SUPPLEMENT TO "ROBUST MEASUREMENT VIA A FUSED LATENT AND GRAPHICAL ITEM RESPONSE THEORY MODEL"

A. Computation via an Alternating Minimization

A.1. Proximal Gradient Descent Update

The proximal gradient descent update is designed for solving nonsmooth convex optimization problems (Parikh and Boyd, 2014). Consider optimization problem

$$\min_x f(x) + g(x), \tag{A.1}$$

where $x \in \mathbb{R}^n$, f is a smooth convex function, and g is a continuous but nonsmooth convex function. Due to the nonsmoothness of g, the traditional gradient descent algorithm cannot be directly applied, because the gradient of g does not always exist. The proximal gradient descent update can be viewed as a variant of the gradient descent update that accounts for the nonsmoothness.

To describe the proximal gradient descent update, we first introduce the proximal operator $\mathbf{P}_{\lambda,g}$: $\mathbb{R}^n \to \mathbb{R}^n$ as

$$\mathbf{P}_{\lambda,g}(v) = \arg\min_{x \in \mathbb{R}^n} \{g(x) + \frac{1}{2\lambda}\|x - v\|^2\}$$

and proximal gradient descent update is

$$x^{t+1} = \mathbf{P}_{\lambda,g}(x^t - \lambda \nabla f(x^t)), \tag{A.2}$$

where x^t and x^{t+1} are the current and updated values. It can be shown that for a sufficiently small λ,

$$f(x^{t+1}) + g(x^{t+1}) < f(x^t) + g(x^t),$$

if x^t is not the an optimal solution. Thus, one can always search for a step size λ, such that the objective function decreases. When function g has a special form, the proximal gradient descent update (A.2) may have a closed form solution, which is indeed the case in our algorithm below.

A.2. An Alternating Minimization Algorithm

We use an alternating minimization algorithm for optimizing (3.3). The positive semidefinite constraint on Σ is not easy to handle in the computation. Therefore, we reparameterize $\Sigma = BB^\top$, where $B = (b_{ij})$ is a $K \times K$ lower triangle matrix. In addition, instead of constraining $\Sigma_{kk} = 1$, we require b_{kk} to be 1. There is a one to one correspondence between the two sets of parametrization and the transformation will be discussed in Remark 4. We let $l(A, S, B) = \log \mathcal{L}(A, S, BB^\top)$ and

$$H_\gamma(A, S, B) = -\frac{1}{N} l(A, S, B) + \gamma \sum_{i \neq j} |s_{ij}|$$

be the objective function. Then the alternating minimization algorithm alternates between updating A, S, and B iteratively, so that the values of (A, S, B), denoted by (A^t, S^t, B^t), satisfy

$$H_\gamma(A^t, S^t, B^t) > H_\gamma(A^{t+1}, S^t, B^t) > H_\gamma(A^{t+1}, S^{t+1}, B^t) > H_\gamma(A^{t+1}, S^{t+1}, B^{t+1}),$$

for all t. Specifically, A and B are updated using a gradient descent method and S is updated using the proximal gradient descent method. We summarize the algorithm as follows, given the current parameter values (A^t, S^t, B^t).

Algorithm 1. An Alternating Minimization Algorithm.

1. Update
$$A^{t+1} \leftarrow A^t - \alpha^t g_A(A^t, S^t, B^t),$$
where $g_A(A^t, S^t, B^t)$ is the gradient of $-l(A, S, B)/N$ with respect to A at (A^t, S^t, B^t). The step size α^t is chosen by line searching, such that $H_\gamma(A^{t+1}, S^t, B^t) < H_\gamma(A^t, S^t, B^t)$.

2. Update
$$S^{t+1} \leftarrow \mathbf{P}_{\lambda^t, h_\gamma}(S^t - \lambda^t g_S(A^{t+1}, S^t)),$$
where $h_\gamma(S) = \gamma \sum_{i \neq j} |s_{ij}|$ is the regularization function, $g_S(A^{t+1}, S^t, B^t))$ is the gradient of $-l(A, S, B)/N$ with respect to S at (A^{t+1}, S^t, B^t). In addition, λ^t is the step size for a proximal gradient operator chosen by line searching, satisfying
$$H_\gamma(A^{t+1}, S^{t+1}, B^t) < H_\gamma(A^{t+1}, S^t, B^t). \tag{A.3}$$

3. Update
$$B^{t+1} \leftarrow B^t - \beta^t g_B(A^{t+1}, S^{t+1}, B^t),$$
where $g_B(A^{t+1}, S^{t+1}, B^t)$ is the gradient of $-l(A, S, B)/N$ with respect to B evaluated at (A^{t+1}, S^{t+1}, B^t). The step size β^t is chosen by line searching, such that $H_\gamma(A^{t+1}, S^{t+1}, B^{t+1}) < H_\gamma(A^{t+1}, S^{t+1}, B^t)$.

4. Iterates between the above three steps until convergence.

We make a few remarks.

Remark 1. $-l(A, S, B)/N$ is a smooth function of (A, S, B) and its gradients g_A, g_S, and g_B have analytic forms.

Remark 2. In Step 2, $H_\gamma(A^{t+1}, S, B^t)$, when viewed as a function of S (with A^{t+1} and B^t fixed), is the sum of a smooth convex function $-l(A^{t+1}, S, B^t)/N$ and a nonsmooth convex function $h_\gamma(S) = \gamma \sum_{i \neq j} |s_{ij}|$. Therefore, according to the dis-

cussion in Appendix A.1, there exists a sufficiently small step size λ^t, such that (A.3) is satisfied.

Remark 3. The proximal operator $\mathbf{P}_{\lambda^t,h_\gamma}(\cdot)$ has a closed form solution. Let $\tilde{S} = S^t - \lambda^t g_S(A^{t+1}, S^t, B^t)$. Then $s_{jj}^{t+1} = \tilde{s}_{jj}$ and s_{ij}^{t+1} is obtained by soft-thresholding

$$s_{ij}^{t+1} = \begin{cases} \frac{1}{2}(\tilde{s}_{ij} + \tilde{s}_{ji}) - \lambda^t\gamma & \text{if } \frac{1}{2}(\tilde{s}_{ij} + \tilde{s}_{ji}) > \gamma\lambda^t; \\ 0 & \text{if } |\frac{1}{2}(\tilde{s}_{ij} + \tilde{s}_{ji})| \leq \gamma\lambda^t; \\ \frac{1}{2}(\tilde{s}_{ij} + \tilde{s}_{ji}) + \lambda^t\gamma & \text{if } \frac{1}{2}(\tilde{s}_{ij} + \tilde{s}_{ji}) < -\gamma\lambda^t. \end{cases}$$

Remark 4. Given estimates $\hat{A}^\gamma$, $\hat{S}^\gamma$, and $\hat{B}^\gamma$ from optimizing $H_\gamma(A, S, B)$, the estimates under the parametrization in (3.3) can be obtained by

$$\hat{A}^\gamma \leftarrow \hat{A}^\gamma D, \quad \hat{S}^\gamma \leftarrow \hat{S}^\gamma, \quad \hat{\Sigma}^\gamma \leftarrow D^{-1}\hat{B}^\gamma(\hat{B}^\gamma)^\top D^{-1},$$

where $D = diag(d_{11}, ..., d_{KK})$ is a $K \times K$ diagonal matrix with

$$d_{kk} = \sqrt{(\hat{B}^\gamma(\hat{B}^\gamma)^\top)_{kk}}.$$

Remark 5. A classical way of updating S, which is a Lasso problem, is via coordinate descent (e.g. Friedman *et al.*, 2010). In the proposed algorithm, we adopt a proximal gradient update instead of coordinate descent for two reasons. First, a coordinate-wise decent update (i.e. optimizing with respect to a single parameter at a time) does not have a closed form, due to the form of the log-likelihood function. Second, for Lasso type problems, better properties have been found for the performance of the proximal gradient algorithm comparing to the coordinate descent algorithm (see Parikh and Boyd (2014) and reference therein).

B. Simulation Study 3

We further provide a simulation study to evaluate the performance of the FLaG-IRT analysis under model misspecification. Specifically, we consider the same simulation setting as Study 1, where the generating model is not a FLaG-IRT model. For ease of presentation, we choose $c = 2$. Sample sizes $N = 500$, 1000, and 3000 are considered. The same evaluation criteria as in Study 2 are adopted. In addition, Kendall's tau correlation between the EAP scores from the selected model and the true nuisance factor scores is computed, as a measure of measurement bias. In particular, the TPR and FPR are calculated by considering $E = \{(i, j) : i, j \leq 5\}$ as the true graph.

Results are shown in Table 1, where the oracle values of the Kendall's tau correlations are based on the true bi-factor model. According to the results, even when data are not from a FLaG-IRT model, under all values of $\rho = 0, 0.25, 0.5$ of the extended Bayesian information criterion, the selected models from the FLaG-IRT analysis have high measurement accuracy and low measurement bias, almost as if the true model is being used. In addition, under this setting, the true positive rate is close to 1, under all sample sizes and all values of ρ of the EBIC. Similar to the results of Study 2, the EBIC with a smaller value of ρ yields a higher FPR. In addition, the FPR decreases as the sample size increases. These results indicate that even under model misspecification, the FLaG-IRT model approximates the latent structure well and the proposed FLaG-IRT analysis provides reliable measurement when the local dependence structure is unknown. In addition, the quality of measurement is robust to the choice of the parameter ρ in the extended Bayesian information criterion.

		$\rho = 0$	$\rho = 0.25$	$\rho = 0.5$	Oracle
	Kendall's tau (accuracy)	0.70 (0.001)	0.70 (0.001)	0.70 (0.001)	0.71
	Kendall's tau (bias)	0.09 (0.004)	0.09 (0.004)	0.08 (0.004)	0.08
$N = 500$	TPR	0.98 (0.004)	0.97 (0.005)	0.96 (0.006)	1
	FPR	0.07 (0.004)	0.03 (0.002)	0.02 (0.002)	0
	Kendall's tau (accuracy)	0.70 (0.001)	0.70 (0.001)	0.71 (0.001)	0.71
	Kendall's tau (bias)	0.08 (0.003)	0.08 (0.002)	0.08 (0.002)	0.08
$N = 1000$	TPR	1.00 (0.001)	0.99 (0.001)	0.99 (0.002)	1
	FPR	0.06 (0.003)	0.03 (0.002)	0.01 (0.001)	0
	Kendall's tau (accuracy)	0.71 (0.000)	0.71 (0.000)	0.71 (0.000)	0.71
	Kendall's tau (bias)	0.08 (0.001)	0.08 (0.001)	0.08 (0.001)	0.08
$N = 3000$	TPR	1.00 (0.000)	1.00 (0.000)	1.00 (0.000)	1
	FPR	0.04 (0.003)	0.02 (0.002)	0.01 (0.001)	0

TABLE 1.
Study 3: Performance of FLaG-IRT analysis under model misspecification. The average of each evaluation measure and its standard error over 100 independent replications are reported.

References

Friedman, J., Hastie, T. and Tibshirani, R. (2010) Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, **33**, 1–22.

Parikh, N. and Boyd, S. P. (2014) Proximal algorithms. *Foundations and Trends in Optimization*, **1**, 127–239.