Supplementary material for Gaussian Variational Estimation for Multidimensional Item Response Theory

In this supplementary document, we present the derivations and proofs omitted from the paper *Gaussian Variational Estimation for Multidimensional Item Response Theory.* Section A and C illustrate the derivation of the GVEM algorithm for M2PL and M3PL models, respectively. In Section B, we present the proof for the Proposition 1. Finally in Section D, we prove the consistency result in Theorem 1.

A Derivation of GVEM in the 2PL model

In this section, we provide a step by step derivation of the GVEM algorithm in 2PL model. Especially, EM steps are described in detail.

E-Step In E step, we evaluate the lower bound of the expected log-likelihood with respect to the variational distributions $q_i(\boldsymbol{\theta}_i)$'s. Recall that we can mathematically write the lower bound as

$$E(\boldsymbol{A},\boldsymbol{B},\boldsymbol{\xi}) := \sum_{i=1}^{N}\int_{\boldsymbol{\theta}_i} l(Y_i,\boldsymbol{\theta}_i,\boldsymbol{\xi}_i \mid \boldsymbol{A},\boldsymbol{B}) \times q_i(\boldsymbol{\theta}_i)d\boldsymbol{\theta}_i. \tag{1}$$

Our main interest is to evaluate the integral in (1) to derive a closed-form expression of the variational lower bound, $E(\boldsymbol{A},\boldsymbol{B},\boldsymbol{\xi})$. In each E step, we iteratively update the lower bound until convergence.

Section 3.1 in the paper discusses the choice of the optimal variational density $q_i(\boldsymbol{\theta}_i)$ as a Gaussian distribution with mean and covariance determined by

$$\begin{aligned}\mu_i &= \Sigma_i \times \sum_{j=1}^{J}\Big\{2\eta(\xi_{i,j})b_j + Y_{ij} - \frac{1}{2}\Big\}(\boldsymbol{\alpha}_j)^\top, &(2)\\ \Sigma_i^{-1} &= (\Sigma_{\boldsymbol{\theta}})^{-1} + 2\sum_{j=1}^{J}\eta(\xi_{i,j})\boldsymbol{\alpha}_j\boldsymbol{\alpha}_j^\top. &(3)\end{aligned}$$

Let $q_i^{(t)}(\boldsymbol{\theta}_i) = q_i(\boldsymbol{\theta}_i \mid \boldsymbol{A}^{(t)},\boldsymbol{B}^{(t)},\Sigma_{\boldsymbol{\theta}}^{(t)},\boldsymbol{\xi}_i^{(t)})$ denote the tth iteration's variational density $q_i(\boldsymbol{\theta}_i)$ with all recent updates of the model parameters $(\boldsymbol{A}^{(t)},\boldsymbol{B}^{(t)},\Sigma_{\boldsymbol{\theta}}^{(t)},\boldsymbol{\xi}_i^{(t)})$. Also, let $E_{q_i}^{(t)}$ denote the expectation with respect to the distribution $q_i^{(t)}(\boldsymbol{\theta}_i)$. Then, we can write the tth iteration's variational lower bound as

$$\begin{aligned}E^{(t)}(\boldsymbol{A},\boldsymbol{B},\boldsymbol{\xi}) &= \sum_{i=1}^{N}\int_{\boldsymbol{\theta}_i} l(Y_i,\boldsymbol{\theta}_i,\boldsymbol{\xi}_i \mid \boldsymbol{A},\boldsymbol{B}) \times q_i^{(t)}(\boldsymbol{\theta}_i)d\boldsymbol{\theta}_i\\ &= \sum_{i=1}^{N}\sum_{j=1}^{J}\Bigg(\log\frac{e^{\xi_{i,j}}}{(1+e^{\xi_{i,j}})} + Y_{ij}(\boldsymbol{\alpha}_j^\top E_{q_i}^{(t)}[\boldsymbol{\theta}_i] - b_j) + \frac{1}{2}(b_j - \boldsymbol{\alpha}_j^\top E_{q_i}^{(t)}[\boldsymbol{\theta}_i] - \xi_{i,j})\\ &\quad -\eta(\xi_{i,j})\{E_{q_i}^{(t)}[(b_j - \boldsymbol{\alpha}_j^\top\boldsymbol{\theta}_i)^2] - \xi_{i,j}^2\}\Bigg) + \frac{N}{2}\log|\Sigma_{\boldsymbol{\theta}}^{-1}| - \sum_{i=1}^{N}\frac{1}{2}E_{q_i}^{(t)}[\boldsymbol{\theta}_i^\top\Sigma_{\boldsymbol{\theta}}^{-1}\boldsymbol{\theta}_i]\end{aligned}$$

Note that the expectation $E_{q_i}^{(t)}$ can be expressed with $\mu_i^{(t)}$ and $\Sigma_i^{(t)}$ since

$$E_{q_i}^{(t)}[\boldsymbol{\theta}_i] = \mu_i^{(t)}, \quad E_{q_i}^{(t)}[(b_j - \boldsymbol{\alpha}_j^\top \boldsymbol{\theta}_i)^2] = b_j^2 - 2b_j \boldsymbol{\alpha}_j^\top \mu_i^{(t)} + (\boldsymbol{\alpha}_j^{(t)})^\top [\Sigma_i^{(t)} + (\mu_i^{(t)})(\mu_i^{(t)})^\top] \boldsymbol{\alpha}_j^{(t)},$$

and

$$E_{q_i}^{(t)}[\boldsymbol{\theta}_i^\top \Sigma_{\boldsymbol{\theta}}^{-1} \boldsymbol{\theta}_i] = E_{q_i}^{(t)}[Tr(\Sigma_{\boldsymbol{\theta}}^{-1} \boldsymbol{\theta}_i \boldsymbol{\theta}_i^\top)] = Tr(\Sigma_{\boldsymbol{\theta}}^{-1} E_{q_i}^{(t)}[\boldsymbol{\theta}_i \boldsymbol{\theta}_i^\top]) = Tr(\Sigma_{\boldsymbol{\theta}}^{-1}[\Sigma_i^{(t)} + (\mu_i^{(t)})(\mu_i^{(t)})^\top]).$$

Therefore, by plugging in we have the following equivalent form.

$$\begin{aligned}
E^{(t)}(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{\xi}) &= \sum_{i=1}^{N} \sum_{j=1}^{J} \Bigg(\log \frac{e^{\xi_{i,j}}}{(1 + e^{\xi_{i,j}})} + Y_{ij}(\boldsymbol{\alpha}_j^\top \mu_i^{(t)} - b_j) + \frac{1}{2}(b_j - \boldsymbol{\alpha}_j^\top \mu_i^{(t)} - \xi_{i,j}) \\
&\quad - \eta(\xi_{i,j}) \{ b_j^2 - 2b_j \boldsymbol{\alpha}_j^\top \mu_i^{(t)} + (\boldsymbol{\alpha}_j^{(t)})^\top [\Sigma_i^{(t)} + (\mu_i^{(t)})(\mu_i^{(t)})^\top] \boldsymbol{\alpha}_j^{(t)} - \xi_{i,j}^2 \} \Bigg) \\
&\quad + \frac{N}{2} \log |\Sigma_{\boldsymbol{\theta}}^{-1}| - \sum_{i=1}^{N} \frac{1}{2} Tr(\Sigma_{\boldsymbol{\theta}}^{-1}[\Sigma_i^{(t)} + (\mu_i^{(t)})(\mu_i^{(t)})^\top]) \\
&= \sum_{i=1}^{N} \sum_{j=1}^{J} \Bigg(\log \frac{e^{\xi_{i,j}}}{(1 + e^{\xi_{i,j}})} + (\frac{1}{2} - Y_{ij}) b_j + (Y_{ij} - \frac{1}{2}) \boldsymbol{\alpha}_j^\top \mu_i^{(t)} - \frac{1}{2} \xi_{i,j} \\
&\quad - \eta(\xi_{i,j}) \{ b_j^2 - 2b_j \boldsymbol{\alpha}_j^\top \mu_i^{(t)} + \boldsymbol{\alpha}_j^\top [\Sigma_i^{(t)} + (\mu_i^{(t)})(\mu_i^{(t)})^\top] \boldsymbol{\alpha}_j - \xi_{i,j}^2 \} \Bigg) \\
&\quad + \frac{N}{2} \log |\Sigma_{\boldsymbol{\theta}}^{-1}| - \sum_{i=1}^{N} \frac{1}{2} Tr(\Sigma_{\boldsymbol{\theta}}^{-1}[\Sigma_i^{(t)} + (\mu_i^{(t)})(\mu_i^{(t)})^\top]). \qquad (4)
\end{aligned}$$

This gives a closed form expression of the expectation function $E^{(t)}(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{\xi})$, i.e the tth iteration's variational lower bound of the marginal likelihood. In every E step, we iteratively update $E^{(t)}(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{\xi})$ (i.e. (4)) with all recently updated model parameters for $t \geq 1$ steps.

M-Step In tth iteration's M step, we maximize the $E^{(t)}(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{\xi})$ to estimate the parameters $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{\xi})$. This is achieved by setting the derivative of $E^{(t)}(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{\xi})$ with respect to $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{\xi})$ to be zero.

First, consider the $\boldsymbol{\alpha}_j$. Setting the derivative with respect to $\boldsymbol{\alpha}_j$ equal to zero, we have

$$\frac{\partial E^{(t)}(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{\xi})}{\partial \boldsymbol{\alpha}_j} = \sum_{i=1}^{N} (Y_{ij} - \frac{1}{2})(\mu_i^{(t)})^\top + 2b_j \eta(\xi_{i,j})(\mu_i^{(t)})^\top - 2\eta(\xi_{i,j})[\Sigma_i^{(t)} + (\mu_i^{(t)})(\mu_i^{(t)})^\top]\boldsymbol{\alpha}_j = 0$$

which implies that $\boldsymbol{\alpha}_j^{(t+1)}$ is updated according to

$$\boldsymbol{\alpha}_j = \frac{1}{2}\Big[\sum_{i=1}^{N} \eta(\xi_{i,j})\Sigma_i^{(t)} + \eta(\xi_{i,j})(\mu_i^{(t)})(\mu_i^{(t)})^\top\Big]^{-1} \sum_{i=1}^{N} \Big[\Big(Y_{ij} - \frac{1}{2} + 2b_j\eta(\xi_{i,j})\Big)(\mu_i^{(t)})^\top\Big]. \tag{5}$$

Similarly, set the derivative with respect to b_j equal to zero and we have

$$\frac{\partial E^{(t)}(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{\xi})}{\partial b_j} = \sum_{i=1}^{N} (\frac{1}{2} - Y_{ij}) - 2\eta(\xi_{i,j})b_j + 2\eta(\xi_{i,j})\boldsymbol{\alpha}_j^\top \mu_i^{(t)} = 0$$

which implies that $b_j^{(t+1)}$ is updated according to

$$b_j = \frac{\sum_{i=1}^{N} \Big[(\frac{1}{2} - Y_{ij}) + 2\eta(\xi_{i,j})\boldsymbol{\alpha}_j^\top \mu_i^{(t)}\Big]}{\sum_{i=1}^{N} 2\eta(\xi_{i,j})}. \tag{6}$$

Setting the derivative with respect to $\xi_{i,j}$ equal to zero, we have

$$\begin{aligned} 0 = \frac{\partial E^{(t)}(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{\xi})}{\partial \xi_{i,j}} &= \frac{1}{(1 + e^{\xi_{i,j}})} - \frac{1}{2} - \eta'(\xi_{i,j})\{E_{q_i}^{(t)}[(b_j - \boldsymbol{\alpha}_j^\top \boldsymbol{\theta}_i)^2] - \xi_{i,j}^2\} + 2\eta(\xi_{i,j})\xi_{i,j} \\ &= -\eta'(\xi_{i,j})\{E_{q_i}^{(t)}[(b_j - \boldsymbol{\alpha}_j^\top \boldsymbol{\theta}_i)^2] - \xi_{i,j}^2\}. \end{aligned}$$

This implies that $\xi_{i,j}^{(t+1)}$ is updated according to the equation

$$\xi_{i,j}^2 = E_{q_i}^{(t)}[(b_j - \boldsymbol{\alpha}_j^\top \boldsymbol{\theta}_i)^2] = b_j^2 - 2b_j \boldsymbol{\alpha}_j^\top \mu_i^{(t)} + \boldsymbol{\alpha}_j^\top [\Sigma_i^{(t)} + (\mu_i^{(t)})(\mu_i^{(t)})^\top]\boldsymbol{\alpha}_j. \tag{7}$$

When there is no constraint for Σ, i.e., all parameters of Σ are free, we set the derivative with respect to $\Sigma_{\boldsymbol{\theta}}^{-1}$ to be 0 and we obtain

$$0 = \frac{N}{2}\frac{\partial \log|\Sigma_{\boldsymbol{\theta}}^{-1}|}{\partial \Sigma_{\boldsymbol{\theta}}^{-1}} - \frac{1}{2}\sum_{i=1}^{N} \frac{\partial Tr(\Sigma_{\boldsymbol{\theta}}^{-1}[\Sigma_i^{(t)} + (\mu_i^{(t)})(\mu_i^{(t)})^\top])}{\partial \Sigma_{\boldsymbol{\theta}}^{-1}} = \frac{N}{2}\Sigma_{\boldsymbol{\theta}} - \frac{1}{2}\sum_{i=1}^{N}[\Sigma_i^{(t)} + (\mu_i^{(t)})(\mu_i^{(t)})^\top],$$

which gives the update of $\Sigma_{\boldsymbol{\theta}}^{(t+1)}$ as

$$\Sigma_{\boldsymbol{\theta}} = \frac{1}{N}\sum_{i=1}^{N}[\Sigma_i^{(t)} + (\mu_i^{(t)})(\mu_i^{(t)})^\top]. \tag{8}$$

Hence, we update $\Sigma_{\boldsymbol{\theta}}$ by (8) in confirmatory factor analysis. However, in exploratory factor analysis we keep $\Sigma_{\boldsymbol{\theta}} = I_K$ and ignore the step (8). Note that if the $\Sigma_{\boldsymbol{\theta}}$ is assumed to be the correlation matrix with diagonals being 1, then we can standardize the estimated $\Sigma_{\boldsymbol{\theta}}$ to get correlation matrix.

B Proof of Proposition 1

Proposition 1 states that the hierarchical formulation of the 3PL model with new latent variable Z_{ij} is equivalent to the general IRT formulation of the 3PL model. This can be proved by showing that the two approaches yield the same distribution, i.e. $P(Y_i \mid \boldsymbol{\theta}_i, M_p)$.

We first start from the hierarchical formulation of the 3PL model. The conditional distribution of Y_{ij} given Z_{ij} and $\boldsymbol{\theta}_i$ can be equivalently written as

$$P(Y_{ij} \mid Z_{ij}, \boldsymbol{\theta}_i, \boldsymbol{\alpha}_j, b_j) = \Big[(\frac{\exp(\boldsymbol{\alpha}_j^\top \boldsymbol{\theta}_i - b_j)}{1+\exp(\boldsymbol{\alpha}_j^\top \boldsymbol{\theta}_i - b_j)})^{Y_{ij}}(\frac{1}{1+\exp(\boldsymbol{\alpha}_j^\top \boldsymbol{\theta}_i - b_j)})^{1-Y_{ij}}\Big]^{Z_{ij}} I(Y_{ij}=1)^{1-Z_{ij}}.$$

Then, the joint distribution of a response Y_{ij} and a latent variable Z_{ij} is

$$\begin{aligned}
& P(Y_{ij}, Z_{ij} \mid \boldsymbol{\theta}_i, \boldsymbol{\alpha}_j, b_j, c_j) \\
= \; & P(Y_{ij} \mid Z_{ij}, \boldsymbol{\theta}_i, \boldsymbol{\alpha}_j, b_j) P(Z_{ij} \mid c_j) \\
= \; & \Big[(\frac{\exp(\boldsymbol{\alpha}_j^\top \boldsymbol{\theta}_i - b_j)}{1+\exp(\boldsymbol{\alpha}_j^\top \boldsymbol{\theta}_i - b_j)})^{Y_{ij}}(\frac{1}{1+\exp(\boldsymbol{\alpha}_j^\top \boldsymbol{\theta}_i - b_j)})^{1-Y_{ij}}\Big]^{Z_{ij}} \times I(Y_{ij}=1)^{1-Z_{ij}}(1-c_j)^{Z_{ij}} c_j^{1-Z_{ij}} \\
= \; & \Big[(\frac{\exp(\boldsymbol{\alpha}_j^\top \boldsymbol{\theta}_i - b_j)}{1+\exp(\boldsymbol{\alpha}_j^\top \boldsymbol{\theta}_i - b_j)})^{Y_{ij}}(\frac{1}{1+\exp(\boldsymbol{\alpha}_j^\top \boldsymbol{\theta}_i - b_j)})^{1-Y_{ij}}(1-c_j)\Big]^{Z_{ij}} \times (I(Y_{ij}=1)c_j)^{1-Z_{ij}}.
\end{aligned}$$

By summing the joint distribution over the domain of Z_{ij}, we recover the general IRT formulation of the 3PL model.

$$\begin{aligned}
& P(Y_{ij} \mid \boldsymbol{\theta}_i, \boldsymbol{\alpha}_j, b_j, c_j) \\
= \; & \sum_{Z_{ij}=0,1} P(Y_{ij}, Z_{ij} \mid \boldsymbol{\theta}_i, \boldsymbol{\alpha}_j, b_j, c_j) \\
= \; & I(Y_{ij}=1)c_j + (1-c_j)\Big[(\frac{\exp(\boldsymbol{\alpha}_j^\top \boldsymbol{\theta}_i - b_j)}{1+\exp(\boldsymbol{\alpha}_j^\top \boldsymbol{\theta}_i - b_j)})^{Y_{ij}}(\frac{1}{1+\exp(\boldsymbol{\alpha}_j^\top \boldsymbol{\theta}_i - b_j)})^{1-Y_{ij}}\Big].
\end{aligned}$$

Hence, the result of Proposition 1 allows us to use the hierarchical formulation of the 3PL model instead of the general IRT formulation for the derivation of GVEM algorithm in the case of M3PL.

C Derivation of GVEM in the 3PL model

In 3PL model, the EM steps are derived in the similar fashion as in 2PL model. We again start with the derivation of the E step.

E-Step As in the M2PL model, we estimate the variational parameters first and then compute the variational lower bound on the expected log-likelihood. As previously discussed in Section 4.1 in the paper, the choice of the optimal variational density for the first latent variable $\boldsymbol{\theta}_i$ is a Gaussian distribution $q_i(\boldsymbol{\theta}_i)$ with mean and covariance determined by

$$\begin{aligned} \mu_i &= \Sigma_i \times \sum_{j=1}^{J} \Big\{ 2\eta(\xi_{i,j})b_j + Y_{ij} - \frac{1}{2} \Big\} (1 - Y_{ij} + s_{ij}Y_{ij})(\boldsymbol{\alpha}_j)^\top, \quad (9) \\ \Sigma_i^{-1} &= (\Sigma_{\boldsymbol{\theta}})^{-1} + 2\sum_{j=1}^{J} \eta(\xi_{i,j})(1 - Y_{ij} + s_{ij}Y_{ij})\boldsymbol{\alpha}_j\boldsymbol{\alpha}_j^\top. \quad (10) \end{aligned}$$

The optimal variational density of the second latent variable Z_{ij} is a Bernoulli distribution $r_{ij}(Z_{ij})$ with the success probability s_{ij} determined by

$$\begin{aligned} s_{ij}^{-1} &= 1 + \frac{c_j}{1-c_j}\frac{1+e^{\xi_{i,j}}}{e^{\xi_{i,j}}} \exp\Big\{ - Y_{ij}(\boldsymbol{\alpha}_j^\top E_{q_i}[\boldsymbol{\theta}_i] - b_j) + \\ &\quad \frac{1}{2}(b_j - \boldsymbol{\alpha}_j^\top E_{q_i}[\boldsymbol{\theta}_i] - \xi_{i,j}) - \eta(\xi_{i,j})\{E_{q_i}[(b_j - \boldsymbol{\alpha}_j^\top \boldsymbol{\theta}_i)^2] - \xi_{i,j}^2\}\Big\}. \quad (11) \end{aligned}$$

Let the tth iteration's variational densities for the latent variables $\boldsymbol{\theta}_i$ and Z_{ij} be denoted as $q_i^{(t)}(\boldsymbol{\theta}_i) = q_i(\boldsymbol{\theta}_i \mid \boldsymbol{A}^{(t)}, \boldsymbol{B}^{(t)}, \boldsymbol{C}^{(t)}, \Sigma_{\boldsymbol{\theta}}^{(t)}, \boldsymbol{\xi}_i^{(t)})$ and $r_{ij}^{(t)}(Z_{ij}) = r_{ij}(Z_{ij} \mid \boldsymbol{A}^{(t)}, \boldsymbol{B}^{(t)}, \boldsymbol{C}^{(t)}, \Sigma_{\boldsymbol{\theta}}^{(t)}, \boldsymbol{\xi}_i^{(t)})$ with all recent updates of the model parameters, respectively. Then, the tth iteration's variational lower bound of the expected log-likelihood is

$$E^{(t)}(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{\xi}) := \sum_{i=1}^{N} \int_{\boldsymbol{\theta}_i} \left[\sum_{\boldsymbol{Z}_i} l(Y_i, \boldsymbol{\theta}_i, \boldsymbol{Z}_i, \boldsymbol{\xi}_i \mid \boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}) \times r_i^{(t)}(\boldsymbol{Z}_i) \right] \times q_i^{(t)}(\boldsymbol{\theta}_i) d\boldsymbol{\theta}_i. \quad (12)$$

where $r_i^{(t)}(\boldsymbol{Z}_i) = \prod_{j=1}^{J} r_{ij}^{(t)}(Z_{ij})$.

With the variational parameters discussed above (i.e. (9), (10), (11)), we can derive a closed form expression of the variational lower bound. Consistent with the notations from the paper, $E_r^{(t)}$ denotes the expectation with respect to the variational distribution $r_{ij}^{(t)}$'s. Now we evaluate the integrals in expectation function $E^{(t)}(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{\xi})$ with respect to $q_i(\boldsymbol{\theta}_i)$'s and $r_{ij}(Z_{ij})$'s.

$$\begin{aligned}
& E^{(t)}(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{\xi}) \\
= \; & \sum_{i=1}^{N} E_{q_i}^{(t)} \Bigg[\sum_{j=1}^{J} (1 - Y_{ij} + E_r^{(t)}[Z_{ij}] Y_{ij}) \Bigg(\log \frac{e^{\xi_{i,j}}}{(1 + e^{\xi_{i,j}})} + Y_{ij}(\boldsymbol{\alpha}_j^\top \boldsymbol{\theta}_i - b_j) \\
& + \frac{1}{2}(b_j - \boldsymbol{\alpha}_j^\top \boldsymbol{\theta}_i - \xi_{i,j}) - \eta(\xi_{i,j})\{(b_j - \boldsymbol{\alpha}_j^\top \boldsymbol{\theta}_i)^2 - \xi_{i,j}^2\} \Bigg) + \\
& \sum_{j=1}^{J} Y_{ij}(1 - E_z[Z_{ij}]) \log I(Y_{ij} = 1) + \log \phi(\boldsymbol{\theta}_i) + \sum_{j=1}^{J} E_r^{(t)}[\log p(Z_{ij}^{'})] \Bigg] \\
= \; & \sum_{i=1}^{N} \sum_{j=1}^{J} (1 - Y_{ij} + s_{ij} Y_{ij}) \Bigg(\log \frac{e^{\xi_{i,j}}}{(1 + e^{\xi_{i,j}})} + (\frac{1}{2} - Y_{ij})(b_j - \boldsymbol{\alpha}_j^\top \mu_i^{(t)}) - \frac{1}{2} \xi_{i,j} \\
& - \eta(\xi_{i,j})\{b_j^2 - 2 b_j \boldsymbol{\alpha}_j^\top \mu_i^{(t)} + (\boldsymbol{\alpha}_j)^\top [\Sigma_i^{(t)} + (\mu_i^{(t)})(\mu_i^{(t)})^\top] \boldsymbol{\alpha}_j - \xi_{i,j}^2\} \Bigg) \\
& + \sum_{i=1}^{N} \sum_{j=1}^{J} Y_{ij}(1 - s_{ij}) \log I(Y_{ij} = 1) + \frac{N}{2} \log |\Sigma_{\boldsymbol{\theta}}^{-1}| - \sum_{i=1}^{N} \frac{1}{2} Tr(\Sigma_{\boldsymbol{\theta}}^{-1} [\Sigma_i^{(t)} + (\mu_i^{(t)})(\mu_i^{(t)})^\top]) \\
& + \sum_{i=1}^{N} \sum_{j=1}^{J} \{(1 - Y_{ij} + s_{ij} Y_{ij}) log(1 - c_j) + Y_{ij}(1 - s_{ij}) log(c_j)\}, \qquad (13)
\end{aligned}$$

since $E_r[Z_{ij}] = s_{ij}$ for the Bernoulli distribution r_{ij}. The equation (13) is the closed form expression of the tth iteration's variational lower bound $E^{(t)}(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{\xi})$.

M-Step As in 2PL case, in tth iteration's M step we maximize the $E^{(t)}(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{\xi})$ to update the model parameters $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{\xi})$. This is again achieved by setting the derivative of $E^{(t)}(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{\xi})$ with respect to $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{\xi})$ to be zero.

For $\boldsymbol{\alpha}_j$, setting the derivative with respect to $\boldsymbol{\alpha}_j$ equal to zero, we have

$$\begin{aligned}\frac{\partial E^{(t)}(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{\xi})}{\partial \boldsymbol{\alpha}_j} &= \sum_{i=1}^{N}(1 - Y_{ij} + s_{ij}Y_{ij})\Big(Y_{ij} - \frac{1}{2} + 2b_j\eta(\xi_{i,j})\Big)(\mu_i^{(t)})^\top \\ &\qquad - 2\eta(\xi_{i,j})(1 - Y_{ij} + s_{ij}Y_{ij})[\Sigma_i^{(t)} + (\mu_i^{(t)})(\mu_i^{(t)})^\top]\boldsymbol{\alpha}_j = 0.\end{aligned}$$

This implies that $\boldsymbol{\alpha}_j^{(t+1)}$ is updated by

$$\begin{aligned}\boldsymbol{\alpha}_j = \frac{1}{2}\Big[\sum_{i=1}^{N}(1 - Y_{ij} + s_{ij}Y_{ij})\eta(\xi_{i,j})[\Sigma_i^{(t)} + (\mu_i^{(t)})(\mu_i^{(t)})^\top]\Big]^{-1} \times \\ \sum_{i=1}^{N}\Big[(1 - Y_{ij} + s_{ij}Y_{ij})\Big(Y_{ij} - \frac{1}{2} + 2b_j\eta(\xi_{i,j})\Big)(\mu_i^{(t)})^\top\Big].\end{aligned} \tag{14}$$

Similarly for b_j, the derivative of the variational lower bound with respect to b_j is

$$\frac{\partial E^{(t)}(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{\xi})}{\partial b_j} = \sum_{i=1}^{N}(1 - Y_{ij} + s_{ij}Y_{ij})\big[(\frac{1}{2} - Y_{ij}) - 2\eta(\xi_{i,j})b_j + 2\eta(\xi_{i,j})\boldsymbol{\alpha}_j^\top\mu_i^{(t)}\big] = 0$$

Setting it equal to 0, we obtain the updating equation for $b_j^{(t+1)}$ as

$$b_j = \frac{\sum_{i=1}^{N}(1 - Y_{ij} + s_{ij}Y_{ij})\Big[(\frac{1}{2} - Y_{ij}) + 2\eta(\xi_{i,j})\boldsymbol{\alpha}_j^\top\mu_i^{(t)}\Big]}{\sum_{i=1}^{N}2(1 - Y_{ij} + s_{ij}Y_{ij})\eta(\xi_{i,j})}. \tag{15}$$

Finally for a guessing parameter c_j, we again take derivate of $E^{(t)}(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{\xi})$ with respect to c_j and set it equal to zero.

$$\frac{\partial E^{(t)}(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{\xi})}{\partial c_j} = \sum_{i=1}^{N} [(1 - Y_{ij} + s_{ij} Y_{ij}) \frac{-1}{1 - c_j} + Y_{ij}(1 - s_{ij}) \frac{1}{c_j}] = 0.$$

This implies that $c_j^{(t+1)}$ is updated according to

$$c_j = \frac{\sum_{i=1}^{N} (Y_{ij} - s_{ij} Y_{ij})}{\sum_{i=1}^{N} (1 - Y_{ij} + s_{ij} Y_{ij}) + \sum_{i=1}^{N} (Y_{ij} - s_{ij} Y_{ij})} = \frac{\sum_{i=1}^{N} Y_{ij}(1 - s_{ij})}{N}. \tag{16}$$

Following the same procedure, it is easy to check that $\boldsymbol{\xi}$ and $\Sigma_{\boldsymbol{\theta}}$ are updated with the same updating rule as in 2PL model (i.e. (7), (8)). Hence this completes the derivation of the M step for the 3PL model. In every M step, we iteratively update the tth iteration's estimate of the model parameters by (7), (8), (14), (15), and (16) until convergence, for $t \geq 1$ steps.

D Proof of Theorem 1

In this section, we provide theoretical bounds on the estimation of the model parameters. We follow the proof of Theorem 1 in Davenport, Plan, Van Den Berg, and Wootters (2014) and Theorem 1 in Chen, Li, and Zhang (2019). Define a matrix $M = [M_{ij}] = [\boldsymbol{\alpha}_j^T \boldsymbol{\theta}_i - b_j]$ and define $f(x)$ to be a logistic sigmoid function. For simplicity, we use the notation $\sum_{ij} = \sum_{i=1}^{N} \sum_{j=1}^{J}$ for the following proof. Then the variational lower bound to the marginal log-likelihood is as follows.

$$
\begin{aligned}
L_E(M) &= \sum_{i=1}^{N} \int_{\boldsymbol{\theta}_i} \log P(Y_i, \boldsymbol{\theta}_i | \boldsymbol{A}, \boldsymbol{B}) q_i(\boldsymbol{\theta}_i) d\boldsymbol{\theta}_i \\
&= \sum_{i=1}^{N} \int_{\boldsymbol{\theta}_i} \Bigg[\sum_{j=1}^{J} Y_{ij} \log(\frac{\exp(\boldsymbol{\alpha}_j^T \boldsymbol{\theta}_i - b_j)}{1 + \exp(a_j^T \boldsymbol{\theta}_i - b_j)}) \\
&\quad + (1 - Y_{ij}) \log(\frac{1}{1 + \exp(\boldsymbol{\alpha}_j^T \boldsymbol{\theta}_i - b_j)}) + \log \phi(\boldsymbol{\theta}_i) \Bigg] q_i(\boldsymbol{\theta}_i) d\boldsymbol{\theta}_i \\
&= \sum_{i=1}^{N} \sum_{j=1}^{J} Y_{ij} E_{q_i} \Bigg[\log(\frac{\exp(\boldsymbol{\alpha}_j^T \boldsymbol{\theta}_i - b_j)}{1 + \exp(a_j^T \boldsymbol{\theta}_i - b_j)}) \Bigg] \\
&\quad + (1 - Y_{ij}) E_{q_i} \Bigg[\log(\frac{1}{1 + \exp(\boldsymbol{\alpha}_j^T \boldsymbol{\theta}_i - b_j)}) \Bigg] + \sum_{i=1}^{N} E_{q_i} \Bigg[\log \phi(\boldsymbol{\theta}_i) \Bigg] \\
&= \sum_{ij} Y_{ij} E_{q_i} \big[\log(f(M_{ij})) \big] + (1 - Y_{ij}) E_{q_i} \big[\log(1 - f(M_{ij})) \big] + \sum_{i=1}^{N} E_{q_i} \big[\log \phi(\boldsymbol{\theta}_i) \big],
\end{aligned}
$$

where E_{q_i} denotes the expectation with respect to the distribution function $q_i(\boldsymbol{\theta}_i)$.

Define $\bar{L}_E(M) = L_E(M) - L_E(\mathbf{0})$ where $\mathbf{0}$ is a zero matrix with the same dimension as M. Then,

$$
\bar{L}_E(M) = \sum_{ij} Y_{ij} E_{q_i} \Bigg[\log \frac{f(M_{ij})}{f(0)} \Bigg] + (1 - Y_{ij}) E_{q_i} \Bigg[\log \frac{1 - f(M_{ij})}{1 - f(0)} \Bigg].
$$

By the Mean Value Theorem of integrals, we can express $E_{q_i}[\log f(M_{ij})] = \log f(\bar{M}_{ij})$ and $E_{q_i}[\log(1 - f(M_{ij}))] = \log(1 - f(\tilde{M}_{ij}))$ for some $\bar{M}_{ij}$ and $\tilde{M}_{ij}$. Since we only observe either

$Y_{ij} = 0$ or 1 for each data point, we then can rewrite $\bar{L}_E(M)$ as

$$\begin{aligned}
\bar{L}_E(M) &= \sum_{ij} Y_{ij}\left[\log\frac{f(\bar{M}_{ij})}{f(0)}\right] + (1 - Y_{ij})\left[\log\frac{1 - f(\tilde{M}_{ij})}{1 - f(0)}\right] \\
&= \sum_{ij} I_{\{Y_{ij}=1\}}\left[\log\frac{f(\bar{M}_{ij})}{f(0)}\right] + \sum_{ij} I_{\{Y_{ij}=0\}}\left[\log\frac{1 - f(\tilde{M}_{ij})}{1 - f(0)}\right] \\
&=: \bar{L}_{E_1}(\bar{M}) + \bar{L}_{E_0}(\tilde{M}). \qquad (17)
\end{aligned}$$

Define $G = \{M \in \mathbb{R}^{N\times J} : \|M\|_* \le C\sqrt{KNJ}\} \subset \mathbb{R}^{N\times J}$ for $C \ge 0$, where $\|M\|_*$ is defined as a nuclear norm of a matrix M. Define

$$H = \{M = [M_{ij}]_{1\le i\le N, 1\le j\le J} : M_{ij} = \boldsymbol{\alpha}_j^\top \boldsymbol{\theta}_i - b_j \text{ st } \|\boldsymbol{\theta}_i\|^2 \le C \text{ and } \|\boldsymbol{\alpha}_j\|^2 \le C \text{ for all } i, j\},$$

which is the set that satisfies the boundedness assumption $(A1)$. As shown in Chen et al. (2019), if $M \in H$ then $M \in G$ since

$$\|M\|_* \le \sqrt{NJ}\sqrt{\text{rank}(M)}\|M\|_\infty \le C\sqrt{KNJ}.$$

Note that rank$(M) = K$ in this proof as we assume that the number of latent traits K is fixed and known. For the following proof, we define $C_b = C_0 C\sqrt{K}\sqrt{NJ(N+J) + NJ\log(NJ)}$ with an absolute constant C_0 for simplicity. Hence we have

$$\begin{aligned}
&\mathbb{P}\left(\sup_{M\in H, \mu_i, \Sigma_i} \left|\bar{L}_E(M) - E[\bar{L}_E(M)]\right| \ge C_b\right) \\
\le\ &\mathbb{P}\left(\sup_{\bar{M}\in H, \tilde{M}\in H} \left|\bar{L}_{E_1}(\bar{M}) - E[\bar{L}_{E_1}(\bar{M})] + \bar{L}_{E_0}(\tilde{M}) - E[\bar{L}_{E_0}(\tilde{M})]\right| \ge C_b\right) \\
\le\ &\mathbb{P}\left(\sup_{M\in G} \left|\bar{L}_E(M) - E[\bar{L}_E(M)]\right| \ge C_b\right) \qquad (18)
\end{aligned}$$

where the first inequality follows from (17). The last expression in (18) satisfies conditions of Lemma A.1 of Davenport et al. (2014). Hence, we achieve the following result in lemma 1 for GVEM.

Lemma 1 *For absolute constants C_0 and C_1,*

$$\mathbb{P}\bigg(\sup_{M \in H, \mu_i, \Sigma_i} \left| \bar{L}_E(M) - E[\bar{L}_E(M)] \right| \geq C_0 C \sqrt{K} \sqrt{NJ(N+J) + NJ \log(NJ)} \bigg) \leq \frac{C_1}{N+J}.$$

We can show with slight modification to page 210 of Davenport et al. (2014) that for any choice of M and $M^{'} \in H$,

$$\begin{aligned} & E[\bar{L}_E(M^{'})] - E[\bar{L}_E(M)] = E[L_E(M^{'}) - L_E(M)] \\ = \; & E\bigg[\sum_{ij} Y_{ij} E_{q_i}\left[\log\left(\frac{f(M^{'}_{ij})}{f(M_{ij})} \right) \right] + (1 - Y_{ij}) E_{q_i}\left[\log\left(\frac{1 - f(M^{'}_{ij})}{1 - f(M_{ij})} \right) \right] \bigg] \\ = \; & \sum_{ij} f(M_{ij}) E_{q_i}\left[\log\left(\frac{f(M^{'}_{ij})}{f(M_{ij})} \right) \right] + (1 - f(M_{ij})) E_{q_i}\left[\log\left(\frac{1 - f(M^{'}_{ij})}{1 - f(M_{ij})} \right) \right] \\ = \; & -NJ E_q[D(f(M)||f(M^{'}))] \qquad (19) \end{aligned}$$

where $D(P\|Q) = \frac{1}{NJ}\sum_{ij} KL(P_{ij}\|Q_{ij})$ for $P, Q \in [0,1]^{N \times J}$ is the KL divergence defined on the matrices of scalar inputs $P_{ij}, Q_{ij} \in [0,1]$ for all i, j as defined in Davenport et al. (2014).

Now, define $\hat{L}_E(\hat{M}) = \bar{L}_E|_{\hat{\mu_i}, \hat{\boldsymbol{A}}, \hat{\boldsymbol{B}}, \hat{\Sigma_i}}$ which is estimated lower bound evaluated at the estimates from GVEM. It can be written as

$$\begin{aligned} \hat{L}_E(\hat{M}) \; = \; & \sum_{ij} Y_{ij} E_{\hat{q}_i}\left[\log\left(\frac{f(\hat{M}_{ij})}{f(0)} \right) \right] + (1 - Y_{ij}) E_{\hat{q}_i}\left[\log\left(\frac{1 - f(\hat{M}_{ij})}{1 - f(0)} \right) \right] \\ = \; & \sum_{ij} Y_{ij} E_{\hat{q}_i}\left[\log\left(\frac{f(\hat{\boldsymbol{\alpha}}_i^{\top} \boldsymbol{\theta}_i - \hat{b}_j)}{f(0)} \right) \right] + (1 - Y_{ij}) E_{\hat{q}_i}\left[\log\left(\frac{1 - f(\hat{\boldsymbol{\alpha}}_i^{\top} \boldsymbol{\theta}_i - \hat{b}_j)}{1 - f(0)} \right) \right]. \end{aligned}$$

Then, we have

$$\begin{aligned}
& \hat{L}_E(\hat{M}) - \bar{L}_E(M) \\
= \;& E[\hat{L}_E(\hat{M}) - \bar{L}_E(M)] + \hat{L}_E(\hat{M}) - E[\hat{L}_E(\hat{M})] - \big(\bar{L}_E(M) - E[\bar{L}_E(M)]\big) \\
\leq \;& E[\hat{L}_E(\hat{M}) - \bar{L}_E(M)] + 2 \sup_{M \in H, \mu_i, \Sigma_i} \left|\bar{L}_E(M) - E[\bar{L}_E(M)]\right| \\
= \;& -NJ \times E_{\hat{q}}[D(f(M)||f(\hat{M}))] + 2 \sup_{M \in H, \mu_i, \Sigma_i} \left|\bar{L}_E(M) - E[\bar{L}_E(M)]\right| \qquad (20)
\end{aligned}$$

where (20) follows from (19). Since the estimates $\hat{M}$ from GVEM should satisfy $\hat{L}_E(\hat{M}) \geq \bar{L}_E(M^*)$ for the true parameter matrix $M^* \in H$,

$$-NJE_{\hat{q}}[D(f(M^*)||f(\hat{M}))] + 2 \sup_{M \in H, \mu_i, \Sigma_i} \left|\bar{L}_E(M) - E[\bar{L}_E(M)]\right| \geq 0. \qquad (21)$$

By Lemma 1, with probability $1 - \frac{C_1}{N+J}$

$$\sup_{M \in H, \mu_i, \Sigma_i} \left|\bar{L}_E(M) - E[\bar{L}_E(M)]\right| \leq C_0 C \sqrt{K} \sqrt{NJ(N+J) + NJ\log(N+J)}. \qquad (22)$$

Combining (21) and (22),

$$E_{\hat{q}}[D(f(M^*)||f(\hat{M}))] \leq \frac{2C_0 C\sqrt{K}}{\sqrt{NJ}} \sqrt{(N+J) + \log(N+J)}. \qquad (23)$$

Note that the KL divergence can be bounded below by the Hellinger distance; $d_H^2(p,q) \leq D(p||q)$. Using this fact with Lemma A.2 from Davenport et al. (2014), we have for $C > 0$

$$||M^* - \hat{M}||_F^2 \leq \frac{8(1+e^C)^2}{e^C} NJ \times D(f(M)||f(\hat{M})) \leq 32e^C NJ \times D(f(M)||f(\hat{M})). \qquad (24)$$

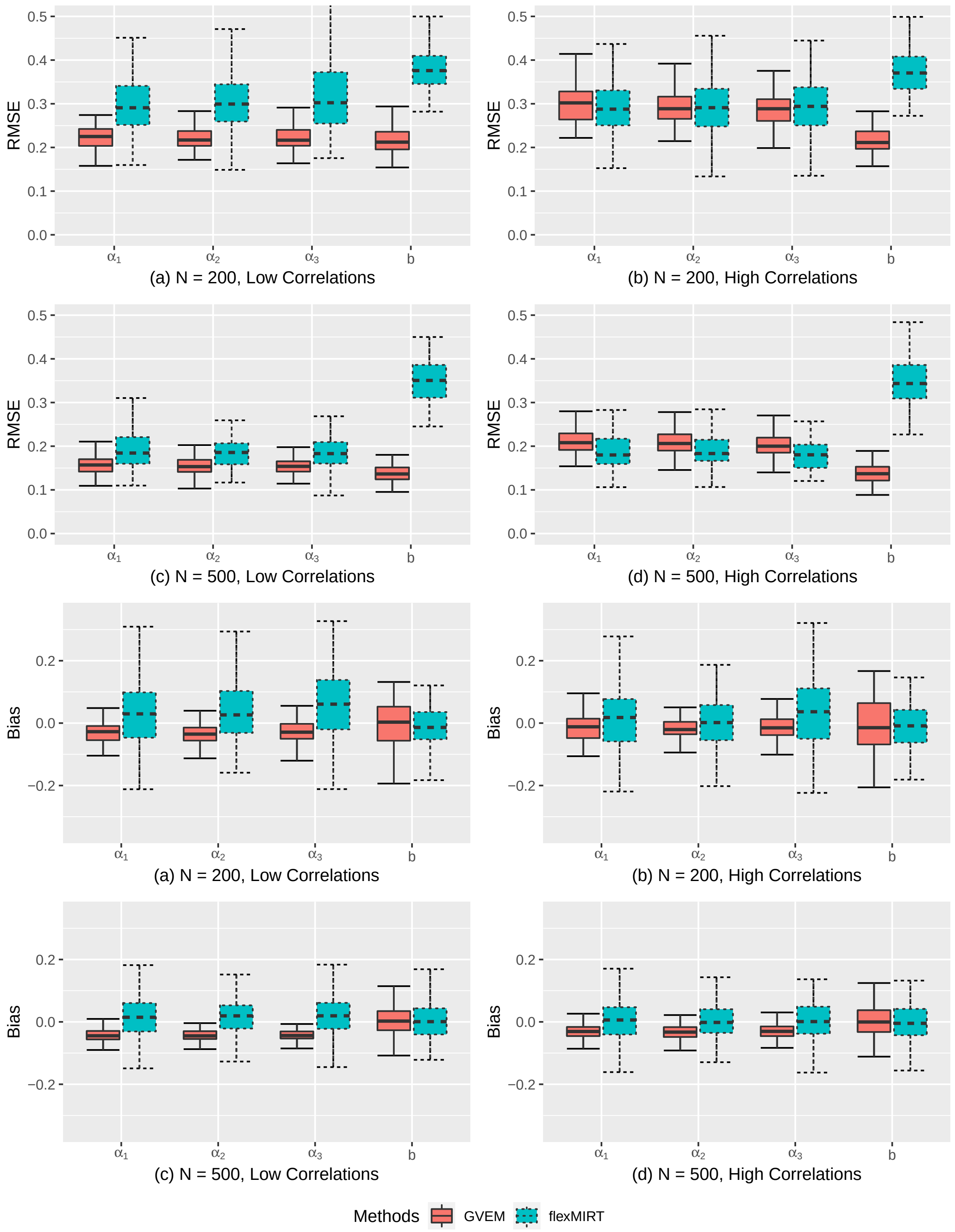
From (23) and (24),

$$\frac{1}{NJ}E_{\hat{q}}[||M^* - \hat{M}||_F^2] \leq 64e^C C_0 C\sqrt{K}\sqrt{\frac{N+J}{NJ}}\sqrt{1+\frac{\log(N+J)}{N+J}}. \tag{25}$$

This completes the proof of Theorem 1.

E Additional Simulation Results

We also tried other tuning methods in flexMIRT and found that a more careful tuning can improve the performance of MHRM as in 'mirt' package; on the other hand, the estimation results can be very sensitive to the tuning, and the optimal tuning of MHRM could vary case by case, depending on the model setting and the correlation of the latent traits. In particular, following one reviewer's kind suggestion, we used the strategy of combining mirt's default Stage 3 setup with flexMIRT's default Stages 1 and 2 setup. Figure 1 shows that it provides slightly better estimation results than the proposed GVEM under the high correlation and between item model setting, while slightly worse under the low correlation. On the other hand, this tuning strategy does not improve the estimation results of the 'mirt' package under the considered within item models, whose results are therefore not reported.



(a) N = 200, Low Correlations

(b) N = 200, High Correlations

(c) N = 500, Low Correlations

(d) N = 500, High Correlations

(a) N = 200, Low Correlations

(b) N = 200, High Correlations

(c) N = 500, Low Correlations

(d) N = 500, High Correlations

Figure 1: Parameter recovery of the between-item M2PL models from exploratory factor analysis

F R code

In this section, we provide R codes that are used for the real data analysis with annotations so that interested readers can apply the GVEM algorithm themselves.

```
[baselinestretch=0.75]
#Helper function
eta <- function(x){
  y <- exp(x)/(exp(x)+1) - 1/2
  eta <- y/(2*x)
  #limit of eta is 0.125.
  eta[abs(eta)< 0.01] <- 0.125
}


#Initialization of model parameters
init <- function(Y, K, method=’EFA’){
  N = dim(Y)[1]
  J = dim(Y)[2]
  if(method == ’dimselect’){
    indic <- diag(K)
    indic[lower.tri(indic)] <- rep(1,sum(1:(K-1)))
    indic <- rbind(indic, matrix(1, J-K, K))
  }else if(method=’EFA’){
    indic = matrix(1, nrow=J, ncol=K)
  }else{
    print(’Wrong method : please check your input argument’)
  }
  num_correct <- rowSums(Y) #number of items responded correctly per examinee
  p = apply(Y, 2, mean) #percentage of examinee who got item j right
  q = 1 - p
  dens = dnorm(qnorm(p,0,1),0,1)
  r = numeric(J)
  s <- sd(num_correct)
  for(j in 1:J){
    x1 = num_correct[which(Y[,j]==1)] #Num of items responded correctly by
    #examinee who has correct response for item j
    x2 = num_correct[which(Y[,j]==0)] #num of items responded correctly by
    #examinee who has incorrect response for item j
    r[j] = (mean(x1) - mean(x2))/(s*dens[j]) * p[j] * q[j]
  }
  #Bound r between 0 and 1 to prevent division by 0
  r[which(r > 0.999)] <- 0.999
  r[which(r < 0 )] <- 0
  A0 <- t(array(1, dim=c(K,1)) %*% (r/sqrt(1-r^2))) * indic
  # colSums(Y) is number of examinees who got item j correct
  B0 <- -qnorm(colSums(Y)/N, 0, 1)/r
  C0 <- runif(0.05, 0.3, J) #additional parameter in M3PL model
```

```
  sig_theta0 <- diag(K)
  theta <- matrix(rnorm(N*K, 0, 1), nrow=N, ncol=K)
  eps0 <- array(1,N)%*%t(B0) - theta%*%t(A0);
  return(list(A0=A0, B0=B0, C0=C0, sig_theta0 = sig_theta0,
              eps0 = eps0, theta0 = theta))
}

#Exploratory Factor Analysis with Gaussian Variaitonal E-M algorithm.
GVEM <- function(old_A, old_B, old_eps, Y, K){
  N = dim(Y)[1]
  J = dim(Y)[2]
  converged <- FALSE
  old_eta <- eta(old_eps)
  while(!converged){
    tryCatch({
      sig_i <- array(0, dim=c(K,K,N)); mu_i <- matrix(0, nrow=K, ncol=N);
      new_eps <- matrix(0, nrow=N, ncol=J)
      for(i in 1:N){
        sigma_sum = matrix(0, nrow=K, ncol=K);
        mu_sum = numeric(K)
        for(j in 1:J){
          sigma_sum <- sigma_sum + old_eta[i,j]*(old_A[j,]%*%t(old_A[j,]))
          mu_sum <- mu_sum + (2*old_eta[i,j]*old_B[j] + Y[i,j] - 0.5)*old_A[j,]
  }
  sigma_i_hat <- solve(diag(K) + 2*sigma_sum)
  mu_i_hat <- sigma_i_hat %*% mu_sum
  sig_i[,,i] <- sigma_i_hat; mu_i[,i] <- mu_i_hat;
  new_eps[i,] <- sqrt(old_B^2-2*old_B*(old_A%*%mu_i_hat) + diag(old_A%*%(
                        sigma_i_hat+mu_i_hat%*%t(mu_i_hat))%*%t(old_A)))
}
new_eta <- eta(new_eps)
new_B <- rep(0, J)
num <- numeric(J)
for(i in 1:N){
  num <- num + (1/2 - Y[i,] + 2*new_eta[i,]*(old_A%*% mu_i[,i]))
}
new_B <- num/colSums(2*new_eta)
new_A <- matrix(0, nrow=J, ncol=K)
for(j in 1:J){
  mat = matrix(0, K, K); vec <- numeric(K)
  for(i in 1:N){
    mat <- mat + new_eta[i,j]*sig_i[,,i] +
                new_eta[i,j]*mu_i[,i] %*% t(mu_i[,i])
    vec <- vec + (Y[i,j] - 1/2 + 2*new_B[j]*new_eta[i,j])*mu_i[,i]
  }
  new_A[j,] <- solve(mat) %*% vec/2
```

```
      }
      #Check if convergence criterion is satisfied
      converged <- (sqrt(sum((new_A - old_A)^2) ) +
                      sqrt(sum((new_B - old_B)^2)) < 0.0001)
      #Replace values with new updates
      old_A <- new_A; old_B <- new_B; old_eps <- new_eps; old_eta <- new_eta;
    }, error=function(e){cat("ERROR :",conditionMessage(e), "\n")})
  }
  #Can choose other rotation methods
  rotated <- Promax(old_A)
  return(list(A_hat = rotated$loadings, B_hat = old_B, mu_i = mu_i,
              sig_i = sig_i, sig_hat = rotated$Phi, eps = old_eps))
}


#Function used for dimension selection
GVEM_dimselect <- function(old_A, old_B, old_eps, Y, K){
  N = dim(Y)[1]
  J = dim(Y)[2]
  #Using Q structure with zero triangular submatrix often helps
  #the estimation in dimension selection. Theoretically there is
  #no difference in estimation results from using such as Q structure.
  indic <- diag(K)
  indic[lower.tri(indic)] <- rep(1,sum(1:(K-1)))
```

```
indic <- rbind(indic, matrix(1, J-K, K))

converged <- FALSE

old_eta <- eta(old_eps)

old_sig = diag(K)

while(!converged){

  tryCatch({

    sig_i <- array(0, dim=c(K,K,N)); mu_i <- matrix(0, nrow=K, ncol=N);

    new_eps <- matrix(0, nrow=N, ncol=J)

    sig_theta_sum <- matrix(0, K, K)

    for(i in 1:N){

      sigma_sum = matrix(0, nrow=K, ncol=K);

      mu_sum = numeric(K)

      for(j in 1:J){

        sigma_sum <- sigma_sum + old_eta[i,j]*(old_A[j,]%*%t(old_A[j,]))

        mu_sum <- mu_sum + (2*old_eta[i,j]*old_B[j] + Y[i,j] - 0.5)*old_A[j,]

      }

      sigma_i_hat <- solve(solve(old_sig) + 2*sigma_sum)

      mu_i_hat <- sigma_i_hat %*% mu_sum

      sig_i[,,i] <- sigma_i_hat; mu_i[,i] <- mu_i_hat;

      new_eps[i,]<-sqrt(old_B^2-2*old_B*(old_A%*%mu_i_hat) + diag(old_A%*%(

                        sigma_i_hat+mu_i_hat%*%t(mu_i_hat))%*%t(old_A)))

      sig_theta_sum <- sig_theta_sum + sigma_i_hat + mu_i_hat%*%t(mu_i_hat)

    }
```

```
new_eta <- eta(new_eps)

new_sig <- sig_theta_sum/N

corr <- matrix(0, nrow=K, ncol=K)

for(t in 1:K){

  for(s in 1:K){

    #rescale to diag=1

    corr[t,s] <- new_sig[t,s]/sqrt(new_sig[t,t]*new_sig[s,s])

  }

}

new_sig <- corr

new_B <- rep(0, J)

num <- numeric(J)

for(i in 1:N){

  num <- num + (1/2 - Y[i,] + 2*new_eta[i,]*(old_A%*% mu_i[,i]))

}

new_B <- num/colSums(2*new_eta)

new_A <- matrix(0, nrow=J, ncol=K)

for(j in 1:J){

  ind <- which(indic[j,] > 0)

  n <- length(ind)

  mat = matrix(0, n, n); vec <- numeric(n)

  for(i in 1:N){

    sub_sigma <- sig_i[ind,ind,i]
```

```
          sub_mu <-  mu_i[ind,i]
          mat <- mat + new_eta[i,j]*sub_sigma +
                    new_eta[i,j]*sub_mu %*% t(sub_mu)
          vec <- vec + (Y[i,j] - 1/2 + 2*new_B[j]*new_eta[i,j])*sub_mu
        }
        new_A[j,ind] <- solve(mat) %*% vec/2
      }
      #Check if convergence criterion is satisfied
      converged <- (sqrt(sum((new_A - old_A)^2) ) +
                      sqrt(sum((new_B - old_B)^2)) < 0.0001)
      #Replace values with new updates
      old_A <- new_A; old_B <- new_B; old_eps <- new_eps;
      old_eta <- new_eta; old_sig <- new_sig
    }, error=function(e){cat("ERROR :",conditionMessage(e), "\n")})
  }
  return(list(A_hat = old_A, B_hat = old_B, mu_i = mu_i,
              sig_i = sig_i, sig_hat = old_sig, eps = old_eps))
}


#Calculate variational lowerbound with parameter estimates
lower_bound <- function(est, Y){
  N <- dim(Y)[1]
  J <- dim(Y)[2]
  A <- est$A_hat; B <- est$B_hat; mu_i <- est$mu_i; sig_i <- est$sig_i
  sig_th <- est$sig_hat; eps <- est$eps; eta_ij <- eta(eps)
  sum1 <- 0; sum2 <- 0
  for(i in 1:N){
    for(j in 1:J){
      sum1 <- sum1 + log(exp(eps[i,j])/(1+exp(eps[i,j]))) + (1/2-Y[i,j])*B[j] +
        (Y[i,j]-1/2)*A[j,]%*%mu_i[,i] - 1/2*eps[i,j] - eta_ij[i,j]*(B[j]^2 -
        2*B[j]*A[j,]%*%mu_i[,i] + A[j,]%*%(sig_i[,,i] +
        mu_i[,i]%*%t(mu_i[,i]))%*%A[j,] - eps[i,j]^2)
    }
    sum2 <- sum2 + tr(solve(sig_th)%*%(sig_i[,,i] + mu_i[,i]%*%t(mu_i[,i])))
  }
  lb <- sum1 - 1/2*sum2 + N/2*log(det(solve(sig_th)))
  return(lb)
}


#Stochastic optimization of GVEM algorithm
stoGVEM <- function(old_A, old_B, old_eps, Y, K, samp_size, forgetrate,
                    initial_samp){
  N = dim(Y)[1]
  J = dim(Y)[2]
  iter = 1
```

```
converged <- FALSE

old_eta <- eta(old_eps)

old_mat = matrix(0, K, K); old_vec <- numeric(K)

old_sum <- numeric(J); old_num <- numeric(J)

while(!converged){

  tryCatch({

    if(iter == 1){

      #For the first interation, use all data points for updates

      dec_step = 1;

      id = sample(N,initial_samp)

    }

    else{

      #Subsample small portion of dataset for efficiency

      dec_step = (iter+1)^(-forgetrate)

      id = sample(N,samp_size)

    }

    sig_i <- array(0, dim=c(K,K,N)); mu_i <- matrix(0, nrow=K, ncol=N);

    new_eps <- matrix(0, nrow=N, ncol=J)

    for(i in id){

      sigma_sum = matrix(0, nrow=K, ncol=K);

      mu_sum = numeric(K)

      for(j in 1:J){

        sigma_sum <- sigma_sum + old_eta[i,j]*(old_A[j,]%*%t(old_A[j,]))
        mu_sum <- mu_sum + (2*old_eta[i,j]*old_B[j] + Y[i,j] - 0.5)*old_A[j,]
      }
      sigma_i_hat <- solve(diag(K) + 2*sigma_sum)
      mu_i_hat <- sigma_i_hat %*% mu_sum
      sig_i[,,i] <- sigma_i_hat; mu_i[,i] <- mu_i_hat;
      new_eps[i,] <- sqrt(old_B^2-2*old_B*(old_A%*%mu_i_hat) + diag(old_A%*%(
                          sigma_i_hat+mu_i_hat%*%t(mu_i_hat))%*%t(old_A)))
    }
    new_eta <- eta(new_eps)
    new_B <- rep(0, J)
    num <- numeric(J)
    for(i in id){
      num <- num + (1/2 - Y[i,] + 2*new_eta[i,]*(old_A%*% mu_i[,i]))
    }
    new_B <- (dec_step*num+(1-dec_step)*old_num)/
                  (dec_step*colSums(2*new_eta)+(1-dec_step)*old_sum)
    new_A <- matrix(0, nrow=J, ncol=K)
    for(j in 1:J){
      mat = matrix(0, K, K); vec <- numeric(K)
      for(i in id){
        mat <- mat + new_eta[i,j]*sig_i[,,i] +
                new_eta[i,j]*mu_i[,i] %*% t(mu_i[,i])
        vec <- vec + (Y[i,j] - 1/2 + 2*new_B[j]*new_eta[i,j])*mu_i[,i]
```

```
        }
        new_A[j,] <- solve(dec_step*mat+(1-dec_step)*old_mat) %*%
          (dec_step*vec+(1-dec_step)*old_vec)/2
      }
      #Check if convergence criterion is satisfied
      converged <- (sqrt(sum((new_A - old_A)^2) ) +
                      sqrt(sum((new_B - old_B)^2)) < 0.0001)
      #Save old weighted averages from next stochastic update
      old_mat = dec_step*mat + (1-dec_step)*old_mat
      old_vec = dec_step*vec + (1-dec_step)*old_vec
      old_sum = dec_step*colSums(2*new_eta) + (1-dec_step)*old_sum
      old_num = dec_step*num + (1-dec_step)*old_num

      #Replace values with new updates
      old_A <- new_A; old_B <- new_B; old_eps <- new_eps; old_eta <- new_eta;
    }, error=function(e){cat("ERROR :",conditionMessage(e), "\n")})
  }
  rotated <- Promax(old_A)
  return(list(A_hat = rotated$loadings, B_hat = old_B, mu_i = mu_i,
              sig_i = sig_i, sig_hat = rotated$Phi, eps = old_eps))
}

# Real Data analysis
```

```
require('psych')

set.seed('FOR REPRODUCIBILITY, CHOOSE SEED NUMBER')

setwd('SET YOUR WORKING DIRECTORY')

NELS <- read.csv("FILE_NAME.csv", header=T)

J = dim(NELS)[2] #Number of test items


#1. Run Exploratory Factor Analysis assuming six latent factors under M2PL

p0 <- init(NELS, K=6)

phat <- GVEM(p0$A0, p0$B0, p0$eps0, NELS, K=6)


#2. One can run stochastic version of GVEM in both 2PL and 3PL if

#there are high number of parameters to update in every EM steps.

#Specify samp_size and forgetrate that you would like to use for

#stochacstic optimization

p0 <- init(NELS, K=6)

phat <- stoGVEM(p0$A0, p0$B0, p0$eps0, NELS, K=6, samp_size,

                forgetrate, intial_samp)


#3. Dimension Selection under M2PL

BIC_hat <- numeric(0); AIC_hat <- numeric(0);

#Specify the range of factor dimensions you would like to test for your data

dimensions = 4:7

for(candidate_k in dimensions){
  p0 <- init(NELS, candidate_k, method='dimselect')

  phat <- GVEM_dimselect(p0$A0, p0$B0, p0$eps0, as.matrix(NELS), candidate_k)

  var_lowerboud <- lower_bound(phat, NELS)

  Q_hat <- (phat$A_hat != 0)*1

  BIC_hat <- append(BIC_hat, log(N)*(sum(Q_hat) + J) - 2*var_lowerboud)

  AIC_hat <- append(AIC_hat, 2*(sum(Q_hat) + J) - 2*var_lowerboud)

}

min(BIC_hat)

min(AIC_hat)
```

References

Chen, Y., Li, X., & Zhang, S. (2019). Joint maximum likelihood estimation for high-dimensional exploratory item factor analysis. *Psychometrika*, *84*(1), 124–146.

Davenport, M. A., Plan, Y., Van Den Berg, E., & Wootters, M. (2014). 1-bit matrix completion. *Information and Inference: A Journal of the IMA*, *3*(3), 189–223.