 3.2 in [33] for more details. For the other unknown factor $\mu_j$'s in $\theta$, we apply standard $\ell 2$ regularization. The code along with instructions to reproduce our experiments can be downloaded from https://tinyurl.com/tvm4332.

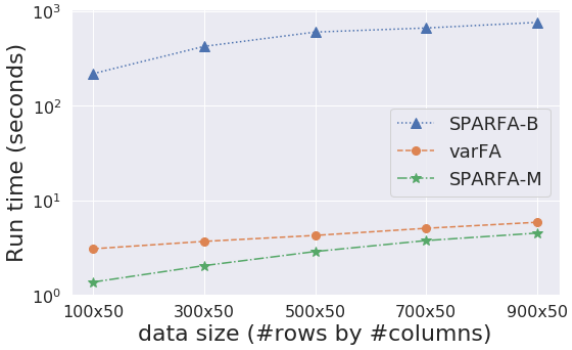



Figure 2: Training run time comparing VarFA, SPARFA-M, and SPARFA-B on synthetic data sets of varying sizes. VarFA performs approximate Bayesian inference almost 100x faster than SPARFA-B and is close to the run time of SPARFA-M. Thus, VarFA enables practical and scalable Bayesian inference for very large data sets.

## 4.1 Synthetic Data Experiments

### 4.1.1 Setup

*Data set generation.* We generate data set according to the canonical FA model specified in Eq. 4, taking into consideration the additional sparsity and nonnegativity assumptions in SPARFA. Specifically, We sample the factors $\mu_j$'s and $\mathbf{c}_i$'s from i.i.d. standard isotropic Gaussian distributions with variance 1 (or identity matrix for $\mathbf{c}_i$'s) and mean sampled from a uniform distribution in range $[-1, 1]$. To simulate sparse and nonnegative factors $\mathbf{m}_j$'s, we first sample an auxiliary variable $s_{kj} \overset{\text{i.i.d.}}{\sim} \text{Bernoulli}(\pi)$, where $\pi$ controls the sparsity, and then sample $m_{kj} \overset{\text{i.i.d.}}{\sim} \text{Exponential}(1)$ when $s_{kj} = 1$ or setting $m_{kj} = 0$ when $s_{kj} = 0$ for each $j$ and $k$. In all synthetic data experiments, we use $\pi = 0.3$ and set the true number of latent skills to $K = 5$.

*Experimental settings.* We vary the size of the data set and use 5 different data matrix sizes: 100×50, 300×50, 500×50, 700×50 and 900×50. We select a data missing rate of 50%, i.e., we randomly choose 50% of all entries in the data matrix as training set and the rest as test set. Experi-

Table 1: Summary statistics of pre-processed real data sets.

| data set | #students | #questions | %observed |
|---|---|---|---|
| assistment | 392 | 747 | 12.93% |
| algebra | 697 | 782 | 13.02% |
| bridge | 913 | 1242 | 11.42% |

mental results for each of the above data sizes are averaged over 5 runs where we randomize over the train/test data split. We train SPARFA with both SPARFA-M and VarFA using the Adam optimizer [30] with learning rate = 0.05 for 100 epochs. Regularization hyper-parameters are chosen by grid search for each experiment. For VarFA, we use a simple 3-layer neural network for the function $f_\phi$ in the variational distribution. We use the hyper-parameter settings for SPARFA-B following Section 4.2 in [33].

*Evaluation metrics.* We evaluate the inference algorithms on their ability to recover (predict) the missing entries in the data matrix given the observed entries. We term this criterion "student answer prediction". Since our data matrix is binary, using prediction accuracy (ACC) alone does not accurately reflect the performance of the inference algorithms under comparison. Thus, we use area under the receiver operating characteristic curve (AUC) and F1 score in addition to ACC, as standard in evaluating binary predictions [23].

#### 4.1.2 Results

Fig. 1 shows bar plots that compare the student answer prediction performance of VarFA with SPARFA-M and SPARFA-B on all three evaluation metrics. We can observe that all three methods perform similarly and that SPARFA-M and SPARFA-B show no statistically significant advantage over VarFA. We further showcase VarFA's scalability by comparing its training run-time with SPARFA-M and SPARFA-B. The results are shown in Fig. 2. VarFA is significantly faster than SPARFA-B and is almost as fast as SPARFA-M.

In summary, with VarFA, we obtain posteriors that allow uncertainty quantification with roughly the *same* computation complexity of computing point estimates and *without* compromising prediction performance.

## 4.2 Real Data Experiments

#### 4.2.1 Setup

*Data sets and pre-processing steps.* We perform experiments on three large-scale, publicly available, real educational data sets including ASSISTments 2009-2010 (Assistment) [24], Algebra I 2006-2007 (algebra) [60] and Bridge to Algebra 2006-2007 (bridge) [61, 62]. The details of the data sets, including data format and data collection procedure can be found in the preceding references. We remove students and questions that have too few answer records from the data sets to reduce the sparsity of the data sets. Specifically, we keep students and questions that have no less than 30, 35 and 40 answer records for Assistment, Algebra and Bridge data sets, respectively. In each data set, each student may provide more than one answer record for each question. Therefore, we also remove student-question answer records except for the first one. Table 1 presents the summary statistics of the resulting pre-processed data matrix for each data set.

Table 2: Student answer prediction erformance comapring VarFA to SPARFA-M on Assistment, Algebra and Bridge data sets. ↑ and ↓ denote higher and lower is better, respectively. VarFA performs better than SPARFA-M on all three data sets and evaluation metrics most of the time. Additionally, VarFA's run time is very close to SPARFA-M.

(a) Assistment

| Metric | Algorithm | |
|---|---|---|
| | SPARFA-M | VarFA |
| ACC ↑ | 0.7074±0.0044 | **0.7101**±0.0048 |
| AUC ↑ | 0.756±0.048 | **0.7635**±0.0036 |
| F1 ↑ | 0.7746±0.0029 | **0.7765**±0.0014 |
| Run time (s) ↓ | **5.3319**±0.2774 | 6.9167±0.1074 |

(b) Algebra

| Metric | Algorithm | |
|---|---|---|
| | SPARFA-M | VarFA |
| ACC ↑ | 0.7735±0.0037 | **0.7774**±0.0031 |
| AUC ↑ | 0.8137±0.003 | **0.8245**±0.002 |
| F1 ↑ | 0.8465±0.0021 | **0.8486**±0.001 |
| Run time (s) ↓ | **8.464**±0.4568 | 10.3335±0.4435 |

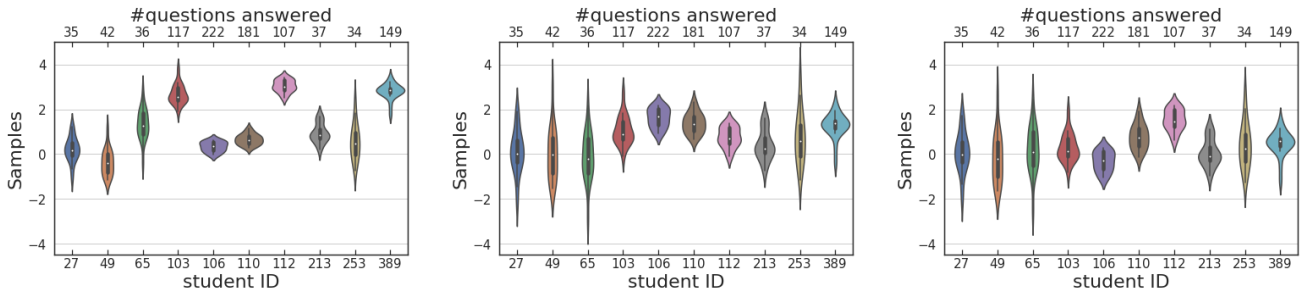
(c) Bridge

| Metric | Algorithm | |
|---|---|---|
| | SPARFA-M | VarFA |
| ACC ↑ | **0.8492**±0.0016 | 0.8468±0.0016 |
| AUC ↑ | 0.837±0.0024 | **0.8419**±0.0028 |
| F1 ↑ | **0.9121**±0.0005 | 0.912±0.0009 |
| Run time (s) ↓ | **15.6048**±0.7314 | 15.8558±1.046 |

*Experimental settings and evaluation metrics.* Optimizer, learning rate, number of epochs, neural network architecture and evaluation metrics are the same as in synthetic data experiments. For real data sets, we use 8 latent skills instead of 5. Other choices of the number of latent skills might result in better performance. However, since we are comparing different inference algorithms for the *same* model, it is a fair to compare the inference algorithms on the same model with the same number of latent dimensions. We use a 80:20 data split, i.e., we randomly sample 80% and 20% of the observed entries in each data set, without replacement, as training and test sets, respectively. Regularization hyper-parameters are selected by grid search and are different for each data set. We only compare VarFA with SPARFA-M because SPARFA-B does not scale to such large data sets.

#### 4.2.2 Results: Performance Comparison

Table 2 shows the average performance on the test set of each data set comparing VarFA and SPARFA-M for all three data sets and additionally run time. We can see that VarFA achieves slightly better student answer prediction on most data sets and on most metrics. Interestingly, recall that, in the preceding synthetic data set experiment,




(a) 3rd latent concept (b) 4th latent concept (c) 7th latent concept

Figure 3: Violin plot showing the mean and standard deviation of the estimated skill mastery levels on 10 selected students on the 3rd, 4th and 7th latent skills that VarFA computes. In each sub-figure, bottom and top axises respectively shows student IDs and top axis shows the number of questions each student answered. The more questions a student answers, the tighter the credible interval. (Best viewed in color.)

SPARFA-M performed better than VarFA most of the time. A possible explanation is that, for synthetic data sets, the underlying data generation process matches the SPARFA model, whereas for real data sets, the underlying data generation process is unknown. VarFA's better performance than SPARFA-M for real data sets implies that VarFA is more robust to the unknown underlying data generation process. As a result, VarFA may be more applicable than SPARFA-M in real-world situations.

Table 2 also shows the run time comparison between VarFA and SPARFA-M; see the last row in each sub-table. We see that both inference algorithms have very similar run time, showing that VarFA is applicable for very large data sets. Notably, VarFA achieves this efficiency while also performing Bayesian inference on the student knowledge level factor.

#### 4.2.3 *Results: Bayesian Inference With VarFA*

We now illustrate VarFA's capability of outputting credible intervals using the Assistment data set. Fig. 3 presents violin plots that show the sampled student latent skill levels for a random subset of 10 students. Plots 3a, 3b and 3c shows the inferred students ability for the 3rd, 4th and 7th latent skill dimension. In each plot, the bottom axis shows the student ID and the top axis shows the total number of questions answered by the corresponding student. For each student, the horizontal width of the violin represents the density of the samples; the skinnier the violin, the more widespread the samples are, implying the model's less certainty on its estimations.

Results in Fig. 3 confirms our intuition that the more questions a student answers, the more certain the model is about its estimation. For example, students with ID 106, 110 and 389 answered 222, 181 and 149 questions, respectively, and the credible intervals of their ability estimation is quite small. In contrast, students with ID 27, 49 and 65 answered far less questions and the credible intervals of their ability estimation is quite large. This result implies that VarFA outputs sensible and interpretable credible intervals. As mentioned in Section 1, such uncertainty quantification may benefit a number of educational applications such as improving adaptive testing algorithms. Note that VarFA is able to compute such credible interval as fast as SPARFA-M, making VarFA potentially useful for even real-time educational systems. In contrast, SPARFA-M is not capable of computing credible intervals. Although we may still obtain confidence interval as uncertainty quantification with SPARFA-M via bootstrapping, i.e., train with SPARFA-M multiple times with random subsets of the data set and then use the point estimates from different random runs to compute confidence intervals. However, this method suffers from two immediate drawbacks: 1) it is slow because we need to run the model multiple times and 2) different runs result in permutation in the estimated factors and thus the averaged estimations loss meaning. Given that VarFA is as efficient as SPARFA-M and the previous drawbacks of SPARFA-M, we recommend VarFA over bootstrapping with SPARFA-M for quantifying model's uncertainty in practice.

#### 4.2.4 *Results: Post-Processing for Improved Interpretability*

SPARFA assumes that each student factor $\nu_i$ identifies a multi-dimensional skill level on a number of "latent" skills (recall that we use 8 latent skills in our experiments). As mentioned earlier, these latent skills are not interpretable without the aid of additional information. To improve interpretability, [33] proposed that, when the skill tags for each question is available in the data set, we can associate each latent skill with skill tags via a simple matrix factorization. Then, we can compute each students' mastery levels on the actual skill tags. We refer readers to Sections 5.1 and 5.2 in [33] for more technical details. Although the above method to improve interpretability has already been proposed, [33] only presented results on private data sets, whereas here we presents results on publicly available data set with code, making our results more transparent and reproducible.

We again use the Assistment data set for illustration. We compute the association of skill tags in the data set with each of the latent skills and show 4 of the latent skills with their top 3 most strongly associated skill tags. We can see that each latent skill roughly identify the same group of skill tags. For example, latent skill 4 clusters skill tags on statistics and probability while latent skill 7 clusters skill tags on geometry. Thus, by simple post-processing, we obtain an interpretation of the latent skills by associating them with known skill tags

Table 3: Illustration of the estimated latent skills with the their top 3 most strongly associated skill tags in the Assistment data set. The percentage in the parenthesis shows the association probability (summed to 1 for each latent skill). We see that the tagged skills associated with each estimated latent skill form intuitive and interpretable groups.

| **Latent Skill 1** | **Latent Skill 3** |
|---|---|
| Division Fractions (29.1%) | Conversion of Fraction Decimals Percents (7.3%) |
| Least Common Multiple (18.1%) | Addition and Subtraction Positive Decimals (6.8%) |
| Write Linear Equation from Ordered Pairs (17.8%) | Probability of a Single Event (5.7%) |
| **Latent Skill 4** | **Latent Skill 7** |
| Pattern Finding (17.4%) | Volume Sphere (13.4%) |
| Histogram as Table or Graph (11.3%) | Volume Cylinder (10.4%) |
| Percent Of (10.5%) | Surface Area Rectangular Prism (10.2%) |

in the data.

We can similarly obtain VarFA's estimations of the students' mastery levels on each skill tags through the above process. In Fig. 4, we compare the predicted mastery level for each skill tag (only for the questions this student answered) with the percent of correct answers for that skill tag. Blue curve shows the empirical student's mastery level on a skill tag by computing the percentage of correctly answered questions belonging to a particular skill tag. Orange curve shows VarFA's estimated student mastery level on a skill tag, normalized to range $[0, 1]$. We can see that, when the student gave more correct answers to questions of a particular skill tag, such as skill tag ID 2, 10, 14 and 30, VarFA also predicts a higher ability score for these skill tags. When the student gave more incorrect answers to questions of a particular skill tag, such as skill tag ID 0, 4 and 27, VarFA also predicts a lower ability score for those skill tags. Although the correspondence is not perfect, VarFA's predicted student's mastery levels match our intuition about student's abilities (using the observed correct answer ratio as an empirical estimate) reasonably well. Thus, by post-processing, we can interpret VarFA's estimated students' latent skill mastery levels using the easily understandable skill tags.

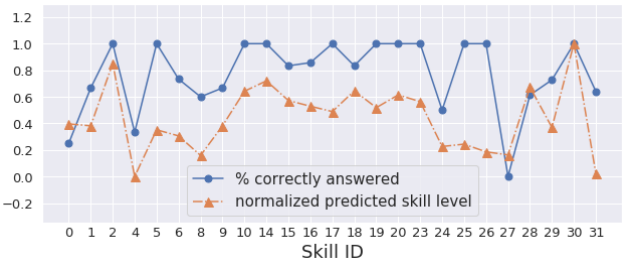



Figure 4: Comparison between the estimated skill mastery levels using VarFA's predictions and using empirical observations for student with ID 110. Even though the two curves show different numeric values, they nevertheless demonstrate similar trends, showing that the predictions reasonably match our intuition about student's skill mastery levels.

## 5. CONCLUSIONS AND FUTURE WORK

We have presented VarFA, a variational inference factor analysis framework to perform efficient Bayesian inference for learning analytics. VarFA is general and can be applied to a wide array of FA models. We have demonstrated the effectiveness of our VarFA using the sparse factor analysis (SPARFA) model as a case study. We have shown that VarFA can very efficiently output interpretable, educationally meaningful information, in particular credible intervals, much faster than classic Bayesian inference methods. Thus, VarFA has potential application in many educational data mining scenarios where efficient credible interval computation is desired, i.e., in adaptive testing and adaptive learning systems. We have also provided open-source code to reproduce our results and facilitate further research efforts.

We outline three possible future research directions and extensions. First, VarFA currently performs Bayesian inference on the student skill mastery level factor. We are working on extending the framework to perform full Bayesian inference for all unknown factors. Extensions to some models such as IRT is straightforward, as studied in [70], because all factors can be reasonably assumed to be Gaussian. However, some other models involve additional modeling assumptions, making it challenging to design distributions that satisfy these assumptions. For example, in SPARFA, one of the factors is assumed to be nonnegative and sparse [33]. No standard distribution fulfill these requirements. We are exploring *mixture* distributions, in particular spike and slab models [27] for Bayesian variable selection, and methods to combine them with variational inference, following [64, 65].

Second, other methods exist that accelerate classic Bayesian inference. Recent works in large-scale Bayesian inference proposed *approximate* MCMC methods that scale to large data sets [73, 38, 59]. Some of these methods apply variational inference to perform the approximation [13, 22]. We are investigating extending VarFA to support approximate MCMC methods.

Finally, the goal of VarFA is to improve real-world educational systems and ultimately improve learning. Therefore, it is necessary to evaluate VarFA beyond synthetic and benchmark data sets. We plan to integrate VarFA into an existing educational system and conduct a case study where students interact with the system in real-time to understand the VarFA's educational implications in the wild.

## Acknowledgements

This work was supported by NSF grants CCF-1911094, IIS-1838177, IIS-1730574, DRL-1631556, IUSE-1842378; ONR grants N00014-18-12571 and N00014-17-1-2551; AFOSR grant FA9550-18-1-0478; and a Vannevar Bush Faculty Fellowship, ONR grant N00014-18-1-2047.

## 6. REFERENCES

[1] M. Abadi et al. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from `tensorflow.org`.

[2] T. A. Ackerman. Using multidimensional item response theory to understand what items and tests are measuring. *Applied Measurement in Education*, 7(4):255–278, 1994.

[3] J. R. Anderson, A. T. Corbett, K. R. Koedinger, and R. Pelletier. Cognitive tutors: Lessons learned. *Journal of the Learning Sciences*, 4(2):167–207, 1995.

[4] R. S. Baker and K. Yacef. The state of educational data mining in 2009: A review and future visions. *Journal of Educational Data Mining*, 1(1):3–17, 2009.

[5] C. M. Bishop. *Pattern recognition and machine learning*. springer, 2006.

[6] H. Cen, K. Koedinger, and B. Junker. Learning factors analysis – a general method for cognitive model evaluation and improvement. In M. Ikeda, K. D. Ashley, and T.-W. Chan, editors, *Proc. Intelligent Tutoring Systems*, pages 164–175, 2006.

[7] H.-H. Chang, Z. Ying, et al. Nonlinear sequential designs for logistic item response theory models with applications to computerized adaptive tests. *The Annals of Statistics*, 37(3):1466–1488, 2009.

[8] T. Q. Chen, X. Li, R. B. Grosse, and D. K. Duvenaud. Isolating sources of disentanglement in variational autoencoders. In *Proc. Advances in Neural Information Processing Systems*, pages 2610–2620, 2018.

[9] M. Chi, K. R. Koedinger, G. J. Gordon, P. Jordon, and K. VanLahn. Instructional factors analysis: A cognitive model for multiple instructional interventions. 2011.

[10] B. Choffin, F. Popineau, Y. Bourda, and J.-J. Vie. Das3h: Modeling student learning and forgetting for optimally scheduling distributed practice of skills. In *Proc. Educational Data Mining*, 2019.

[11] K. Chrysafiadi and M. Virvou. Student modeling approaches: A literature review for the last decade. *Expert Systems with Applications*, 40(11):4715–4729, 2013.

[12] E. Creager, D. Madras, J.-H. Jacobsen, M. Weis, K. Swersky, T. Pitassi, and R. Zemel. Flexibly fair representation learning by disentanglement. In K. Chaudhuri and R. Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proc. Machine Learning Research*, pages 1436–1445, Long Beach, California, USA, 09–15 Jun 2019. PMLR.

[13] N. De Freitas, P. Højen-Sørensen, M. I. Jordan, and S. Russell. Variational mcmc. In *Proc. conference on Uncertainty in Artificial Intelligence*, pages 120–127. Morgan Kaufmann Publishers Inc., 2001.

[14] M. P. Driscoll. *Psychology of learning for instruction.* Allyn & Bacon, 1994.

[15] J.-P. Fox. *Bayesian item response modeling: Theory and applications*. Springer Science & Business Media, 2010.

[16] J.-P. Fox and C. A. Glas. Bayesian estimation of a multilevel irt model using gibbs sampling. *Psychometrika*, 66(2):271–288, 2001.

[17] A. Gelman, J. B. Carlin, H. S. Stern, D. B. Dunson, A. Vehtari, and D. B. Rubin. *Bayesian data analysis*. Chapman and Hall/CRC, 2013.

[18] Z. Ghahramani and M. J. Beal. Variational inference for bayesian mixtures of factor analysers. In *Proc. Advances in Neural Information Processing Systems*, pages 449–455, 2000.

[19] W. R. Gilks, S. Richardson, and D. Spiegelhalter. *Markov chain Monte Carlo in practice*. Chapman and Hall/CRC, 1995.

[20] W. Gong, S. Tschiatschek, R. Turner, S. Nowozin, and J. M. Hernández-Lobato. Icebreaker: Element-wise active information acquisition with bayesian deep latent gaussian model. In *Proc. Advances in Neural Information Processing Systems*, 2019.

[21] S. Graf and Kinshuk. *Personalized Learning Systems*, pages 2594–2596. Springer US, Boston, MA, 2012.

[22] R. Habib and D. Barber. Auxiliary variational mcmc. *Proc. International Conference on Learning Representations*, 2019.

[23] T. Hastie, R. Tibshirani, and J. Friedman. *The elements of statistical learning: data mining, inference, and prediction*. Springer Science & Business Media, 2009.

[24] N. T. Heffernan and C. Heffernan. The assistments ecosystem: Building a platform that brings scientists and teachers together for minimally invasive research on human learning and teaching. *International Journal of Artificial Intelligence in Education*, 24:470–497, 2014.

[25] N. T. Heffernan and C. L. Heffernan. The assistments ecosystem: Building a platform that brings scientists and teachers together for minimally invasive research on human learning and teaching. *International Journal of Artificial Intelligence in Education*, 24(4):470–497, 2014.

[26] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *Proc. International Conference on Learning Representations*, 2017.

[27] H. Ishwaran, J. S. Rao, et al. Spike and slab variable selection: frequentist and bayesian strategies. *The Annals of Statistics*, 33(2):730–773, 2005.

[28] J. L. W. V. Jensen et al. Sur les fonctions convexes et les inégalités entre les valeurs moyennes. *Acta Mathematica*, 30:175–193, 1906.

[29] H. Kim and A. Mnih. Disentangling by factorising. In *Proc. International Conference on Machine Learning*, volume 80, pages 2649–2658, 2018.

[30] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *Proc. International Conference on Learning Representations*, 2015.

[31] D. P. Kingma and M. Welling. Auto-encoding

variational bayes. In *Proc. International Conference on Learning Representations*, 2013.

[32] D. P. Kingma, M. Welling, et al. An introduction to variational autoencoders. *Foundations and Trends® in Machine Learning*, 12(4):307–392, 2019.

[33] A. S. Lan, A. E. Waters, C. Studer, and R. G. Baraniuk. Sparse factor analysis for learning and content analytics. *Journal of Machine Learning Research*, 15:1959–2008, 2014.

[34] D. Liang, R. G. Krishnan, M. D. Hoffman, and T. Jebara. Variational autoencoders for collaborative filtering. In *Proc. International Conference on World Wide Web*, pages 689–698, 2018.

[35] R. V. Lindsey, J. D. Shroyer, H. Pashler, and M. C. Mozer. Improving studentsâĂŹ long-term knowledge retention through personalized review. *Psychological Science*, 25(3):639–647, 2014.

[36] C. Ma, W. Gong, J. M. Hernández-Lobato, N. Koenigstein, S. Nowozin, and C. Zhang. Partial vae for hybrid recommender system. In *Proc. NeurIPS Workshop on Bayesian Deep Learning*, 2018.

[37] C. Ma, S. Tschiatschek, K. Palla, J. M. Hernández-Lobato, S. Nowozin, and C. Zhang. Eddi: Efficient dynamic discovery of high-value information with partial vae. In *Proc. International Conference on Machine Learning*, 2019.

[38] Y.-A. Ma, T. Chen, and E. Fox. A complete recipe for stochastic gradient mcmc. In *Proc. Advances in Neural Information Processing Systems*, pages 2917–2925, 2015.

[39] D. J. MacKay and D. J. Mac Kay. *Information theory, inference and learning algorithms*. Cambridge University Press, 2003.

[40] E. Mathieu, T. Rainforth, N. Siddharth, and Y. W. Teh. Disentangling disentanglement in variational autoencoders. In *Proc. International Conference on Machine Learning*, pages 4402–4412, Jun 2019.

[41] A. Merceron and K. Yacef. Educational data mining: a case study. In *Proc. Artificial Intelligence in Education*, pages 467–474, 2005.

[42] E. C. Merkle. A comparison of imputation methods for bayesian factor analysis models. *Journal of Educational and Behavioral Statistics*, 36(2):257–276, 2011.

[43] S. Minn, Y. Yu, M. C. Desmarais, F. Zhu, and J.-J. Vie. Deep knowledge tracing and dynamic student classification for knowledge tracing. In *Proc. IEEE International Conference on Data Mining*, pages 1182–1187. IEEE, 2018.

[44] R. J. Mislevy. Bayes modal estimation in item response models. *Psychometrika*, 51(2):177–195, 1986.

[45] M. C. Mozer and R. V. Lindsey. Predicting and improving memory retention: Psychological theory matters in the big data era. In *Big Data in Cognitive Science*, pages 43–73. Psychology Press, 2016.

[46] A. Nazabal, P. M. Olmos, Z. Ghahramani, and I. Valera. Handling incomplete heterogeneous data using vaes. *arXiv preprint arXiv:1807.03653*, 2018.

[47] B. D. Nye, A. C. Graesser, and X. Hu. Autotutor and family: A review of 17 years of natural language tutoring. *International Journal of Artificial Intelligence in Education*, 24(4):427–469, 2014.

[48] Z. A. Pardos, N. T. Heffernan, B. Anderson, C. L. Heffernan, and W. P. Schools. Using fine-grained skill models to fit student performance with bayesian networks. *Handbook of Dducational Data Mining*, 417, 2010.

[49] A. Paszke et al. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Proc. Advances in Neural Information Processing Systems*, pages 8024–8035. Curran Associates, Inc., 2019.

[50] P. I. Pavlik, H. Cen, and K. R. Koedinger. Performance factors analysis –a new alternative to knowledge tracing. In *Proc. Artificial Intelligence in Education*, page 531âĂŞ538, NLD, 2009. IOS Press.

[51] C. Piech, J. Bassen, J. Huang, S. Ganguli, M. Sahami, L. J. Guibas, and J. Sohl-Dickstein. Deep knowledge tracing. In *Proc. Advances in Neural Information Processing Systems*, pages 505–513. 2015.

[52] A. N. Rafferty, M. M. LaMar, and T. L. Griffiths. Inferring learners' knowledge from their actions. *Cognitive Science*, 39(3):584–618, 2015.

[53] D. J. Rezende, S. Mohamed, and D. Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *Proc. International Conference on Machine Learning*, ICMLâĂŹ14, page IIâĂŞ1278âĂŞIIâĂŞ1286. JMLR.org, 2014.

[54] C. Romero and S. Ventura. Educational data mining: a review of the state of the art. *IEEE Transactions on Systems, Man, and Cybernetics*, 40(6):601–618, 2010.

[55] R. Salakhutdinov, A. Mnih, and G. Hinton. Restricted boltzmann machines for collaborative filtering. In *Proc. International Conference on Machine Learning*, pages 791–798, 2007.

[56] S. Sedhain, A. K. Menon, S. Sanner, and L. Xie. Autorec: Autoencoders meet collaborative filtering. In *Proc. International Conference on World Wide Web*, pages 111–112, 2015.

[57] V. J. Shute and J. Psotka. Intelligent tutoring systems: Past, present, and future. Technical report, 1994.

[58] N. Siddharth, B. Paige, J.-W. Van de Meent, A. Desmaison, N. Goodman, P. Kohli, F. Wood, and P. Torr. Learning disentangled representations with semi-supervised deep generative models. In *Proc. Advances in Neural Information Processing Systems*, pages 5925–5935, Dec. 2017.

[59] J. Song, S. Zhao, and S. Ermon. A-nice-mc: Adversarial training for mcmc. In *Proc. Advances in Neural Information Processing Systems*, pages 5140–5150, 2017.

[60] J. Stamper, A. Niculescu-Mizil, S. Ritter, G. Gordon, and K. Koedinger. Algebra i 2006-2007. *Development data set from KDD Cup 2010 Educational Data Mining Challenge*, 2010.

[61] J. Stamper, A. Niculescu-Mizil, S. Ritter, G. Gordon, and K. Koedinger. Bridge to algebra 2006-2007. *Development data set from KDD Cup 2010 Educational Data Mining Challenge*, 2010.

[62] J. C. Stamper and Z. A. Pardos. The 2010 kdd cup competition dataset: Engaging the machine learning community in predictive learning analytics. 2016.

[63] K. K. Tatsuoka. Rule space: An approach for dealing with misconceptions based on item response theory. *Journal of Educational Measurement*, 20(4):345–354, 1983.

[64] M. K. Titsias and M. Lázaro-Gredilla. Spike and slab variational inference for multi-task and multiple kernel learning. In *Proc. Advances in neural information processing systems*, pages 2339–2347, 2011.

[65] F. Tonolini, B. S. Jensen, and R. Murray-Smith. Variational sparse coding, July 2019.

[66] W. J. van der Linden and R. K. Hambleton. *Handbook of modern item response theory*. Springer Science & Business Media, 2013.

[67] K. VanLehn. Student modeling. *Foundations of Intelligent Tutoring Systems*, 55:78, 1988.

[68] J.-J. Vie and H. Kashima. Knowledge tracing machines: Factorization machines for knowledge tracing. In *Proc. AAAI Conference on Artificial Intelligence*, volume 33, pages 750–757, 2019.

[69] K. H. Wilson, Y. Karklin, B. Han, and C. Ekanadham. Back to the basics: Bayesian extensions of irt outperform neural networks for proficiency estimation. In *Proc. Educational Data Mining*, 2016.

[70] M. Wu, R. L. Davis, B. W. Domingue, C. Piech, and N. Goodman. Variational item response theory: Fast, accurate, and expressive. *arXiv preprint arXiv:2002.00276*, 2020.

[71] D. Yan, A. A. Von Davier, and C. Lewis. *Computerized multistage testing: Theory and applications*. CRC Press, 2016.

[72] J. Zhang, X. Shi, I. King, and D.-Y. Yeung. Dynamic key-value memory networks for knowledge tracing. In *Proc. International Conference on World Wide Web*, pages 765–774, 2017.

[73] R. Zhang, C. Li, J. Zhang, C. Chen, and A. G. Wilson. Cyclical stochastic gradient mcmc for bayesian deep learning. *Proc. International Conference on Learning Representations*, 2020.

[74] J.-h. Zhao and L. Philip. A note on variational bayesian factor analysis. *Neural Networks*, 22(7):988–997, 2009.

[75] S. Zhao, J. Song, and S. Ermon. Infovae: Balancing learning and inference in variational autoencoders. In *Proc. AAAI Conference on Artificial Intelligence*, volume 33, pages 5885–5892, Feb. 2019.