Siliang Zhang, Yunxiao Chen and Yang Liu

An improved stochastic EM algorithm for large-scale full-information item factor analysis

Article (Accepted version) (Refereed)

An Improved Stochastic EM Algorithm for Large-Scale Full-information Item Factor Analysis

Abstract

In this paper, we explore the use of the stochastic EM algorithm (Celeux & Diebolt, 1985) for large-scale full-information item factor analysis. Innovations have been made on its implementation, including (1) an adaptive-rejection-based Gibbs sampler for the stochastic E step, (2) a proximal gradient descent algorithm for the optimization in the M step, and (3) diagnostic procedures for determining the burn-in size and the stopping of the algorithm. These developments are based on the theoretical results of Nielsen (2000), as well as advanced sampling and optimization techniques. The proposed algorithm is computationally efficient and virtually tuning-free, making it scalable to large-scale data with many latent traits (e.g. more than five latent traits) and easy to use for practitioners. Standard errors of parameter estimation are also obtained based on the missing information identity (Louis, 1982). The performance of the algorithm is evaluated through simulation studies and an application to the analysis of the IPIP-NEO personality inventory. Extensions of the proposed algorithm to other latent variable models are discussed.

Keywords: Multidimensional item response theory, stochastic EM algorithm, full-information item factor analysis, Gibbs sampler, rejection sampling, proximal gradient descent

An Improved Stochastic EM Algorithm for Large-Scale Full-information Item Factor Analysis

1 Introduction

Large-scale data, which contain large numbers of participants and manifest variables, are often collected in psychology, education, and other social science disciplines that involve measuring many latent variables (e.g., personalities, emotional distress, etc.) and explicating the relationship thereof. When the survey is composed of items, multidimensional item response theory (MIRT; e.g., Y. Liu, Magnus, Quinn, & Thissen, in press; Reckase, 2009), also known as item factor analysis (IFA; e.g., Wirth & Edwards, 2007), provides a unified framework and convenient statistical tools for item analysis and scoring. The increasing scale and complexity of survey designs call for MIRT models with many latent traits. For example, the revised version of the Minnesota Multiphasic Personality Inventory (Butcher, Dahlstrom, Graham, Tellegen, & Kaemmer, 1989) consists of over 30 clinical subscales (traits) formed by in total more than 550 discretely scored items. The full-information maximum likelihood estimation has been regarded as the "gold standard" method for IFA parameter estimation with many desirable and well-known statistical properties[1]. It is referred to as the *large-scale full-information item factor analysis*, when analyzing large-scale item response data based on an MIRT model with many latent traits.

The classical approach for the maximum likelihood estimation for MIRT models is through the Expectation-Maximization algorithm (EM; Bock & Aitkin, 1981; Dempster, Laird, & Rubin, 1977). Even when the number of latent traits K is only moderately large (e.g. $K \geq 5$), the computational burden of the EM algorithm becomes high, because the complexity of evaluating K-dimensional numerical integrals in the E step grows

[1]It is noted that limited-information approaches such as the weight least square estimator (Muthén, 1978, 1984, 1993) and the bivariate composite likelihood estimator (Jöreskog & Moustaki, 2001; Zhao & Joe, 2005), albeit computationally more economical, do not yield asymptotically efficient solutions; therefore, they are not further considered in the current work.

exponentially with K. Various methods have been proposed, including adaptive Gaussian quadrature EM algorithms (Rabe-Hesketh, Skrondal, & Pickles, 2005; Schilling & Bock, 2005), Laplace approximation methods (Huber, Ronchetti, & Victoria-Feser, 2004; Kass & Steffey, 1989; Thomas, 1993), Monte Carlo EM algorithms (Meng & Schilling, 1996; Song & Lee, 2005), Markov chain Monte Carlo methods (Albert, 1992; Béguin & Glas, 2001; Edwards, 2010; Patz & Junker, 1999a, 1999b; Shi & Lee, 1998), and stochastic approximation methods (Cai, 2010a, 2010b; Delyon, Lavielle, & Moulines, 1999; Gu & Kong, 1998; von Davier & Sinharay, 2010). Readers are referred to Cai (2010a) for a comprehensive review of the advantages and weaknesses of those methods. In particular, the Metroplis-Hastings Robbins-Monro (MH-RM) algorithm (Cai, 2010a, 2010b), which is a stochastic approximation method, has been widely applied in education and psychology due to its flexibility and computational efficiency.

In this paper, we investigate an alternative method, the Stochastic EM algorithm (StEM; Celeux & Diebolt, 1985; Ip, 2002), which has been applied to IRT and low-dimensional MIRT models (Diebolt & Ip, 1996; Fox, 2003; Ip, 1994). Similar to the EM algorithm, the StEM algorithm iterates between two steps, the stochastic expectation (StE) step and the maximization (M) step. This algorithm avoids calculating the conditional expectation required in the E step of the classical EM algorithm by Monte Carlo simulations in its StE-step. The StEM algorithm is closely related to the stochastic approximation Newton-Raphson (SA-NR) algorithm (Gu & Kong, 1998) and the MH-RM algorithm (Cai, 2010a, 2010b) which adopt a data generation step similar to the StE step of the StEM algorithm. The major difference is that instead of solving a maximization problem exactly in the M-step, the SA-NR and MH-RM algorithms update the parameter estimates by a Robbins-Monro update rule. The StEM algorithm is also similar to the stochastic approximation EM algorithm (SA-EM; Camilli & Fox, 2015; Delyon et al., 1999). They differ by that the SAEM algorithm uses a Robbins-Monro update in its E-step which is avoided in the StE step of the StEM algorithm.

The major contribution of this paper is to propose an improved StEM algorithm whose implementation is tailored to large-scale full-information item factor analysis. This algorithm is developed under a general family of MIRT models, the multidimensional generalized partial credit (MGPC) model (Yao & Schwarz, 2006). The MGPC model handles ordinal response data and is a generalization of the multidimensional two-parameter logistic (M2PL) model (e.g. Reckase, 2009), one of the most widely used compensatory MIRT models. For this family of MIRT models, we develop (1) an adaptive-rejection-based Gibbs sampler (Gilks & Wild, 1992) for the StE step, (2) a proximal gradient descent algorithm (Parikh, Boyd, et al., 2014) for the optimization in the M step, and (3) diagnostic procedures for determining the burn-in size and the stopping based on the asymptotic properties of the StEM algorithm (Nielsen, 2000). These new developments lead to an efficient and virtually tuning-free algorithm that is scalable to large-scale data with many latent traits (e.g. more than five latent traits) and easy to use for practitioners. This algorithm can be easily generalized to many other MIRT models, such as the multidimensional graded response model (Muraki & Carlson, 1995), partially compensatory MIRT models (Sympson, 1978), etc.

The proposed method tends to perform more stably for large-scale IFA than other stochastic algorithms, including the SA-NR, MH-RM, and SA-EM algorithms. This is because, the proposed algorithm is virtually tuning-free, while the SA-NR, MH-RM, and SA-EM algorithms tend to be sensitive to tuning. Specifically, the SA-NR, MH-RM, and SA-EM algorithms algorithms involve Robbins-Monro update, which is sensitive to the choice of a decaying step size (Nemirovski, Juditsky, Lan, & Shapiro, 2009; Spall, 2005). In addition, the MH-RM algorithm can also be sensitive to the choice of step size in its random-walk Metropolis-Hastings (MH) sampler. Based on the simulation results, our algorithm tends to outperform the MH-RM algorithm implemented in the **flexMIRT** software (Cai, 2013) for larger-scale IFA, and perform similarly for small- to median-scale problems.

The rest of the paper is organized as follows. The problem of full-information maximum likelihood estimation for item factor analysis is formally stated in Section 2. Then in Section 3, an improved stochastic EM algorithm is proposed. Section 4 discusses the advantages and the extensions of the proposed algorithm and compares it with the MH-RM algorithm. Sections 5 and 6 present simulation studies and an application to the IPIP-NEO personality inventory. Finally, concluding remarks are provided in Section 7. The implementation details are provided in the appendix.

2 Full-Information Item Factor Analysis

2.1 MGPC Model

We consider N respondents answering J items. Let Y_{ij} be a random variable, denoting the response from respondent i to item j, and y_{ij} be its realization. The responses are assumed to be ordinal, $Y_{ij} \in \{0, 1, \ldots, m_j\}$. We denote $\mathbf{Y}_i = (Y_{i1}, \ldots, Y_{iJ})$ and $\mathbf{y}_i = (y_{i1}, \ldots, y_{iJ})$ the vectors of responses. Each respondent is represented by a K-dimensional latent vector, $\boldsymbol{\theta}_i = (\theta_{i1}, \ldots, \theta_{iK})$. Specifically, we consider item factor analysis under the multidimensional generalized partial credit (MGPC) model (Yao & Schwarz, 2006), one of the most popular IFA model for ordinal response data. In the special case where all the items are dichotomously scored (i.e., $m_j = 1$), the MGPC model becomes the multidimensional two-parameter logistic (M2PL) model (e.g. Reckase, 2009), one of the most widely used compensatory MIRT models. The MGPC model assumes an adjacent categories logit model (Chapter 6, Agresti, 1996) when regressing the response Y_{ij} on the latent traits $\boldsymbol{\theta}_i$, that is,

$$P(Y_{ij} = y|\boldsymbol{\theta}_i, Y_{ij} \in \{y-1, y\}) = \frac{\exp(\boldsymbol{\theta}_i \cdot \mathbf{a}_j + d_{jy})}{1 + \exp(\boldsymbol{\theta}_i \cdot \mathbf{a}_j + d_{jy})}, \ y = 1, \ldots, m_j, \tag{1}$$

where $\mathbf{a}_j = (a_{j1}, \ldots, a_{jK})$ and $\mathbf{d}_j = (d_{j1}, \ldots, d_{jm_j})$ are known as the slope parameters and the intercept parameters, respectively, and $\boldsymbol{\theta}_i \cdot \mathbf{a}_j = \sum_{k=1}^{K} a_{jk}\theta_{ik}$ denotes the inner product of two vectors. Equation (1) implies that, when restricted to two adjacent categories y and $y+1$, the conditional probability of the response in the higher category takes the same

form as the item response function of the multidimensional two-parameter logistic (M2PL) model. Equation (1) implies that the item response function takes the form

$$P(Y_{ij} = y|\boldsymbol{\theta}_i) = \frac{\exp(y\mathbf{a}_j \cdot \boldsymbol{\theta}_i + \sum_{l=1}^{y} d_{jl})}{1 + \sum_{n=1}^{m_j} \exp(n\mathbf{a}_j \cdot \boldsymbol{\theta}_i + \sum_{l=1}^{n} d_{jl})}, y = 0, 1, \ldots, m_j, \tag{2}$$

where $\sum_{l=1}^{y} d_{jl} = 0$ when $y = 0$. The local independence assumption is adopted. That is, given the latent trait $\boldsymbol{\theta}_i$, Y_{i1}, ..., Y_{iJ} are conditionally independent. Moreover, $\boldsymbol{\theta}_i$ is assumed to follow a K-variate normal distribution, with mean zero and covariance matrix $\Sigma = (\sigma_{kk'})_{K\times K}$. The multivariate normality of the latent variable is typically assumed in MIRT (see Chapter 6, Reckase, 2009) by default; however, we are aware of the recent development aiming to relax the assumption (see e.g., Monroe, 2014). While the proposed estimation algorithm can be extended to accommodate a more flexible density for $\boldsymbol{\theta}_i$, we focus on the multivariate normal case in the current paper. We denote $\phi(\boldsymbol{\theta}|\Sigma)$ the density function of $\boldsymbol{\theta}_i$, that is,

$$\phi(\boldsymbol{\theta}|\Sigma) = \frac{1}{\sqrt{|2\pi\Sigma|}} \exp\left(-\frac{1}{2}\boldsymbol{\theta}^\top \Sigma^{-1}\boldsymbol{\theta}\right).$$

To identify the scale of the latent traits, σ_{kk}s are set to be 1, $k = 1, \ldots, K$. Under the confirmatory setting, a_{jk} is constrained to be zero if item j does not measure latent trait k. More precisely, the measurement design is indicated by a pre-specified matrix $Q = (q_{jk})_{J\times K}$, each entry of which takes value 0 or 1. In particular, $q_{jk} = 1$ indicates that item j measures latent trait k and therefore a_{jk} is freely estimated, and $q_{jk} = 0$ indicates that item j does not measure latent trait k and therefore a_{jk} is set to zero. We assume that the test is well designed, so that the latent factors cannot be freely rotated given the zero constraints and consequently the model is identifiable. We refer the readers to Anderson and Rubin (1956) for sufficient conditions on Q that anchor the rotation. For ease of exposition, we use Ψ to denote all the unknown parameters, including a_{jk}s, d_{jy}s, and $\sigma_{kk'}$s.

2.2 Marginal Maximum Likelihood Estimation

Full-information item factor analysis relies on the marginal maximum likelihood estimator (MMLE). More precisely, the MMLE is defined as

$$\hat{\Psi}^{\text{MLE}} = \arg\max_{\Psi} l(\Psi), \tag{3}$$

where

$$l(\Psi) = \sum_{i=1}^{N} \log \left(\int \prod_{j=1}^{J} \frac{\exp(y_{ij}\mathbf{a}_j \cdot \boldsymbol{\theta} + \sum_{l=1}^{y_{ij}} d_{jl})}{1 + \sum_{n=1}^{m_j} \exp(n\mathbf{a}_j \cdot \boldsymbol{\theta} + \sum_{l=1}^{n} d_{jl})} \phi(\boldsymbol{\theta}|\Sigma) d\boldsymbol{\theta} \right)$$

is the marginal log-likelihood of observed data where the latent traits have been marginalized out. Traditionally, the optimization in (3) is solved by the EM algorithm (Bock & Aitkin, 1981; Dempster et al., 1977), which is an iterative algorithm that requires to evaluate N K-dimensional integrals in each iteration. When K is large, numerical integration becomes computationally infeasible, because the computational complexity grows exponentially as the dimensionality of the latent traits increases.

3 Stochastic EM Algorithm

3.1 Stochastic EM Algorithm

Similar to other stochastic algorithms, including the Monte Carlo EM, MH-RM, and SAEM, the StEM algorithm avoids the numerical integration in the optimization of (3) by Monte Carlo simulations. The algorithm iterates between two steps, i.e. the StE step and the M step. Let $\Psi^{(0)}$ be the initial parameter values and $\tilde{\boldsymbol{\theta}}_i^{(0)}, i = 1, \ldots, N$ be the initial values of person parameters. In each step t ($t \geq 1$), the following StE step and M step are performed.

StE step: Sample $\tilde{\boldsymbol{\theta}}_i^{(t)}$ from the conditional density $f(\boldsymbol{\theta}|\mathbf{y}_i, \Psi^{(t-1)})$, where

$$f(\boldsymbol{\theta}|\mathbf{y}_i, \Psi^{(t-1)}) \propto \phi(\boldsymbol{\theta}|\Sigma^{(t-1)}) \prod_{j=1}^{J} \frac{\exp(y_{ij}\mathbf{a}_j^{(t-1)} \cdot \boldsymbol{\theta} + \sum_{l=1}^{y_{ij}} d_{jl}^{(t-1)})}{1 + \sum_{n=1}^{m_j} \exp(n\mathbf{a}_j^{(t-1)} \cdot \boldsymbol{\theta} + \sum_{l=1}^{n} d_{jl}^{(t-1)})}, \tag{4}$$

and the notation "$\propto$" means that the two sides of (4) only differ by a constant that does not depend on $\boldsymbol{\theta}$.

M step: Obtain parameter estimate

$$\Psi^{(t)} = \arg\max_{\Psi} \sum_{i=1}^{N} l(\mathbf{y}_i, \tilde{\boldsymbol{\theta}}_i^{(t)}; \Psi), \tag{5}$$

where

$$\begin{aligned} & l(\mathbf{y}_i, \tilde{\boldsymbol{\theta}}_i^{(t)}; \Psi) \\ &= \log(\phi(\tilde{\boldsymbol{\theta}}_i^{(t)}|\Sigma)) + \sum_{j=1}^{J} \left\{ y_{ij}\tilde{\boldsymbol{\theta}}_i^{(t)} \cdot \mathbf{a}_j + \sum_{l=1}^{y_{ij}} d_{jl} - \log\left(1 + \sum_{n=1}^{m_j} \exp\left(n\tilde{\boldsymbol{\theta}}_i^{(t)} \cdot \mathbf{a}_j + \sum_{l=1}^{n} d_{jl}\right)\right)\right\} \end{aligned}$$

is the complete data log-likelihood of a single observation.

The final estimate of Ψ is given by the average of $\Psi^{(t)}$s from the last m iterations, i.e.,

$$\hat{\Psi} = \frac{1}{m} \sum_{t=T+1}^{T+m} \Psi^{(t)}, \tag{6}$$

for sufficiently large values of T and m. Choosing T and m is the major tuning aspect of the algorithm, as discussed in Section 3.2 and Appendix C. As pointed out by Ip (2002), the single draw at the StE-step could be translated into practical computational saving, as compared to other methods such as the Monte Carlo EM.

The theoretical properties of the StEM algorithm has been studied comprehensively in Nielsen (2000). According to Nielsen (2000), the stochastic EM algorithm has the following properties under suitable regularity conditions.

1. When the observed data $\mathbf{y}_1, \ldots, \mathbf{y}_N$ are viewed as fixed (i.e. conditioned upon), the estimates $\Psi^{(t)}$, $t = 1, 2, \ldots$, obtained from the M step, form a time-homogeneous Markov chain. Moreover, the Markov chain is ergodic.

2. For ease of exposition, we add subscript N to the output from each M step, $\Psi_N^{(t)}$, to emphasize its dependence on data. As the number of iterations t goes to infinity, $\Psi_N^{(t)}$ converges in distribution to a random variable $\tilde{\Psi}_N$, satisfying $\tilde{\Psi}_N = \Psi^* + O_p(\frac{1}{\sqrt{N}})$, where Ψ^* is the true parameter.

3. Moreover, for a fixed large value of m and sufficiently large burn-in size $T = T_N$, the

final estimate (6), $\hat{\Psi}_N = \frac{1}{m}\sum_{t=T_N+1}^{T_N+m} \Psi_N^{(t)}$, performs similarly as the MMLE for sufficiently large N. More precisely, $\sqrt{N}(\hat{\Psi}_N - \Psi^*)$ is asymptotically normal (as $N \to \infty$) with mean zero and variance $V(\Psi^*)$ satisfying

$$\|V(\Psi^*) - \mathcal{I}(\Psi^*)^{-1}\| \leq \frac{C^*}{m}. \tag{7}$$

Here, $\mathcal{I}(\Psi^*)$ is the Fisher information under the marginal likelihood based on one observation and thus $\mathcal{I}^{-1}(\Psi^*)$ is the asymptotic variance of $\sqrt{N}(\hat{\Psi}_N^{\text{MLE}} - \Psi^*)$, where $\hat{\Psi}_N^{\text{MLE}}$ is the MMLE. C^* is a positive constant that only depends on the true model parameters and the norm is the matrix L_2 norm. Equation (7) implies that the asymptotic variance of $\hat{\Psi}$ (after scaled by $\sqrt{N}$) is close to that of the MMLE, with a gap bounded by C^*/m. The gap comes from the use of simulation in the StE step. For sufficiently large m, this gap is negligible and $\hat{\Psi}$ can be used as the MMLE $\hat{\Psi}_N^{\text{MLE}}$.

3.2 Implementation Details

In what follows, we describe the implementation details of the algorithm.

Gibbs sampler. In the StE-step of each iteration, we sample $\tilde{\boldsymbol{\theta}}_i^{(t)}$ from the conditional distribution $F(\boldsymbol{\theta}|\mathbf{y}_i, \Psi^{(t-1)})$ by one Gibbs sampling iteration. That is, we iterate over $k = 1, \ldots, K$. For each k, sample $\tilde{\theta}_{ik}^{(t)}$ from $f(\theta_k|\mathbf{y}_i, \Psi^{(t-1)}, \tilde{\boldsymbol{\theta}}_{i,-k}^{(t)})$, where $f(\theta_k|\mathbf{y}_i, \Psi^{(t-1)}, \tilde{\boldsymbol{\theta}}_{i,-k}^{(t)})$ is the conditional distribution of θ_k given $\mathbf{Y}_i = \mathbf{y}_i$ and $\boldsymbol{\theta}_{i,-k} = \tilde{\boldsymbol{\theta}}_{i,-k}^{(t)} = (\tilde{\theta}_{i1}^{(t)}, \ldots, \tilde{\theta}_{i,k-1}^{(t)}, \tilde{\theta}_{i,k+1}^{(t-1)}, \ldots, \tilde{\theta}_{i,K}^{(t-1)})$ and parameters $\Psi^{(t-1)}$ are from the previous step. Adaptive rejection sampling (Gilks & Wild, 1992) is used, where a piecewise log-linear density function is constructed adaptively as the proposal distribution (i.e. envelop). This adaptive rejection sampler is computationally efficient, because the proposal distribution is effectively constructed by making use of the fact that $f(\theta_k|\mathbf{y}_i, \Psi^{(t-1)}, \tilde{\boldsymbol{\theta}}_{i,-k}^{(t)})$ is log-concave thanks to the exponential family form of (2). The details of this adaptive rejection sampler are provided in Appendix A.

Optimization in the M step. The M step optimizes

$$\Psi^{(t)} = \arg\max_{\Psi} \sum_{i=1}^{N} l(\mathbf{y}_i, \tilde{\boldsymbol{\theta}}_i^{(t)}; \Psi),$$

where $\Psi^{(t)}$ includes covariance $\sigma_{kk'}$s, slope a_{jk}s and intercept parameters d_js. Due to the separable form of the objective function, the optimization splits into the following problems:

$$\hat{\Sigma}^{(t)} = \arg\max_{\Sigma} \sum_{i=1}^{N} \log(\phi(\tilde{\boldsymbol{\theta}}_i^{(t)}|\Sigma)),\ \Sigma \succeq 0,\ \sigma_{kk} = 1, k = 1, \ldots, K, \tag{8}$$

$$\begin{aligned}(\mathbf{a}_j^{(t)}, \mathbf{d}_j^{(t)}) = &\arg\max_{\mathbf{a}_j, \mathbf{d}_j} \left\{ \sum_{i=1}^{N} y_{ij} \tilde{\boldsymbol{\theta}}_i^{(t)} \cdot \mathbf{a}_j + \sum_{l=1}^{y_{ij}} d_{jl} - \log\left(1 + \sum_{n=1}^{m_j} \exp\left(n\tilde{\boldsymbol{\theta}}_i^{(t)} \cdot \mathbf{a}_j + \sum_{l=1}^{n} d_{jl}\right)\right)\right\}, \\ &\text{s.t. } a_{jk}^{(t)} = 0,\ \text{if } q_{jk} = 0, k = 1, \ldots, K, j = 1, \ldots, J\end{aligned} \tag{9}$$

where $\Sigma \succeq 0$ denotes that Σ is positive semi-definite and q_{jk}s are entries of the pre-specified design matrix Q. Optimization problem (9) is a smooth low-dimensional convex optimization problem, with trivial constraints that some a_{jk}s are set to be zero due to the measurement design. Such a problem can be handled by many standard numerical solvers and in particular, we solve it by the limited-memory BFGS algorithm (D. C. Liu & Nocedal, 1989).

The optimization of (8) is a convex optimization problem with equality constraints and a positive semi-definite constraint. It is known as a semidefinite programming (SDP) problem in convex optimization. Making use of recent advances in convex optimization, we propose a proximal gradient descent algorithm (Parikh et al., 2014) for solving (8). The details of the proposed proximal gradient descent algorithm are provided in Appendix B.

Determining T and m. As mentioned earlier, the estimates $\Psi^{(t)}$, $t = 1, 2, \ldots,$ obtained from the M-steps form an ergodic Markov chain, similar to the posterior samples of parameters from a Markov Chain Monte Carlo (MCMC) algorithm under a Bayesian setting. Under this connection, we call the value of T in (6) as the burn-in size. This is because in our final estimate $\hat{\Psi}$, an initial portion ($\Psi^{(1)}$, $\ldots$, $\Psi^{(T)}$) of the Markov chain is discarded, so that the effect of the initial value is minimized. Methods for determining the

burn-in of MCMC algorithms can be adopted here for determining the value of T, such as the Geweke statistic (Geweke, 1992) and the Gelman-Rubin statistic (Gelman & Rubin, 1992). In our implementation, the value of T is determined using a batch procedure based on the Geweke statistic. See Appendix C for the details.

We determine the value of m based on a similar batch procedure, after the determination of burn-in size T. More precisely, we choose m such that the conditional variance of each entry of $\hat{\Psi}_N = \frac{1}{m}\sum_{t=T_N+1}^{T_N+m} \Psi_N^{(t)}$ given the the observed data $\mathbf{y}_1, \ldots \mathbf{y}_N$ falls below a pre-specified threshold. The conditional variance is estimated by a batch variance procedure (Roberts, 1996), a standard approach in the MCMC literature that takes into account the autocorrelation between samples from the Markov chain. See Appendix C for the details.

3.3 Standard Error of Parameter Estimation

Once the final point estimate $\hat{\Psi}$ is obtained from the StEM algorithm, we use the missing information identity (Louis, 1982) to compute the Fisher information of observed data, based on which standard errors of parameter estimates are obtained. According to the missing information identity, the observed data Fisher information is

$$\begin{aligned} I(\hat{\Psi}) = -\frac{\partial^2 l(\Psi)}{\partial\Psi\partial\Psi^T}|_{\Psi=\hat{\Psi}} = &\sum_{i=1}^{N} E(H(\mathbf{y}_i, \boldsymbol{\theta}_i; \hat{\Psi}) - s(\mathbf{y}_i, \boldsymbol{\theta}_i; \Psi)[s(\mathbf{y}_i, \boldsymbol{\theta}_i; \hat{\Psi})]^\top | \mathbf{Y}_i = \mathbf{y}_i) \\ &+ E(s(\mathbf{y}_i, \boldsymbol{\theta}_i; \hat{\Psi})|\mathbf{Y}_i = \mathbf{y}_i) E([s(\mathbf{y}_i, \boldsymbol{\theta}_i; \hat{\Psi})]^\top | \mathbf{Y}_i = \mathbf{y}_i), \end{aligned} \tag{10}$$

where

$$H(\mathbf{y}_i, \boldsymbol{\theta}_i; \Psi) = -\frac{\partial^2 l(\mathbf{y}_i, \boldsymbol{\theta}_i; \Psi)}{\partial\Psi\partial\Psi^\top} \text{ and } s(\mathbf{y}_i, \boldsymbol{\theta}_i; \Psi) = \frac{\partial l(\mathbf{y}_i, \boldsymbol{\theta}_i; \Psi)}{\partial\Psi}$$

are the complete data information matrix and score of observation i, respectively. The expectation is with respect to the conditional distribution of $\boldsymbol{\theta}_i$ given $\mathbf{Y}_i = \mathbf{y}_i$, under the model with parameters $\hat{\Psi}$. In particular, we use Monte Carlo integrations to evaluate the conditional expectation in (10), based on posterior samples of $\boldsymbol{\theta}_i$ from the Gibbs sampler described in Section 3.2 and Appendix A. Notice that standard errors of parameter estimates can be obtained by making use of (10), because $I(\hat{\Psi})/N$ converges to $\mathcal{I}(\Psi^*)$

when $\hat{\Psi}$ is a consistent estimate. According to our discussion after equation (7), $\mathcal{I}(\Psi^*)$ is the per-observation Fisher information based on a single observation, so that $\mathcal{I}^{-1}(\Psi^*)$ is the asymptotic variance of $\sqrt{N}(\hat{\Psi}_N^{\text{MLE}} - \Psi^*)$.

4 Discussions

4.1 Alternative Gibbs Sampler

The use of adaptive rejection sampling in the StE step relies on the log-concavity of the conditional distribution $f(\theta_k|\mathbf{y}_i, \Psi^{(t-1)}, \tilde{\boldsymbol{\theta}}_{i,-k}^{(t)})$, which is true for many popular IRT and MIRT models, such as the Rasch model (Rasch, 1960), two-parameter logistic model (Birnbaum, 1968), the partial credit model (Masters, 1982), the nominal response model (Bock, 1972), and their multidimensional extensions (Revuelta, 2014; Yao & Schwarz, 2006). When the log-concavity is not satisfied, for example, when noncompensatory MIRT models (e.g., Chapter 4, Reckase, 2009) are considered, alternative methods are available for sampling from $f(\theta_k|\mathbf{y}_i, \Psi^{(t-1)}, \tilde{\boldsymbol{\theta}}_{i,-k}^{(t)})$. In particular, the slice sampler may be a good choice, which is also virtually tuning free and tailors to the form of the target distribution (Neal, 2003). See Neal (2003) for the implementation details.

4.2 Comparison with MH-RM Algorithm

The StEM algorithm is similar to the MH-RM algorithm to some extent. Specifically, both algorithms avoid the numerical integration in the classical EM algorithm by Monte Carlo simulation. In what follows, we list the key differences between the two methods, based on which we argue that the proposed StEM algorithm may be a better choice when N, J, and K are all relatively large. The advantage of the proposed algorithm is further confirmed by simulation results in Section 5.

1. The MH-RM algorithm iteratively updates the parameter estimates by a Robbins-Monro procedure (Robbins & Monro, 1951) and use the parameter estimates in the last update upon convergence as the final estimate. This procedure requires the specification of the step size γ_t in each iteration t, where γ_t satisfies $\sum_{t=1}^{\infty} \gamma_t = \infty$

and $\sum_{t=1}^{\infty} \gamma_t^2 < \infty$. In particular, γ_t is set to be $1/t$ in Cai (2010a). As reported by many studies on the Robbins-Monro stochastic approximation method, algorithms based on the Robbins-Monro update are sensitive to the choice of the step size and thus its performance can be unstable upon implementation (Nemirovski et al., 2009; Spall, 2005). On the other hand, the StEM algorithm does not involve a Robbins-Monro update and thus does not suffer from this issue.

2. In addition, little research has been done on the stopping rule of Robbins-Monro type algorithms (e.g. Wada & Fujisaki, 2015). Since the step size γ_t converges to zero as the number of iteration t grows, early stopping may occur if one terminates the algorithm when the difference between two subsequent updates falls below a certain threshold. On the other hand, the stopping rule of the proposed StEM algorithm is supported by the asymptotic property of the outputs $\Psi^{(t)}$ from the M steps.

3. The MH-RM algorithm samples from $f(\boldsymbol{\theta}|\mathbf{y}_i, \Psi^{(t-1)})$ using a random-walk Metropolis-Hastings (MH) sampler. The random-walk MH sampler requires the specification of a tuning parameter, which is the step-size of the random walk. The performance of the algorithm is sensitive to the choice of this tuning parameter and choosing a good tuning parameter is not an easy task (Neal, 2003). The Gibbs sampler adopted here avoids this issue by making use of adaptive rejection sampling.

4. Finally, as discussed in Sections 4.4-4.6, the StEM algorithm can be easily extended to more complex settings which require solving optimization problems with constraints or nonsmooth penalties. On the other hand, the generalization of the MH-RM algorithm to such problems is more challenging, since the Robbins-Monro update does not handle inequality constraints or nonsmooth objective functions. Specifically, the MH-RM algorithm implemented in the **mirt** R package fails when the latent dimension $K = 20$, because non-positive definite estimates of the latent variable covariance matrix are produced in intermediate Robbins-Monro iterations.

For the current problem, one may modify the MH-RM algorithm by reparameterizing $\Sigma = BB^\top$, where B is a lower triangle matrix, and estimate B instead[2]. However, this solution is not flexible enough. For example, it does not handle confirmatory IFA with additional zero constraints (i.e., independence between some traits) in the covariance matrix, while as discussed in Section 4.5, the StEM algorithm can be easily adapted to deal with such constraints.

4.3 Parallel Computing

Thanks to the special model structure of MIRT models, computational algorithms for MIRT estimation can typically be speeded up through parallel computing (Cai, 2013; von Davier, 2016). Similar to the MH-RM algorithm, both the StE-step and M step of this StEM algorithm can be paralleled and therefore the algorithm can be substantially speeded up by parallel computing. More precisely, in the StE step, $\tilde{\boldsymbol{\theta}}_i^{(t)}$s can be sampled in parallel for different individuals and in the M step, $(\mathbf{a}_j, \mathbf{d}_j)$, $j = 1, \ldots, J$, and Σ can all be updated in parallel. This is known as an "embarrassingly parallel" structure, for which little effort is needed to separate the problem into a number of parallel tasks (Herlihy & Shavit, 2011). This feature of the algorithm makes it scalable to very large-scale data when the computing environment supports multiple processors. For the proposed algorithm, parallel computing is implemented through an Open Multi-Processing (OpenMP; Dagum & Menon, 1998) application programming interface.

4.4 Extension: Exploratory IFA

Thanks to the simple procedure of the StEM algorithm, the algorithm can be generalized to solving many other problems. In particular, an StEM algorithm can be used to solve the optimization for the L_1 regularized estimator for exploratory IFA (Sun, Chen, Liu, Ying, & Xin, 2016). Specifically, under the exploratory IFA setting, no Q-matrix is pre-specified and thus no constraint is imposed on the slope parameters a_{jk}. To impose a

[2]Since this reparametrization does not guarantee $\sigma_{kk} = 1$, $k = 1, \ldots, K$, some of the slope parameters have to be constrained to be 1 to ensure the identifiability of the model.

simple structure on the slopes, Sun et al. (2016) propose a L_1 regularized maximum likelihood estimator, under which many of the a_{jk}s are estimated to be zero. In other words, the L_1 regularized estimator automatically rotates the factors to achieve a sparse slope structure. More precisely, under the MGPC model, the L_1 regularized maximum likelihood estimator is obtained by solving the following optimization problem

$$\begin{aligned} \hat{\Psi}^{\lambda} = \arg\max_{\Psi} \quad & l(\Psi) - \lambda \sum_{j=1}^{J} \sum_{k=1}^{K} |a_{jk}|, \\ s.t.\ & \Sigma \succeq 0, \sigma_{kk} = 1, k = 1, 2, \ldots, K, \end{aligned} \tag{11}$$

where λ is a positive tuning parameter.

With slight modification, the algorithm described in Section 3.1 can be used to solve (11). Specifically, the StE step and the optimization for Σ in the M step remain the same. The optimization for the item parameters in (9) becomes an L_1 regularized regression problem, which can be solved using either a coordinate decent algorithm (Friedman, Hastie, & Tibshirani, 2010) or a proximal algorithm (Parikh et al., 2014).

4.5 Extension: Constraints on Σ

In confirmatory IFA, it is usually of interest to study the relationship between the latent traits by comparing models where different zero constraints are imposed on the covariance Σ. For example, one may test the independence between traits 1 and 2 by comparing two models, where one constrains $\sigma_{12} = \sigma_{21} = 0$ and the other does not. Such constraints can be easily handled using an StEM algorithm.

More precisely, let E denote the set of constraints $E = \{(k, k') : \sigma_{kk'} = 0\}$. To incorporate these constraints in the estimator, only the update of Σ in (8) of the M-step needs to be modified. Instead of solving (8), we solve

$$\begin{aligned} \max_{\Sigma} & \sum_{i=1}^{N} \log(\phi(\tilde{\boldsymbol{\theta}}_i^{(t)} | \Sigma)), \\ s.t.\ & \Sigma \succeq 0,\ \sigma_{kk} = 1, k = 1, \ldots, K, \\ & \sigma_{kk'} = 0, (k, k') \in E, \end{aligned}$$

which is again an SDP problem that can be solved similarly using a proximal gradient descent algorithm.

4.6 Extension: Latent Regression

The StEM algorithm can also be extended to solving latent regression item response theory models, which are two-level latent variable models in which covariates serve as predictors of the conditional distribution of the latent traits (Camilli & Fox, 2015; von Davier & Sinharay, 2010). In such models, each respondent is associated with p covariates, denoted by $\mathbf{x}_i = (x_{i1}, \ldots, x_{ip})$. When adopting the MGPC model as the measurement model, the latent regression item response theory model differs from the model described in Section 2.1 by assuming

$$\boldsymbol{\theta}_i \sim N(\Gamma \mathbf{x}_i, \Sigma),$$

where Γ is $K \times p$ matrix containing all the regression coefficients. Moreover, when p is large, L_1 and/or L_2 regularization can be imposed on Γ, for the purposes of model selection, handling the collinearity among covariates, etc. The StEM algorithm can be easily modified to estimate $\mathbf{a}_j$, $\mathbf{d}_j$, $j = 1, \ldots, J$, Σ and Γ under this model, even with the presence of L_1 and/or L_2 regularization.

5 Simulation Study[3]

5.1 Study I

In this study, we verify the theoretical properties of the StEM algorithm using a small Monte Carlo simulation study. In particular, we compare the StEM algorithm with the classical EM algorithm (Bock & Aitkin, 1981) in terms of parameter recovery.

We consider a single latent trait ($K = 1$), ten items ($J = 10$), and sample sizes $N = 500, 1000$, and 2000. We further assume all the responses are binary (i.e., $m_j = 1$) and thus the MGPC model becomes the M2PL model. For ease of exposition, we simplify the

[3]All simulations, except for the comparison between the proposed algorithm and the MH-RM algorithm implemented in the **flexMIRT** software in Section 5.2, are conducted on Intel(R) machines with the specifications: Xeon(R) CPU E5-2687W v4 @ 3.00GHz; R version 3.4.1 (2017-06-30); gcc version 4.8.5 20150623 (Red Hat 4.8.5-16).

notation, $a_j = a_{j1}$ and $d_j = d_{j1}$. The parameters a_j and d_j are generated once from the uniform distribution over the interval $(0.5, 1.5)$ and the standard normal distribution, respectively; see Table 1 for their true values. For each sample size, we generate 500 independent data sets and fit each data set with the proposed StEM algorithm and the classical EM algorithm. We adopt the implementation of the EM algorithm in the `mirt` package (Chalmers, 2012) in statistical software R. Moreover, the EM algorithm is implemented with 100 quadrature points for the E step, so that the approximation error in the numerical integrals of the E step is negligible.

Table 1
True values of parameters in study I.

	a_1^*	a_2^*	a_3^*	a_4^*	a_5^*	a_6^*	a_7^*	a_8^*	a_9^*	a_{10}^*
True	1.39	1.40	0.73	0.55	0.71	1.05	0.92	0.50	0.68	0.90
	d_1^*	d_2^*	d_3^*	d_4^*	d_5^*	d_6^*	d_7^*	d_8^*	d_9^*	d_{10}^*
True	1.55	-1.04	0.60	1.10	0.27	0.19	-0.67	0.45	0.13	1.20

Results are summarized in Figures 1-3. In Figure 1, the mean squared errors (MSE) for the a_j and d_j parameters over 500 replications are presented. For example, the MSE for a_1 is computed as

$$\frac{1}{500}\sum_{i=1}^{500}(\hat{a}_1^{(i)} - a_1^*)^2,$$

where $\hat{a}_1^{(i)}$ is the estimate of a_1 from the ith replication. The left, middle, and right panels of Figure 1 correspond to the three sample sizes, $N = 500$, 1000, and 2000, respectively. In each panel, the boxplots with labels "StEM.A", and "StEM.d" are based on the MSEs for $a_1, \ldots, a_{10}$ and $d_1, \ldots, d_{10}$, respectively, from the StEM algorithm. Similarly, the boxplots with labels "EM.A", and "EM.d" are based on results from the EM algorithm. According to this plot, it is found that the MSEs of the parameter estimation based on the StEM algorithm are very close to, if slightly larger than, the oracle ones based on the EM algorithm.

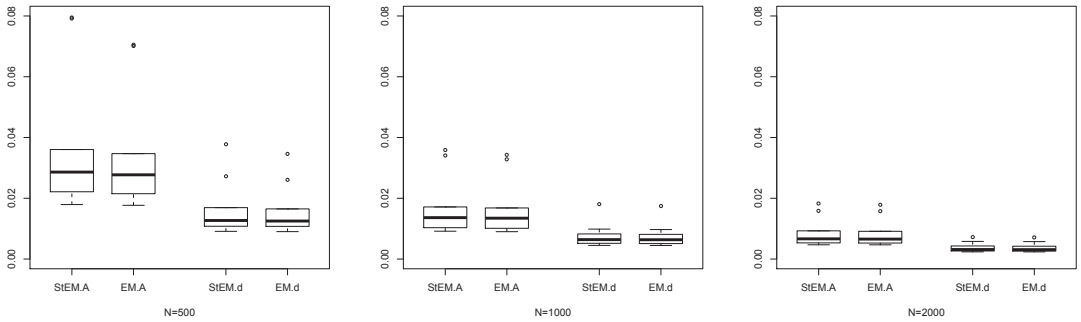

Figure 1. The boxplot of MSEs of slope parameters a_j and intercept parameters d_j for the StEM and the EM algorithm. Left: $N = 500$; Middle: $N = 1000$; Right: $N = 2000$.

Second, Figure 2 shows the mean squared differences (MSD) between the estimate from the StEM algorithm and that from the EM algorithm. For example, the MSD for parameter a_1 is defined as

$$\frac{1}{500}\sum_{i=1}^{500}(a_{1,StEM}^{(i)} - a_{1,EM}^{(i)})^2,$$

where $a_{1,StEM}^{(i)}$ and $a_{1,EM}^{(i)}$ denote the StEM and EM estimates of a_1, respectively, based on data from the ith replication. The left panel of Figure 2 corresponds to the MSDs for the a_j parameters and the right panel corresponds to those for the d_j parameters. In each panel, a boxplot corresponds to a sample size, as indicated by its label. According to these boxplots, the MSDs are negligible comparing to the values of MSEs given in Figure 1. According to Figures 1-2, under all sample sizes and for all the model parameters, the point estimation given by the StEM algorithm and that given by the EM algorithm are almost the same.

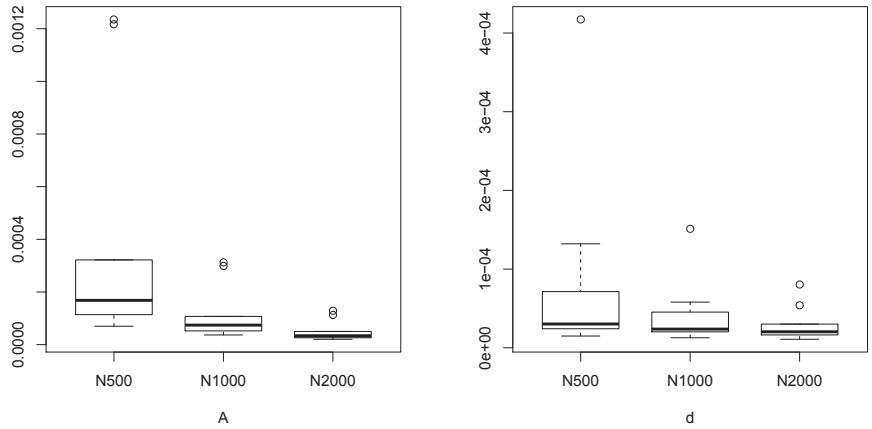

Figure 2. The boxplot of mean squared differences of slope parameters a_j and intercept parameters d_j between the StEM and the EM algorithm under different sample size N. Left: mean squared differences of slope parameters. Right: mean squared differences of intercept parameters.

Finally, for the purpose of illustration, we present the results on the selection of T and m, two important parameters of the StEM algorithm that affect the accuracy of parameter estimation. In Figures 3 and 4, the histograms of the selected m and T are presented, respectively, under different sample sizes. According to these figures, both the selected m and T vary in different replications.

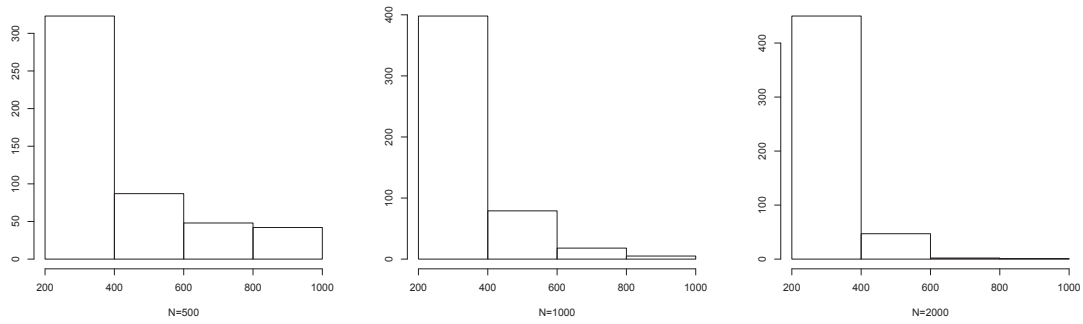

Figure 3. The histogram of selected m in the StEM algorithm. Left: $N = 500$; Middle: $N = 1000$; Right: $N = 2000$.

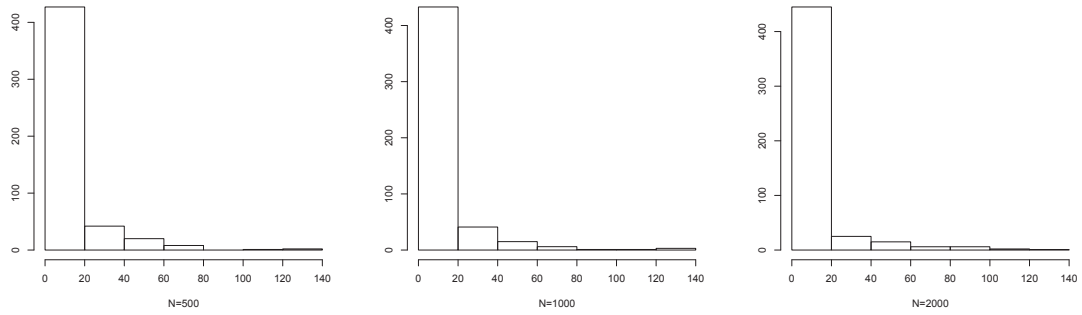

Figure 4. The histogram of selected burn in length T in the StEM algorithm. Left: $N = 500$; Middle: $N = 1000$; Right: $N = 2000$.

5.2 Study II

In the second study, we compare the proposed StEM algorithm and the MH-RM algorithm under settings where the dimensionality of the latent space is high ($K = 10, 20$), where two implementations of the MH-RM algorithm are adopted including the one implemented in the **mirt** R package that is mainly written in programming language R and the other implemented in the **flexMIRT** software that is written in C++. In this comparison, the proposed StEM algorithm is implemented in R with core functions written in C++. For these high-dimensional settings, the classical EM algorithm is computationally infeasible. For fairness, all the algorithms are compared within a single core, which does not allow for parallel computing.

We first compare the proposed algorithm and the MH-RM algorithm implemented in the **mirt** package, under two settings: (1) $K = 10, J = 100$, and $N = 2000$, and (2) $K = 20, J = 200$, and $N = 2000$. We consider a simple confirmatory design, where each latent trait is measured by 10 items. That is, items 1-10 measure latent trait 1, items 11-20 measure latent trait 2, and so forth. The true nonzero slope parameters a_{jk} and the intercept parameters d_j are generated once from the uniform distribution over the interval (0.5, 1.5) and the standard normal distribution, respectively. The latent traits $\boldsymbol{\theta}_i$ are generated from a multivariate normal distribution with mean zero and covariance matrix

Σ, where $\sigma_{kk} = 1$, and $\sigma_{kk'} = 0.6$, $k \neq k'$, $k, k' = 1, \ldots, K$.

The comparison between the two algorithms is based on 100 replications for each setting. The results on the estimation precision and computation time are presented in Figures 5 and 6. According to these results, the two algorithms have similar accuracy on the slope and intercept parameters, but the proposed algorithm is substantially more accurate in estimating the correlations among the latent traits. In addition, to achieve this accuracy, the StEM algorithm is about 2.5 times faster than the MH-RM implementation in the **mirt** package. For setting (2), no result is obtained for the MH-RM algorithm for any of the replications, with an error message "MH sampler failed", which is possibly due to that non-positive definite estimates of the latent variable covariance matrix Σ are produced in intermediate Robbins-Monro iterations. On the other hand, valid results are obtained from the StEM algorithm for all the replications; See Figures 7-8 for the results on its accuracy and computation time.

We then compare the proposed algorithm with the MH-RM algorithm implemented in the **flexMIRT** software. Since the **flexMIRT** software can only be run in the Microsoft Windows system, the comparison is conducted under a Windows system based on five replications for each setting[4]. Results are given in Table 2 and Figure 9-10. It is worth noting that the MH-RM algorithm implemented in **flexMIRT** is able to provide valid estimates even when $K = 20$, which may be due to that the **flexMIRT** has a better implementation of the MH-RM algorithm that takes the positive definiteness of the Σ matrix into account. When $K = 10$, based on the five replications, the two algorithms achieve similar accuracy within comparable computation time. When $K = 20$, our algorithm takes substantially less time to achieve a similar accuracy level.

[4]Both algorithms are conducted on a personal computer with specifications: Processor 2.2 GHz Intel Core i7; Memory 8 GB 1600 MHz DDR3.

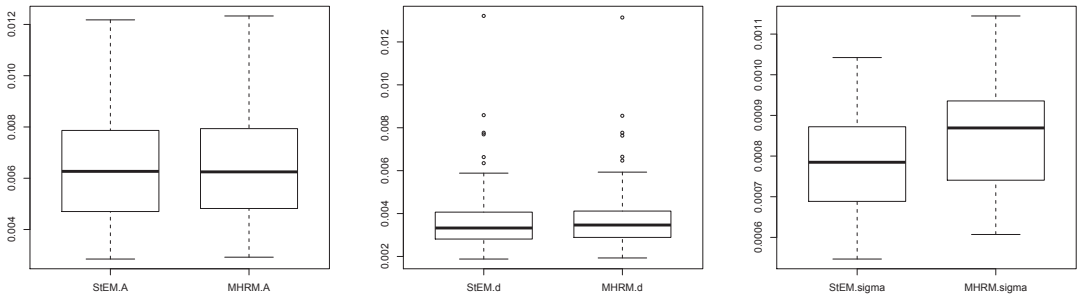

Figure 5. The boxplot of MSEs of parameters estimation for the StEM and the MH-RM algorithm implemented in the **mirt** package when the latent dimension is 10. Left: MSEs of non-zero slope parameters a_j; Middle: MSEs of intercept parameters d_j; Right: MSEs of correlation parameters σ_{ij}.

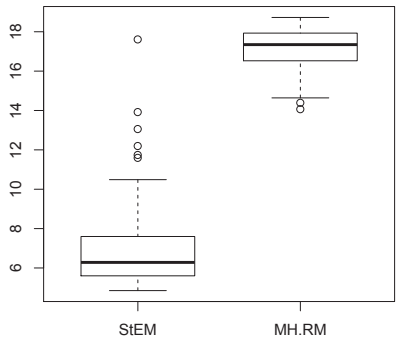

Figure 6. The boxplot of elapsed time (minutes) of the estimation procedure for the StEM and the MH-RM algorithm implemented in the **mirt** package when the latent dimension is 10.

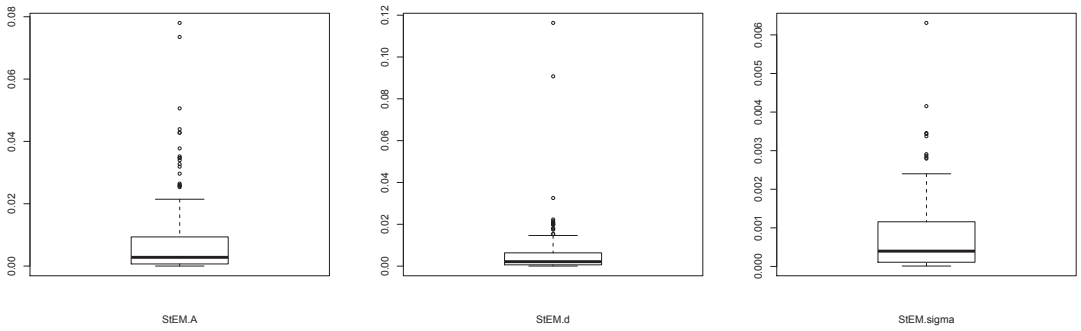

Figure 7. The boxplot of MSEs of parameters estimation for the StEM when the latent dimension is 20. Left: MSEs of non-zero slope parameters a_j; Middle: MSEs of intercept parameters d_j; Right: MSEs of correlation parameters σ_{ij}.

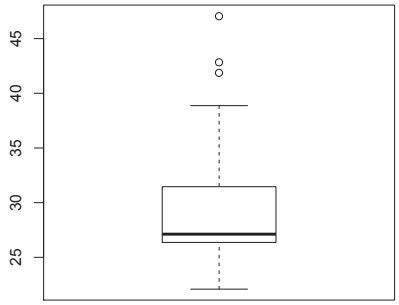

Figure 8. The boxplot of elapsed time (minutes) of the estimation procedure for the StEM when the latent dimension is 20.

Table 2
The computation time (minutes) for the estimation based on the StEM and MH-RM algorithms implemented in **flexMIRT**.

	1	2	3	4	5
StEM (K=10)	2.5	3.2	5.8	2.9	3.1
MH-RM (K=10)	3.2	2.2	6.0	2.4	4.8
StEM (K=20)	20.3	19.0	18.7	15.4	17.1
MH-RM (K=20)	82.8	48.9	93.3	70.7	64.6

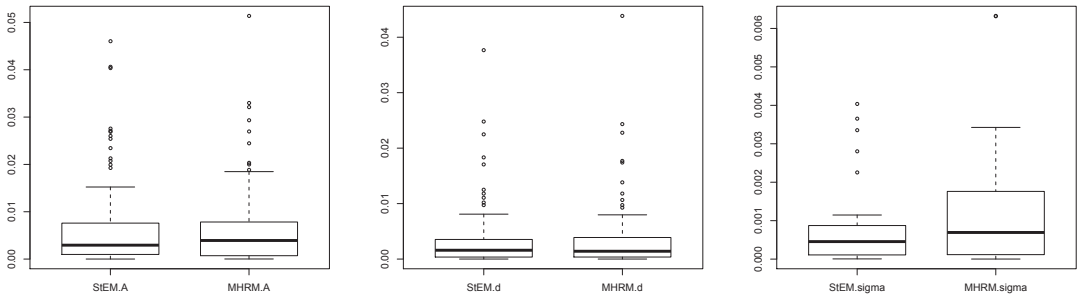

Figure 9. The boxplot of MSEs of the estimated parameters for the StEM and the MH-RM algorithm implemented in the **flexMIRT** software when the latent dimension is 10. Left: MSEs of non-zero slope parameters a_j; Middle: MSEs of intercept parameters d_j; Right: MSEs of correlation parameters σ_{ij}.

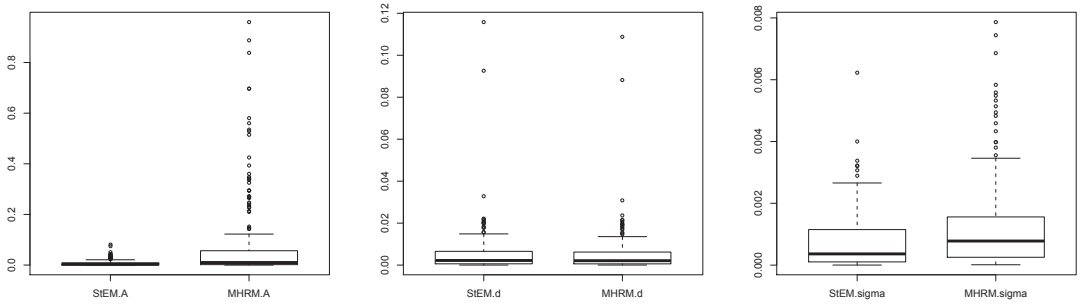

Figure 10. The boxplot of MSEs of the estimated parameters for the StEM and the MH-RM algorithm implemented in the **flexMIRT** software when the latent dimension is 20. Left: MSEs of non-zero slope parameters a_j; Middle: MSEs of intercept parameters d_j; Right: MSEs of correlation parameters σ_{ij}.

6 Application to Big Five Personality Test

We further illustrate the use of the proposed algorithm through an application to a personality assessment dataset based on an International Personality Item Pool (IPIP) NEO personality inventory (Johnson, 2014). This inventory is a public-domain version of the widely used NEO personality inventory (Costa & McCrae, 1985), which is designed to measure the big five personality factors, including Neuroticism (N), Agreeableness (A), Extraversion (E), Openness to experience (O), and Conscientiousness (C). According to

(Johnson, 2014), each personality factor can be further split into six personality facets, resulting in 30 facets. For example, the Neuroticism factor is split into (N1) anxiety, (N2) anger, (N3) depression, (N4) self-consciousness, (N5) immoderation, and (N6) vulnerability. A list of the thirty personality facets is provided in Table 3.

The dataset was collected via the Web (Johnson, 2005), containing 20,993 participants and 300 items[5]. We analyzed a subset of this dataset, containing data from 7,325 participants who completed all the items. All the 30 personality facets are measured, with each facet measured by 10 items. All the items are on a five-category rating scale. An example item is "Worry about things", and the response categories are "Very Inaccurate", "Moderately Inaccurate", "Neither Accurate nor Inaccurate", "Moderately Accurate", and "Very Accurate". Reverse-worded items were reversely recorded $(1 \to 5, 2 \to 4, 4 \to 2, 5 \to 1)$ at the time the respondent completed the inventory. Based on the structure of data, we fit a thirty dimensional MGPC model where each factor represents a facet. The path diagram of the model is visualized in Figure 11.

Table 3
Interpretation of the thirty personality facets in IPIP-NEO inventory.

Facet		Facet		Facet	
N1	Anxiety	A1	Trust	E1	Friendliness
N2	Anger	A2	Morality	E2	Gregariousness
N3	Depression	A3	Altruism	E3	Assertiveness
N4	Self-Consciousness	A4	Cooperation	E4	Activity Level
N5	Immoderation	A5	Modesty	E5	Excitement-Seeking
N6	Vulnerability	A6	Sympathy	E6	Cheerfulness
Facet		Facet			
O1	Imagination	C1	Self-Efficacy		
O2	Artistic Interests	C2	Orderliness		
O3	Emotionality	C3	Dutifulness		
O4	Adventurousness	C4	Achievement-Striving		
O5	Intellect	C5	Self-Discipline		
O6	Liberalism	C6	Cautiousness		

[5]The dataset and items can be downloaded from `https://osf.io/tbmh5/`.

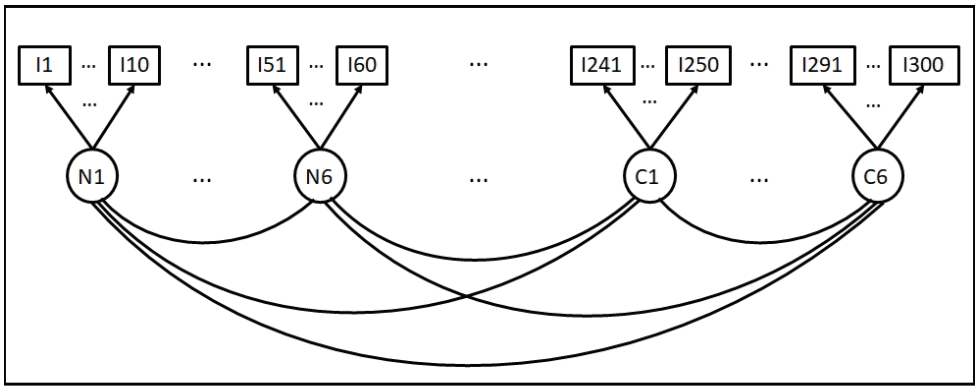

Figure 11. Visualization of an MIRT model with thirty latent traits fitted to the NEO data.

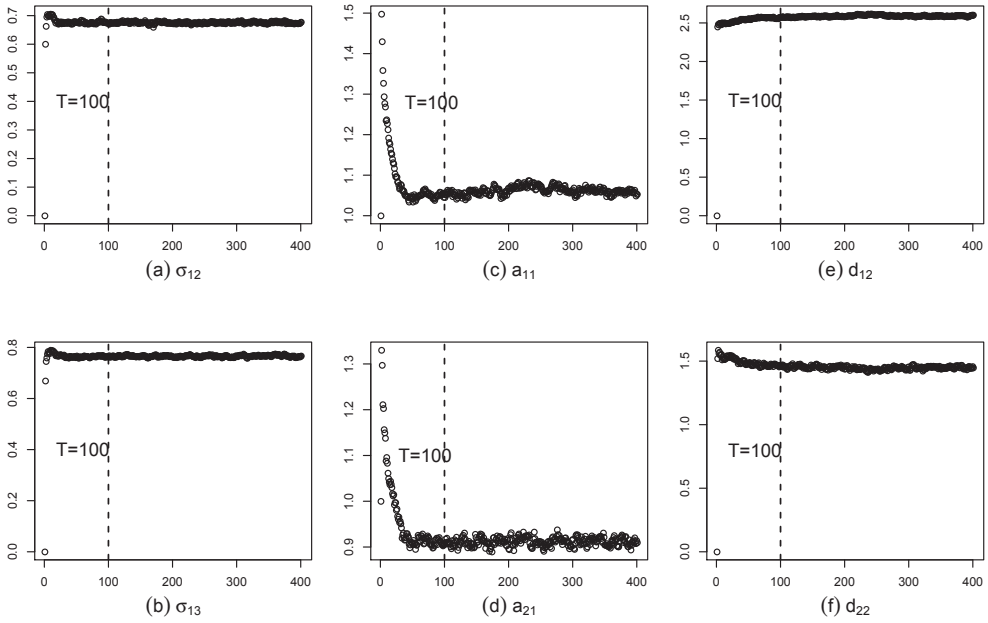

Figure 12. The Markov chain of estimated parameters. Left (a)-(b): correlation parameters σ_{12}, σ_{13}; Middle (c)-(d): slope parameters a_{11}, a_{21}; Right (e)-(f): slope parameters d_{21}, d_{22}.

We provide a discussion on the details of the results. Using parallel computing, the algorithm converges within 32 minutes on a computer cluster with 24 cores[6]. The burn-in size T and the length m of the effective Markov chain are chosen as $T = 100, m = 300$ by

[6]The real data analysis is conducted on an Intel(R) machine with the specifications: Xeon(R) CPU E5-2687W v4 @ 3.00GHz; R version 3.4.1 (2017-06-30); gcc version 4.8.5 20150623 (Red Hat 4.8.5-16).

the proposed procedure, respectively. The dynamic process of the StEM algorithm is visualized in Figure 12, where the Markov chains of the correlation σ_{12} and σ_{13}, the slope parameters a_{11} and a_{21} and the intercept parameters d_{12} and d_{22} are shown as illustrative examples in panels (a)-(f), respectively. In these plots, the x-axis shows the iteration number t and the y-axis shows the parameter value. As we can see, all these Markov chains stabilize quickly after a few StEM iterations. After the burn-in size $T = 100$, which is chosen by the proposed procedure, the effect of the starting points seems to be negligible.

As shown in Table 4, all the estimated slopes are positive, which is consistent with the confirmatory design of the measurement scale. The values of estimated slope parameters vary substantially, implying the heterogenous psychometric properties of the items. The estimated correlation matrix among the 30 latent traits is visualized in Figure 13. It is observed that bubbles within the diagonal blocks formed by facets belonging to the same personality factor are mostly large and in black color, indicating strong positive associations thereof. The Neuroticism facets tend to be negatively correlated with most of the facets within the other four factors, with only a few exceptions (e.g., with A5 "Modesty" and O3 "Emotionality"). In addition, most of the Agreeableness and Conscientiousness facets are positively correlated, and most of the Extraversion and Openness facets are positively correlated as well. These overall patterns of inter-factor correlations echo existing findings in the literature of Big-five personality (e.g., Steel, Schmidt, & Shultz, 2008).

To the best of our knowledge, this is the first time that the 30 facets of the NEO personality inventory are simultaneously analyzed at the item level, while previous studies analyzed the latent structure of the 30 facets based on the total scores of the corresponding scales (e.g., Johnson, 2005). Our analysis has a few advantages. First, by making use of item level data and taking into account the ordinal nature of the items, the 30 facets may be better measured, which may further lead to a better estimation of the facet-facet correlations. Second, estimates of the item parameters are available from our analysis,

Table 4
The estimated slope parameters of the NEO dataset.

Latent trait	Loading										Description
N1	1.06	0.91	1.01	1.68	1.15	0.95	0.92	0.73	0.74	0.72	Item 1-10.
N2	1.91	1.70	1.53	0.97	2.10	1.86	1.58	1.48	1.09	0.63	Item 11-20.
N3	1.42	1.76	1.99	1.77	0.85	1.16	0.62	1.11	1.78	1.58	Item 21-30.
N4	0.81	0.65	1.13	0.94	0.74	0.48	0.68	0.56	0.79	0.80	Item 31-40.
N5	0.54	0.42	0.48	0.74	0.46	1.01	0.84	1.04	0.39	0.54	Item 41-50.
N6	1.16	1.02	1.15	0.58	1.00	1.19	0.66	1.21	0.77	1.10	Item 51-60.
A1	1.41	1.67	1.72	0.92	1.13	0.53	2.26	0.99	0.76	0.92	Item 61-70.
A2	0.43	0.62	0.43	1.00	0.70	1.07	0.66	0.61	1.24	0.84	Item 71-80.
A3	0.98	0.88	1.25	1.31	0.68	0.80	1.17	0.67	0.98	1.06	Item 81-90.
A4	0.39	0.30	0.44	0.72	0.73	0.53	0.91	1.14	0.82	0.55	Item 91-100.
A5	0.22	0.33	0.29	0.26	0.58	4.19	4.41	0.37	0.33	0.35	Item 101-110.
A6	1.00	1.21	0.47	0.66	1.00	0.86	0.35	0.70	0.60	0.49	Item 111-120.
E1	1.29	1.07	2.09	1.79	0.94	0.75	1.35	1.62	0.59	1.12	Item 121-130.
E2	1.25	0.99	0.99	0.69	0.64	0.98	1.03	1.39	1.63	1.06	Item 131-140.
E3	1.73	1.41	0.74	0.56	1.57	1.28	1.08	0.82	0.64	0.74	Item 141-150.
E4	1.24	1.33	0.88	0.60	0.35	0.53	0.33	0.40	0.28	0.41	Item 151-160.
E5	1.04	1.37	1.26	0.75	1.08	1.17	0.62	1.02	0.49	0.47	Item 161-170.
E6	1.16	1.60	0.54	0.76	1.26	1.26	0.88	0.73	0.66	0.54	Item 171-180.
O1	1.01	1.06	1.49	1.41	0.76	0.68	1.50	0.96	0.91	0.98	Item 181-190.
O2	1.67	0.49	0.93	0.49	0.81	1.80	0.91	1.54	0.26	0.49	Item 191-200.
O3	1.41	0.67	0.37	0.15	0.37	1.41	1.46	0.72	0.87	0.93	Item 201-210.
O4	0.67	0.55	0.47	0.77	1.02	2.33	2.23	0.74	0.28	0.72	Item 211-220.
O5	0.56	0.89	0.91	0.72	0.77	1.04	1.27	1.39	1.40	1.05	Item 221-230.
O6	0.61	0.33	0.83	0.25	0.70	0.47	0.75	0.94	1.25	0.31	Item 231-240.
C1	0.91	1.04	1.19	0.86	0.94	1.20	0.79	0.85	0.97	0.54	Item 241-250.
C2	1.27	1.35	0.50	1.28	0.87	0.85	1.02	0.84	0.64	0.92	Item 251-260.
C3	0.95	0.68	0.38	0.93	0.67	1.07	0.77	0.85	0.93	0.79	Item 261-270.
C4	0.74	1.30	0.91	0.91	1.23	0.65	0.45	0.60	1.17	1.23	Item 271-280.
C5	0.91	0.81	1.55	1.16	1.08	1.62	1.15	1.32	1.78	0.70	Item 281-290.
C6	0.41	0.43	0.32	1.82	1.47	0.80	1.73	0.70	2.05	0.55	Item 291-300.

providing diagnostic information about the items. For example, the estimated slope of item 204 ("Enjoy examining myself and my life") is relatively small ($\hat{a}_{204} = 0.15$). It suggests that this item contains a relatively small amount of information about the corresponding facet, O3 (Emotionality). Such an item may be removed when developing a short scale. Finally, although not considered in this analysis, it is much easier to compare different hypotheses of the personality structure under the adopted full-information IFA framework. In particular, the full-information IFA framework turns the different hypotheses into different MIRT models and then compares them using standard statistical inference tools.

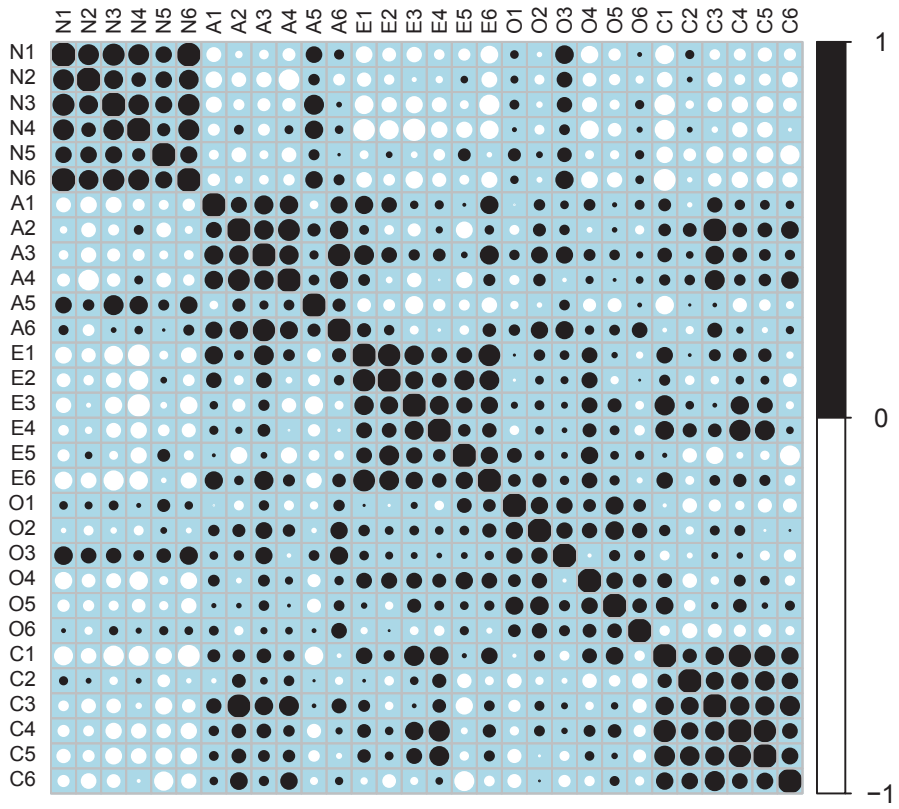

Figure 13. Visualization of the estimated correlation matrix of the 30 latent traits for the IPIP-NEO dataset. Positive and negative correlations are shown in black and white circles, respectively. The size of the circle is proportional to the absolute value of the correlation.

7 Concluding Remarks

In this paper, we propose an improved stochastic EM algorithm for estimating MIRT model parameters. Thanks to the asymptotic properties of the stochastic EM algorithm, as well as advanced sampling and optimization techniques, the developed algorithm not only produces a point estimator that closely resembles the MMLE but is also computationally efficient and virtually tuning-free. As discussed in Section 4, this algorithm can be easily generalized to the estimation of latent variable models with constraints and nonsmooth penalties, including L_1 regularized estimation of slope parameters in exploratory IFA, confirmatory IFA with constrained covariance matrix, and estimation of multilevel latent

variable models. Our simulation studies suggest that the performance of this algorithm is comparable to the popular MH-RM algorithm. Moreover, when the dimensionality of the latent space is high, our algorithm tends to outperform the MH-RM algorithm for greater computational efficiency and less tuning burden. These evidence suggest that the proposed StEM algorithm has the potential to become a popular research and operational tool.

This current study will be extended along the following directions in future research. First, the performance of the StEM algorithm on solving a regularized estimator for exploratory IFA will be investigated and its statistical properties, such as the consistency in parameter estimation and model selection, will be studied. Second, a generic StEM algorithm will be developed for the estimation of general structural equation models, such as the latent regression IRT analysis, which can be very useful in educational and psychological research for analyzing structural equation models with many latent variables. Finally, the potential applications of the StEM algorithm for analyzing latent variable models with more complex structures will be investigated, including the latent Dirichlet allocation (Blei, Ng, & Jordan, 2003) for natural language processing and the mixed membership stochastic blockmodels (Airoldi, Blei, Fienberg, & Xing, 2008) for network data analysis.

Appendix A

Adaptive Rejection Sampling in StE Step

We elaborate on the adaptive rejection sampler (Gilks & Wild, 1992) used in the StE-step, for sampling from $f(\theta_k|\mathbf{y}_i, \Psi^{(t-1)}, \tilde{\boldsymbol{\theta}}^{(t)}_{i,-k})$. The problem becomes to sampling from a probability density function $g(x)/C$, where C is an unknown normalizing constant and $g(x)$ is log-concave, continuous and differentiable everywhere. Consequently, $h(x) = \log g(x)$ is concave, continuous and differentiable everywhere. The adaptive rejection sampler consists of the following three steps.

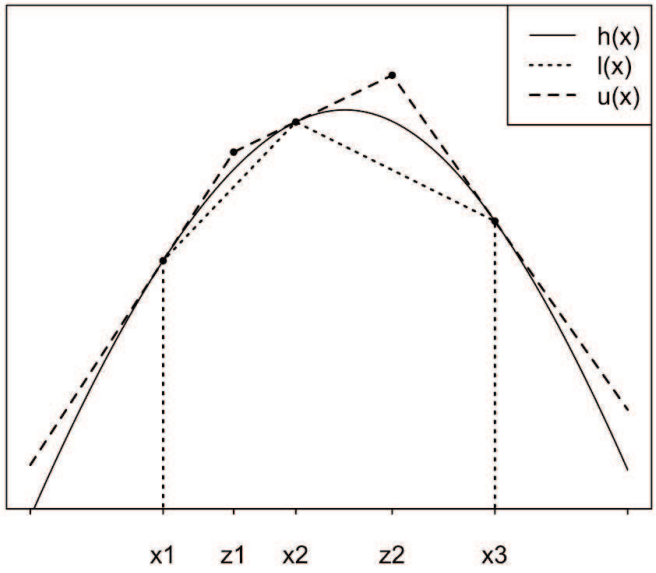

Figure A1. An illustration of the construction of the proposal distribution in the adaptive rejection sampler. $h(x) = \log(g(x))$, where $g(x)/\int g(x')dx'$ is the target distribution to be sampled from. $u(x)$ is a piecewise linear upper bound of $h(x)$ and $l(x)$ is a piecewise linear lower bound of $h(x)$.

1. **Construction of envelope**. Let set $D = \{x_i : i = 1, ..., M\}$, satisfying $x_1 < \cdots < x_M$, $h'(x_1) > 0$ and $h'(x_M) < 0$, where $h'(x)$ is the derivative of $h(x)$. Calculate the following for the starting points in D:

   (a) $u(x)$, the piecewise linear upper bound formed by the tangents to $h(x)$ at each

point in D. More precisely, let

$$z_i = \frac{h(x_{i+1}) - h(x_i) - x_{i+1}h'(x_{i+1}) + x_i h'(x_i)}{h'(x_i) - h'(x_{i+1})},$$

for $i = 1, \ldots, M-1$, and $z_0 = -\infty$ and $z_M = \infty$. Then

$$u(x) = h(x_j) + (x - x_j)h'(x_j), \ x \in [z_{j-1}, z_j], \quad j = 1, \ldots, M.$$

(b) $s(x) = \exp(u(x))/\int \exp(u(x'))dx'$.

(c) $l(x)$, the piecewise linear lower bound formed by the chords between adjacent points in D. More precisely,

$$l(x) = \frac{(x_{i+1} - x_i)h(x_i) + (x - x_i)h(x_{i+1})}{x_{i+1} - x_i},$$

for $x \in [x_i, x_{i+1}]$, $i = 1, \ldots, M-1$, and $l(x) = -\infty$ for $x < x_1$ and $x > x_M$.

Note that $u(x)$, $s(x)$, and $l(x)$ all have analytical forms. Note that $s(x)$ is a cumulative distribution function, which serves as the cumulative distribution function. See Figure A1 for an illustrative example, where $M = 3$.

2. **Sampling**. Sample a value x^* from $s(x)$ and a value u^* independently from uniform distribution over the interval (0, 1). If $u^* \leq \exp\{l(x^*) - u(x^*)\}$ then accept x^*, otherwise evaluate $h(x^*)$ and $h'(x^*)$. If further $u^* \leq \exp\{h(x^*) - u_M(x^*)\}$ then accept x^*, otherwise reject x^*.

3. **Updating**. If $h(x^*)$ and $h'(x^*)$ are evaluated in the previous sampling step, include x^* in D. Relabel the elements of D in ascending order and reconstruct functions $u(x)$, $s(x)$, and $l(x)$.

Our sampler iterates among the three steps, until one sample has been accepted. In our implementation, we use $M = 3$ and $D = \{-5, 0, 5\}$ as the default starting point.

Appendix B

Optimization in the M Step

We then consider the optimization of (8), which is a convex optimization problem with a positive semi-definite constraint. It follows after some simple algebra that(8) is equivalent to

$$\begin{aligned} &\min_{\Sigma} \langle \Sigma^{-1}, \hat{\Sigma} \rangle + \log\det(\Sigma) \\ &s.t.\ \Sigma \succeq 0, \sigma_{kk} = 1, k = 1, 2, \ldots, K, \end{aligned} \tag{12}$$

where $\hat{\Sigma} = (\hat{\sigma}_{k_1k_2})_{K\times K}$, $\hat{\sigma}_{k_1k_2} = \frac{1}{N}\sum_{i=1}^{N} \tilde{\theta}_{ik_1}^{(t)} \tilde{\theta}_{ik_2}^{(t)}$. We solve the estimation problem by developing a proximal gradient descent algorithm (Parikh et al., 2014). Denote $C = \{M_{K\times K} : m_{kk} = 1\}$. The optimization problem becomes:

$$\min_{\Sigma} \langle \Sigma^{-1}, \hat{\Sigma} \rangle + \log\det(\Sigma)\ s.t.\ \Sigma \in C,\ \Sigma \succeq 0. \tag{13}$$

We have the following procedure:

1. **Initialization**. Given $\Theta = (\tilde{\boldsymbol{\theta}}_1^{(t)}, \tilde{\boldsymbol{\theta}}_2^{(t)}, \ldots, \tilde{\boldsymbol{\theta}}_N^{(t)})$ which is sampled from StE step, set initial value $\Sigma^{(0)}$ as the sample correlation of Θ.

2. **Proximal gradient descent**. For $s = 1, 2, \ldots,$ update

$$\Sigma^{(s+1)} = Prox_C\left(\Sigma^{(s)} - \lambda^s \nabla f(\Sigma^{(s)})\right),$$

   where $\nabla f(\Sigma) = -\Sigma^{-1}\hat{\Sigma}\Sigma^{-1} + \Sigma^{-1}$, the gradient of

$$f(\Sigma) = \langle \Sigma^{-1}, \hat{\Sigma} \rangle + \log\det(\Sigma),$$

   $Prox_C(\cdot)$ is a matrix operator that set the diagnal elements of a matrix to be 1.

   $\lambda^s > 0$ is a step size obtained by line search which grantees that $f(\Sigma^{(s+1)}) < f(\Sigma^{(s)})$.

3. **Output**. Iterate step 2 until convergence. Output $\Sigma^{(S)}$, where S is the last iteration number.

Due to the presence of the logarithm of determinant term in the objective function, the positive semi-definite constraint is satisfied automatically and thus $\Sigma^{(S)}$ is strictly positive definite given that the initial value $\Sigma^{(0)}$ is strictly positive definite.

Appendix C

Details on Determining T and m

Determining T. We determine the burn-in size T using a batch procedure based on the Geweke statistic (Gelman & Rubin, 1992). Let batch size be B, where B is chosen as 20 as the default value in our implementation. We also use M batches as a moving window for the Markov chain of $\{\Psi^{(t)} : t = 1, 2, \dots\}$, based on which the Geweke statistics are computed; in our implementation, $M = 10$ is chosen as the default value. We denote the number of parameters in Ψ as p. More precisely, we have the following batch procedure:

1. **Initialization.** Set iteration number $k = 0$. Run MB iterations of the StEM algorithm, and obtain $\Psi^{(1)}, \dots, \Psi^{(MB)}$.

2. **Check stationarity.** For each entry j of Ψ, we compute the Geweke statistic z_j based on the Markov chain $\{\Psi_j^{(kB+1)}, \dots, \Psi_j^{((k+M)B)}\}$, based on the mean difference between first 10% and last 50% part of chain. We regard stationary being reached when all $|z_j|$s are sufficiently small. In the implementation, we terminate the burn-in procedure if
$$\sum_{j=1}^{p} z_j^2 < p\epsilon_1,$$
where ϵ_1 is chosen as 1.5 in the implementation.

3. **Updating.** If burn-in has not been terminated, we increase iteration number k by 1, discard the first batch in the current moving window, and run an additional batch (B iterations) of the StEM algorithm.

We iterate between Steps 2 and 3, until burn-in is terminated according to Step 2. Upon stopping, we set burn-in size $T = kB$.

Determining m. We determine the value of m based on a similar batch procedure, after the determination of burn-in size T.

1. **Initialization.** Once T has been determined, we have a Markov chain of length MB, $\{\Psi^{(T+1)}, \ldots, \Psi^{(T+MB)}\}$. We start with this initial chain and initialize the number of batches for averaging as $n = M$.

2. **Check convergence.** For each parameter Ψ_j, we estimate the variance of

$$\hat{\Psi}_j(n) = \frac{1}{nB} \sum_{t=T+1}^{T+nB} \Psi_j^{(t)}$$

by the batch variance procedure (Roberts, 1996),

$$\hat{\delta}_j(n) = \frac{\sum_{i=1}^{n} (\bar{\Psi}_j(i) - \hat{\Psi}_j(n))^2}{(n-1)n},$$

where $\bar{\Psi}_j(i)$ is the mean of the ith batch. This estimate adjusts for the autocorrelation among the $\Psi_j^{(t)}$s. We declare convergence when $\hat{\delta}_j(n) < \epsilon_2/N$ for all $j = 1, \ldots, p$. In our implementation, $\epsilon_2 = 0.4$ is chosen as the default value.

3. **Updating.** If convergence has not been reached, we increase n by 1 and run an additional batch (B iterations) of the StEM algorithm. We then obtain a Markov chain of length nB, $\Psi^{(T+1)}$, $\ldots$, $\Psi^{(T+nB)}$.

We iterate between Steps 2 and 3, until convergence has been reached according to Step 2. Upon stopping, we set $m = nB$.

References

Agresti, A. (1996). *An introduction to categorical data analysis*. New York, NY: Wiley.

Airoldi, E. M., Blei, D. M., Fienberg, S. E., & Xing, E. P. (2008). Mixed membership stochastic blockmodels. *Journal of Machine Learning Research*, *9*, 1981–2014.

Albert, J. H. (1992). Bayesian estimation of normal ogive item response curves using Gibbs sampling. *Journal of Educational Statistics*, *17*, 251–269. doi: 10.2307/1165149

Anderson, T. W., & Rubin, H. (1956). Statistical inference in factor analysis. In J. Neyman (Ed.), *Proceedings of the third berkeley symposium on mathematical statistics and probability* (pp. 111–150). Los Angeles, CA: University of California Press.

Béguin, A. A., & Glas, C. A. (2001). MCMC estimation and some model-fit analysis of multidimensional IRT models. *Psychometrika*, *66*, 541-561. doi: 10.1007/bf02296195

Birnbaum, A. (1968). Some latent trait models and their use in inferring an examinee's ability. In F. M. Lord & M. R. Novick (Eds.), *Statistical theories of mental test scores* (p. 397 - 472). Reading, MA: Addison-Wesley Publishing.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research*, *3*, 993–1022.

Bock, R. D. (1972). Estimating item parameters and latent ability when responses are scored in two or more nominal categories. *Psychometrika*, *37*, 29–51. doi: 10.1007/bf02291411

Bock, R. D., & Aitkin, M. (1981). Marginal maximum likelihood estimation of item parameters: Application of an EM algorithm. *Psychometrika*, *46*, 443–459. doi: 10.1007/bf02294168

Butcher, J. N., Dahlstrom, W., Graham, J., Tellegen, A., & Kaemmer, B. (1989). *MMPI-2: Manual for administration and scoring*. Minneapolis, MN: University of Minnesota Press.

Cai, L. (2010a). High-dimensional exploratory item factor analysis by a Metropolis–Hastings Robbins–Monro algorithm. *Psychometrika*, *75*, 33–57. doi:

10.1007/s11336-009-9136-x

Cai, L. (2010b). Metropolis-Hastings Robbins-Monro algorithm for confirmatory item factor analysis. *Journal of Educational and Behavioral Statistics*, *35*, 307–335. doi: 10.3102/1076998609353115

Cai, L. (2013). *flexMIRT: Flexible multilevel multidimensional item analysis and test scoring (version 2.0) [computer software]*. Chapel Hill, NC: Vector Psychometric Group.

Camilli, G., & Fox, J.-P. (2015). An aggregate IRT procedure for exploratory factor analysis. *Journal of Educational and Behavioral Statistics*, *40*, 377–401. doi: 10.3102/1076998615589185

Celeux, G., & Diebolt, J. (1985). The SEM algorithm: A probabilistic teacher algorithm derived from the EM algorithm for the mixture problem. *Computational Statistics Quarterly*, *2*, 73–82.

Chalmers, R. P. (2012). mirt: A multidimensional item response theory package for the R environment. *Journal of Statistical Software*, *48*, 1–29. doi: 10.1007/978-0-387-89976-3

Costa, P. T., & McCrae, R. R. (1985). *The NEO personality inventory*. Odessa, FL: Psychological Assessment Resources.

Dagum, L., & Menon, R. (1998). OpenMP: An industry standard API for shared-memory programming. *Computational Science & Engineering, IEEE*, *5*, 46–55. doi: 10.1109/99.660313

Delyon, B., Lavielle, M., & Moulines, E. (1999). Convergence of a stochastic approximation version of the EM algorithm. *Annals of Statistics*, *27*, 94–128. doi: 1018031103

Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, *39*, 1–38. doi: 10.1142/9789812388759_0028

Diebolt, J., & Ip, E. H. (1996). Stochastic EM: Method and application. In W. R. Gilks,

S. Richardson, & D. Spiegelhalter (Eds.), *Markov chain Monte Carlo in practice* (pp. 259–273). New York, NY: CRC Press.

Edwards, M. C. (2010). A Markov chain Monte Carlo approach to confirmatory item factor analysis. *Psychometrika*, *75*, 474–497. doi: 10.1007/s11336-010-9161-9

Fox, J.-P. (2003). Stochastic EM for estimating the parameters of a multilevel IRT model. *British Journal of Mathematical and Statistical Psychology*, *56*, 65–81. doi: 10.1348/000711003321645340

Friedman, J., Hastie, T., & Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, *33*, 1-22. doi: 10.18637/jss.v033.i01

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

Geweke, J. (1992). Evaluating the accuracy of sampling-based approaches to the calculation of posterior moments. In J. Bernardo, J. Berger, A. Dawid, & A. Smith (Eds.), *Bayesian statistics* (pp. 169–193). Oxford, England: Oxford University Press.

Gilks, W. R., & Wild, P. (1992). Adaptive rejection sampling for Gibbs sampling. *Applied Statistics*, *10.2307/2347565*, 337–348. doi: 10.2307/2347565

Gu, M. G., & Kong, F. H. (1998). A stochastic approximation algorithm with Markov chain Monte-Carlo method for incomplete data estimation problems. *Proceedings of the National Academy of Sciences*, *95*, 7270–7274. doi: 10.1073/pnas.95.13.7270

Herlihy, M., & Shavit, N. (2011). *The art of multiprocessor programming.* Burlington, MA: Morgan Kaufmann.

Huber, P., Ronchetti, E., & Victoria-Feser, M.-P. (2004). Estimation of generalized linear latent variable models. *Journal of the Royal Statistical Society: Series B (Methodology)*, *66*, 893–908. doi: 10.1111/j.1467-9868.2004.05627.x

Ip, E. H. (1994). *A stochastic EM estimator in the presence of missing data: Theory and applications* (Unpublished doctoral dissertation). Department of Statistics, Stanford

University.

Ip, E. H. (2002). On single versus multiple imputation for a class of stochastic algorithms estimating maximum likelihood. *Computational Statistics*, *17*, 517–524. doi: 10.1007/s001800200124

Johnson, J. A. (2005). Ascertaining the validity of individual protocols from web-based personality inventories. *Journal of Research in Personality*, *39*, 103–129. doi: 10.1016/j.jrp.2004.09.009

Johnson, J. A. (2014). Measuring thirty facets of the Five Factor Model with a 120-item public domain inventory: Development of the IPIP-NEO-120. *Journal of Research in Personality*, *51*, 78–89. doi: 10.1016/j.jrp.2014.05.003

Jöreskog, K. G., & Moustaki, I. (2001). Factor analysis of ordinal variables: A comparison of three approaches. *Multivariate Behavioral Research,*, *36*, 347–387. doi: 10.1207/s15327906347-387

Kass, R. E., & Steffey, D. (1989). Approximate Bayesian inference in conditionally independent hierarchical models (parametric empirical Bayes models). *Journal of the American Statistical Association*, *84*, 717-726. doi: 10.1080/01621459.1989.10478825

Liu, D. C., & Nocedal, J. (1989). On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, *45*, 503–528. doi: 10.1007/BF01589116

Liu, Y., Magnus, B., Quinn, H., & Thissen, D. (in press). Multidimensional item response theory. In D. Hughes, P. Irwing, & T. Booth (Eds.), *Handbook of psychometric testing.* Wiley-Blackwell.

Louis, T. A. (1982). Finding the observed information matrix when using the EM algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, *44*, 226–233.

Masters, G. N. (1982). A Rasch model for partial credit scoring. *Psychometrika*, *47*, 149–174. doi: 10.1007/BF02296272

Meng, X.-L., & Schilling, S. (1996). Fitting full-information item factor models and an

empirical investigation of bridge sampling. *Journal of the American Statistical Association*, *91*, 1254–1267. doi: 10.1080/01621459.1996.10476995

Monroe, S. L. (2014). *Multidimensional item factor analysis with semi-nonparametric latent densities* (Unpublished doctoral dissertation). UCLA.

Muraki, E., & Carlson, J. E. (1995). Full-information factor analysis for polytomous item responses. *Applied Psychological Measurement*, *19*, 73–90. doi: 10.1177/014662169501900109

Muthén, B. (1978). Contributions to factor analysis of dichotomous variables. *Psychometrika*, *43*, 551–560. doi: 10.1007/bf02293813

Muthén, B. (1984). A general structural equation model with dichotomous, ordered categorical, and continuous latent variable indicators. *Psychometrika*, *49*, 115–132. doi: 10.1007/bf02294210

Muthén, B. (1993). Goodness of fit with categorical and other nonnormal variables. In K. A. Bollen & J. S. Long (Eds.), *Testing structural equation models* (pp. 205–234). Newbury Park, CA: Sage.

Neal, R. M. (2003). Slice sampling. *Annals of Statistics*, *31*, 705–741. doi: 1056562461

Nemirovski, A., Juditsky, A., Lan, G., & Shapiro, A. (2009). Robust stochastic approximation approach to stochastic programming. *SIAM Journal on Optimization*, *19*, 1574–1609. doi: 10.1137/070704277

Nielsen, S. F. (2000). The stochastic EM algorithm: Estimation and asymptotic results. *Bernoulli*, *6*, 457–489. doi: 10.2307/3318671

Parikh, N., Boyd, S., et al. (2014). Proximal algorithms. *Foundations and Trends® in Optimization*, *1*, 127–239. doi: 10.1561/2400000003

Patz, R. J., & Junker, B. W. (1999a). Applications and extensions of MCMC in IRT: Multiple item types, missing data, and rated responses. *Journal of Educational and Behavioral Statistics*, *24*, 342–366. doi: 10.3102/10769986024004342

Patz, R. J., & Junker, B. W. (1999b). A straightforward approach to Markov chain Monte

Carlo methods for item response models. *Journal of Educational and Behavioral Statistics*, *24*, 146–178. doi: 10.3102/10769986024002146

Rabe-Hesketh, S., Skrondal, A., & Pickles, A. (2005). Maximum likelihood estimation of limited and discrete dependent variable models with nested random effects. *Journal of Econometrics*, *128*, 301–323. doi: 10.1016/j.jeconom.2004.08.017

Rasch, G. (1960). *Probabilistic models for some intelligence and achievement tests.* Copenhagen, Denmark: Nielsen and Lydiche.

Reckase, M. (2009). *Multidimensional item response theory.* New York, NY: Springer. doi: 10.1007/978-0-387-89976-3_4

Revuelta, J. (2014). Multidimensional item response model for nominal variables. *Applied Psychological Measurement*, *38*, 549–562. doi: 10.1177/0146621614536272

Robbins, H., & Monro, S. (1951). A stochastic approximation method. *The Annals of Mathematical Statistics*, *22*, 400–407. doi: 10.1007/978-1-4612-5110-1_9

Roberts, G. O. (1996). Markov chain concepts related to sampling algorithms. In W. R. Gilks, S. Richardson, & D. Spiegelhalter (Eds.), *Markov chain Monte Carlo in practice.* New York, NY: CRC Press.

Schilling, S., & Bock, R. D. (2005). High-dimensional maximum marginal likelihood item factor analysis by adaptive quadrature. *Psychometrika*, *70*, 533–555. doi: 10.1007/s11336-003-1141-x

Shi, J.-Q., & Lee, S.-Y. (1998). Bayesian sampling-based approach for factor analysis models with continuous and polytomous data. *British Journal of Mathematical and Statistical Psychology*, *51*, 233–252. doi: 10.1111/j.2044-8317.1998.tb00679.x

Song, X.-Y., & Lee, S.-Y. (2005). A multivariate probit latent variable model for analyzing dichotomous responses. *Statistica Sinica*, *15*, 645–664.

Spall, J. C. (2005). *Introduction to stochastic search and optimization: Estimation, simulation, and control.* New York, NY: John Wiley & Sons.

Steel, P., Schmidt, J., & Shultz, J. (2008). Refining the relationship between personality

and subjective well-being. *Psychological Bulletin*, *134*, 138-161. doi: 10.1037/0033-2909.134.1.138

Sun, J., Chen, Y., Liu, J., Ying, Z., & Xin, T. (2016). Latent variable selection for multidimensional item response theory models via L_1 regularization. *Psychometrika*, *81*, 921–939. doi: 10.1007/s11336-016-9529-6

Sympson, J. B. (1978). A model for testing with multidimensional items. In D. Weiss (Ed.), *Proceedings of the 1977 computerized adaptive testing conference* (p. 82-98). Minneapolis, MN.

Thomas, N. (1993). Asymptotic corrections for multivariate posterior moments with factored likelihood functions. *Journal of Computational and Graphical Statistics*, *2*, 309–322. doi: 10.2307/1390648

von Davier, M. (2016). New results on an improved parallel EM algorithm for estimating generalized latent variable models. In L. van der Ark, M. Wiberg, S. Culpepper, J. Douglas, & W. Wang (Eds.), *The annual meeting of the psychometric society* (pp. 1–8). New York, NY: Springer. doi: 10.1007/978-3-319-56294-0_1

von Davier, M., & Sinharay, S. (2010). Stochastic approximation methods for latent regression item response models. *Journal of Educational and Behavioral Statistics*, *35*, 174–193. doi: 10.3102/1076998609346970

Wada, T., & Fujisaki, Y. (2015). A stopping rule for stochastic approximation. *Automatica*, *60*, 1–6. doi: 10.1016/j.automatica.2015.06.029

Wirth, R., & Edwards, M. C. (2007). Item factor analysis: Current approaches and future directions. *Psychological Methods*, *12*, 58 - 79. doi: 10.1037/1082-989X.12.1.58

Yao, L., & Schwarz, R. D. (2006). A multidimensional partial credit model with associated item and test statistics: An application to mixed-format tests. *Applied Psychological Measurement*, *30*, 469–492. doi: 10.1177/0146621605284537

Zhao, Y., & Joe, H. (2005). Composite likelihood estimation in multivariate data analysis. *Canadian Journal of Statistics*, *33*, 335–356. doi: 10.1002/cjs.5540330303