Variational Item Response Theory:
Fast, Accurate, and Expressive

Mike Wu♦, Richard L. Davis◇, Benjamin W. Domingue◇, Chris Piech♦, Noah Goodman♦,■
Department of Computer Science (♦), Education (◇), and Psychology (■)
Stanford University
{wumike,rldavis,bdomingu,cpiech,ngoodman}@stanford.edu

ABSTRACT

Item Response Theory is a ubiquitous algorithm used around the world to understand humans based on their responses to questions in fields as diverse as education, medicine and psychology. However, for medium to large datasets, contemporary solutions pose a tradeoff: either have bayesian, interpretable, accurate estimates or have fast computation. We introduce variational inference and deep generative models to Item Response Theory to offer the best of both worlds. The resulting algorithm is (a) orders of magnitude faster when inferring on the classical model, (b) naturally extends to more complicated input than binary correct/incorrect, and more expressive deep bayesian models of responses. Applying this method to five large-scale item response datasets from cognitive science and education, we find improvements in imputing missing data and better log likelihoods. The open-source algorithm is immediately usable.

1. INTRODUCTION

The task of estimating human ability from stochastic responses to a series of questions has been studied since the 1950s in thousands of papers and spans several domains. The corresponding Item Response Theory algorithms are used every day around the world, and are currently deployed in many human critical contexts including college admissions test, school-system assessment, survey analysis, popular questionnaires, and medical diagnosis.

Despite a large body of literature, contemporary IRT methods fall short – it remains surprisingly difficult to estimate human ability from stochastic responses. One crucial bottleneck is that the state of the art Bayesian inference algorithms are prohibitively slow to run on reasonable sized datasets even with very rudimentary models of response. This leaves practitioners with a choice: either have interpretable, nuanced Bayesian models with appropriate inference or have timely computation.

In the field of artificial intelligence a revolution in deep gen-erative models via "variational inference" have demonstrated an impressive ability to perform fast inference on complex Bayesian models. In this paper, we present a novel application of variational AI to the problem of IRT. We present necessary derivations for the corresponding inference algorithms and apply the algorithms both to synthetic experimentation as well as real world datasets.

Specifically, our contributions are as follows:

1. **Variational inference for IRT:** We derive a novel optimization objective to perform inference for Item Response Theory. To do so, we derive a variational posterior distribution over ability and item characteristics that is (importantly) amortized over subjects.

2. **Faster inference:** We find our inference approach to be orders of magnitude faster than previous inference techniques and be usable on much larger datasets with no loss in accuracy.

3. **More expressive:** Our inference approach is naturally compatible with deep generative models and, as such, we enable the novel extension of Bayesian IRT models to use neural network based representations for inputs, predictions, and student ability. We develop the first deep generative IRT models.

4. **Real world application:** We demonstrate the impact of faster, expressive inference by applying our algorithms to datasets including: PISA, DuoLingo and Gradescope. We achieve up to 200 times speedup and show improved accuracy at imputing hidden responses.

The potential real world impact of this extension is two-fold. More accurate models can be constructed for existing applications of Item Response Theory, such as for visual acuity tests [40], or educational assessments such as PISA [16]. Second, Item Response Theory can be extended for students who produce work where their responses go beyond binary correct/incorrect (imagine students writing text, drawing pictures, or even learning to code) encouraging educators to give students more engaging items without having to give up Bayesian inference.

2. BACKGROUND

We briefly review the many variations of item response theory and the fundamental principles of approximate inference, focusing on modern variational inference.

2.1 Item Response Theory

Imagine answering a series of multiple choice questions. For example, consider a personality survey, a homework assignment, or a medical examination. As a subject, selecting a response to each question is an interaction between your ability and the characteristics of the question, such as its difficulty. The goal in examination analysis is to gauge this unknown ability of each student and the unknown item characteristics based only on his or her responses. Early procedures [14] defaulted to very simple methods such as counting the number of correct responses, ignoring differences in question quality. In reality, we understand that not all questions are created equal: some may be hard to understand while others may test more difficult concepts. To capture these nuances, Item Response Theory (IRT) was developed as a mathematical framework to reason formally about people's responses to items. The core principle underlying IRT is that the probability of a correct response to an item is a logistic function of only the person ability and item characteristics.

Being widely popular, the IRT model plays an impactful role in many large institutions. It is the preferred method for estimating ability in several state assessments in the United States, for international assessments gauging educational competency across countries [21], and for the National Assessment of Educational Programs (NAEP), a large-scale measurement of literacy and reading comprehension in the US [42]. Beyond education, IRT is a classical method widely used in cognitive science and psychology, especially with regards to studies of language acquisition and development [22, 34, 17, 10].
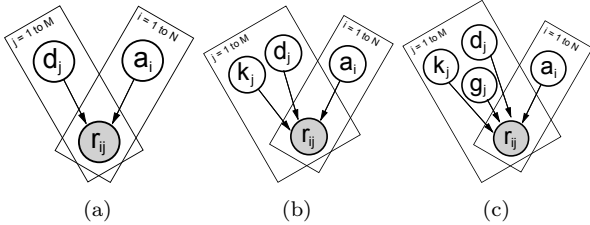
IRT has many forms, of which we review a few:

Figure 1: Graphical models for the (a) 1PL, (b) 2PL, and (c) 3PL Item Response Theories. Observed variables are shaded whereas latent variables are white. Arrows represent dependency between random variables and each rectangle represents a plate. Note nested plate as there is a response for each person and item.

The simplest class of IRT summarizes the ability of a person with a single parameter. This class contains three versions: 1PL, 2PL, and 3PL IRT, each of the differ by the number of free variables used to characterize an item. The 1PL IRT model, also called the Rasch model [41], is given in Eq. 1,

$$p(r_{ij}|a_i, d_j) = \frac{1}{1 + e^{-(a_i - d_j)}} \qquad (1)$$

where $i = 1 \ldots N$ persons and $j = 1 \ldots M$ items with r_{ij} being the response by the i-th person to the j-th question. Each item in the 1PL model is characterized by a single number representing difficulty, d_j. Notice that the func-

tional form of the 1PL model is a logistic function. Thus, a higher difficulty requires a higher ability in order to respond correctly. The 2PL IRT model adds a *discrimination* parameter, k_j for each item that controls the slope (or scale) of the logistic curve whereas the 3PL IRT model also adds a *pseudo-guessing* parameter, g_j for each item that sets the asymptotic minimum of the logistic curve. We can expect a higher discrimination to more heavily separate people of low and high ability. We can also interpret pseudo-guessing as the probability of success if the respondent were to make an educated guess on this question. [1] The 2PL and 3PL IRT models are shown in Eq. 2. Fig. 1 shows the graphical models for each of the IRTs.

$$p(r_{ij}|a_i, \boldsymbol{d}_j) = \frac{1}{1 + e^{-k_j a_i - d_j}} \ \text{ or } \ g_j + \frac{1 - g_j}{1 + e^{-k_j a_i - d_j}} \quad (2)$$

where $\boldsymbol{d}_j = \{k_j, d_j\}$ for 2PL and $\boldsymbol{d}_j = \{g_j, k_j, d_j\}$ for 3PL.

How good of an approximation is IRT to the true underlying function for human cognitive reasoning? The answer varies, and is dependent by the dimensionality of the construct. For instance, if we are measuring a person's understanding on elementary arithmetic, then a single dimension may suffice in capturing the majority of the variance: the simplicity of the domain is limited to four basic operations. However, if we are instead measuring a person's general mathematics ability, a single real number no longer seems sufficient. Even if we bound the domain to middle school mathematics, there are several factors that contribute to "mathematical understanding" (e.g. proficiency in algebra versus geometry). Summarizing a person with a single number in this setting would result in a fairly loose approximation.

For these cases where multiple facets of ability (e.g. intellectual versus socio-emotional) contribute to performance, we consider *multidimensional* item response theory [1, 43, 35]. For simplicitly, we show only 2PL multidimensional IRT:

$$p(r_{ij}|\boldsymbol{a}_i, \boldsymbol{k}_j, d_j) = \frac{1}{1 + e^{-\mathbf{a}_i^T \boldsymbol{k}_j - d_j}} \qquad (3)$$

where we use bolded notation $\boldsymbol{a}_i = (a_i^{(1)}, a_i^{(2)}, \ldots a_i^{(K)})$ to represent a K dimensional vector. Notice that the discrimination is also a vector of equal size to ability. The 1PL and 2PL MIRT models follow from Eq. 3.

In this paper, we make the following simplifying assumptions common in standard item response theory [43]: (1) person abilities cannot change over the course of answering questions, (2) the response by a person to one test item is independent of the responses to any other item, (3) the relation between ability and item characteristics is linear, and (4) the responses are binary. Here, we will explore both nonlinear generalizations to IRT as well as polytomous responses. Other extensions to knowledge tracing, teslets, nonparameteric and unfolding models should be straightforward and are left as future work.

In practice, we are given a dataset of responses and must

---

[1] We default to 2PL as the pseudo-guessing parameter introduces several invariances in the model. This requires far more data (in the millions) to infer ability accurately, as measured by our synthetic experiments. For practitioners, we warn against using 3PL for small to medium datasets.

"fit" one of the many IRT models. In other words, given a $N \times M$ matrix of observed responses, we want to *infer* the ability of all N persons (multi-dimensional or not) and the characteristics of all M items presented. Often, a portion of the $N \times M$ entries are missing. For instance, a student may have skipped a question on the exam or perhaps different versions of an assessment were given to different population groups. In the next section, we provide a brief overview of inference in item response theory followed by a technical introduction to modern approximate inference techniques.

2.2 Inference in Item Response Theory

We compare and contrast the three popular methods used to do inference for IRT in research and industry.

Joint Maximum Likelihood. The most straightforward approach is to jointly pick the most likely ability and item features given the observed responses. We optimize the following objective:

$$\mathcal{L}_{\text{MLE}} = \max_{\{\boldsymbol{a}_i\}_{i=1}^N, \{\boldsymbol{d}_j\}_{j=1}^M} \sum_{i=1}^N \sum_{j=1}^M \log p(r_{i,j}|\boldsymbol{a}_i, \boldsymbol{d}_j) \quad (4)$$

$$= \sum_{i=1}^N \sum_{j=1}^M \max_{\boldsymbol{a}_i} \max_{\boldsymbol{d}_j} \log p(r_{i,j}|\boldsymbol{a}_i, \boldsymbol{d}_j) \quad (5)$$

with any standard optimizer e.g. stochastic gradient descent. The symbol $\boldsymbol{d}_j$ represents all item features e.g. $\boldsymbol{d}_j = \{d_j, \boldsymbol{k}_j\}$ for 2PL. Eq. 5 is often called the Joint Maximum Likelihood Estimator [3, 15], abbreviated MLE. MLE poses inference as a supervised regression problem in which we choose the most likely unknown variables to match known dependent variables. While MLE is simple to understand and implement, we lack any measure of uncertainty which have important consequences when responses are missing.

Marginal Maximum Likelihood. In classic works, many have pointed that when using MLE, the number of unknown parameters (that we must optimize for) unhealthily increases with the number of persons [8, 20]. In particular, [20] shows that in practical settings with a finite (and often small) number of items, standard convergence theorems do not hold for MLE as the number of persons grows.

To remedy this, the authors instead view ability as a nuisance parameter: one that should be marginalized out and removed from consideration [8, 9]. Brock et. al. introduces an Expectation-Maximization (EM) [13] algorithm to iterate between (1) updating beliefs about item characteristics and (2) using the updated beliefs to define a marginal distribution (without ability) $p(r_{ij}|\boldsymbol{d}_j)$ by numerical integration of $\boldsymbol{a}_i$. Appropriately, this algorithm is referred to as Maximum Marginal Likelihood Estimation, abbreviated as EM. Eq. 7 shows the E and M steps for EM.

$$\text{E step:} \quad p(r_{ij}|\boldsymbol{d}_j^{(t)}) = \int_{\boldsymbol{a}_i} p(r_{ij}|\boldsymbol{a}_i, \boldsymbol{d}_j^{(t)}) p(\boldsymbol{a}_i) d\boldsymbol{a}_i \quad (6)$$

$$\text{M step:} \quad \boldsymbol{d}_j^{(t+1)} = \arg\max_{\boldsymbol{d}_j} \sum_{i=1}^N \log p(r_{ij}|\boldsymbol{d}_j^{(t)}) \quad (7)$$

where (t) represents the iteration count. We often choose $p(\boldsymbol{a}_i)$ to be a simple prior distribution like standard Normal. In general, the integral in the E-step is intractable: EM uses a Gaussian-Hermite quadrature to discretely approximate $p(r_{ij}|\boldsymbol{d}_j^{(t)})$. See [23] for a closed form expression for $\boldsymbol{d}_j^{(t+1)}$ in the M step. As a property of EM, this method finds the maximum a posteriori (MAP) estimates for item characteristics. Interestingly EM does not infer ability as it is "ignored" in the model, which may be a limitation in some use cases. The common practice is to use EM to infer item characteristics, then fit ability using a second auxiliary model. In practice, EM has grown to be very popular in industry especially as it is fast for small to medium sized datasets. However, we expect that EM may scale poorly to large datasets and multi-dimensional IRT models as numerical integration requires far more points to properly measure high dimensional volume.

Bayesian Item Response Theory. EM begins to incorporate uncertainty into its inference by using a prior distribution along with the conditional, thereby by Bayes rule, is proportional to the posterior. However, it draws *only* the most likely sample from the posterior, which ignores variance and may struggle with multi-modal distributions. Ideally, we would like to draw i.i.d. samples from the posterior $p(\boldsymbol{a}_i, \boldsymbol{d}_j|\boldsymbol{r}_{ij})$ or perhaps even derive a closed form expression. To achieve the former, MCMC methods have been applied to item response theory. While theoretically appealing, these approaches have not yet gained widespread popularity due to their cost, often several magnitudes more expensive than EM or MLE. In the next subsection, we review MCMC and explore a more modern approach to approxmiate Bayesian inference: variational methods. We hypothesize that variational approaches may preserve some of the theoretical benefits of Bayesian IRT without the overt cost of MCMC.

2.3 Approximate Inference

We will present approximate inference in a general manner and revisit its exact applications to IRT in Sec. 3.

Let $p(\boldsymbol{x}, \boldsymbol{z})$ be a joint distribution over a set of latent variables $\boldsymbol{z}$ and observed variables $\boldsymbol{x}$. The task of *inference* is to compute posterior beliefs over the latent variables after incorporating evidence into the prior distribution: $p(\boldsymbol{z}|\boldsymbol{x}) = p(\boldsymbol{x}, \boldsymbol{z})/p(\boldsymbol{x})$. This quantity, however, is usually intractable as the marginal likelihood $p(\boldsymbol{x}) = \int_{\boldsymbol{z}} p(\boldsymbol{x}, \boldsymbol{z}) d\boldsymbol{z}$ requires solving an infeasible integral. Since we are also not guaranteed to know the form of $p(\boldsymbol{z}|\boldsymbol{x})$, even direct sampling is difficult. Thus, we are forced to approximate the posterior instead.

Sampling Methods. Largely, there are two classes of approximate inference techniques. The first, known as Markov Chain Monte Carlo (MCMC) sampling [24, 18] draws samples from the desired distribution by constructing a Markov chain carefully designed such that $p(\boldsymbol{z}|\boldsymbol{x})$ is the equilibrium distribution. By mixing the chain, the more steps taken, the more closely the distibution of drawn samples matches the true posterior. A primary critique of the MCMC algorithm is that it struggles in high dimensions where samples drawn from the current proposal distribution are commonly rejected, resulting in very slow mixing. To address this issue,

the Hamiltonian Monte Carlo (HMC) algorithm [39, 38, 26] improves upon the MCMC algorithm by choosing the next sample in a more sophisticated manner (rather than a random walk procedure). It does so by solving a Hamiltonian dynamical system (via numerical integration) to define the next proposal distribution. Intuitively, doing so reduces the correlation between successive samples, ensuring a consistently higher probability of acceptance via properties of the Hamiltonian dynamics. We recommend reading [26] for a good review of HMC. The strengths of these algorithms are that the samples are (mostly) guaranteed to be drawn from the true posterior $p(\boldsymbol{z}|\boldsymbol{x})$; correlation issues aside, we can be confident in using the samples for downstream computation. However, the main drawbacks of sampling algorithms are computation — mixing an MCMC chain or using the Leapfrog integrator require signficant compute, meaning these algorithms scale poorly to large datasets and high dimensions. In the context of IRT, with new datasets of upwards of a million people, such limitations can be debilitating.

Given their lengthy existence, MCMC based approaches have been the de-factor approach in psychology, cognitive science, and education. However, there exist a second class of approximate inference techniques that have gained significant attention in the machine learning community. Next, we provide a careful review of this second class of algorithms, named *variational inference*.

Variational Methods. In variational inference [28, 51, 7], or VI, we introduce a family of tractable distributions $\mathcal{Q}$ over the latent variables (such that we can easily sample and score). We wish to find the member $q_{\psi^*(\boldsymbol{x})} \in \mathcal{Q}$ that minimizes the Kullback-Leibler (KL) divergence between itself and the true posterior:

$$q_{\psi^*(\boldsymbol{x})}(\boldsymbol{z}) = \arg\min_{q_{\psi(\boldsymbol{x})}} D_{\mathrm{KL}}(q_{\psi(\boldsymbol{x})}(\boldsymbol{z})||p(\boldsymbol{z}|\boldsymbol{x})) \qquad (8)$$

where $\psi(\boldsymbol{x})$ are parameters that define each distribution. For example, $\psi(\boldsymbol{x})$ would be the mean and scale for a Gaussian distribution. Since the "best" approximate posterior $q_{\psi^*(\boldsymbol{x})}$ depends on the observed variables, its parameters have $\boldsymbol{x}$ as a dependent variable. To be clear, there is one approximate posterior for every possible value of the observed variables.

Frequently, we need to do inference for many different values of the observed variables $\boldsymbol{x}$. Let $p_{\mathcal{D}}(\boldsymbol{x})$ be an empirical distribution over the observed variables, which is equivalent to the marginal $p(\boldsymbol{x})$ if the generative model is correctly specified. Then, the average quality of the variational approximations is measured by

$$\mathbb{E}_{p_{\mathcal{D}}(\boldsymbol{x})}\left[\max_{\psi(\boldsymbol{x})} \mathbb{E}_{q_{\psi(\boldsymbol{x})}(\boldsymbol{z})}\left[\log \frac{p(\boldsymbol{x},\boldsymbol{z})}{q_{\psi(\boldsymbol{x})}(\boldsymbol{z})}\right]\right] \qquad (9)$$

In practice, $p_{\mathcal{D}}(\boldsymbol{x})$ is unknown but assume access to a training dataset $\mathcal{D}$ of examples i.i.d. sampled from $p_{\mathcal{D}}(\boldsymbol{x})$. Since we never need to score, this is sufficient to evaluate Eq. 9. We have essentially converted inference to an optimization problem of picking the best parameters to maximize an objective function. We introduce a few additional details of VI that will be useful in the Methods.

Amortization. As presented in Eq. 9, we must learn an approximate posterior for each $\boldsymbol{x} \in \mathcal{D}$. For a large dataset $\mathcal{D}$, this can quickly grow to be unwieldly. One such solution to this scalability problem is *amortization* [19], which reframes the per-observation optimization problem as a supervised regression task. Consider learning a single deterministic mapping $f_\phi : \mathcal{X} \to \mathcal{Q}$ to predict $\psi^*(x)$ or equivalently $q_{\psi^*(\boldsymbol{x})} \in \mathcal{Q}$ as a function of the observation $\boldsymbol{x}$. Often, we choose f_ϕ to be a conditional distribution, denoted by $q_\phi(\boldsymbol{z}|\boldsymbol{x}) = f_\phi(\boldsymbol{x})(\boldsymbol{z})$.

The benefit of amortization is a large reduction in computational cost: the number of parameters is vastly smaller than learning a per-observation posterior. Additionally, if we manage to learn a good regressor, then the amortized approximate posterior $q_\phi(\boldsymbol{z}|\boldsymbol{x})$ could generalize to new observations $\boldsymbol{x} \notin \mathcal{D}$ unseen in training. This strength has made this approach popular with modern latent variable models like Variational Autoencoders [31].

The drawback of this approach is that it introduces an *amortization gap*, since we are technically using a less flexible inference model. Instead of Eq. 9, we instead get the following optimization problem,

$$\max_\phi \mathbb{E}_{p_{\mathcal{D}}(\boldsymbol{x})}\left[\mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x})}\left[\log \frac{p(\boldsymbol{x},\boldsymbol{z})}{q_\phi(\boldsymbol{z}|\boldsymbol{x})}\right]\right] \qquad (10)$$

The gap comes from the cost of sharing the variational parameters over all examples in the dataset (mathematically represented as pulling the max out of the outer expectation) as opposed to individual parameters per example.

Model Learning. So far we have assumed a fixed generative model $p(\boldsymbol{x},\boldsymbol{z})$. However, often we can only specify a family of possible models $p_\theta(\boldsymbol{x}|\boldsymbol{z})$ parameterized by θ. The symmetric challenge (to approximate inference) is to choose θ whose model best explains the evidence. Naturally, we do so by maximizing the log marginal likelihood of the data

$$\mathbb{E}_{p_{\mathcal{D}}(\boldsymbol{x})}[\log p_\theta(\boldsymbol{x})] = \mathbb{E}_{p_{\mathcal{D}}(\boldsymbol{x})}\left[\log \int_{\boldsymbol{z}} p_\theta(\boldsymbol{x},\boldsymbol{z})d\boldsymbol{z}\right] \qquad (11)$$

Using what we did in Eq. 10, we derive the Evidence Lower Bound (ELBO) with $q_\phi(\boldsymbol{z}|\boldsymbol{x})$ as our inference model

$$\mathbb{E}_{p_{\mathcal{D}}(\boldsymbol{x})}[\log p_\theta(\boldsymbol{x})] \geq \mathbb{E}_{p_{\mathcal{D}}(\boldsymbol{x})}\left[\mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x})}\left[\log \frac{p_\theta(\boldsymbol{x},\boldsymbol{z})}{q_\phi(\boldsymbol{z}|\boldsymbol{x})}\right]\right] \quad (12)$$

We can jointly optimize ϕ and θ to maximize the ELBO. In practice, we often parameterize $p_\theta(\boldsymbol{x}|\boldsymbol{z})$ and $q_\phi(\boldsymbol{z}|\boldsymbol{x})$ with deep neural networks, as is common with the VAE [31].

Stochastic Gradient Estimation. The gradient of the ELBO (Eq. 12) with respect to ϕ and θ are:

$$\nabla_\theta \mathrm{ELBO} = \mathbb{E}_{p_{\mathcal{D}}(\boldsymbol{x})}[\mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x})}[\nabla_\theta \log p_\theta(\boldsymbol{x},\boldsymbol{z})]] \qquad (13)$$

$$\nabla_\phi \mathrm{ELBO} = \mathbb{E}_{p_{\mathcal{D}}(\boldsymbol{x})}[\nabla_\phi \mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x})}[\log p_\theta(\boldsymbol{x},\boldsymbol{z})]] \qquad (14)$$

Eq. 13 can be estimated using Monte Carlo samples. However, as it stands, Eq. 14 is difficult to estimate as we cannot distribute the gradient inside the inner expectation. For certain families $\mathcal{Q}$, we can use a reparameterization trick.

Reparameterization Estimators. Reparameterization is the technique of removing sampling from the gradient computation graph [31, 45]. In particular, if we can reduce sampling $\boldsymbol{z} \sim q_\phi(\boldsymbol{z}|\boldsymbol{x})$ to sampling from a parameter-free distribution $\epsilon \sim p(\epsilon)$ plus a deterministic function application, $\boldsymbol{z} = g_\phi(\epsilon)$, then we may rewrite Eq. 14 as

$$\nabla_\phi \text{ELBO} = \mathbb{E}_{p_{\mathcal{D}(\boldsymbol{x})}}[\mathbb{E}_{p(\epsilon)}[\nabla_{\boldsymbol{z}} \log \frac{p_\theta(\boldsymbol{x}, \boldsymbol{z}(\epsilon))}{q_\phi(\boldsymbol{z}(\epsilon)|\boldsymbol{s})} \nabla_\phi g_\phi(\epsilon)]] \quad (15)$$

which now can be estimated efficiently by Monte Carlo (the gradient is inside the expectation). A benefit of reparameterization over alternative estimators (e.g. score estimator [36] or REINFORCE [53]) is lower variance while remaining unbiased. A common example is if $q_\phi(\boldsymbol{z}|\boldsymbol{x})$ is Gaussian $\mathcal{N}(\mu, \sigma^2)$ and we choose $p(\epsilon)$ to be $\mathcal{N}(0, 1)$, then $g(\epsilon) = \epsilon * \sigma + \mu$.

Normalizing Flows. A critique of VI is the need to specify the family of distributions $\mathcal{Q}$ to be reparameterizable, of which the most popular is the Normal family. If for example, the true posterior is multi-modal, our best Normal approximation will only capture one mode. To access a richer family of distributions, we consider transforming samples from our Normal distribution to a more complex one.

Let $g : \mathbb{R}^d \rightarrow \mathbb{R}^d$ be a continuous and invertible mapping. Given $\boldsymbol{z}^{(0)} \sim q_\phi(\boldsymbol{z}^{(0)}|\boldsymbol{x})$, define $\boldsymbol{z}^{(1)} = g(\boldsymbol{z}^{(0)})$. Then, $q_\phi(\boldsymbol{z}^{(1)}|\boldsymbol{x})$ has the following probability distribution:

$$q_\phi(\boldsymbol{z}^{(1)}|\boldsymbol{x}) = q_\phi(\boldsymbol{z}^{(0)}|\boldsymbol{x}) \left| \det \frac{\partial g^{-1}}{\partial \boldsymbol{z}^{(0)}} \right| = q_\phi(\boldsymbol{z}^{(0)}|\boldsymbol{x}) \left| \det \frac{\partial g}{\partial \boldsymbol{z}^{(0)}} \right|^{-1} \quad (16)$$

where $\left| \det \frac{\partial \cdot}{\partial \cdot} \right|$ is the Jacobian determinant. K such mappings can be composed together: $\boldsymbol{z}^{(K)} = g^{(K)} \circ \cdots \circ g^{(1)}(\boldsymbol{z}^{(0)})$. In this case, the log likelihood of a transformed sample is

$$\log q_\phi^{(K)}(\boldsymbol{z}^{(K)}|\boldsymbol{x}) = \log q_\phi^{(0)}(\boldsymbol{z}^{(0)}|\boldsymbol{x}) - \sum_{k=1}^{K} \log \left| \det \frac{\partial g^{(k)}}{\partial \boldsymbol{z}^{(k-1)}} \right|$$

Then, $\boldsymbol{z}^{(K)} \sim q^{(K)}(\boldsymbol{z}^{(K)}|\boldsymbol{x})$, an arbitrarily complex (potentially multi-modal) distribution. Computing the Jacobian determinant is intractable for most functions g. Several works [44, 49, 4] have posed tractable families of functions for g in which the Jacobian determinant is either 1 or a computable sparse matrix. In our discussion, we will explore planar flows [44] with VI.

$$g(\boldsymbol{z}) = \boldsymbol{z} + \boldsymbol{u}h(\boldsymbol{w}^T \boldsymbol{z} + b) \quad (17)$$

$$\left| \det \frac{\partial g}{\partial \boldsymbol{z}} \right| = \left| 1 + \boldsymbol{u}^T h'(\boldsymbol{w}^T \boldsymbol{z} + b)\boldsymbol{w} \right| \quad (18)$$

where $\boldsymbol{u}, \boldsymbol{w} \in \mathbb{R}^d$ and $b \in \mathbb{R}$ are learnable parameters; h is an elementwise nonlinearity and h' its derivative.

3. METHODS

Having viewed the major principles of variational inference, we are ready to present variational inference for IRT, which will adapt many of the same ideas above. We assume that we are given an empirical dataset of binary responses from N persons to M items. This dataset may or may not contain missing entries. We use the index i to enumerate different persons and the index j to enumerate different
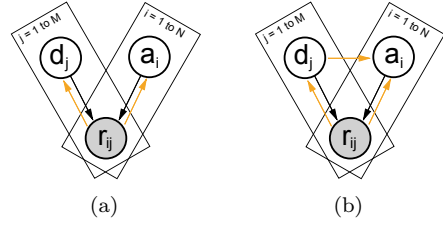

Figure 2: Graphical models with reverse arrows showing inference for VIRTU$^\perp$ (a) and VIRTU (b).

items. For example, $\boldsymbol{a}_i = (a_i^{(1)}, a_i^{(2)}, \ldots)$ refers to the (possibly multidimensional) ability for person i. Likewise, we will use the notation $\boldsymbol{d}_j$ to represent all the characteristics of item j. For instance, for a 3PL unidimensional IRT model, we have $\boldsymbol{d}_j = (d_j, k_j, g_j)$. First, we derive two novel lower bound on the log joint distribution for a single person $\log p_\theta(\boldsymbol{r}_{i,1:M}, \boldsymbol{a}_i, \boldsymbol{d}_{1:M})$ where $\boldsymbol{r}_{i,1:M}$ are the i-th person's responses to all M questions. The generative model, unless otherwise specified, $p(\boldsymbol{r}_{i,1:M}|\boldsymbol{a}_i, \boldsymbol{d}_{1:M})$ is an IRT model and thus includes no additional parameters.

THEOREM 3.1. *Let $\boldsymbol{a}_i$ be the ability for person $i \in [1, N]$ and $\boldsymbol{d}_j$ be the characteristics for item $j \in [1, M]$. We use the shorthand notation $\boldsymbol{d}_{1:M} = (\boldsymbol{d}_1, \ldots, \boldsymbol{d}_M)$. Let $r_{i,j}$ be the binary response for item j by person i. We write $\boldsymbol{r}_{i,1:M} = (r_{i,1}, \ldots r_{i,M})$. If we defined the following terms,*

$$\mathcal{L}_{recon} = \mathbb{E}_{q_\phi(\boldsymbol{a}_i, \boldsymbol{d}_{1:M}|\boldsymbol{r}_{i,1:M})} [\log p(\boldsymbol{r}_{i,1:M}|\boldsymbol{a}_i, \boldsymbol{d}_{1:M})]$$
$$D_{ability} = D_{\text{KL}}(q_\phi(\boldsymbol{a}_i|\boldsymbol{r}_{i,1:M})||p(\boldsymbol{a}_i))$$
$$D_{item} = D_{\text{KL}}(q_\phi(\boldsymbol{d}_{1:M}|\boldsymbol{r}_{i,1:M})||p(\boldsymbol{d}_{1:M}))$$

then the following inequality is true:

$$\log p(\boldsymbol{r}_{i,1:M}) \geq \mathcal{L}_{recon} + D_{ability} + D_{item}$$

where the joint distribution $q_\phi(\boldsymbol{a}_i, \boldsymbol{d}_{1:M}|\boldsymbol{r}_{i,1:M})$ factors as $q_\phi(\boldsymbol{a}_i|\boldsymbol{r}_{i,1:M})q_\phi(\boldsymbol{d}_{1:M}|\boldsymbol{r}_{i,1:M})$.

PROOF. See Sec. A for proof. $\square$

COROLLARY 3.1.1. *Using the definitions in Thm. 3.1, if the variational distribution factors as follows:*

$$q_\phi(\boldsymbol{a}_i, \boldsymbol{d}_{1:M}|\boldsymbol{r}_{i,1:M}) = q_\phi(\boldsymbol{a}_i|\boldsymbol{d}_{1:M}, \boldsymbol{r}_{i,1:M})q_\phi(\boldsymbol{d}_{1:M}|\boldsymbol{r}_{i,1:M})$$

Then the following lower bound holds

$$\log p(\boldsymbol{r}_{i,1:M}) \geq \mathcal{L}_{recon} + \mathbb{E}_{q_\phi(\boldsymbol{d}_{1:M}|\boldsymbol{r}_{i,1:M})}[D_{ability}] + D_{item}$$

where $D_{ability} = D_{\text{KL}}(q_\phi(\boldsymbol{a}_i|\boldsymbol{d}_{1:M}, \boldsymbol{r}_{i,1:M})||p(\boldsymbol{u}_i))$.

PROOF. Near identical to proof of Thm. 3.1. See Sec. A. $\square$

The primary difference between Thm. 3.1 and Coro. 3.1.1 is how the variational joint distribution $q(\boldsymbol{a}_i, \boldsymbol{d}_{1:M}|\boldsymbol{r}_{i,1:M})$ factors. As practitioners, we are free to choose the form of q. In Thm. 3.1, we assume $\boldsymbol{a}_i$ and $\boldsymbol{d}_{1:M}$ are independent conditioned on $\boldsymbol{r}_{i,1:M}$ whereas we do not make that assumption in Coro. 3.1.1. In the supplement, we provide evidence that the independence assumption is too strong and decreases the

quality of inference dramatically. We call the lower bound in Coro. 3.1.1 VI. We will call the lower bound in Thm. 3.1 $\text{VI}^{\perp}$ to make its assumptions clear. Fig. 2 shows graphical models distinguishing the two. Mathematically, we can write the VI objective as:

$$\text{VI} \triangleq \mathbb{E}_{p_{\mathcal{D}}(\boldsymbol{r}_{i,1:M})}[\mathcal{L}_{\text{recon}} + D_{\text{ability}} + D_{\text{item}}] \qquad (19)$$

where $p_{\mathcal{D}}(\boldsymbol{r}_{i,1:M})$ is an empirical distribution over person responses to the M items. As mentioned, we are only given a dataset $\mathcal{D} = \{\boldsymbol{r}_{i,1:M}\}_{i=1}^{N}$ of N persons responding to M items. However, such a dataset is sufficient for computing the VI objective. The gradients of the VI loss with respect to ϕ can be estimated with Monte Carlo samples as:

$$\nabla_{\phi}\text{VI} = \mathbb{E}_{p_{\mathcal{D}}}[\nabla_{\phi}\mathbb{E}_{q_{\phi}(\boldsymbol{d}_{1:M}|\boldsymbol{r}_{i,1:M})}[D_{\text{ability}}] + \nabla_{\phi}D_{\text{item}}] \quad (20)$$

Like in Eq. 15, we may wish to move the gradient inside the KL divergences by reparameterization to reduce variance. To allow easy reparameterization, we define all variational distributions $q_{\phi}(\cdot|\cdot)$ as Normal distributions with diagonal covariance. In practice, we find that estimating the gradient in Eq. 20 with a single simple is sufficient. With this setup, the VI loss can be optimized using stochastic gradient descent to learn an amortized inference model that maximizes the marginal probability of observed data.

The posterior $q_{\phi}(\boldsymbol{a}_i|\boldsymbol{d}_{1:M}, \boldsymbol{r}_{i,1:M})$ needs to be robust to missing data as often not every person answers every question. To do so, we explore the following design:

$$q_{\phi}(\boldsymbol{a}_i|\boldsymbol{d}_{1:M}, \boldsymbol{r}_{i,1:M}) = \prod_{j=1}^{M} q_{\phi}(\boldsymbol{a}_i|\boldsymbol{d}_j, \boldsymbol{r}_{i,j}) \qquad (21)$$

If we assume each component $q_{\phi}(\boldsymbol{a}_i|\boldsymbol{d}_j, \boldsymbol{r}_{i,j})$ is Gaussian, then $q_{\phi}(\boldsymbol{a}_i|\boldsymbol{d}_{1:M}, \boldsymbol{r}_{i,1:M})$ is Gaussian as well, being a Product-Of-Experts [25, 54]. If item j is missing, we replace its term in the product with the prior, $p(\boldsymbol{a}_i)$ representing no added information. We found this design to outperform averaging over non-missing entries: $\frac{1}{M}\sum_{j=1}^{M} q_{\phi}(\boldsymbol{a}_i|\boldsymbol{d}_j, \boldsymbol{r}_{i,j})$.

4. RELATED WORK

We have already covered a variety of methods for parameter estimation in IRT such as MLE, EM, and MCMC. The benefits and drawbacks of these methods are well-documented [32], so we need not discuss them here. Instead, we focus specifically on methods that utilize deep neural networks or variational inference to estimate IRT parameters.

We are aware of two approaches that incorporate deep neural networks into Item Response Theory: Deep-IRT [55] and DIRT [11]. Deep-IRT is a modification of the Dynamic Key-Value Memory Network (DKVMN) [56] that treats data as longitudinal, processing items one-at-a-time using a recurrent architecture. Deep-IRT produces point estimates of ability and item difficulty at each time step, which are then passed into a 1PL IRT function to produce the probability of answering the item correctly. The main difference between DIRT and Deep-IRT is the choice of neural network: instead of the DKVMN, DIRT uses an LSTM with attention [50]. In our experiments, we compare our approach to Deep-IRT and find that we outperform it by up to 30% on the accuracy of missing response imputation.

While variational inference has been suggested as a promising

alternative to other inference approaches for IRT [32], there has been surprisingly little work in this area. In an exploration of Bayesian prior choice for IRT estimation, Natesan et al. [37] posed a variational approximation to the posterior:

$$p(\boldsymbol{a}_i, \boldsymbol{d}_j|r_{i,j}) \approx q(\boldsymbol{a}_i, \boldsymbol{d}_j) = q(\boldsymbol{a}_i)q(\boldsymbol{d}_j) \qquad (22)$$

We note a few differences between Eq. 22 and our variational approach: First, we amortize over persons, meaning our variational posterior is dependent on the response of person i to M items. This is critical to good performance as it prevents overfitting to the observed dataset (one can imagine that using Eq. 22 is problematic for values of $\boldsymbol{a}_i$ or $\boldsymbol{d}_j$ that were not seen in training) and more importantly, Eq. 22 requires fitting a new posterior for each person i and item j. This would result in marginal speed ups (if any) compared to more expensive inference techniques. Second, Eq. 22 has the same form as $\text{VI}^{\perp}$, which we show in the supplement to dramatically degrade performance. The insight to decompose $q(\boldsymbol{a}_i, \boldsymbol{d}_{1:M}|\boldsymbol{r}_{i,1:M}) = q(\boldsymbol{a}_i|\boldsymbol{d}_{1:M}, \boldsymbol{r}_{i,1:M})q(\boldsymbol{d}_{1:M}|\boldsymbol{r}_{i,1:M})$ instead of assuming independence is both theoretically [52] and experimentally supported.

Curi et al. [12] used a Variational Autoencoder (VAE) to estimate IRT parameters in a 28-question synthetic dataset. However, this approach only modeled ability as a latent variable, ignoring the full joint posterior incorporating item features. In this work, our analogue to the VAE builds on the IRT graphical model, incorporating both ability and item characteristics in a principled manner. This could explain why Curi et. al. report the VAE requiring substantially more data to recover the true parameters when compared to MCMC whereas we find comparable performance of our variational lower bound to MCMC.

5. DATASETS

We explore one toy dataset to build intuiton and five large scale applications of IRT to measure ability in either language acquision or educational assessments.

Table 1: Dataset Statistics

	# PERSONS	# ITEMS	MISSING DATA?
CRITLANGACQ	669498	95	N
WORDBANK	5520	797	N
DUOLINGO	2587	2125	Y
GRADESCOPE	1254	98	Y
PISA	519334	183	Y

Synthetic IRT. To sanity check that VI performs as well as other inference techniques, we synthetically generate a dataset of responses using a 2PL IRT model: sample $\boldsymbol{a}_i \sim p(\boldsymbol{a}_i)$, $\boldsymbol{d}_j \sim p(\boldsymbol{d}_j)$ where the prior distributions are spherical Gaussian (recall that $\boldsymbol{d}_j$ contains all item characteristics). Given ability and item characteristics, IRT-2PL determines a Bernoulli distribution over responses to item j by person i. We sample once from this Bernoulli distribution to "generate" an observation. We vary N and M to stress test VI.

Second Language Acquisition. This dataset contains native and non-native English speakers answering questions to

(a) Synthetic: Correlation with True Ability (b) Synthetic: Missing Data Imputation

(c) Real World: Missing Data (d) Real World: Time Cost (e) Nonlinear: Missing Data (f) Nonlinear: Log Likelihoods
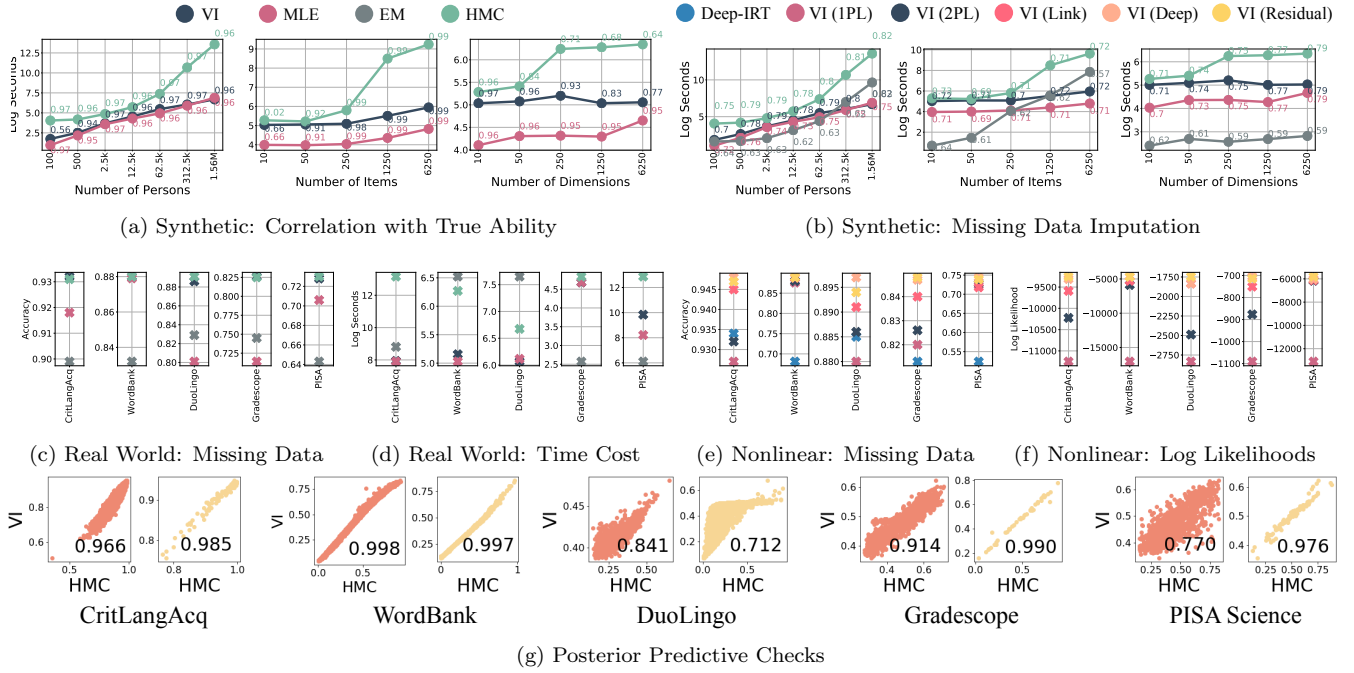
(g) Posterior Predictive Checks

Figure 3: (a) Correlation of inferred ability with true ability (known in synthetic setting) for VI, MLE, and HMC. The y-axis reports the log seconds (e.g. 10 log seconds is 6.1 hours). (b) Missing data imputation (shown as printed numbers) for VI, MLE, EM, and HMC in a synthetic dataset. (c) Missing data imputation for real world data as well as (d) cost in log seconds. (e) Missing data imputation and (f)log likelihoods of training data for various nonlinear IRT models trained with VI. (e) Samples statistics from the predictive posterior defined using HMC and VI. Subfigures in red show the average number of items answered correctly for each person. Subigures in yellow show the average number of persons who answered each item correctly.

a grammar quiz[2], which upon completion would return a prediction of the user's native language. Using social media, over half a million users of varying ages and demographics completed the quiz. Quiz questions often contain both visual and linguistic components. For instance, a quiz question could ask the user to "choose the image where the dog is chased by the cat" and present two images of animals where only one of image agrees with the caption. Every response is thus binary, marked as correct or incorrect. In total, there are 669,498 persons with 95 items and no missing data. The creators of this dataset use it to study the presence of absence of a "critical period" for second language acquisition [22]. We will refer to this dataset as CRITLANGACQ.

WordBank: Vocabulary Development. The MacArthur-Bates Communicative Development Inventories (CDIs) are a widely used metric for early language acquisition in children, testing concepts in vocabulary comprehension, production, gestures, and grammar. The WordBank [17] database archives several independently collected CDI datasets across several languages and research laboratories[3]. The database consists of a matrix of persons against vocabulary words where the (i, j) entry is 1 if person i has knowledge of word j when tested and 0 otherwise. If child i was never quizzed on word j, we consider the entry missing. For simplicity, we ignore difference in children demographics and native

languages (each child is treated as an independent entry). In total, there are 5,520 children responding to 797 items.

DuoLingo: App-Based Language Learning. We examine the 2018 DuoLingo Shared Task on Second Language Acquisition Modeling (SLAM)[4] [46]. This dataset contains anonymized user data from the popular education application, DuoLingo. In the application, users must choose the correct vocabulary word among a list of distractor words. We focus on subset of native English speakers learning Spanish and only consider lesson sessions. Each user has a timeseries of responses to a list of vocabulary words, each of which is shown several times. We repurpose this dataset for IRT: the goal being to infer the user's language proficiency from his or her errors. As such, we average over all times a user has seen each vocabulary item. For example, if the user was presented "habla" 10 times and correctly identified the word 5 times, he or she would be given a response score of 0.5. We then round it to be 0 or 1. We will revisit a continuous version in the discussion. After processing, we are left with 2587 users and 2125 vocabulary words with missing data as users frequently drop out. We ignore user and syntax features.

Gradescope: Course Exam Data. Gradescope [47] is a popular course application that assists teachers in grading student assignments. This dataset contains in total 105,218 reponses from 6,607 assignments in 2,748 courses

---

and 139 schools. All assignments are instructor-uploaded, fixed-template assignments, with at least 3 questions, with the majority being examinations. We focus on course 102576, randomly chosen from the courses with the most students. In the supplement, we provide results for 5 other courses from Gradescope (Table 12). We remove students who did not respond to any questions. We binarize the response scores by rounding up partial credit. In total, there are 1254 students with 98 items with missing entries for course 102576.

PISA 2015: International Assessment. The Programme for International Student Assessment (PISA) is an international exam that measures 15-year-old students' reading, mathematics, and science literacy every three years. It is run by the Organization for Economic Cooperation and Development (OECD). The OECD released anonymized data from PISA '15 for students from 80 countries and education systems[5]. We focus on the science component. Using IRT to access student performance is part of the pipeline the OECD uses to compute holistic literacy scores for different countries. As part of our processing, we binarize responses, rounding any partial credit to 1. In total, there are 519,334 students and 183 questions. Not every student answers every question as many versions of the computer exam exist.

6. FAST AND ACCURATE INFERENCE
We will show that VI is as accurate as HMC but nearly as fast as MLE/EM, making Bayesian IRT a realistic, even optimal, option for modern applications.

6.1 Evaluation
We compare VI to HMC, EM[6], and MLE using unidimensional IRT-2PL. For evaluation, we measure the wall-clock cost of each inference algorithm. However, speed only matters assuming good performance. To measure the latter, we use various approaches, the most popular being accuracy of imputing missing data by artificially holding out 10% of the responses. We use the inferred ability and item characteristics to generate responses, thereby populating missing entries. This is a very general metric and can be used in almost all cases. For the synthetic dataset specifically, because we know the true ability, we can measure the correlation between the it and the inferred ability under each algorithm (with the exception of EM). A correlation of 1.0 would indicate perfect inference. For real world data, we cannot do the same so we settle for a proxy. To compare HMC and VI in particular, we use posterior predictive checks. Recall that the predictive posterior is defined as:

$$p(\tilde{\boldsymbol{r}}_{i,1:M}|\boldsymbol{r}_{i,1:M}) = \mathbb{E}_{p(\boldsymbol{a}_i, \boldsymbol{d}_{1:M}|\boldsymbol{r}_{i,1:M})}[p(\tilde{\boldsymbol{r}}_{i,1:M}|\boldsymbol{a}_i, \boldsymbol{d}_{1:M})]$$

With HMC, we are given samples from the posterior. With VI, replace $p(\boldsymbol{a}_i, \boldsymbol{d}_{1:M}|\boldsymbol{r}_{i,1:M})$ with $q_\phi(\boldsymbol{a}_i, \boldsymbol{d}_{1:M}|\boldsymbol{r}_{i,1:M})$. Given K samples $\{\tilde{\boldsymbol{r}}_{i,1:M}\}_{k=1}^K$ for all such persons $i = 1$ to N, we measure (1) the average number of items answered correctly per person and (2) the average number of persons who answered item j correctly. This practice is common in statistical psychology and cognitive science [48].

6.2 Results and Analysis
With synthetic experiments we are free to vary N and M to extremes to stress test the inference algorithms: first, we range from 100 to 1.5 million persons and 10 to 6k items ; second, we vary the number of dimensions from 1 to 5. Fig. 3(a,b) plot the show the correlation and missing data accuracy of VI, MLE, HMC, and EM. In both subplots, the y-axis represents log seconds: a small difference in log seconds is a dramatic difference in wall clock time. For example, 5 log seconds is about 3 minutes, 10 log seconds is about 6 hours, and 15 log seconds is about 900 hours.

We find several promising patterns. First, the computational cost of VI is only slightly higher than MLE (the backwards pass of VI is more expensive each step), both of which are magnitudes lower than HMC. For instance, with 1.56 million persons, HMC takes 217 hours whereas VI takes 800 seconds. Similarly with 6250 times, HMC takes 4.3 hours whereas VI takes 385 seconds. Although EM is known for being cheap, it does not scale as well as VI for large datasets. For example, with 1.56M persons, EM takes 4.3 hours; with 6250 items, EM takes 43 minutes. With low to medium sized datasets, EM remains the fastest. However, this speed comes at a price. Properly marginalizing out ability is difficult to do numerically, especially as the number of dimensions increases. In all settings, we find the EM's accuracy of imputing missing data to be lower by 10%. As the dimensionality increases, the performance of EM continues to drop, approaching 59% in 5d whereas VI, MLE, and HMC have an accuracy of 79%. HMC and VI are almost identical in performance in both correlation and missing data accuracy, which suggests that here, the variational posterior is a good approximation of the true one. The performance of MLE is close to VI and HMC although consistently lower by 0.5 to 1% in accuracy. This is likely due to MLE overfitting to the observed responses as it is a greedy algorithm.

We also find the limitations of VI. Because we chose to amortize inference over persons, VI will not perform very well given very few individuals. For instance,with 100 people, VI only achieves 0.56 correlation with the true ability where MLE and HMC do much better. However, amortized VI does not require too large of a dataset: with 500 people, we find equal performance between the inference algorithms. Under scarce data, we would recommend not amortizing VI (Eq. 8).

Fig. 3(c) show the accuracy of missing data imputation on real world datasets. We find the same patterns as we did in the synthetic experiments. Running HMC on CritLangAcq or PISA takes roughly 120 hours whereas VI takes 50 minutes for CritLangAcq and 5 hours for PISA, the latter being more expensive because of additional computation required for missing data. In comparison, EM is sometimes faster than VI (e.g. Gradescope, PISA) and sometimes slower. With respect to accuracy, VI and HMC are again identical, outperforming EM by up to 8% in missing data imputation. Interestingly, we find the "overfitting" of MLE to be more pronounced here. If we focus on the DuoLingo and Gradescope, the two datasets with pre-existing large portions of missing values, MLE is surpassed by EM, with VI achieving an accuracy 10% higher. Finally, Fig. 3(e) shows posterior predictive checks between VI and HMC. We find that the two algorithms strongly agree about the average number of correct persons

and items in all datasets. The lowest agreement occurs with DuoLingo: since the number of items is greater than the number of persons, there is naturally more uncertainty.

In summary, through careful experimentation on synthetic and real world data, we find VI to be fast and accurate, matching HMC in performance and MLE in speed.

7. DEEP ITEM RESPONSE THEORY

IRT is a surprisingly good model for item responses despite its simplicity. However, we may not believe the true idiosyncracies of every individual's response is captured by a logistic function. If we were able to use nonlinear functions, we may be able to more faithfully capture the density of the data. We test this intuition here by merging item response theory with the modern machinery of deep learning.

7.1 Nonlinear Generalizations of IRT

We have assumed that $p(\boldsymbol{r}_{i,1:M}|\boldsymbol{a}_i, \boldsymbol{d}_{1:M})$ is a deterministic IRT model defining a Bernoulli distribution for the response to each item. However, nothing limits us from replacing it with a more expressive generative model. With the speed and versatility of VI, we can support much more heavily parameterized models. Here, we consider three different alternatives with varying levels of expressivity that help define a class of more powerful nonlinear IRT.

Learning a Linking Function. We replace the logistic function in standard IRT with a nonlinear linking function. As such, it preserves the linear relationships between items and persons. We call this VI (Link). For person i and item j, the 2PL link generative model is:

$$p(r_{ij}|\boldsymbol{a}_i, \boldsymbol{d}_j) = f_\theta(-\boldsymbol{a}_i^T \boldsymbol{k}_j - d_j) \qquad (23)$$

where f_θ is a nonlinear transformation followed by a sigmoid to constrain the output to be between $[0, 1]$.

Learning a Neural Network. Here, we no longer preserve the linear relationships between items and persons and instead feed the ability and item characteristics directly into a neural network, which will combine the inputs nonlinearly. We call this version VI (Deep). For person i and item j, the deep generative model is:

$$p(r_{ij}|\boldsymbol{a}_i, \boldsymbol{d}_j) = f_\theta(\boldsymbol{a}_i, \boldsymbol{d}_j) \qquad (24)$$

where again f_θ includes a Sigmoid function at the very end to preserve the correct output signatures. This is an even more expressive model than VI (Link).

Learning a Residual Correction. Although clearly a powerful model, we might fear that VI (Deep) becomes too uninterpretable. So, for the third and final nonlinear model, we use the standard IRT but add a nonlinear residual component that can correct for any inaccuracies. We call this version VI (Residual). For person i and item j, the 2PL residual generative model is:

$$p(r_{ij}|\boldsymbol{a}_i, \boldsymbol{k}_j, d_j) = \frac{1}{1 + e^{-\boldsymbol{a}_i^T \boldsymbol{k}_j - d_j + f_\theta(\boldsymbol{a}_i, \boldsymbol{k}_j, d_j)}} \qquad (25)$$

During optimization, we initialize the weights of the residual network to 0, thus ensuring its initial output is 0. This encourages the model to stay close to IRT, using the residual only when necessary.

Since $p_\theta(\boldsymbol{r}_{i,1:M}|\boldsymbol{a}_i, \boldsymbol{d}_{1:M})$ now has free parameters, we can jointly optimize it with the variational parameters ϕ as in Sec. 2.3. The gradient of the VI loss with respect to θ is straightforward to estimate by sampling without any need for reparameterization. Precisely, $\nabla_\theta \text{VI}$ is equal to:

$$\mathbb{E}_{p_\mathcal{D}(\boldsymbol{r}_{i,1:M})}[\mathbb{E}_{q_\phi(\boldsymbol{a}_i, \boldsymbol{d}_{1:M}|\boldsymbol{r}_{i,1:M})}[\nabla_\theta \log p_\theta(\boldsymbol{r}_{i,1:M}|\boldsymbol{a}_i, \boldsymbol{d}_{1:M})]]$$

Again, a single sample should suffice.

7.2 Evaluation

A better generative model should better explain the data density. In other words, observations in the training dataset should have low surprisal to the model. To capture this intuition, we estimate the log marginal likelihood $\log p(r_{1:N,1:M})$ of the training dataset. A higher number (closer to 0) is better. Because VI is amortized, we can evaluate the quality of inference on unseen data points. We held-out a portion of the generated dataset for a separate test dataset. With M questions, there are 2^M unique combinations for responses. We designed the test dataset to contain combinations unseen in training. An undesirable model might overfit to the training combinations, thereby getting poor log likelihoods on the test set. For a single person, the log marginal likelihood of his or her M responses can be computed as:

$$\log \mathbb{E}_{q_\phi(\boldsymbol{a}_i, \boldsymbol{d}_{1:M}|\boldsymbol{r}_{i,1:M})} \left[\frac{p_\theta(\boldsymbol{r}_{i,1:M}, \boldsymbol{a}_i, \boldsymbol{d}_{1:M})}{q_\phi(\boldsymbol{a}_i, \boldsymbol{d}_{1:M}|\boldsymbol{r}_{i,1:M})} \right] \qquad (26)$$

We use 1000 samples to estimate Eq. 26.

Additionally, we measure accuracy on missing data imputation as we did for the speed experiments above. A more powerful generative model that is more descriptive of the data should be better fill in missing values.

7.3 Results and Analysis

Fig. 3(d) show two plots: the left compares the accuracy of imputing missing data; the right compares the log likelihoods of observed data. We find a consistent trend: the more powerful generative models achieve a higher log likelihood (closer to 0) and a higher accuracy. This is perhaps unsurprising as a higher log likelihood indicates that the model better describes the data density, which would be make missing data imputation easier. In particular, we find very large increases in log likelihood moving from IRT to Link, spanning 100 to 500 log points depending on the dataset. Further, from Link to Deep and Residual, we find another increase of 100 to 200 log points. In some cases, we find Residual to outperform Deep despite that the two are equally parameterized, suggesting that initialization with IRT finds better local optima. These gains in log likelihood translate to a consistent 1 to 2% increase in accuracy for Deep/Residual over IRT.

We also compare our various deep generative IRT models with purely deep learning approaches, namely Deep-IRT [56]. Unlike traditional IRT models, Deep-IRT was built for knowledge tracing and assumed sequential responses. To make our datasets amenable to Deep-IRT, we assume an ordering of

responses from $j = 1$ to $j = M$. As shown in Fig. 3(d), our models outperform Deep-IRT in all 5 datasets by as much as 30% in missing data imputation (see WordBank).

8. DISCUSSION
We explore some features of VI and possible extensions that may be interesting for several real world uses.

8.1 Importance of Amortization
We stress the importance of amortization in our variational approach, both in terms of generalization and efficiency. Consider a dataset with 100 items for a total of 2^{100} possible responses for any given person. Even with a dataset of a million persons, almost all the possible responses remain unseen. Without amortization, we fit a variational posterior for each "unique" individual as characterized by their responses. Thus, we will only be able to do inference for the observed dataset. If we were to deploy such a system, it is quite likely that we receive a previously unseen set of responses, making inference difficult. With amortization, as we are learning a regressor from responses to posterior parameters, we may be able to generalize to a larger subset of the 2^{100} possible responses using only the million observed responses for training. In the supplement (Table 10), we show results measuring the log likelihoods of unseen responses under our inference model, in which we find similar log likelihoods to observed responses, suggesting a notion of generalization.

Table 2: Time Costs with and without Amoritzation

Dataset	Amortized	Un-Amortized
CritLangAcq	2.8k	43.2k
WordBank	176.4	657.1
DuoLingo	429.9	717.9
Gradescope	114.5	511.1
PISA	25.2k	125.8k

For efficiency, we find that amortized VI is much cheaper computationally, especially in circumstances in which the number of possible response vectors $\boldsymbol{r}_{1:M}$ is very large (e.g. 2^{95} for CritLangAcq). Table 2 compares the wall clock time (sec.) for the 5 datasets. While VI is comparable to MLE and EM, unamortized VI is 2 to 15 times more expensive.

8.2 Extending to Polytomous Responses
Thus far, we have been only working with binary responses, true or false. However, many questionnaires and examinations are not binary: responses can be multiple choice (e.g. Likert scale) or even real valued (e.g. 92% on a course test). As we have posed IRT as a generative model, we have prescribed a Bernoulli distribution over the i-th person's response to the j-th item. Yet nothing prevents us from choosing a different distribution, such as Categorical for multiple choice or Normal for real-values. The DuoLingo dataset contains partial credit, computed as a fraction of times an individual gets a word correct. A more granular treatment of these polytomous values should yield a more faithful model that can better capture the differences between people. So, we retrain DuoLingo where $p(\boldsymbol{r}_{i,1:M}|\boldsymbol{a}_i, \boldsymbol{d}_{1:M})$ is a (truncated) Normal distribution over responses with fixed variance. Table 3 show the log densities: we observe

Table 3: DuoLingo with Polytomous Responses

Inf. Alg.	Train	Test
VI (IRT)	-22038.07	-21582.03
VI (Link)	-17293.35	-16588.06
VI (Deep)	$\mathbf{-15349.84}$	$\mathbf{-14972.66}$
VI (Res.)	-15350.66	-14996.27

the same patterns among the nonlinear models. In principle, modeling continuous outputs should be more powerful as it can distinguish between a person with response 0.01 and a person with response 0.49 whereas binarized models could not. In summary, we emphasize the flexibility of our generative framework to support varying data formats.

8.3 Beyond Gaussian Posterior Assumptions
In our experiments, we have chosen the variational family $\mathcal{Q}$ to be Gaussian with diagonal covariance, mainly for use. However, for certain distributions, such a simple family may not suffice. There is a rich body of work in the machine learning community dedicated to richer posterior distributions for variational inference [44, 49, 4, 27, 30]. For demonstration, we compare log likelihoods using a Gaussian posterior versus a transformed posterior using planar flows [44]. We compose

Table 4: Effect of Richer Variational Posteriors

Dataset	VI	VI-NF
CritLangAcq	-10224.09	-9945.30
WordBank	-5882.55	-5510.67
DuoLingo	-2488.36	-2174.10
Gradescope	-876.75	-800.68
PISA	-6169.55	-5661.3

10 invertible flows to samples from both $q(\boldsymbol{d}_{1:M}|\boldsymbol{r}_{i,1:M})$ and $q(\boldsymbol{a}_i|\boldsymbol{d}_{1:M}, \boldsymbol{r}_{i,1:M})$. We can then rewrite the VI objective (Eq. 19) with K planar flows as:

$$\text{VI-NF} \triangleq \mathbb{E}_{p_{\mathcal{D}}(\boldsymbol{r}_{i,1:M})}[\mathcal{L}_{\text{recon}} + \mathcal{L}_{\text{ability}} + \mathcal{L}_{\text{item}}] \quad (27)$$

where the components are defined as:

$$\mathcal{L}_{\text{recon}} = \mathbb{E}_{q_\phi^{(0)}(\boldsymbol{a}_i^{(0)}, \boldsymbol{d}_{1:M}^{(0)}|\boldsymbol{r}_{i,1:M})}\left[\log p(\boldsymbol{r}_{i,1:M}|\boldsymbol{a}_i^{(K)}, \boldsymbol{d}_{1:M}^{(K)})\right]$$

$$\mathcal{L}_{\text{ability}} = \mathbb{E}_{q_\phi^{(0)}(\boldsymbol{a}_i^{(0)}, \boldsymbol{d}_{1:M}^{(0)}|\boldsymbol{r}_{i,1:M})}\left[\log \frac{q_\phi^{(K)}(\boldsymbol{a}_i^{(K)}|\boldsymbol{d}_{1:M}^{(k)}, \boldsymbol{r}_{i,1:M})}{p(\boldsymbol{a}_i^{(K)})}\right]$$

$$\mathcal{L}_{\text{item}} = \mathbb{E}_{q_\phi^{(0)}(\boldsymbol{d}_{1:M}^{(0)}|\boldsymbol{r}_{i,1:M})}\left[\log \frac{q_\phi^{(K)}(\boldsymbol{d}_{1:M}^{(K)}|\boldsymbol{r}_{i,1:M})}{p(\boldsymbol{d}_{1:M}^{(K)})}\right]$$

By Eq. 16 and 18, we know $q_\phi^{(K)}(\boldsymbol{a}_i^{(K)}|\boldsymbol{d}_{1:M}^{(K)}, \boldsymbol{r}_{i,1:M})$ equals $q_\phi^{(0)}(\boldsymbol{a}_i^{(0)}|\boldsymbol{d}_{1:M}^{(0)}, \boldsymbol{r}_{i,1:M}) \prod_{k=1}^K \log \left|\det \frac{\partial g^{(k)}}{\partial \boldsymbol{a}^{(k-1)}}\right|^{-1}$ where $q_\phi^{(0)}$ is the usual Gaussian posterior. A similar definition holds for $q_\phi^{(K)}(\boldsymbol{d}_{1:M}|\boldsymbol{r}_{i,1:M})$. By the form of the planar flow (Eq. 18), there is a closed form expression for the Jacobian determinant. Since the expectation is over $q_\phi^{(0)}$ in Eq. 27, we can reparameterize to estimate gradients. One can show that VI-NF is still a lower bound on the expected log marginal $\mathbb{E}[\log p(\boldsymbol{r}_{i,1:M})]$. Table 4 shows results for a 2PL IRT generative model using VI. We find that VI-NF has better log

likelihoods across all datasets. Future work may investigate more complex flow functions.

8.4 Interpreting the Linking Function

With nonlinear models, we face an unfortunate tradeoff between interpretability and expressivity. In domains like education, practitioners greatly value the interpretability of IRT where predictions can be directly attributed to ability or item features. With VI (Deep), our most expressive model, predictions use a black box neural network, making it hard to understand the interactions between persons and items.
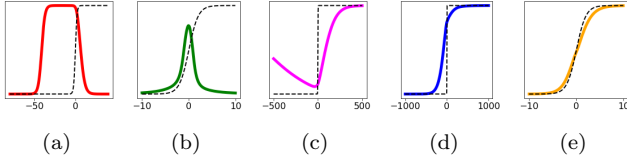

(a) (b) (c) (d) (e)

Figure 4: Learned link functions for (a) CritLangAcq, (b) WordBank, (c) DuoLingo, (d) Gradescope, and (e) PISA. The dotted black line shows the default logistic function.

Fortunately, with VI (Link), we can maintain a degree of interpretability along with power. The "Link" generative model is identical to IRT, only differing in the linking function (i.e. item response function). Each subfigure in Fig. 4 shows the learned response function for one of the real world datasets; the dotted black line represents the standard linking function, a sigmoid. We find three types linking functions: (1) for Gradescope and PISA, the learned function stays near a Sigmoid. In these cases, we find the log densities for VI (Link) to be close to VI (IRT) in Table 10; (2) For WordBank and CritLangAcq, the response function is no longer monotonic. In fact, it closely resembles an unfolding model [33, 2], which encodes a more nuanced interaction between ability and item characteristics: higher scores are related to higher ability only if the ability and item characteristics are "nearby" in latent space. (3) For DuoLingo, we find a piecewise function that resembles a sigmoid for positive values and a negative linear function for negative values. In cases (2) and (3) we find much greater differences between VI (IRT) and VI (Link) in Table 10. For DuoLingo, VI (Link) matches the log density of more expressive models, suggesting that most of the benefit of nonlinearity is exactly in the linking function.

8.5 When to use what?

An outstanding question a practitioner might have after reading this paper is: *Which inference algorithm is best for what setting?* Not wanting to add to the deluge of IRT strategies, we humbly suggest the following policy: If your data size under 10k, use HMC (or EM if speed is of utmost importance); if your data size is under 100k, use VI (or MLE if no missing data); anything beyond 100k, use VI and consider nonlinear IRT. If you are working with multidimensional IRT, use VI or HMC. If the number of items is past 500, use VI (or MLE if no missing data).

9. CONCLUSION

Item Response Iheory is a paradigm for reasoning about the scoring of tests, surveys, and similar instruments for measuring aptitude. Inferring ability and item characteristics poses

a technical challenge: balancing efficiency against accuracy. While the most principled approach, Bayesian IRT, has existed, it posed such a large computational cost that it was unrealistic to use. In this paper, we posed an alternative algorithm for Bayesian IRT that was magnitudes faster and as performant as MCMC. The effectivness of variational inference led to a discussion on nonlinear IRT, which we found gave performance gains.

10. REFERENCES

[1] 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.

[2] D. Andrich and G. Luo. A hyperbolic cosine latent trait model for unfolding dichotomous single-stimulus responses. *Applied Psychological Measurement*, 17(3):253–276, 1993.

[3] A. A. Béguin and C. A. Glas. Mcmc estimation and some model-fit analysis of multidimensional irt models. *Psychometrika*, 66(4):541–561, 2001.

[4] R. v. d. Berg, L. Hasenclever, J. M. Tomczak, and M. Welling. Sylvester normalizing flows for variational inference. *arXiv preprint arXiv:1803.05649*, 2018.

[5] M. Betancourt. A conceptual introduction to hamiltonian monte carlo. *arXiv preprint arXiv:1701.02434*, 2017.

[6] E. Bingham, J. P. Chen, M. Jankowiak, F. Obermeyer, N. Pradhan, T. Karaletsos, R. Singh, P. Szerlip, P. Horsfall, and N. D. Goodman. Pyro: Deep universal probabilistic programming. *The Journal of Machine Learning Research*, 20(1):973–978, 2019.

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

[8] R. D. Bock and M. Aitkin. Marginal maximum likelihood estimation of item parameters: Application of an em algorithm. *Psychometrika*, 46(4):443–459, 1981.

[9] R. D. Bock, R. Gibbons, and E. Muraki. Full-information item factor analysis. *Applied Psychological Measurement*, 12(3):261–280, 1988.

[10] M. Braginsky, D. Yurovsky, V. A. Marchman, and M. C. Frank. Developmental changes in the relationship between grammar and the lexicon. In *CogSci*, pages 256–261, 2015.

[11] S. Cheng, Q. Liu, E. Chen, Z. Huang, Z. Huang, Y. Chen, H. Ma, and G. Hu. Dirt: Deep learning enhanced item response theory for cognitive diagnosis. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 2397–2400, 2019.

[12] M. Curi, G. A. Converse, J. Hajewski, and S. Oliveira. Interpretable variational autoencoders for cognitive models. In *2019 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2019.

[13] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22, 1977.

[14] F. Y. Edgeworth. The statistics of examinations.

Journal of the Royal Statistical Society, 51(3):599–635, 1888.

[15] S. E. Embretson and S. P. Reise. *Item response theory.* Psychology Press, 2013.

[16] O. for Economic Co-operation and D. (OECD). Pisa 2015 database. 2016.

[17] M. C. Frank, M. Braginsky, D. Yurovsky, and V. A. Marchman. Wordbank: An open repository for developmental vocabulary data. *Journal of child language*, 44(3):677–694, 2017.

[18] A. E. Gelfand and A. F. Smith. Sampling-based approaches to calculating marginal densities. *Journal of the American statistical association*, 85(410):398–409, 1990.

[19] S. Gershman and N. Goodman. Amortized inference in probabilistic reasoning. In *Proceedings of the annual meeting of the cognitive science society*, volume 36, 2014.

[20] S. J. Haberman. Maximum likelihood estimates in exponential response models. *The Annals of Statistics*, pages 815–841, 1977.

[21] W. Harlen. The assessment of scientific literacy in the oecd/pisa project. 2001.

[22] J. K. Hartshorne, J. B. Tenenbaum, and S. Pinker. A critical period for second language acquisition: Evidence from 2/3 million english speakers. *Cognition*, 177:263–277, 2018.

[23] M. R. Harwell, F. B. Baker, and M. Zwarts. Item parameter estimation via marginal maximum likelihood and an em algorithm: A didactic. *Journal of Educational Statistics*, 13(3):243–271, 1988.

[24] W. K. Hastings. Monte carlo sampling methods using markov chains and their applications. 1970.

[25] G. E. Hinton. Products of experts. 1999.

[26] M. D. Hoffman and A. Gelman. The no-u-turn sampler: adaptively setting path lengths in hamiltonian monte carlo. *Journal of Machine Learning Research*, 15(1):1593–1623, 2014.

[27] C.-W. Huang, D. Krueger, A. Lacoste, and A. Courville. Neural autoregressive flows. *arXiv preprint arXiv:1804.00779*, 2018.

[28] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. K. Saul. An introduction to variational methods for graphical models. *Machine learning*, 37(2):183–233, 1999.

[29] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[30] D. P. Kingma, T. Salimans, R. Jozefowicz, X. Chen, I. Sutskever, and M. Welling. Improving variational inference with inverse autoregressive flow.(nips), 2016. *URL http://arxiv. org/abs/1606.04934.*

[31] D. P. Kingma and M. Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[32] W. J. v. d. Linden. *Handbook of Item Response Theory: Volume 2: Statistical Tools.* CRC Press, 2017.

[33] C.-W. Liu and R. P. Chalmers. Fitting item response unfolding models to likert-scale data using mirt in r. *PloS one*, 13(5), 2018.

[34] L. Magdalena, E. Haman, S. A. Lotem, B. Etenkowski, F. Southwood, D. Andjelkovic, E. Bloom, T. Boerma, S. Chiat, P. E. de Abreu, et al. Ratings of age of acquisition of 299 words across 25 languages: Is there a cross-linguistic order og words? *Behavior Research Methods (print Edition)*, 48(3):1154–1177, 2016.

[35] R. P. McDonald. A basis for multidimensional item response theory. *Applied Psychological Measurement*, 24(2):99–114, 2000.

[36] A. Mnih and K. Gregor. Neural variational inference and learning in belief networks. *arXiv preprint arXiv:1402.0030*, 2014.

[37] P. Natesan, R. Nandakumar, T. Minka, and J. D. Rubright. Bayesian prior choice in irt estimation using mcmc and variational bayes. *Frontiers in psychology*, 7:1422, 2016.

[38] R. M. Neal. An improved acceptance procedure for the hybrid monte carlo algorithm. *Journal of Computational Physics*, 111(1):194–203, 1994.

[39] R. M. Neal et al. Mcmc using hamiltonian dynamics. *Handbook of markov chain monte carlo*, 2(11):2, 2011.

[40] C. Piech, A. Malik, L. M. Scott, R. T. Chang, and C. Lin. The stanford acuity test: A precise vision test using bayesian techniques and a discovery in human visual response. *arXiv preprint arXiv:1906.01811*, 2019.

[41] G. Rasch. Studies in mathematical psychology: I. probabilistic models for some intelligence and attainment tests. 1960.

[42] D. Ravitch. *National standards in American education: A citizen's guide.* ERIC, 1995.

[43] M. D. Reckase. Multidimensional item response theory models. In *Multidimensional item response theory*, pages 79–112. Springer, 2009.

[44] D. J. Rezende and S. Mohamed. Variational inference with normalizing flows. *arXiv preprint arXiv:1505.05770*, 2015.

[45] D. J. Rezende, S. Mohamed, and D. Wierstra. Stochastic backpropagation and approximate inference in deep generative models. *arXiv preprint arXiv:1401.4082*, 2014.

[46] B. Settles, C. Brust, E. Gustafson, M. Hagiwara, and N. Madnani. Second language acquisition modeling. In *Proceedings of the thirteenth workshop on innovative use of NLP for building educational applications*, pages 56–65, 2018.

[47] A. Singh, S. Karayev, K. Gutowski, and P. Abbeel. Gradescope: a fast, flexible, and fair system for scalable assessment of handwritten work. In *Proceedings of the fourth (2017) acm conference on learning@ scale*, pages 81–88. ACM, 2017.

[48] S. Sinharay, M. S. Johnson, and H. S. Stern. Posterior predictive assessment of item response theory models. *Applied Psychological Measurement*, 30(4):298–321, 2006.

[49] J. M. Tomczak and M. Welling. Improving variational auto-encoders using householder flow. *arXiv preprint arXiv:1611.09630*, 2016.

[50] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[51] M. J. Wainwright, M. I. Jordan, et al. Graphical models, exponential families, and variational inference.

Foundations and Trends® in Machine Learning, 1(1–2):1–305, 2008.

[52] S. Webb, A. Golinski, R. Zinkov, S. Narayanaswamy, T. Rainforth, Y. W. Teh, and F. Wood. Faithful inversion of generative models for effective amortized inference. In *Advances in Neural Information Processing Systems*, pages 3070–3080, 2018.

[53] R. J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256, 1992.

[54] M. Wu and N. Goodman. Multimodal generative models for scalable weakly-supervised learning. In *Advances in Neural Information Processing Systems*, pages 5575–5585, 2018.

[55] C.-K. Yeung. Deep-irt: Make deep learning based knowledge tracing explainable using item response theory. *arXiv preprint arXiv:1904.11738*, 2019.

[56] J. Zhang, X. Shi, I. King, and D.-Y. Yeung. Dynamic key-value memory networks for knowledge tracing. In *Proceedings of the 26th international conference on World Wide Web*, pages 765–774, 2017.

APPENDIX

A. PROOFS OF THEOREM AND COROLLARY

In this section, we will restate the main Theorem and Corollary from the main text as well as provide proofs of the statements.

THEOREM A.1. *Let $\boldsymbol{a}_i$ be the ability for person $i \in [1, N]$ and $\boldsymbol{d}_j$ be the characteristics for item $j \in [1, M]$. We use the shorthand notation $\boldsymbol{d}_{1:M} = (\boldsymbol{d}_1, \ldots, \boldsymbol{d}_M)$. Let $r_{i,j}$ be the binary response for item j by person i. We write $\boldsymbol{r}_{i,1:M} = (r_{i,1}, \ldots r_{i,M})$. If we defined the following terms,*

$$\mathcal{L}_{recon} = \mathbb{E}_{q(\boldsymbol{a}_i, \boldsymbol{d}_{1:M} | \boldsymbol{r}_{i,1:M})} \left[\log p(\boldsymbol{r}_{i,1:M} | \boldsymbol{a}_i, \boldsymbol{d}_{1:M}) \right]$$

$$D_{ability} = D_{\mathrm{KL}}(q(\boldsymbol{a}_i | \boldsymbol{r}_{i,1:M}) || p(\boldsymbol{a}_i))$$

$$D_{item} = D_{\mathrm{KL}}(q(\boldsymbol{d}_{1:M} | \boldsymbol{r}_{i,1:M}) || p(\boldsymbol{d}_{1:M}))$$

then the inequality is true:

$$\log p(\boldsymbol{r}_{i,1:M}) \geq \mathcal{L}_{recon} + D_{ability} + D_{item}$$

where $q(\boldsymbol{a}_i, \boldsymbol{d}_{1:M} | \boldsymbol{r}_{i,1:M}) = q(\boldsymbol{a}_i | \boldsymbol{r}_{i,1:M}) q(\boldsymbol{d}_{1:M} | \boldsymbol{r}_{i,1:M})$.

PROOF. Begin by expanding the marginal distribution and applying Jensen's Inequality.

$$\log p(\boldsymbol{r}_{i,1:M}) = \log \int_{\boldsymbol{a}_i} \int_{\boldsymbol{d}_{1:M}} p(\boldsymbol{r}_{i,1:M}, \boldsymbol{a}_i, \boldsymbol{d}_{1:M}) d\boldsymbol{d}_{1:M} d\boldsymbol{a}_i$$

$$= \log \int_{\boldsymbol{a}_i} \int_{\boldsymbol{d}_{1:M}} q(\boldsymbol{a}_i, \boldsymbol{d}_{1:M} | \boldsymbol{r}_{i,1:M}) \frac{p(\boldsymbol{r}_{i,1:M}, \boldsymbol{a}_i, \boldsymbol{d}_{1:M})}{q(\boldsymbol{a}_i, \boldsymbol{d}_{1:M} | \boldsymbol{r}_{i,1:M})} d\boldsymbol{d}_{1:M} d\boldsymbol{a}_i$$

$$= \log \mathbb{E}_{q(\boldsymbol{a}_i, \boldsymbol{d}_{1:M} | \boldsymbol{r}_{i,1:M})} \left[\frac{p(\boldsymbol{r}_{i,1:M}, \boldsymbol{a}_i, \boldsymbol{d}_{1:M})}{q(\boldsymbol{a}_i, \boldsymbol{d}_{1:M} | \boldsymbol{r}_{i,1:M})} \right]$$

$$\geq \mathbb{E}_{q(\boldsymbol{a}_i, \boldsymbol{d}_{1:M} | \boldsymbol{r}_{i,1:M})} \left[\log \frac{p(\boldsymbol{r}_{i,1:M}, \boldsymbol{a}_i, \boldsymbol{d}_{1:M})}{q(\boldsymbol{a}_i, \boldsymbol{d}_{1:M} | \boldsymbol{r}_{i,1:M})} \right]$$

$$= \mathbb{E}_{q(\boldsymbol{a}_i, \boldsymbol{d}_{1:M} | \boldsymbol{r}_{i,1:M})} \left[\log p(\boldsymbol{r}_{i,1:M} | \boldsymbol{a}_i, \boldsymbol{d}_{1:M}) \right] + \mathbb{E}_{q(\boldsymbol{a}_i, \boldsymbol{d}_{1:M} | \boldsymbol{r}_{i,1:M})} \left[\log \frac{p(\boldsymbol{a}_i) p(\boldsymbol{d}_{1:M})}{q(\boldsymbol{a}_i | \boldsymbol{r}_{i,1:M}) q(\boldsymbol{d}_{1:M} | \boldsymbol{r}_{i,1:M})} \right]$$

$$= \mathcal{L}_{recon} + \mathcal{L}_{remainder}$$

Focus on $\mathcal{L}_{remainder}$ below.

$$\mathcal{L}_{remainder} = \int_{\boldsymbol{a}_i} \int_{\boldsymbol{d}_{1:M}} q(\boldsymbol{a}_i | \boldsymbol{r}_{i,1:M}) q(\boldsymbol{d}_{1:M} | \boldsymbol{r}_{i,1:M}) \log \frac{p(\boldsymbol{a}_i) p(\boldsymbol{d}_{1:M})}{q(\boldsymbol{a}_i | \boldsymbol{r}_{i,1:M}) q(\boldsymbol{d}_{1:M} | \boldsymbol{r}_{i,1:M})} d\boldsymbol{d}_{1:M} d\boldsymbol{a}_i$$

$$= \int_{\boldsymbol{a}_i} q(\boldsymbol{a}_i | \boldsymbol{r}_{i,1:M}) \left(\int_{\boldsymbol{d}_{1:M}} q(\boldsymbol{d}_{1:M} | \boldsymbol{r}_{i,1:M}) \log \frac{p(\boldsymbol{a}_i)}{q(\boldsymbol{a}_i | \boldsymbol{r}_{i,1:M})} + q(\boldsymbol{d}_{1:M} | \boldsymbol{r}_{i,1:M}) \log \frac{p(\boldsymbol{d}_{1:M})}{q(\boldsymbol{d}_{1:M} | \boldsymbol{r}_{i,1:M})} d\boldsymbol{d}_{1:M} \right) d\boldsymbol{a}_i$$

$$= \int_{\boldsymbol{a}_i} q(\boldsymbol{a}_i | \boldsymbol{r}_{i,1:M}) \left(\log \frac{p(\boldsymbol{a}_i)}{q(\boldsymbol{a}_i | \boldsymbol{r}_{i,1:M})} \int_{\boldsymbol{d}_{1:M}} q(\boldsymbol{d}_{1:M} | \boldsymbol{r}_{i,1:M}) d\boldsymbol{d}_{1:M} + \mathbb{E}_{q(\boldsymbol{d}_{1:M} | \boldsymbol{r}_{i,1:M})} \left[\frac{p(\boldsymbol{d}_{1:M})}{q(\boldsymbol{d}_{1:M} | \boldsymbol{r}_{i,1:M})} \right] \right) d\boldsymbol{a}_i$$

$$= \int_{\boldsymbol{a}_i} q(\boldsymbol{a}_i | \boldsymbol{r}_{i,1:M}) \log \frac{p(\boldsymbol{a}_i)}{q(\boldsymbol{a}_i | \boldsymbol{r}_{i,1:M})} d\boldsymbol{a}_i + \mathbb{E}_{q(\boldsymbol{d}_{1:M} | \boldsymbol{r}_{i,1:M})} \left[\frac{p(\boldsymbol{d}_{1:M})}{q(\boldsymbol{d}_{1:M} | \boldsymbol{r}_{i,1:M})} \right] \int_{\boldsymbol{a}_i} q(\boldsymbol{a}_i | \boldsymbol{r}_{i,1:M}) d\boldsymbol{a}_i$$

$$= \mathbb{E}_{q(\boldsymbol{a}_i | \boldsymbol{r}_{i,1:M})} \left[\log \frac{p(\boldsymbol{a}_i)}{q(\boldsymbol{a}_i | \boldsymbol{r}_{i,1:M})} \right] + \mathbb{E}_{q(\boldsymbol{d}_{1:M} | \boldsymbol{r}_{i,1:M})} \left[\frac{p(\boldsymbol{d}_{1:M})}{q(\boldsymbol{d}_{1:M} | \boldsymbol{r}_{i,1:M})} \right]$$

$$= D_{\mathrm{KL}}(q(\boldsymbol{a}_i | \boldsymbol{r}_{i,1:M}) || p(\boldsymbol{a}_i)) + D_{\mathrm{KL}}(q(\boldsymbol{d}_{1:M} | \boldsymbol{r}_{i,1:M}) || p(\boldsymbol{d}_{1:M}))$$

□

COROLLARY A.1.1. *Using the definitions in Thm. A.1, if $q(\boldsymbol{a}_i, \boldsymbol{d}_{1:M} | \boldsymbol{r}_{i,1:M}) = q(\boldsymbol{a}_i | \boldsymbol{d}_{1:M}, \boldsymbol{r}_{i,1:M}) q(\boldsymbol{d}_{1:M} | \boldsymbol{r}_{i,1:M})$, then the following lower bound holds*

$$\log p(\boldsymbol{r}_{i,1:M}) \geq \mathcal{L}_{recon} + \mathbb{E}_{q(\boldsymbol{d}_{1:M} | \boldsymbol{r}_{i,1:M})} [D_{ability}] + D_{item}$$

where $D_{ability} = D_{\mathrm{KL}}(q(\boldsymbol{a}_i | \boldsymbol{d}_{1:M}, \boldsymbol{r}_{i,1:M}) || p(\boldsymbol{u}_i))$.

PROOF. The proof is very similar to the one for Thm. A.1 so we may skip a few steps. We begin by again expanding the marginal and Jensen's.

$$\log p(\boldsymbol{r}_{i,1:M}) \geq \mathbb{E}_{q(\boldsymbol{a}_i, \boldsymbol{d}_{1:M} | \boldsymbol{r}_{i,1:M})} \left[\log p(\boldsymbol{r}_{i,1:M} | \boldsymbol{a}_i, \boldsymbol{d}_{1:M}) \right] + \mathbb{E}_{q(\boldsymbol{a}_i, \boldsymbol{d}_{1:M} | \boldsymbol{r}_{i,1:M})} \left[\log \frac{p(\boldsymbol{a}_i) p(\boldsymbol{d}_{1:M})}{q(\boldsymbol{a}_i | \boldsymbol{d}_{1:M}, \boldsymbol{r}_{i,1:M}) q(\boldsymbol{d}_{1:M} | \boldsymbol{r}_{i,1:M})} \right]$$

$$= \mathcal{L}_{recon} + \mathcal{L}_{remainder}$$

Notice that $\mathcal{L}_{\mathrm{remainder}}$ looks slightly different in this version since $q(\boldsymbol{a}_i|\boldsymbol{d}_{1:M}, \boldsymbol{r}_{i,1:M}) \neq q(\boldsymbol{a}_i|\boldsymbol{r}_{i,1:M})$. Now, again, we proceed to look only at $\mathcal{L}_{\mathrm{remainder}}$.

$$\mathcal{L}_{\mathrm{remainder}} = \int_{\boldsymbol{a}_i} \int_{\boldsymbol{d}_{1:M}} q(\boldsymbol{a}_i|\boldsymbol{d}_{1:M}, \boldsymbol{r}_{i,1:M}) q(\boldsymbol{d}_{1:M}|\boldsymbol{r}_{i,1:M}) \left(\log \frac{p(\boldsymbol{a}_i)}{q(\boldsymbol{a}_i|\boldsymbol{d}_{1:M}, \boldsymbol{r}_{i,1:M})} + \log \frac{p(\boldsymbol{d}_{1:M})}{q(\boldsymbol{d}_{1:M}|\boldsymbol{r}_{i,1:M})} \right) d\boldsymbol{d}_{1:M} d\boldsymbol{a}_i$$

$$= \int_{\boldsymbol{a}_i} \int_{\boldsymbol{d}_{1:M}} q(\boldsymbol{a}_i|\boldsymbol{d}_{1:M}, \boldsymbol{r}_{i,1:M}) q(\boldsymbol{d}_{1:M}|\boldsymbol{r}_{i,1:M}) \log \frac{p(\boldsymbol{a}_i)}{q(\boldsymbol{a}_i|\boldsymbol{d}_{1:M}, \boldsymbol{r}_{i,1:M})} d\boldsymbol{d}_{1:M} d\boldsymbol{a}_i$$

$$+ \int_{\boldsymbol{a}_i} \int_{\boldsymbol{d}_{1:M}} q(\boldsymbol{a}_i|\boldsymbol{d}_{1:M}, \boldsymbol{r}_{i,1:M}) q(\boldsymbol{d}_{1:M}|\boldsymbol{r}_{i,1:M}) \log \frac{p(\boldsymbol{d}_{1:M})}{q(\boldsymbol{d}_{1:M}|\boldsymbol{r}_{i,1:M})} d\boldsymbol{d}_{1:M} d\boldsymbol{a}_i$$

$$= A + B$$

Let us look at A and B individually.

$$A = \int_{\boldsymbol{d}_{1:M}} q(\boldsymbol{d}_{1:M}|\boldsymbol{r}_{i,1:M}) \left(\int_{\boldsymbol{a}_i} q(\boldsymbol{a}_i|\boldsymbol{d}_{1:M}, \boldsymbol{r}_{i,1:M}) \log \frac{p(\boldsymbol{a}_i)}{q(\boldsymbol{a}_i|\boldsymbol{d}_{1:M}, \boldsymbol{r}_{i,1:M})} d\boldsymbol{a}_i \right) d\boldsymbol{d}_{1:M}$$

$$= \mathbb{E}_{q(\boldsymbol{d}_{1:M}|\boldsymbol{r}_{i,1:M})} \left[D_{\mathrm{KL}}(q(\boldsymbol{a}_i|\boldsymbol{d}_{1:M}, \boldsymbol{r}_{i,1:M}) || p(\boldsymbol{a}_i)) \right]$$

$$B = \int_{\boldsymbol{d}_{1:M}} q(\boldsymbol{d}_{1:M}|\boldsymbol{r}_{i,1:M}) \log \frac{p(\boldsymbol{d}_{1:M})}{q(\boldsymbol{d}_{1:M}|\boldsymbol{r}_{i,1:M})} \left(\int_{\boldsymbol{a}_i} q(\boldsymbol{a}_i|\boldsymbol{d}_{1:M}, \boldsymbol{r}_{i,1:M}) d\boldsymbol{a}_i \right) d\boldsymbol{d}_{1:M}$$

$$= \mathbb{E}_{q(\boldsymbol{d}_{1:M}|\boldsymbol{r}_{i,1:M})} \left[\log \frac{p(\boldsymbol{d}_{1:M})}{q(\boldsymbol{d}_{1:M}|\boldsymbol{r}_{i,1:M})} \right]$$

$$= D_{\mathrm{KL}}(q(\boldsymbol{d}_{1:M}|\boldsymbol{r}_{i,1:M}) || p(\boldsymbol{d}_{1:M}))$$

Thus, $\mathcal{L}_{\mathrm{remainder}} = \mathbb{E}_{q(\boldsymbol{d}_{1:M}|\boldsymbol{r}_{i,1:M})} \left[D_{\mathrm{KL}}(q(\boldsymbol{a}_i|\boldsymbol{d}_{1:M}, \boldsymbol{r}_{i,1:M}) || p(\boldsymbol{a}_i)) \right] + D_{\mathrm{KL}}(q(\boldsymbol{d}_{1:M}|\boldsymbol{r}_{i,1:M}) || p(\boldsymbol{d}_{1:M}))$. $\square$

B. TRAINING DETAILS

For reproducibility, we specify the hyperparameters the design choices for the experiments in the main text.

B.1 Fast Inference Experiments

When using HMC [39, 5], we use the No-U-Turn sampler [26] and take 100 warmup steps followed by 200 sampling steps. To reduce cost, we use only a single chain. A public implementation of HMC can be found in Pyro [6]. For VI, we use the Adam optimizer [29] with a learning rate of 5e-3 and train for 200 epochs (one epoch being a full pass over the dataset). Thus, with a larger dataset, we take more gradient steps, which we find necessary for good performance. However, for larger datasets, we increase the batch size. For 1k persons, we use a batch size of 16; for 10k, we use 128; for 100k, we use 512. For the ability inference networks, we use 3 layer MLPs with 64 hidden units each and ELU nonlinearities. For the item inference networks, we use a Embedding layers to match each item index j to free parameters. All models are trained with a single Titan X GPU.

B.2 Deep IRT Experiments

We split each dataset randomly into a training and test portion in ratios of 80% and 20%. Like above, we optimize with Adam for 200 epochs with a learning rate of 5e-3. The batch size is dependent on the dataset: we use 512 for CritLangAcq and PISA, 64 for WordBank, and 16 for the remaining two. When training VI, we only use the conditional posterior $q(\boldsymbol{a}_i|\boldsymbol{d}_{1:M}, \boldsymbol{r}_{i,1:M})$. By default, we use 3PL IRT for the generative model. When training nonlinear generative models, we use the same architectures as in the synthetic experiments with all hidden layers having 64 units and ELU nonlinearity. To estimate log marginal probabilities, we use 1000 samples. We use 1 ability dimension when comparing HMC and VI and 5 ability dimensions for nonlinear models.

B.3 Conditional Inference Model

In the main text, we made reference to a result that using $\mathrm{VI}^{\perp}$ is not as good as VI itself. To support this experimentally, we took various synthetically generated datasets using a 2PL IRT model with varying number of people, items, and dimensions. For each synthetic dataset, we compute the correlation between the true ability (known since we generated the dataset) and the inferred ability by VI, $\mathrm{VI}^{\perp}$, and HMC. A better inference algorithm should return a correlation closer to 1. As Fig. 5 shows, the performance of $\mathrm{VI}^{\perp}$ is far lower than VI and HMC, struggling as the response data grows larger in both size and dimensionality. We find this phenomena quite intriguing as the design of $\mathrm{VI}^{\perp}$ is perhaps the more intuitive one (we just "flipped" the arrows of the generative model). However, the dependency of ability on items was extremely important to model. There has been some theoretical studies of this phenomena [52] as "faithful inversions of generative models" i.e. how do you pick the best inference model design given the graphical model of a generative model. Pure inversion is not the correct strategy. In fact, the correct strategy in [52] would result in VI's design. Table 5 show the results averaged over three trials.
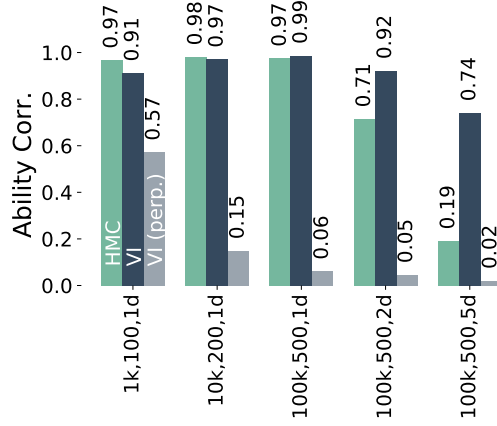
Figure 5: Comparison of correlation between inferred and true ability in a synthetic dataset between VI, VI$^\perp$, and HMC.

Table 5: Inference Quality as a function of Persons, Items, and Dim. (IRT-2PL)

Inf. Alg.	# Persons	# Items	Dim.	Ability Corr.	Item Feat. Corr.	Cost (sec.)
HMC	1K	100	1	$96.72 \pm 5 \times 10^{-5}$	$99.48 \pm 7 \times 10^{-5}$	341.0 ± 3.4
VI	1K	100	1	91.24 ± 0.01	$98.03 \pm 1 \times 10^{-3}$	82.8 ± 6.2
VI$^\perp$	1K	100	1	$57.23 \pm 2 \times 10^{-4}$	68.57 ± 0.30	94.6 ± 1.8
HMC	10K	200	1	$98.18 \pm 2 \times 10^{-5}$	$99.93 \pm 2 \times 10^{-6}$	5179.9 ± 36.7
VI	10K	200	1	$97.19 \pm 4 \times 10^{-4}$	$99.45 \pm 1 \times 10^{-4}$	172.6 ± 5.8
VI$^\perp$	10K	200	1	$14.98 \pm 2 \times 10^{-6}$	52.27 ± 0.01	162.0 ± 4.4
HMC	100K	500	1	$97.52 \pm 4 \times 10^{-4}$	$99.75 \pm 2 \times 10^{-4}$	15327.5 ± 148.5
VI	100K	500	1	98.76 ± 0.001	$99.91 \pm 8 \times 10^{-5}$	1211.4 ± 20.4
VI$^\perp$	100K	500	1	$6.14 \pm 3 \times 10^{-6}$	60.91 ± 0.01	1072.4 ± 32.2
HMC	100K	500	2	71.58 ± 0.001	$89.26 \pm 3 \times 10^{-4}$	18978.0 ± 234.6
VI	100K	500	2	92.25 ± 0.02	97.48 ± 0.02	1827.4 ± 61.5
VI$^\perp$	100K	500	2	$4.64 \pm 7 \times 10^{-6}$	62.63 ± 0.02	1526.7 ± 29.5
HMC	100K	500	5	$19.17 \pm 2 \times 10^{-4}$	$28.30 \pm 7 \times 10^{-4}$	24178.6 ± 190.8
VI	100K	500	5	74.13 ± 0.02	86.91 ± 0.02	1952.64 ± 12.6
VI$^\perp$	100K	500	5	$2.17 \pm 1 \times 10^{-4}$	$63.48 \pm 2 \times 10^{-3}$	1569.7 ± 13.3
VIRTU	1M	1K	5	86.24 ± 0.01	$94.04 \pm 1 \times 10^{-3}$	16101.87 ± 88.2

C. RESULTS IN TABLE FORMAT

For all of the plots in the main text, we include tables here with the numbers for more precise comparison. Additionally, due to lack of space, we only showed the results of nonlinear VI for a single Gradescope course. Below, we show log likelihoods for 5 more courses, chosen from the Gradescope dataset by largest number of students. We find consistent patterns.

Table 6: Inference Quality for Synthetic IRT Datasets

# Person	# Item	Dim.	Time Cost (Sec.)			Corr. with True Ability		
			VI	MLE	HMC	VI	MLE	HMC
100	100	1	5.7	2.6	57.1	0.561	0.973	0.972
500	100	1	12.4	8.8	65.2	0.939	0.953	0.957
2.5к	100	1	39.9	35.8	128.3	0.965	0.966	0.969
12.5к	100	1	94.6	73.3	303.1	0.961	0.961	0.964
62.5к	100	1	238.1	139.3	1.6к	0.965	0.963	0.966
312.5к	100	1	422.2	365.6	43.6к	0.967	0.959	0.967
1.56M	100	1	804.1	986.4	784.7к	0.960	0.956	0.964
10к	10	1	153.6	54.2	195.5	0.658	0.655	0.024
10к	50	1	157.1	53.3	184.3	0.913	0.911	0.922
10к	250	1	163.2	56.6	329.3	0.984	0.985	0.985
10к	1250	1	246.5	77.9	4.9к	0.997	0.997	0.997
10к	6250	1	382.8	124.7	10.3к	0.999	0.999	0.999
10к	100	1	153.7	60.6	197.2	0.966	0.961	0.964
10к	100	2	160.3	74.1	223.7	0.957	0.958	0.838
10к	100	3	181.7	74.9	513.5	0.926	0.947	0.713
10к	100	4	153.8	73.4	532.6	0.833	0.953	0.679
10к	100	5	157.0	104.9	568.2	0.776	0.946	0.643

Table 7: 10% Missing Data Imputation for Synthetic IRT Datasets

# Person	# Item	Dim.	Time Cost (Sec.)				Accuracy			
			VI	MLE	EM	HMC	VI	MLE	EM	HMC
100	100	1	6.3	2.9	5.1	58.3	0.699	0.720	0.635	0.751
500	100	1	14.0	8.5	5.4	65.9	0.776	0.756	0.630	0.785
2.5к	100	1	40.1	35.8	8.0	121.5	0.791	0.740	0.628	0.792
12.5к	100	1	89.7	73.9	23.5	306.9	0.777	0.728	0.619	0.783
62.5к	100	1	241.2	137.6	81.9	1.6к	0.794	0.754	0.630	0.798
312.5к	100	1	424.6	365.3	1.1к	43.5к	0.803	0.754	0.624	0.809
1.56M	100	1	802.3	990.0	15.5к	784.5к	0.815	0.754	0.629	0.815
10к	10	1	152.0	52.5	2.0	198.3	0.717	0.711	0.644	0.726
10к	50	1	161.6	54.7	4.3	185.7	0.710	0.689	0.606	0.690
10к	250	1	160.2	61.1	57.6	333.5	0.704	0.707	0.621	0.707
10к	1250	1	250.3	81.3	261.2	4.9к	0.715	0.710	0.624	0.710
10к	6250	1	385.3	119.9	2.6к	15.3к	0.717	0.713	0.573	0.716
10к	100	1	150.0	56.4	11.1	193.4	0.705	0.701	0.621	0.707
10к	100	2	163.6	78.2	14.8	222.2	0.737	0.731	0.608	0.737
10к	100	3	181.7	78.7	13.1	515.8	0.747	0.745	0.590	0.747
10к	100	4	150.3	71.8	14.7	534.3	0.772	0.772	0.594	0.773
10к	100	5	152.6	106.3	16.8	572.9	0.787	0.785	0.592	0.787

Table 8: Inference Quality for Real World Datasets (IRT-2PL, Dim. 1)

Dataset	Time Cost (Sec.)			Corr. with HMC Ability	
	VI	MLE	HMC	VI	MLE
CritLangAcq	2.8к	2.5к	514.1к	0.966	0.712
WordBank	176.4	150.7	527.8	0.998	0.996
DuoLingo	429.9	453.7	784.7	0.841	0.275
Gradescope	114.5	116.3	95.2	0.914	0.135
PISA	25.2к	3.1к	383.5к	0.976	0.276

Table 9: 10% Missing Data Imputation for Real World Datasets (IRT-2PL, Dim. 1)

DATASET	TIME COST (SEC.)				ACCURACY			
	VI	MLE	EM	HMC	VI	MLE	EM	HMC
CRITLANGACQ	2.8K	2.7K	6.7K	517.2K	0.932	0.918	0.899	0.931
WORDBANK	173.5	152.7	679.2	528.1	0.880	0.879	0.832	0.880
DUOLINGO	431.2	453.1	2.1K	792.1	0.886	0.801	0.829	0.891
GRADESCOPE	107.7	108.1	13.1	125.8	0.826	0.714	0.745	0.825
PISA	18.7K	3.7K	448.8	387.1K	0.728	0.706	0.643	0.730

Table 10: Log Likelihoods of Various Deep Generative IRT Models (Ability Dim. 5)

DATASET	INF. ALG.	TRAIN LOG MARGINAL	TEST LOG MARGINAL
CRITLANGACQ	VI (MIRT-1PL)	-11249.85 ± 7.55	-11246.25 ± 8.22
	VI (MIRT-2PL)	-10224.09 ± 7.12	-10226.36 ± 9.10
	VI (LINK-2PL)	-9590.33 ± 2.13	-9593.02 ± 6.57
	VI (DEEP-2PL)	-9311.26 ± 5.13	-9313.33 ± 1.89
	VI (RESIDUAL-2PL)	$\mathbf{-9254.13} \pm 4.78$	$\mathbf{-9256.22} \pm 5.52$
WORDBANK	VI (MIRT-1PL)	-17047.28 ± 4.32	-17039.03 ± 2.11
	VI (MIRT-2PL)	-5882.55 ± 0.78	-5672.97 ± 3.51
	VI (LINK-2PL)	-5268.04 ± 6.95	-5327.52 ± 0.52
	VI (DEEP-2PL)	$\mathbf{-4658.44} \pm 3.94$	$\mathbf{-4385.49} \pm 1.93$
	VI (RESIDUAL-2PL)	-4681.45 ± 2.19	-4421.22 ± 6.18
DUOLINGO '15 SLAM (EN-ES)	VI (MIRT-1PL)	-2833.39 ± 0.66	-2631.76 ± 0.57
	VI (MIRT-2PL)	-2488.36 ± 1.35	-2411.56 ± 0.12
	VI (LINK-2PL)	-1833.93 ± 0.27	-1773.18 ± 0.53
	VI (DEEP-2PL)	-1834.20 ± 1.27	-1774.31 ± 1.83
	VI (RESIDUAL-2PL)	$\mathbf{-1745.41} \pm 4.68$	$\mathbf{-1699.37} \pm 5.20$
GRADESCOPE COURSE 102576	VI (MIRT-1PL)	-1090.73 ± 2.88	-1078.39 ± 3.01
	VI (MIRT-2PL)	-876.75 ± 3.47	-858.35 ± 3.75
	VI (LINK-2PL)	-750.82 ± 0.04	-728.99 ± 0.05
	VI (DEEP-2PL)	$\mathbf{-705.05} \pm 0.54$	$\mathbf{-683.41} \pm 1.84$
	VI (RESIDUAL-2PL)	-715.33 ± 2.66	-694.41 ± 1.98
PISA '15 SCIENCE	VI (MIRT-1PL)	-13104.28 ± 5.12	-13093.58 ± 4.88
	VI (MIRT-2PL)	-6169.55 ± 4.77	-6158.98 ± 4.12
	VI (LINK-2PL)	-6120.12 ± 1.28	-6105.58 ± 2.11
	VI (DEEP-2PL)	-6030.22 ± 3.33	-6020.17 ± 1.91
	VI (RESIDUAL-2PL)	$\mathbf{-5807.38} \pm 4.24$	$\mathbf{-5797.81} \pm 2.65$

Table 11: Missing Data Imputation (Ability Dim. 5)

Dataset	Inf. Alg.	Accuracy
CritLangAcq	Deep-IRT	0.934
	VI (IRT-1PL)	0.927
	VI (IRT-2PL)	0.932
	VI (Link-2PL)	0.945
	VI (Deep-2PL)	**0.948**
	VI (Residual-2PL)	0.947
WordBank	Deep-IRT	0.681
	VI (IRT-1PL)	0.876
	VI (IRT-2PL)	0.880
	VI (Link-2PL)	0.888
	VI (Deep-2PL)	0.889
	VI (Residual-2PL)	**0.889**
DuoLingo '15 SLAM (en-es)	Deep-IRT	0.884
	VI (IRT-1PL)	0.880
	VI (IRT-2PL)	0.886
	VI (Link-2PL)	0.891
	VI (Deep-2PL)	**0.897**
	VI (Residual-2PL)	0.894
Gradescope Course 102576	Deep-IRT	0.813
	VI (IRT-1PL)	0.820
	VI (IRT-2PL)	0.826
	VI (Link-2PL)	0.840
	VI (Deep-2PL)	0.847
	VI (Residual-2PL)	**0.848**
PISA '15 Science	Deep-IRT	0.524
	VI (IRT-1PL)	0.723
	VI (IRT-2PL)	0.728
	VI (Link-2PL)	0.718
	VI (Deep-2PL)	**0.744**
	VI (Residual-2PL)	0.739

Table 12: Nonlinear VIRTU models on 6 Gradescope Course Assignments

Dataset	Inf. Alg.	Train Log Marginal	Test Log Marginal
Gradescope Course 104450	VI (IRT-2PL)	-498.38	-504.35
	VI (Link-2PL	-498.98	-501.25
	VI (Deep-2PL)	$\mathbf{-436.41}$	$\mathbf{-440.03}$
	VI (Residual-2PL)	-441.71	-445.54
Gradescope Course 99332	VI (IRT-2PL)	-491.95	-495.62
	VI (Link-2PL)	-491.47	-495.19
	VI (Deep-2PL)	$\mathbf{-433.04}$	$\mathbf{-435.29}$
	VI (Residual-2PL)	-439.29	-441.30
Gradescope Course 104394	VI (IRT-2PL)	-957.98	-961.24
	VI (Link-2PL)	-974.54	-971.88
	VI (Deep-2PL)	$\mathbf{-856.87}$	$\mathbf{-860.25}$
	VI (Residual-2PL)	-869.14	-870.63
Gradescope Course 94342	VI (IRT-2PL)	-80.96	-81.49
	VI (Link-2PL)	-83.28	-82.84
	VI (Deep-2PL)	$\mathbf{-74.49}$	$\mathbf{-74.92}$
	VI (Residual-2PL)	-74.96	-75.39
Gradescope Course 104417	VI (IRT-2PL)	-290.34	-284.65
	VI (Link-2PL)	-289.67	-283.35
	VI (Deep-2PL)	$\mathbf{-254.85}$	$\mathbf{-249.51}$
	VI (Residual-2PL)	-258.85	-253.32
Gradescope Course 102576	VI (IRT-2PL)	-873.27	-854.59
	VI (Link-2PL)	-750.85	-729.04
	VI (Deep-2PL)	$\mathbf{-704.51}$	$\mathbf{-681.57}$
	VI (Residual-2PL)	-717.99	-696.39