Published in final edited form as:

J Comput Graph Stat. 2017 ; 26(4): 814–825. doi:10.1080/10618600.2017.1340891.

Sequential Co-Sparse Factor Regression

Aditya Mishra, Dipak K. Dey, and Kun Chen*

Department of Statistics, University of Connecticut

Abstract

In multivariate regression models, a sparse singular value decomposition of the regression component matrix is appealing for reducing dimensionality and facilitating interpretation. However, the recovery of such a decomposition remains very challenging, largely due to the simultaneous presence of orthogonality constraints and co-sparsity regularization. By delving into the underlying statistical data generation mechanism, we reformulate the problem as a supervised co-sparse factor analysis, and develop an efficient computational procedure, named sequential factor extraction via co-sparse unit-rank estimation (SeCURE), that completely bypasses the orthogonality requirements. At each step, the problem reduces to a sparse multivariate regression with a unit-rank constraint. Nicely, each sequentially extracted sparse and unit-rank coefficient matrix automatically leads to co-sparsity in its pair of singular vectors. Each latent factor is thus a sparse linear combination of the predictors and may influence only a subset of responses. The proposed algorithm is guaranteed to converge, and it ensures efficient computation even with incomplete data and/or when enforcing exact orthogonality is desired. Our estimators enjoy the oracle properties asymptotically; a non-asymptotic error bound further reveals some interesting finite-sample behaviors of the estimators. The efficacy of SeCURE is demonstrated by simulation studies and two applications in genetics.

Keywords

multivariate analysis; reduced-rank regression; regularization; singular value decomposition

1 Introduction

We are interested in studying the predictive relationship between a multivariate response $\mathbf{y} \in \mathbb{R}^q$ and a multivariate predictor $\mathbf{x} \in \mathbb{R}^p$, using the multivariate linear regression model

$$\mathbf{y} = \mathbf{C}^{*T} \mathbf{x} + \mathbf{e}, \quad (1)$$

*Corresponding author.

Supplementary Materials

The online Supplementary Materials include additional simulation results, a biclustering example using gene expression data, more results in the yeast cycle data analysis, details on handling incomplete data and exact orthogonality, and all the technical proofs. Implementations of the proposed methods are available in the R package *secure* (R Development Core Team, 2017), which can be accessed at <https://CRAN.R-project.org/package=secure>.

where $\mathbf{C}^* \in \mathbb{R}^{p \times q}$ is an unknown coefficient matrix, and $\mathbf{e} \in \mathbb{R}^q$ is a random error vector of zero mean. Given n independent observations $(\mathbf{y}_i, \mathbf{x}_i)$, $i = 1, \dots, n$, let $\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_n]^T \in \mathbb{R}^{n \times q}$ be the response matrix, $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_n]^T \in \mathbb{R}^{n \times p}$ the predictor/design matrix, and $\mathbf{E} = [\mathbf{e}_1, \dots, \mathbf{e}_n]^T \in \mathbb{R}^{n \times q}$ the error matrix. Then the sample model can be expressed as

$$\mathbf{Y} = \mathbf{X}\mathbf{C}^* + \mathbf{E}. \quad (2)$$

The responses and the predictors are assumed to be centered, so there is no intercept term. In many applications, it is desirable to assume that $\mathbf{C}^*$ admits certain low-dimensional structures, the exploration of which could utilize the multivariate dependency, alleviate the curse of dimensionality, and facilitate model interpretation. In particular, $\mathbf{C}^*$ could be of low rank, i.e., $r^* = \text{rank}(\mathbf{C}^*) \leq \min(r_x, q)$, where $r_x = \text{rank}(\mathbf{X})$.

To recover the potential low-rank structure in $\mathbf{C}^*$, the reduced-rank regression (RRR) (Anderson, 1951; Reinsel and Velu, 1998) was formulated as a rank-constrained estimation,

$$\min_{\mathbf{C}} \|\mathbf{Y} - \mathbf{X}\mathbf{C}\|_F^2, \quad \text{s.t.} \quad \text{rank}(\mathbf{C}) \leq r, \quad (3)$$

for $1 \leq r \leq \min(r_x, q)$, where $\|\cdot\|_F$ denotes the Frobenius norm. This reduces the

effective number of regression parameters from $r_x q$ to $(r_x + q - r)r$, and the reduction can be substantial when r is much smaller than both r_x and q . Bunea *et al.* (2011) formulated the rank-constrained estimation as a regularized regression problem in which the penalty is proportional to the ℓ_1 norm of the singular values of $\mathbf{C}$. The convex nuclear norm penalization approach achieved rank reduction through penalizing the ℓ_1 norm of the singular values of $\mathbf{C}$ (Yuan *et al.* 2007; Negahban and Wainwright, 2011; Lu *et al.* 2012). See also, Mukherjee and Zhu (2011), Chen *et al.* (2013) and Zhou and Li (2014).

It is fair to say that the reduced-rank structure is one of the most critical ingredients in multivariate learning. The RRR naturally connects to factor analysis (FA) and principal component analysis (PCA): both FA/PCA and RRR aim to identify certain low-dimensional subspace to represent $\mathbf{Y}$, and the main difference is that the latter conducts a supervised search in the column space of $\mathbf{X}$ while the search in the former is unsupervised. Motivated by these connections, a sparse SVD representation of $\mathbf{X}\mathbf{C}^*$ is quite appealing,

$$\frac{1}{\sqrt{n}}\mathbf{X}\mathbf{C}^* = \left(\frac{1}{\sqrt{n}}\mathbf{X}\mathbf{U}^*\right)\mathbf{D}^*\mathbf{V}^{*T}, \quad \text{s.t.} \quad \left(\frac{1}{\sqrt{n}}\mathbf{X}\mathbf{U}^*\right)^T \left(\frac{1}{\sqrt{n}}\mathbf{X}\mathbf{U}^*\right) = \mathbf{V}^{*T}\mathbf{V}^* = \mathbf{I}_{r^*}, \quad (4)$$

where $\mathbf{C}^* = \mathbf{U}^* \mathbf{D}^* \mathbf{V}^{*\top}$, $\mathbf{I}_{r^*}$ denotes the $r^* \times r^*$ identity matrix, and $\mathbf{D}^* = \text{diag} \{d_1^*, \dots, d_{r^*}^*\}$ is an $r^* \times r^*$ diagonal matrix of the singular values. Here both $\mathbf{U}^* = [\mathbf{u}_1^*, \dots, \mathbf{u}_{r^*}^*] \in \mathbb{R}^{p \times r^*}$ and $\mathbf{V}^* = [\mathbf{v}_1^*, \dots, \mathbf{v}_{r^*}^*] \in \mathbb{R}^{q \times r^*}$ are assumed to be sparse, which we refer to as the “*co-sparsity*” structure. There are several prominent features in this co-sparse and low-rank formulation: (I) $(1/\sqrt{n})\mathbf{X}\mathbf{U}^*$ gives r^* latent predictors/factors, i.e., $(1/\sqrt{n})\mathbf{X}\mathbf{u}_k^*$, $k = 1, \dots, r^*$, each of which is constructed as a linear combination of a subset of the original predictors (factor-specific predictor selection); (II) each latent factor is allowed to be related to a subset of the response variables as each $\mathbf{v}_k^*$ is also sparse (factor-specific response selection); (III) the sample latent factors are uncorrelated with each other, due to the orthogonality of $(1/\sqrt{n})\mathbf{X}\mathbf{U}^*$ and the parameter identifiability is ensured with the orthogonality of $\mathbf{V}^*$; (IV) the singular values d_k^* directly indicate the strengths and importance of the associations between $\mathbf{Y}$ and the latent factors. In recent years, although several works considered sparse and low-rank models, but neither of them could achieve (I)-(IV) together. In Chen *et al.* (2012), the coefficient matrix $\mathbf{C}^*$, rather than the regression component $\mathbf{X}\mathbf{C}^*$, was assumed to admit a sparse SVD, so their method failed to achieve (III) and (IV). In Chen and Huang (2012), Bunea *et al.* (2012) and Ma and Sun (2014), all the latent factors were constructed from the same subset of predictors and were related to all the responses, so their methods did not fully achieve (I) and did not consider (II). In Chen *et al.* (2013), the decomposition of the regression components $\mathbf{X}\mathbf{C}^*$ were considered mainly for constructing a class of computationally efficient low-rank estimators, but they did not pursue its potential sparsity. So far, how to efficiently recover the model structure in (4) remains a challenging problem, largely due to the simultaneous presence of orthogonality and co-sparsity requirements.

To tackle this problem, we explore the underlying data generating mechanism targeted by the sparse SVD structure in (4), which enables us to reformulate the problem as a *co-sparse factor regression* and develop a novel sequential extraction procedure. At each sequential step, the estimation problem nicely reduces to a sparse multivariate regression with a unit-rank constraint, in which the orthogonality constraint is not needed at all. Interestingly, each estimated sparse and unit-rank coefficient matrix automatically leads to co-sparsity in its two singular vectors. As such, each step extracts a new latent factor as a sparse linear combination of the original predictors, which aims to best predict the responses after controlling for the effects of the previously extracted factors. The proposed method completely bypasses the orthogonality constraints in the recovery of the co-sparse and low-rank structure, which, in contrast, would have been inevitable in any joint estimation approach due to the need of parameter identifiability. Most recently, a similar sparse and low-rank model was considered in Taha Bahadori *et al.* (2016), in which a two-step estimation procedure was used, i.e., solve sparse generalized eigenvalue problems to obtain initial estimates, and then apply thresholding for refining the co-sparsity pattern. Comparing to their method, our approach achieves factor extraction and co-sparsity recovery together, allowing the two tasks to inform and reinforce each other. Our regression formulation also allows convenient handling of incomplete response data, which is bound to occur in big data problems.

We propose the co-sparse factor regression model in Section 2. A sequential extraction procedure for model estimation is presented in Section 3. We then develop a co-sparse unit-rank estimation algorithm in Section 4 with convergence guarantee. In Section 5, the sequentially extracted estimators are shown to enjoy the oracle properties asymptotically, and a non-asymptotic error bound reveals interesting finite-sample behaviors of the estimators. Simulation studies in Section 6 and an application in Section 7 further showcase the effectiveness of the proposed approach. Some concluding remarks are provided in Section 8.

2 Co-Sparse Factor Regression Model

To motivate, we start with the familiar factor analysis (FA) (Anderson and Rubin, 1956). For a continuous variable $y \in \mathbb{R}^q$ the model can be expressed as $y = \tilde{\mathbf{V}}\mathbf{z} + \mathbf{e}$ where $\mathbf{z} \in \mathbb{R}^{r^*}$ consists of a set of unobserved latent variables, $\tilde{\mathbf{V}} \in \mathbb{R}^{q \times r^*}$ is an unknown loading matrix, and $\mathbf{e} \in \mathbb{R}^q$ is a random error vector of zero mean. The r^* latent factors satisfy $E(\mathbf{z}) = \mathbf{0}$ and $\text{cov}(\mathbf{z}) = \mathbf{I}_{r^*}$, the error satisfies $E(\mathbf{e}) = \mathbf{0}$ and $\text{cov}(\mathbf{e}) = \mathbf{\Sigma}$, usually a diagonal matrix, and $\mathbf{e}$ and $\mathbf{z}$ are independent. Here we assume $E(\mathbf{y}) = \mathbf{0}$, so there is no intercept. It is common to parameterize $\tilde{\mathbf{V}} = \mathbf{V}^*\mathbf{D}^*$ such that $\mathbf{V}^{*T}\mathbf{V}^* = \mathbf{I}_{r^*}$ and $\mathbf{D}^* = \text{diag}\{d_1^*, \dots, d_{r^*}^*\}$.

Now consider the RRR model in (1). Write the coefficient matrix $\mathbf{C}^*$ as $\mathbf{C}^* = \mathbf{U}^*\tilde{\mathbf{V}}^T$, for some $\mathbf{U}^* \in \mathbb{R}^{p \times r^*}$ and $\tilde{\mathbf{V}} \in \mathbb{R}^{q \times r^*}$. Let $\mathbf{x} \in \mathbb{R}^p$ be the multivariate predictor variable, with $E(\mathbf{x}) = \mathbf{0}$ and $\text{cov}(\mathbf{x}) = \mathbf{\Gamma}$. It is immediately clear that RRR can be regarded as a supervised factor analysis in which the latent factors are some linear projections of $\mathbf{x}$, i.e.,

$$\mathbf{y} = \mathbf{C}^{*T}\mathbf{x} + \mathbf{e} = \tilde{\mathbf{V}}(\mathbf{U}^{*T}\mathbf{x}) + \mathbf{e}, \quad (5)$$

where $\tilde{\mathbf{V}} = \mathbf{V}^*\mathbf{D}^*$ with $\mathbf{V}^{*T}\mathbf{V}^* = \mathbf{I}$ and $\text{cov}(\mathbf{U}^{*T}\mathbf{x}) = \mathbf{U}^{*T}\mathbf{\Gamma}\mathbf{U}^* = \mathbf{I}_{r^*}$. With n independent observations, (5) then leads to the following sample model,

$$\begin{aligned} \mathbf{Y} &= \mathbf{X}\mathbf{U}^*\mathbf{D}^*\mathbf{V}^{*T} + \mathbf{E} = \sum_{k=1}^{r^*} d_k^*(\mathbf{X}\mathbf{u}_k^*)\mathbf{v}_k^{*T} + \mathbf{E} = \sum_{k=1}^{r^*} \mathbf{X}\mathbf{C}_k^* + \mathbf{E}, \text{ s.t. } \mathbf{U}^{*T}\mathbf{\Gamma}\mathbf{U}^* = \mathbf{I}_{r^*}, \mathbf{V}^{*T}\mathbf{V}^* \\ &= \mathbf{I}_{r^*}. \end{aligned}$$

(6)

where

$$\mathbf{C}^* = \mathbf{U}^*\mathbf{D}^*\mathbf{V}^{*T}, \quad \mathbf{U}^* = [\mathbf{u}_1^*, \dots, \mathbf{u}_{r^*}^*] \in \mathbb{R}^{p \times r^*}, \quad \mathbf{V}^* = [\mathbf{v}_1^*, \dots, \mathbf{v}_{r^*}^*] \in \mathbb{R}^{q \times r^*}, \quad \mathbf{D}^* = \text{diag}\{d_1^*, \dots, d_{r^*}^*\},$$

$\mathbf{C}_k^* = d_k^*\mathbf{u}_k^*\mathbf{v}_k^{*T}$. The d_k^* s are assumed to be distinct and are placed in a descending order, i.e.,

$d_1^* > \dots > d_{r^*}^* > 0$. By further assuming that both $\mathbf{U}^*$ and $\mathbf{V}^*$ in (6) are sparse, we arrive at our proposed *co-sparse factor regression* model (SFAR).

It is now clear that the decomposition in (6) can be viewed as the population version of the sample decomposition in (4). Consequently, the above SFAR model possesses several prominent features, e.g., latent model interpretation, predictor selection, response selection, etc, as discussed in Section 1. In particular, the model reveals that there are r^* distinct uncorrelated latent factors of descending importance relating the responses to the predictors. In the k th latent association, $k = 1, \dots, r^*$, the elements in $\mathbf{u}_k^*$ give the weights for constructing the factor, the elements in $\mathbf{v}_k^*$ provide the relative effects of the k th factor on the responses, and d_k^* indicates the importance of the k th factor. Due to the co-sparsity on $\mathbf{u}_k^*$ and $\mathbf{v}_k^*$, each latent factor may be constructed from only a subset of the predictors and may only influence a subset of the responses.

At the first glance, it is convenient to conduct model estimation by solving

$$\min_{\mathbf{U}, \mathbf{D}, \mathbf{V}} \left\{ \frac{1}{2} \left\| \mathbf{Y} - \mathbf{XUDV}^T \right\|_F^2 + \lambda_1 \rho_1(\mathbf{U}) + \lambda_2 \rho_2(\mathbf{V}) \right\}, \quad \text{s.t. } \mathbf{U}^T \left(\frac{\mathbf{X}^T \mathbf{X}}{n} \right) \mathbf{U} = \mathbf{I}, \mathbf{V}^T \mathbf{V} = \mathbf{I}, \quad (7)$$

where $\mathbf{U} \in \mathbb{R}^{p \times r}$, $\mathbf{V} \in \mathbb{R}^{q \times r}$ and $\mathbf{D} = \text{diag}\{d_1, \dots, d_r\}$, for some $1 \leq r \leq \min(r_x, q)$. Here ρ_1, ρ_2 are some sparsity-inducing penalty functions (Tibshirani, 1996) with λ_1, λ_2 being their tuning parameters. Since $\mathbf{\Gamma}$ is unknown, it is replaced with $\mathbf{X}^T \mathbf{X}/n$. Unfortunately, there are several difficulties with this joint estimation approach. The rank r has to be specified in advance. The method requires that the sample realizations of the latent factors are exactly uncorrelated, which is necessary to ensure parameter identifiability in (7). However, this requirement is too stringent in view of the true model in (6). Moreover, to the best of our knowledge, existing optimization methods for solving orthogonality constrained regularized estimation are mostly heuristic and do not scale well to large problems. The simultaneous presence of the high-dimensional low-rank structure, the co-sparsity regularization and the orthogonality constraints makes the estimation challenging. This motivates us to explore new avenues for conducting model estimation in SFAR.

3 Co-Sparsity Recovery via Sequential Extraction

3.1 Sequential Extraction in Reduced-Rank Estimation

The SFAR model in (6) aims to approximate $\mathbf{Y}$ using a few supervised latent factors, which are uncorrelated and ordered in descending importance based on their predictive power. A promising way is to extract the latent factors in a *sequential fashion*. This idea of sequential extraction is not new, but it was mainly studied in unsupervised learning from machine learning perspectives; see, e.g., Mackey (2009) and Journée et al. (2010).

To lay the foundation of our approach, we first show in Proposition 3.1 that the classical RRR can be performed via a sequential unit-rank extraction procedure.

Proposition 3.1.—*The reduced-rank estimator $\tilde{\mathbf{C}}(r)$, which solves (3) for some $1 \leq r \leq \min(r_x, q)$, can be obtained by sequentially performing unit-rank regression,*

$$\tilde{\mathbf{C}}_k = \underset{\mathbf{C}}{\operatorname{argmin}} \|\mathbf{Y}_k - \mathbf{X}\mathbf{C}\|_F^2, \quad \text{s.t. } \operatorname{rank}(\mathbf{C}) \leq 1, \quad (8)$$

each time with $\mathbf{Y}_k$ being the current residual matrix, i.e.,

$$\mathbf{Y}_1 = \mathbf{Y}, \mathbf{Y}_k = \mathbf{Y} - \mathbf{X} \sum_{h=1}^{k-1} \tilde{\mathbf{C}}_h, k = 2, \dots, r. \text{ That is, } \tilde{\mathbf{C}}(r) = \sum_{k=1}^r \tilde{\mathbf{C}}_k$$

Moreover, $\tilde{\mathbf{C}}_k = \tilde{d}_k \tilde{\mathbf{u}}_k \tilde{\mathbf{v}}_k^T$ such that $\tilde{\mathbf{u}}_k = (1/\tilde{d}_k)(\mathbf{X}^T \mathbf{X})^- \mathbf{X}^T \mathbf{Y}_k$, where $()^-$ denotes the Moore-Penrose inverse, and $\tilde{\mathbf{v}}_k$ is the normalized eigenvector corresponding to the k^{th} largest eigenvalue $\tilde{d}_k^2$ of $(1/n) \mathbf{Y}^T \mathbf{X}(\mathbf{X}^T \mathbf{X})^- \mathbf{X}^T \mathbf{Y}$. Therefore, the estimators satisfy $(1/n) \tilde{\mathbf{U}}^T \mathbf{X}^T \mathbf{X} \tilde{\mathbf{U}} = \mathbf{I}$ and $\tilde{\mathbf{V}}^T \tilde{\mathbf{V}} = \mathbf{I}$, where $\tilde{\mathbf{U}} = [\tilde{\mathbf{u}}_1, \dots, \tilde{\mathbf{u}}_r]$ and $\tilde{\mathbf{V}} = [\tilde{\mathbf{v}}_1, \dots, \tilde{\mathbf{v}}_r]$.

In Proposition 3.1, there is no orthogonality constraint ever imposed. Interestingly, the sequentially extracted solutions automatically correspond to the decomposition in (6) and automatically satisfy the exact orthogonality! Although this “free orthogonality” may no longer hold when sparse regularization on $\mathbf{U}$ and $\mathbf{V}$ is imposed, Proposition 3.1 suggests that it is promising that the co-sparse $\mathbf{U}^*$ and $\mathbf{V}^*$ can be consistently recovered via sequential extraction without explicit orthogonality constraints.

Proposition 3.1 also further reveals that the proposed decomposition in (6) is quite special and plays a key role to ensure the potential success of sequential recovery. To understand this, recall that there are infinite number of ways to decompose $\mathbf{C}^*$ to the form $\mathbf{C}^* = \mathbf{U}_c \mathbf{V}_c^T$ for some $\mathbf{U}_c \in \mathbb{R}^{p \times r^*}$ and $\mathbf{V}_c \in \mathbb{R}^{q \times r^*}$ simply because $\mathbf{C}^* = \mathbf{U}_c \mathbf{V}_c^T = (\mathbf{U}_c \mathbf{Q}^T)(\mathbf{V}_c \mathbf{Q}^{-1})^T$ for any non-singular matrix $\mathbf{Q} \in \mathbb{R}^{r^* \times r^*}$. Among these infinite possibilities, the decomposition in (6) is the special one that can produce a set of uncorrelated latent factors and can be recovered by sequential unit-rank extraction.

3.2 Sequential Extraction in SFAR—Motivated by (8), to extract the first latent factor and its effects on the responses, we formulate the task as the following unit-rank penalized least squares problem,

$$\hat{\mathbf{C}}_1^{(\lambda)} = \underset{\mathbf{C}}{\operatorname{argmin}} \left\{ \frac{1}{2} \|\mathbf{Y} - \mathbf{X}\mathbf{C}\|_F^2 + \rho(\mathbf{C}; \lambda) \right\}, \quad \text{s.t. } \operatorname{rank}(\mathbf{C}) \leq 1, \quad (9)$$

where $\rho(\cdot; \lambda)$ is certain sparsity-inducing penalty term with tuning parameter λ . Comparing to (8), the only difference in (9) is that additional sparsity regularization on $\mathbf{C}$ is introduced. From matrix approximation point of view, this new criterion aims to produce the best unit-rank approximation of $\mathbf{Y}$ in the column space of $\mathbf{X}$ subject to a given level of sparsity in $\mathbf{C}$. As $\hat{\mathbf{C}}_1^{(\lambda)}$ is of unit rank, it can be decomposed as $\hat{\mathbf{C}}_1^{(\lambda)} = \hat{d}_1^{(\lambda)} \hat{\mathbf{u}}_1^{(\lambda)} \hat{\mathbf{v}}_1^{(\lambda)T}$ with,

$\tilde{d}_1^{(\lambda)} \geq 0, \hat{\mathbf{u}}_1^{(\lambda)T} \mathbf{X}^T \mathbf{X} \hat{\mathbf{u}}_1^{(\lambda)} / n = 1$ and $\hat{\mathbf{v}}_1^{(\lambda)T} \hat{\mathbf{v}}_1^{(\lambda)} = 1$. Similar to SVD, this decomposition is unique up to the signs of the vectors as long as $\hat{\mathbf{C}}_1^{(\lambda)}$ is not a zero matrix.

It is important to realize that the sparsity structure in a unit-rank matrix is very special, in that its sparsity can directly translate to the co-sparsity of its pairs of singular vectors! Proposition 3.2 fully characterizes this correspondence. This simple yet interesting observation is the key that enables us to “kill two birds with one stone”: by promoting the entrywise sparsity of the unit-rank coefficient matrix, solving (9) yields the sparse factor coefficients $\hat{\mathbf{u}}_1^{(\lambda)}$ and the sparse factor effects $\hat{\mathbf{v}}_1^{(\lambda)}$ simultaneously.

Proposition 3.2.: Suppose $\mathbf{C} \in \mathbb{R}^{p \times q}$ is of unit rank and $\mathbf{C} \neq \mathbf{0}$. Write

$\mathbf{C} = [\mathbf{c}_{ij}] = [\mathbf{c}_1, \dots, \mathbf{c}_q] = [\tilde{\mathbf{c}}_1^T, \dots, \tilde{\mathbf{c}}_q^T]^T$. Then $\mathbf{C}$ can be decomposed as $\mathbf{C} = \mathbf{d}\mathbf{u}\mathbf{v}^T$, for some $d > 0$, $\mathbf{u} = [u_i] \in \mathbb{R}^p$ and $\mathbf{v} = [v_j] \in \mathbb{R}^q$. Moreover, for any $i = 1, \dots, p, j = 1, \dots, q$, it holds that (a) $c_{ij} = 0 \Leftrightarrow$ Either $\tilde{\mathbf{c}}_i = \mathbf{0}$ or $\mathbf{c}_j = \mathbf{0}$; (b) $u_i = 0 \Leftrightarrow \tilde{\mathbf{c}}_i = \mathbf{0}$; and (c) $v_j = 0 \Leftrightarrow \mathbf{c}_j = \mathbf{0}$.

We use the adaptive elastic net penalty (Zou and Hastie, 2005; Zou and Zhang, 2009),

$$\rho(\mathbf{C}; \lambda) = \rho(\mathbf{C}; \mathbf{W}_1, \lambda, \alpha) = \alpha \lambda \|\mathbf{W}_1 \circ \mathbf{C}\|_1 + (1 - \alpha) \lambda \|\mathbf{C}\|_F^2 = \alpha \lambda \sum_{i=1}^p \sum_{j=1}^q w_{ij1} |c_{ij}| + (1 - \alpha) \lambda \sum_{i=1}^p \sum_{j=1}^q c_{ij}^2. \quad (10)$$

Here $\|\cdot\|_1$ denotes the ℓ_1 norm, the operator “ $\circ$ ” stands for the Hadamard product,

$\mathbf{W}_1 = [w_{ij1}]_{p \times q}$ is a pre-specified weighting matrix, λ is a tuning parameter controlling the overall amount of regularization, and $\alpha \in (0, 1)$ controls the relative weights between the two penalty terms. We set $\mathbf{W}_1 = |\tilde{\mathbf{C}}_1|^{-\gamma}$ such that $w_{ij1} = w_1^{(d)} w_{i1}^{(u)} w_{j1}^{(v)}$, with

$$w_1^{(d)} = |\tilde{d}_1|^{-\gamma}, \mathbf{w}_1^{(u)} = [w_{11}^{(u)}, \dots, w_{p1}^{(u)}]^T = |\tilde{\mathbf{u}}_1|^{-\gamma}, \mathbf{w}_1^{(v)} = [w_{11}^{(v)}, \dots, w_{q1}^{(v)}]^T = |\tilde{\mathbf{v}}_1|^{-\gamma}, \quad (11)$$

where $\tilde{\mathbf{C}}_1 = \tilde{d}_1 \tilde{\mathbf{u}}_1 \tilde{\mathbf{v}}_1^T$ is the first set of unit-rank RRR estimator given in Proposition 3.1, and γ is a non-negative constant with $|\cdot|^{-\gamma}$ componentwisely defined. As suggested by Zou (2006), we set $\gamma = 2$. Since we mainly focus on sparse estimation, we fix α as a constant,

i.e., $\alpha = 0.95$. Comparing to lasso, one advantage of elastic net is that the additional ridge penalty improves the convexity of the problem and can enhance the stability of optimization (see Section 4). For simplicity, we may write $\rho(\mathbf{C}; \mathbf{W}, \lambda, \alpha) = \rho(\mathbf{C}; \mathbf{W}, \lambda)$.

To focus on the sequential extraction procedure, we defer the details of optimization to Section 4. Assuming the solution path of (9) can be fitted, the tuning parameter λ can be chosen based on either cross validation or some information criterion. This yields the parameter estimates $\hat{\mathbf{C}}_1$ or equivalently, $(\hat{d}_1, \hat{\mathbf{u}}_1, \hat{\mathbf{v}}_1)$.

The search of the subsequent latent factor proceeds by replacing $\mathbf{Y}$ as the current residual matrix. Specifically, the estimation of the k th factor, for $k = 2, \dots, r$, becomes

$$\hat{\mathbf{C}}_k^{(\lambda)} = \underset{\mathbf{C}}{\operatorname{argmin}} \left\{ \frac{1}{2} \left\| \mathbf{Y}_k - \mathbf{X}\mathbf{C} \right\|_F^2 + \rho(\mathbf{C}; \mathbf{W}_k, \lambda) \right\}, \text{ s.t. } \operatorname{rank}(\mathbf{C}) \leq 1, \quad (12)$$

where $\mathbf{W}_k$ is similarly constructed as in (11) with $(\tilde{d}_k, \tilde{\mathbf{u}}_k, \tilde{\mathbf{v}}_k)$ from the sequential RRR, and

$$\mathbf{Y}_k = \mathbf{Y} - \sum_{l=1}^{k-1} \mathbf{X}\hat{\mathbf{C}}_l = \mathbf{Y} - \sum_{l=1}^{k-1} \hat{d}_l \mathbf{X}\hat{\mathbf{u}}_l \hat{\mathbf{v}}_l^T, \quad (13)$$

with $\hat{\mathbf{C}}_l = \hat{d}_l \hat{\mathbf{u}}_l \hat{\mathbf{v}}_l^T$, $\hat{d}_l \geq 0$, $\hat{\mathbf{u}}_l^T \mathbf{X}^T \mathbf{X} \hat{\mathbf{u}}_l / n = 1$ and $\hat{\mathbf{v}}_l^T \hat{\mathbf{v}}_l = 1$. Upon defining $\mathbf{Y}_1 = \mathbf{Y}$, (12) also subsumes the first-step problem in (9).

We sequentially perform regularized estimation analysis using (12) to obtain $\hat{\mathbf{C}}_k$ or equivalently $(\hat{d}_k, \hat{\mathbf{u}}_k, \hat{\mathbf{v}}_k)$ for $k = 1, \dots, r$, $r \leq \min(r_x, q)$. The estimators up to rank r for $\mathbf{C}$ are given by $\hat{\mathbf{C}}(k) = \sum_{l=1}^k \hat{\mathbf{C}}_l$, for $k = 1, \dots, r$, which are nested to each other. We refer to the generic problem in (12) together with the tuning process as *co-sparse unit-rank estimation* (CURE), i.e., $\text{CURE}(\mathbf{C}; \mathbf{Y}_k, \mathbf{X}, \mathbf{W}_k)$. Algorithm 1 summarizes the proposed computation procedure, i.e., *sequential factor extraction via co-sparse unit-rank estimation* (SeCURE).

4 Computation

4.1 Co-Sparse Unit-Rank Estimation Algorithm—For simplicity, we drop the index k and write the generic form of the problem in (12) as

$$\min_{\mathbf{C}} \left\{ \frac{1}{2} \left\| \mathbf{Y} - \mathbf{X}\mathbf{C} \right\|_F^2 + \rho(\mathbf{C}; \mathbf{W}, \lambda) \right\}, \text{ s.t. } \operatorname{rank}(\mathbf{C}) \leq 1, \quad (14)$$

where $\rho(\mathbf{C}; \mathbf{W}, \lambda)$ is the adaptive elastic net penalty defined in (10). When $\lambda = 0$, the problem reduces to (8), which admits explicit solution as given in Proposition 3.1.

For the general case $\lambda > 0$, we reformulate (14) by expressing $\mathbf{C} = d\mathbf{u}\mathbf{v}^T$, with $d \geq 0$, $\mathbf{u}^T \mathbf{X}^T \mathbf{X} \mathbf{u} / n = 1$ and $\mathbf{v}^T \mathbf{v} = 1$, corresponding to (6). Then (14) is equivalent to

$$\min_{(d, \mathbf{u}, \mathbf{v})} \left\{ Q(d, \mathbf{u}, \mathbf{v}; \lambda) = \frac{1}{2} \|\mathbf{Y} - d\mathbf{X}\mathbf{u}\mathbf{v}^T\|_F^2 + \rho(d\mathbf{u}\mathbf{v}^T; \mathbf{W}, \lambda) \right\}, \text{ s. t. } d \geq 0, \mathbf{u}^T \mathbf{X}^T \mathbf{X} \mathbf{u} = n, \mathbf{v}^T \mathbf{v} = 1, \quad (15)$$

where $\rho(d\mathbf{u}\mathbf{v}^T; \mathbf{W}, \lambda) = \alpha\lambda \sum_{i=1}^p \sum_{j=1}^p w_{ij} |du_i v_j| + (1-\alpha)\lambda \sum_{i=1}^p \sum_{j=1}^p (du_i v_j)^2$. That is, $(\hat{d}^{(\lambda)}, \hat{\mathbf{u}}^{(\lambda)}, \hat{\mathbf{v}}^{(\lambda)})$ solves (15) if and only if $\hat{\mathbf{C}}^{(\lambda)} = \hat{d}^{(\lambda)} \hat{\mathbf{u}}^{(\lambda)} \hat{\mathbf{v}}^{(\lambda)T}$ solves (14).

We propose to solve (15) by a block coordinate descent algorithm with two overlapping blocks of parameters $(d, \mathbf{u})$ and $(d, \mathbf{v})$, motivated by Chen *et al.* (2012). Consider the update of $(d, \mathbf{u})$ with fixed $\mathbf{v}$ satisfying $\mathbf{v}^T \mathbf{v} = 1$. The relevant constraints are $\mathbf{u}^T \mathbf{X}^T \mathbf{X} \mathbf{u} = n$ and $d \geq 0$. The key is to recognize that the objective function is a function of $(d, \mathbf{u})$ only through their product $\tilde{\mathbf{u}} = d\mathbf{u}$, so both constraints are avoided when optimizing with respect to $\tilde{\mathbf{u}}$. It boils down to solving the following problem with respect to $\tilde{\mathbf{u}}$ (referred to as the **U**-step):

$$\min_{\tilde{\mathbf{u}}} \left\{ Q_{\mathbf{u}}(\tilde{\mathbf{u}}; \mathbf{v}, \lambda) = \frac{1}{2} \|\mathbf{y} - \mathbf{X}^{(u)} \tilde{\mathbf{u}}\|_2^2 + \lambda_1^{(u)} \sum_{i=1}^p w_i |\tilde{u}_i| + \lambda_2^{(u)} \sum_{i=1}^p \tilde{u}_i^2 \right\}, \quad (16)$$

where $\mathbf{y} = \text{vec}(\mathbf{Y})$, $\mathbf{X}^{(u)} = \mathbf{v} \otimes \mathbf{X}$, $\lambda_1^{(u)} = \alpha\lambda w^{(d)} \left(\sum_{j=1}^q w_j^{(v)} |v_j| \right)$, and $\lambda_2^{(u)} = (1-\alpha)\lambda \sum_{j=1}^q v_j^2$. Here $\text{vec}(\cdot)$ is the vectorization operator, and $\otimes$ denotes the Kronecker product. This problem is a convex elastic net regression, which can be efficiently solved. Once we obtain the solution $\tilde{\mathbf{u}}$, we update $d = \|\mathbf{X}\tilde{\mathbf{u}}\|_2 / \sqrt{n}$ and $\mathbf{u} = \tilde{\mathbf{u}}/d$ to satisfy the constraints whenever $\tilde{\mathbf{u}} \neq \mathbf{0}$. Here we remark that $\tilde{\mathbf{u}} = \mathbf{0}$ implies that $\|\mathbf{X}\tilde{\mathbf{u}}\|_2 \neq 0$, because otherwise $Q_{\mathbf{u}}(\mathbf{0}; \mathbf{v}, \lambda) \leq Q_{\mathbf{u}}(\tilde{\mathbf{u}}; \mathbf{v}, \lambda)$ which contradicts with the optimality of $\tilde{\mathbf{u}}$. When $\tilde{\mathbf{u}} = \mathbf{0}$, $\mathbf{u}$ is no longer identifiable; we then update $d = 0$ and terminate the algorithm.

Similarly, for fixed $\mathbf{u}$ satisfying $\mathbf{u}^T \mathbf{X}^T \mathbf{X} \mathbf{u} = n$, the minimization of (15) with respect to $(d, \mathbf{v})$ becomes minimization with respect to $\tilde{\mathbf{v}} = d\mathbf{v}$ (referred to as the **V**-step):

$$\min_{\tilde{\mathbf{v}}} \left\{ Q_{\mathbf{v}}(\tilde{\mathbf{v}}; \mathbf{u}, \lambda) = \frac{1}{2} \|\mathbf{y} - \mathbf{X}^{(v)} \tilde{\mathbf{v}}\|_2^2 + \lambda_1^{(v)} \sum_{j=1}^q w_j |\tilde{v}_j| + \lambda_2^{(v)} \sum_{j=1}^q \tilde{v}_j^2 \right\}, \quad (17)$$

where $\mathbf{X}^{(v)} = \mathbf{I}_q \otimes (\mathbf{X}\mathbf{u})$, $\lambda_1^{(v)} = \alpha\lambda w^{(d)} \left(\sum_{i=1}^P w_i^{(u)} |u_i| \right)$, and $\lambda_2^{(v)} = (1 - \alpha)\lambda \sum_{i=1}^P u_i^2$. Once $\tilde{\mathbf{v}}$ is obtained, we update $d = \|\tilde{\mathbf{v}}\|_2$ and $\mathbf{v} = \tilde{\mathbf{v}}/d$ whenever $\tilde{\mathbf{v}} \neq 0$. Similarly, when $\tilde{\mathbf{v}} = 0$ we update $d = 0$ and terminate the algorithm.

Our CURE algorithm is summarized in Algorithm 2. Without loss of generality, we start from fixed $\mathbf{v} = \mathbf{v}_\lambda^0$. It generates a sequence $\{(\tilde{d}_\lambda^s, d_\lambda^s, \mathbf{u}_\lambda^s, \mathbf{v}_\lambda^s)\}_{s \in \mathbb{N}}$ until reaching convergence.

4.2 Convergence Analysis and Generation of Solution Paths—Our CURE algorithm is closely connected to the Alternating Convex Search (ACS) method for biconvex optimization (Gorski *et al.*, 2007; Luenberger and Ye, 2008). The main difference between CURE and a standard ACS is that the two blocks of parameters in CURE, $(d, \mathbf{u})$ and $(d, \mathbf{v})$, are overlapping to each other. We are able to establish the convergence behavior of CURE to some coordinatewise minimum point, as summarized in Theorem 4.1.

Theorem 4.1.: Consider the optimization problem in (15) with $\lambda > 0$ and $0 < \alpha < 1$. Assume the weights $\{w_{ij}\}$ and the data $(\mathbf{Y}, \mathbf{X})$ are finite, and the initial value $\mathbf{v}_\lambda^0$ satisfies

$\arg\min_{\tilde{\mathbf{u}}} Q_{\tilde{\mathbf{u}}}(\tilde{\mathbf{u}}, \mathbf{v}_\lambda^0, \lambda) \neq 0$ i.e., $\tilde{d}_\lambda^1 \neq 0$. Then the sequence $\{(\tilde{d}_\lambda^s, d_\lambda^s, \mathbf{u}_\lambda^s, \mathbf{v}_\lambda^s)\}_{s \in \mathbb{N}}$ generated by the CURE algorithm is uniformly bounded and has at least one accumulation point. Moreover, all accumulation points are coordinatewise minimum points and have the same objective value, and $Q(\tilde{d}_\lambda^s, \mathbf{u}_\lambda^s, \mathbf{v}_\lambda^s; \lambda)$ converges monotonically to $Q(d^{(\lambda)}, \mathbf{u}^{(\lambda)}, \mathbf{v}^{(\lambda)}; \lambda)$ for some coordinatewise minimum point $(d^{(\lambda)}, \mathbf{u}^{(\lambda)}, \mathbf{v}^{(\lambda)})$.

Due to the nonconvex nature of the problem, Theorem 4.1 does not guarantee any global optimality of the solution. Nevertheless, the established convergence results are adequate to ensure stable computation for practical use (Luenberger and Ye, 2008; Bunea *et al.*, 2012).

In Theorem 4.1, it is required that the initial value $\mathbf{v}_\lambda^0$ does not produce null solution in the first step of the algorithm. The reasons and implications are as follows. One key to establish the convergence results is to ensure that each sub-problem in either (16) or (17) always produces a unique solution in terms of the pair $(d, \mathbf{u})$ or $(d, \mathbf{v})$ along the iterations. Owing to the use of the elastic net penalty, these sub-problems are strictly convex in either $\tilde{\mathbf{u}} = d\mathbf{u}$ or $\tilde{\mathbf{v}} = d\mathbf{v}$ whenever $\lambda > 0$ and $0 < \alpha < 1$; consequently they always produce unique solutions of $d\mathbf{u}$ or $d\mathbf{v}$. However, the complication occurs when $\tilde{\mathbf{u}} = \mathbf{0}$ or $\tilde{\mathbf{v}} = \mathbf{0}$ where the solution in terms of the pair $(d, \mathbf{u})$ or $(d, \mathbf{v})$ are not unique anymore. Therefore, the occurrence of the null solution during iterations needs special care. Nicely, we are able to show that as long as the initial value $\mathbf{v}^0$ does not produce null solution, it is guaranteed that the null solution will not occur later along the iterations. Moreover, suppose $d^{(\lambda)} \neq 0$ where $(d^{(\lambda)}, \mathbf{u}^{(\lambda)}, \mathbf{v}^{(\lambda)})$ is the solution of (15) with tuning parameter λ . Then we show that when solving (15) with a smaller tuning parameter than λ , the null solution can again be avoided by using $\mathbf{v}^{(\lambda)}$ as the initial value. Theorem 4.2 summarizes these results.

Theorem 4.2.: Consider solving (15) with $\lambda > 0$ and $0 < \alpha < 1$ using the CURE algorithm.

If $\arg\min_{\mathbf{u}} Q_{\mathbf{u}}(\tilde{\mathbf{u}}; \mathbf{v}_{\lambda}^0, \lambda) \neq \mathbf{0}$, i.e., $\tilde{d}_{\lambda}^1 \neq 0$ then $d^{(\lambda_2)} \neq 0$ for any $s = 1$. Let $(d^{(\lambda)}, \mathbf{u}^{(\lambda)}, \mathbf{v}^{(\lambda)})$ denote the solution of (15) with tuning parameter λ . If $d^{(\lambda_1)} \neq 0$ then $d^{(\lambda_2)} \neq 0$ for any $\lambda_2 \leq \lambda_1$. Moreover, if $d^{(\lambda_1)} \neq 0$, then setting $\mathbf{v}_{\lambda_2}^0 = \mathbf{v}^{(\lambda_1)}$ ensures that $\tilde{d}_{\lambda_2}^s \neq 0$ and $d_{\lambda_2}^s \neq 0$ for any $s = 1$.

It is now in order to consider the generation of the solution paths using CURE. Theorem 4.2 clearly suggests to fit the model for a sequence of descending λ values with a warm-start strategy, i.e., the estimate from the previous larger λ value is used as the initial value for the next smaller λ value. It then remains to determine the relevant range of λ values, for which it boils down to determine $\lambda_{\max}$, the smallest λ value for which (15) produces the null solution. Nicely, similar to Chen *et al.* (2012), both $\lambda_{\max}$ and the first set of non-zero solution on the paths can be explicitly determined. Denote $\mathbf{Y} = [\mathbf{y}_{(1)}, \dots, \mathbf{y}_{(q)}]$ and $\mathbf{X} = [\mathbf{x}_{(1)}, \dots, \mathbf{x}_{(p)}]$ and let $(i^0, j^0) = \arg\max_{(i, j)} \{\|\mathbf{x}_{(i)}^T \mathbf{y}_{(j)}\| / w_{ij}\}$ assuming no ties. Then, for any $0 < \alpha \leq 1$, we have $\lambda_{\max} = \|\mathbf{x}_{(i^0)}^T \mathbf{y}_{(j^0)}\| / (\alpha w_{i^0 j^0})$, and the initial $\mathbf{v}$ vector, denoted as $\mathbf{v}^0 \in \mathbb{R}^q$ is a standard basis vector with $v_{j^0}^0 = 1$ and $v_j^0 = 0$ for any $j \neq j^0$.

In practice, we fit the model for a sequence of 100 descending λ values equally spaced on the log-scale between $[\lambda_{\max} - \epsilon, \lambda_{\min}]$ to produce a spectrum of co-sparsity patterns. Here ϵ is a small positive number, and $\lambda_{\min}$ is taken as either 0 or a fraction of $\lambda_{\max}$ at which the model has excessive number of non-zero coefficients (we set $\lambda_{\min} = \lambda_{\max} \times 10^{-5}$). With the above strategies, for each fixed λ within the range, we do not observe any early termination of the algorithm due to the production of null solution, and we always observe a unique limit point. The computation of the solution paths is efficient and stable. To save space, all the proofs are provided in the Supplementary Materials.

4.3 Tuning and Rank Selection: For the genetic problem in (14) or (15), once the solution path is obtained, we need to choose the optimal solution along the path. For small-scale problems, λ can be chosen by cross validation (Stone, 1974). Alternatively, various information criteria have been widely used due to their computational efficiency. In all our numerical studies, we use the following hybrid information criterion because of its superior performance in sparse learning,

$$\text{IC}(\lambda) = \begin{cases} \log\{\text{SSE}(\lambda)\} + \{\log(qn)/(nq)\}df(\lambda) & \text{when } p < n; \\ \log\{\text{SSE}(\lambda)\} + \{\log\log(nq)\log(pq)/(nq)\}df(\lambda) & \text{when } p \geq n, \end{cases}$$

where $\text{SSE}(\lambda) = \|\mathbf{Y} - \hat{\mathbf{d}}^{(\lambda)} \mathbf{X} \hat{\mathbf{u}}^{(\lambda)} \hat{\mathbf{v}}^{(\lambda)T}\|_F^2$, and $df(\lambda)$ is the model degrees of freedom, which is estimated by $\hat{df}(\lambda) = \sum_{i=1}^p \mathbf{I}(\hat{u}_i^{(\lambda)} \neq 0) + \sum_{j=1}^q \mathbf{I}(\hat{v}_j^{(\lambda)} \neq 0) - 1$ with $\mathbf{I}(\cdot)$ being the indicator function. The criterion corresponds to the familiar Bayesian Information Criterion (BIC)

when $p < n$ and the generalized information criterion proposed by Fan and Tang (2013) when $p \geq n$. The SeCURE does not require the rank or the number of factors to be specified in advance. The sequential unit-rank extraction proceeds until $\hat{d}_k = 0$ for some k .

4.4 Extensions to Incomplete Data and Exact Orthogonality: Here we explore some useful extensions of SeCURE. Let $\mathbf{H} = \{(i, j); y_{ij} \text{ is observed}, i = 1, \dots, n, j = 1, \dots, q\}$ be an index set collecting the indices of all the observed entries in $\mathbf{Y}$. Let $\tilde{\mathbf{Y}} = P_{\mathcal{H}}(\mathbf{Y})$ be the projection of $\mathbf{Y}$ onto $\mathbf{H}$ such that $\tilde{y}_{ij} = y_{ij}$ for any $(i, j) \in \mathbf{H}$ and $\tilde{y}_{ij} = 0$ We extend SeCURE by modifying (12) as

$$\min_{\mathbf{C}} \left\{ \frac{1}{2} \left\| P_{\mathcal{H}}(\mathbf{Y}_k) - P_{\mathcal{H}}(\mathbf{XC}) \right\|_F^2 + \rho(\mathbf{C}; \mathbf{W}_k, \lambda) \right\}, \quad \text{s.t. } \text{rank}(\mathbf{C}) \leq 1, \mathbf{A}_k \mathbf{C} = \mathbf{0}, \mathbf{B}_k \mathbf{C}^T = \mathbf{0}.$$

where $\mathbf{A}_1 = \mathbf{0}$, $\mathbf{B}_1 = \mathbf{0}$ and for $k \geq 2$, $\mathbf{A}_k$ and $\mathbf{B}_k$ are formed from the estimated factors as $\mathbf{A}_k = \hat{\mathbf{U}}_{k-1}^T \mathbf{X}^T \mathbf{X}$, $\mathbf{B}_k = \hat{\mathbf{V}}_{k-1}^T$, with $\hat{\mathbf{U}}_{k-1} = [\hat{\mathbf{u}}_1, \dots, \hat{\mathbf{u}}_{k-1}]$ and $\hat{\mathbf{V}}_{k-1} = [\hat{\mathbf{v}}_1, \dots, \hat{\mathbf{v}}_{k-1}]$ and all the other terms are defined the same as before. Only the observed entries contribute to the loss function, which follows the same spirit as matrix completion (Candès and Recht, 2009). The linear constraints enforce the orthogonality among the estimated factors.

Accordingly, the generic CURE problem in (15) can be modified to the form

$$\min_{d, \mathbf{u}, \mathbf{v}} \left\{ \frac{1}{2} \left\| P_H(\mathbf{Y}) - P_H(d \mathbf{X} \mathbf{u} \mathbf{v}^T) \right\|_F^2 + \rho(d \mathbf{u} \mathbf{v}^T; \mathbf{W}, \lambda, \alpha) \right\},$$

$$\text{s.t. } d \geq 0, \mathbf{u}^T \mathbf{X}^T \mathbf{X} \mathbf{u} = n, \mathbf{v}^T \mathbf{v} = 1; \mathbf{A} \mathbf{u} = \mathbf{0}, \mathbf{B} \mathbf{v} = \mathbf{0}.$$

Nicely, the CURE optimization can still be conducted via the block coordinate descent algorithm. The problem in either the $\mathbf{U}$ -step or the $\mathbf{V}$ -step becomes a linear-constrained adaptive elastic net regression, which remains convex and admits efficient computation (James et al., 2013; Lin et al., 2014). In the Supplementary Materials, we provide our implemented algorithm based on the ideas of coordinate descent and Lagrangian multipliers.

5 Theoretical Properties

5.1 Asymptotic Results: In our asymptotic analysis, we consider the model in (5) and (6) with fixed p, q . The following assumptions on the design, the random error, and the singular values are used.

A1. $\|(1/n) \mathbf{X}^T \mathbf{X} - \mathbf{\Gamma}\|_F = O_p(1/\sqrt{n})$ where $\mathbf{\Gamma}$ is a fixed, positive definite matrix.

A2. The error vectors $\mathbf{e}_i, i = 1, \dots, n$, are independent and identically distributed (i.i.d.) normal random variables with $E(\mathbf{e}_i) = \mathbf{0}$ and $\text{cov}(\mathbf{e}_i) = \mathbf{\Sigma}$, a positive definite matrix.

A3. $d_1^* > \dots > d_{r^*}^* > 0$ First, Proposition 5.1 shows that the sequential RRR estimators are consistent. This justifies our approach of using them for constructing the adaptive weights.

Proposition 5.1.: Consider the sequential reduced-rank estimators $(\tilde{d}_k, \tilde{\mathbf{u}}_k, \tilde{\mathbf{v}}_k), k = 1, \dots, r$ presented in Proposition 3.1. Under conditions A1-A3, as $n \rightarrow \infty$

i. $\sqrt{n}(\tilde{d}_k - d_k^*) = O_p(1), \sqrt{n}(\tilde{\mathbf{u}}_k - \mathbf{u}_k^*) = O_p(1)$ and $\sqrt{n}(\tilde{\mathbf{v}}_k - \mathbf{v}_k^*) = O_p(1)$ for $k = 1, \dots, r^*$.

ii. $\sqrt{n}\tilde{d}_k = O_p(1)$, for $k = r^* + 1, \dots, r$.

Recall that we have parameterized the true coefficient matrix $\mathbf{C}^*$ in (6); we have also written $\mathbf{C}_k^* = d_k^* \mathbf{u}_k^* \mathbf{v}_k^{*T}$ and $\mathbf{C}^* = \sum_{k=1}^{r^*} \mathbf{C}_k^*$. To simplify the problem, we consider a reparameterization of $\mathbf{C}^*$, by letting each singular value d_k^* be absorbed to each pair of vectors $(\mathbf{u}_k^*, \mathbf{v}_k^*)$

Specifically, for each $\mathbf{v}_k^*, k = 1, \dots, r^*$, there must exist at least one entry that is nonzero, i.e.,

$v_{\ell_k}^* \neq 0$ for some $1 \leq \ell_k \leq q$. Accordingly, we reparameterize $\mathbf{C}_k^*$ as

$\mathbf{C}_k^* = \mathbf{u}_k^* \mathbf{v}_k^{*T}$, with $v_{\ell_k}^* = 1$. Here, with some abuse of notation, $(\mathbf{u}_k^*, \mathbf{v}_k^*)$ have been redefined, and they are re-scaled versions of their original counterparts, i.e., the new pair now satisfies $(\mathbf{u}_k^{*T} \Gamma \mathbf{u}_k^*)(\mathbf{v}_k^{*T} \mathbf{v}_k^*) = d_k^*, k = 1, \dots, r^*$. Consequently, we parameterize $\mathbf{C}^*$ as

$$\mathbf{C}^* = \mathbf{U}^* \mathbf{V}^{*T}, \text{ s.t. } \mathbf{U}^{*T} \Gamma \mathbf{U}^* \text{ and } \mathbf{V}^{*T} \mathbf{V}^* \text{ are diagonal; } v_{\ell_k}^* = 1, k = 1, \dots, r^*. \quad (18)$$

Now all the elements in $\mathbf{U}^*$ and $\mathbf{V}^*$ are uniquely defined and hence identifiable. For $k > r^*$, it is obvious that $C_k^* = 0$ and its corresponding singular vectors become unidentifiable; as such, we can set $\mathbf{u}_k^* = 0$ but still choose $\mathbf{v}_k^*$ to be a nonzero vector with some $v_{\ell_k}^* = 1$.

For the sequential reduced-rank estimators, let $\tilde{d}_k$ be absorbed to $(\tilde{\mathbf{u}}_k, \tilde{\mathbf{v}}_k)$ similar to (18), such that $\tilde{\mathbf{C}}_k = \tilde{d}_k \tilde{\mathbf{u}}_k \tilde{\mathbf{v}}_k^T$ in Proposition 3.1 now becomes $\tilde{\mathbf{C}}_k = \tilde{\mathbf{u}}_k \tilde{\mathbf{v}}_k^T$ with $\tilde{v}_{\ell_k} = 1$. With some abuse of notation, here we have still used $(\tilde{\mathbf{u}}_k, \tilde{\mathbf{v}}_k)$ to denote the rescaled versions. Then, as $n \rightarrow \infty$ the consistency results in Proposition 5.1 are alternatively expressed as

i. $\sqrt{n}(\tilde{\mathbf{u}}_k - \mathbf{u}_k^*) = O_p(1)$ and $\sqrt{n}(\tilde{\mathbf{v}}_k - \mathbf{v}_k^*) = O_p(1)$, for $k = 1, \dots, r^*$.

ii. $\sqrt{n}\tilde{d}_k = O_p(1)$, where $\tilde{d}_k = (1/n) \left(\tilde{\mathbf{u}}_k^T \mathbf{X}^T \mathbf{X} \tilde{\mathbf{u}}_k \right) \left(\tilde{\mathbf{v}}_k^T \tilde{\mathbf{v}}_k \right)$, for $k = r^* + 1, \dots, r$.

Now, consider the proposed SeCURE approach. Under the new parameterization in (18), the objective function in the k th step of SeCURE can be written as

$$Q_k^{(n)}(\mathbf{u}, \mathbf{v}) = \frac{1}{2} \left\| \mathbf{Y}_k - \mathbf{X} \mathbf{u} \mathbf{v}^T \right\|_F^2 + \alpha \lambda_k^{(n)} \sum_{i=1}^p \sum_{j=1}^q w_{ijk} |u_i v_j| + (1 - \alpha) \lambda_k^{(n)} \sum_{i=1}^p \sum_{j=1}^q (u_i v_j)^2,$$

where $\mathbf{u} \in \mathbb{R}^p, \mathbf{v} \in \mathbb{R}^q$ with $v_{\ell_k} = 1$, and $\mathbf{Y}_1 = \mathbf{Y}, \mathbf{Y}_k = \mathbf{Y} - \sum_{1 \leq h \leq k-1} \mathbf{X} \hat{\mathbf{u}}_h \hat{\mathbf{v}}_h^T$, where $(\hat{\mathbf{u}}_h, \hat{\mathbf{v}}_h) = \arg\min Q_h^{(n)}(\mathbf{u}, \mathbf{v})$. Here $w_{ijk} = w_{ik} w_{jk}$ where $w_{ik} = |\tilde{u}_{ik}|^{-\gamma}$ and $w_{jk} = |\tilde{v}_{jk}|^{-\gamma}$ with some fixed $\gamma > 0$. Here $\alpha \in (0, 1)$ is considered as a fixed constant.

We show that our SeCURE estimators enjoy consistency, asymptotic normality, variable selection consistency and rank selection consistency. Our main results are summarized in Theorems 5.2–5.4 below. The following assumption on the choice of $\lambda_k^{(n)}$ is required.

$$\text{A4. } \frac{\lambda_k^{(n)}}{\sqrt{n}} \rightarrow 0 \text{ and } \frac{\lambda_k^{(n)}}{\sqrt{n}} n^{\gamma/2} \rightarrow \infty \text{ as } n \rightarrow \infty, \text{ for } k = 1, \dots, r^*. \text{ Also, } \frac{\lambda_k^{(n)}}{\sqrt{n}} n^{\gamma/2} \rightarrow \infty \text{ as } n \rightarrow \infty, \\ \text{for } k = r^* + 1, \dots, r.$$

Theorem 5.2.: (Existence of Local Minimum). Suppose A1-A3 are satisfied, and $\lambda_k^{(n)}/\sqrt{n} \rightarrow \lambda_k \geq 0$ as $n \rightarrow \infty$. Then there is a local minimizer $(\hat{\mathbf{u}}_k, \hat{\mathbf{v}}_k)$ of $Q_k^{(n)}(\mathbf{u}, \mathbf{v})$ such that

$$\text{i. } \|\hat{\mathbf{u}}_k - \mathbf{u}_k^*\| = O_p(n^{-1/2}) \text{ and } \|\hat{\mathbf{v}}_k - \mathbf{v}_k^*\| = O_p(n^{-1/2}), \text{ for } k = 1, \dots, r^* \\ \text{ii. } |\hat{d}_k| = O_p(n^{-1/2}) \text{ where } \hat{d}_k = (1/n)(\hat{\mathbf{u}}_k^T \mathbf{X}^T \mathbf{X} \hat{\mathbf{u}}_k)(\hat{\mathbf{v}}_k^T \hat{\mathbf{v}}_k), \text{ for } k = r^* + 1, \dots, r.$$

Theorem 5.3.: (Asymptotic Normality) Suppose A1-A4 are satisfied. Let $\mathcal{A}_k = \{i; u_{ik}^* \neq 0\}$ and $\mathcal{B}_k = \{j; v_{jk}^* \neq 0, j \neq \ell_k\}$, for $k = 1, \dots, r^*$; for $k = 1, \dots, r^*$; denote their complements as $\mathcal{A}_k^c$ and $\mathcal{B}_k^c$. Accordingly, define the subvectors $\mathbf{u}_{k\mathcal{A}_k}^*, \hat{\mathbf{u}}_{k\mathcal{A}_k}, \mathbf{v}_{k\mathcal{B}_k}^*, \hat{\mathbf{v}}_{k\mathcal{B}_k}$, etc. Then

- i. $\sqrt{n}(\hat{\mathbf{u}}_{k\mathcal{A}_k} - \mathbf{u}_{k\mathcal{A}_k}^*)$ and $\sqrt{n}(\hat{\mathbf{v}}_{k\mathcal{B}_k} - \mathbf{v}_{k\mathcal{B}_k}^*)$ are jointly asymptotically normally distributed with zero mean.
- ii. $\sqrt{n}(\hat{\mathbf{u}}_{k\mathcal{A}_k} - \mathbf{u}_{k\mathcal{A}_k}^*) \rightarrow_d 0$ and $\sqrt{n}(\hat{\mathbf{v}}_{k\mathcal{B}_k} - \mathbf{v}_{k\mathcal{B}_k}^*) \rightarrow_d 0$ as $n \rightarrow \infty$.

Theorem 5.4.: (Selection Consistency and Rank Consistency) Suppose A1-A4 are satisfied. Let

$$\mathcal{A} = \{(i, k): u_{ik}^* \neq 0, k = 1, \dots, r^*\} \text{ and } \mathcal{B} = \{(j, k): v_{jk}^* \neq 0, (j, k) \neq (\ell_k, k), k = 1, \dots, r^*\}, \text{ and let } \\ \mathcal{A}^{(n)} = \{(i, k): \hat{u}_{ik} \neq 0, k = 1, \dots, r^*\} \text{ and } \mathcal{B}^{(n)} = \{(j, k): \hat{v}_{jk} \neq 0, (j, k) \neq (\ell_k, k), k = 1, \dots, r^*\}. \text{ Let } \\ \hat{d}_k = (1/n)(\hat{\mathbf{u}}_k^T \mathbf{X}^T \mathbf{X} \hat{\mathbf{u}}_k)(\hat{\mathbf{v}}_k^T \hat{\mathbf{v}}_k)$$

be the k th estimated singular value, for $k = r^* + 1, \dots, r$. Then

- i. $P(\mathcal{A}^{(n)} = \mathcal{A}) \rightarrow 1$ and $P(\mathcal{B}^{(n)} = \mathcal{B}) \rightarrow 1$ as $n \rightarrow \infty$
- ii. $P(\hat{d}_k = 0) \rightarrow 1$ as $n \rightarrow \infty$, for $k = r^* + 1, \dots, r$.

5.2 A Non-Asymptotic Error Bound: We briefly discuss the non-asymptotic behaviors of the SeCURE estimators. The asymptotic convergence of $\mathbf{X}^T \mathbf{X}/n$ in **A1** is no longer assumed. Consequently, we consider the model

$$\mathbf{Y} = \mathbf{X}\mathbf{C}^* + \mathbf{E} = \mathbf{X}\mathbf{U}^* \mathbf{D}^* \mathbf{V}^{*T} + \mathbf{E}, \quad \text{s.t. } \mathbf{U}^{*T} \mathbf{X}^T \mathbf{X} \mathbf{U}^*/n = \mathbf{I}_{r^*}, \mathbf{V}^{*T} \mathbf{V}^* = \mathbf{I}_{r^*}. \quad (19)$$

To focus on the fundamentals, we study SeCURE estimation using the non-adaptive elastic net penalty, i.e., $w_{ijk} = 1$. Let $\mathbf{C}^* = \sum_{k=1}^{r^*} \mathbf{C}_k^*$, with $\mathbf{C}_k^* = d_k^* \mathbf{u}_k^* \mathbf{v}_k^{*T}$, and denote the corresponding SeCURE estimators by solving (12) as $\hat{\mathbf{C}}_k$. Denote $\lambda_1 = \lambda \alpha$ and $\lambda_2 = \lambda(1 - \alpha)$. Let $\mathbf{P}_\mathbf{X}$ denote the projection matrix unto the column space of $\mathbf{X}$. Let $\lambda_{\min}(\mathbf{H})$ and $\lambda_{\max}(\mathbf{H})$ denote the minimum and maximum eigenvalues of a square matrix $\mathbf{H}$. Let $d_k(\mathbf{H})$ denote the k th largest singular value of a matrix $\mathbf{H}$. We consider the following conditions.

B1. $b \leq \lambda_{\min}(\mathbf{X}^T \mathbf{X}/n) \leq \lambda_{\max}(\mathbf{X}^T \mathbf{X}/n) \leq B$, where b and B are two positive constants.

B2. The error matrix $\mathbf{E}$ has independent $N(0, \sigma^2)$ entries.

Our main result is presented in Theorem 5.5.

Theorem 5.5. Suppose **B1** and **A3** are satisfied. For $k = 1, \dots, r^*$,

$$\begin{aligned} \|\hat{\mathbf{C}}_k - \mathbf{C}_k^*\|_F &\leq \frac{\lambda_1 \sqrt{pq} + 2\sqrt{2} d_1(\mathbf{P}_\mathbf{X} \mathbf{E}) \sqrt{\lambda_{\max}} + 2\sqrt{2} \sqrt{nd_{k+1}^*} \sqrt{\lambda_{\max}} + 2\sqrt{2} \lambda_2 \|\mathbf{C}_k^*\|_F}{\lambda_{\min} + \lambda_2} \\ &+ \frac{2\sqrt{2} \lambda_{\max}}{\lambda_{\min} + \lambda_2} \left(\sum_{h=1}^{k-1} \|\hat{\mathbf{C}}_h - \mathbf{C}_h^*\|_F \right). \end{aligned} \quad (20)$$

This bound reveals some interesting properties of SeCURE estimators. First, it shows that $\sqrt{nd_{k+1}^*}$ plays similar role as the stochastic term $d_1(\mathbf{P}_\mathbf{X} \mathbf{E})$ in the estimation of $\mathbf{C}_k^*$. The former term measures the size of the left-over signal in the model, while the latter term measures the size of the stochastic error; they both appear in the error bound because they both act as “errors” in the estimation of the k th factor. Moreover, the second term in the bound shows the phenomenon of “noise accumulation” that arises naturally due to the sequential procedure itself. This is caused by the fact in each step the ideal response matrix to use, i.e., $\mathbf{Y} - \sum_{h=1}^{k-1} \mathbf{X} \mathbf{C}_h^*$, is clearly not available, which has to be estimated by $\mathbf{Y} - \sum_{h=1}^{k-1} \mathbf{X} \hat{\mathbf{C}}_h$. The stochastic term $d_1(\mathbf{P}_\mathbf{X} \mathbf{E})$ can be treated similarly as in Bunea et al. (2011). Theorem 5.5 can then be used to establish the consistency of SeCURE with diverging model dimensions. To save space, more details and all the proofs are provided in the Supplementary Materials.

6 Simulation

6.1 Setups: We compare estimation, prediction and co-sparsity recovery performance of ordinary least squares (OLS), reduced-rank regression (RRR), sparse reduced-rank

regression using adaptive group lasso (SRRR) by Chen and Huang (2012), reduced-rank regression with sparse singular value decomposition using adaptive lasso (RSSVD) by Chen et al. (2012), and several versions of our proposed SeCURE approach. We consider both adaptive elastic net and adaptive lasso penalties, and the corresponding methods are denoted as SeCURE(AE) and SeCURE(AL), respectively. We also show that SeCURE can optionally enforce the exact orthogonality; the corresponding methods are denoted as SeCURE(AE*) and SeCURE(AL*).

The true rank of $\mathbf{C}^* \in \mathbb{R}^{p \times q}$ is set as $r^* = 3$, and we set $d_1^* = 20, d_2^* = 15, d_3^* = 10$. Model I is a low dimensional example with $p = 50, q = 25$ and $n = 400$. The $\mathbf{u}_k^*$ is generated as

$\mathbf{u}_k^* = \tilde{\mathbf{u}}_k / \|\tilde{\mathbf{u}}_k\|$, where

$$\tilde{\mathbf{u}}_1 = [\text{unif}(\mathcal{A}_u, 8), \text{rep}(0, 42)]^T, \tilde{\mathbf{u}}_2 = [\text{rep}(0, 5), \text{unif}(\mathcal{A}_u, 9), \text{rep}(0, 36)]^T \text{ and } \tilde{\mathbf{u}}_3 = [\text{rep}(0, 11), \text{unif}(\mathcal{A}_u, 9), \text{rep}(0, 30)]^T; \mathbf{v}_k^*$$

generated as $\tilde{\mathbf{v}}_1$ where $\tilde{\mathbf{v}}_1 =$

$$[\text{unif}(\mathcal{A}_v, 5), \text{rep}(0, 20)]^T, \tilde{\mathbf{v}}_2 = [\text{rep}(0, 5), \text{unif}(\mathcal{A}_v, 5), \text{rep}(0, 15)]^T, \text{ and } \tilde{\mathbf{v}}_3. \text{ The notation}$$

$$= [\text{rep}(0, 10), \text{unif}(\mathcal{A}_v, 5), \text{rep}(0, 10)]^T$$

$\text{unif}(A, b)$ denotes a vector of length b whose entries are i.i.d. uniformly distributed on set A ; we use $\mathcal{A}_u = \pm 1, \mathcal{A}_v = [-1, -0.3] \cup [0.3, 1]$. The notation $\text{rep}(a, b)$ denotes a vector of length b , whose entries are all equal to a . Model II is a high dimensional example with $p = 500, q = 200$ and $n = 400$, in which the $\mathbf{U}^*$ and $\mathbf{V}^*$ are obtained by appending the $\mathbf{u}_k^*$ and $\mathbf{v}_k^*$ vectors generated from Model I with zeros.

The predictor matrix $\mathbf{X} \in \mathbb{R}^{n \times p}$ is generated from the SFAR model in (5) and (6).

Specifically, let $\mathbf{x} \sim \mathcal{N}(\mathbf{0}, \Gamma)$ where $\Gamma = [\gamma_{ij}]_{p \times p}$ with $\gamma_{ij} = 0.5^{|i-j|}$. Denote $\mathbf{x}_1 = \mathbf{U}^T \mathbf{x}$; then

$\mathbf{x}_1 \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{r^*})$. As such, we first generate $\mathbf{X}_1 \in \mathbb{R}^{n \times r^*}$ by drawing n random samples from

$\mathcal{N}(\mathbf{0}, \mathbf{I}_{r^*})$. Given a generated $\mathbf{U}^*$, we can find a $\mathbf{U}_\perp^* \in \mathbb{R}^{p \times (p-r^*)}$ such that

$\mathbf{P} = [\mathbf{U}^*, \mathbf{U}_\perp^*] \in \mathbb{R}^{p \times p}$ and $\text{rank}(\mathbf{P}) = p$. It follows that $\mathbf{P}^T \mathbf{x} \sim \mathcal{N}(\mathbf{0}, \mathbf{P}^T \Gamma \mathbf{P})$. Let $\mathbf{x}_2 = \mathbf{U}_\perp^{*T} \mathbf{x}$; we can

then generate $\mathbf{X}_2 \in \mathbb{R}^{n \times (p-r^*)}$ by drawing n random samples from the conditional

distribution of $\mathbf{x}_2$ given $\mathbf{x}_1$. Finally, we obtain $\mathbf{X} = [\mathbf{X}_1, \mathbf{X}_2] \mathbf{P}^{-1}$.

The rows of the error matrix $\mathbf{E}$ are generated as i.i.d samples from $\mathcal{N}(0, \sigma^2 \Delta)$, where

$$\Delta = [\delta_{ij}]_{q \times q} \text{ with } \delta_{ij} = \rho^{|i-j|} \text{ and we experiment with } \rho \in \{0, 0.3, 0.5\}. \text{ The response matrix}$$

$\mathbf{Y}$ is then generated by $\mathbf{Y} = \mathbf{X} \mathbf{C}^* + \mathbf{E}$. We set σ according to a given signal to noise ratio

(SNR) defined as $\text{SNR} = \|\mathbf{d}_{r^*}^* \mathbf{X} \mathbf{u}_{r^*}^* \mathbf{v}_{r^*}^{*T}\|_2 / \|\mathbf{E}\|_F$, i. e.. $\text{SNR} \in \{0.25, 0.5\}$. The missing entries in

$\mathbf{Y}$ are randomly selected, and we consider missing proportions $M\% \in \{0\%, 15\%, 30\%\}$. The experiment was replicated 300 times under each setup.

The estimation accuracy is measured by

$Er(\mathbf{C}) = \|\hat{\mathbf{C}} - \mathbf{C}\|_{F/(pq)}^2$, $Er(\mathbf{XC}) = \left\| r^{\frac{1}{2}} \hat{\mathbf{C}} - \mathbf{C} \right\|_{F/(nq)}^2$. The sparsity recovery in the decomposition of the coefficient matrix is characterized by the false positive rate (FPR) and the false negative rate (FNR), calculated from comparing the sparsity pattern of $(\hat{\mathbf{u}}_k, \hat{\mathbf{v}}_k)$ to that of $(\mathbf{u}_k^*, \mathbf{v}_k^*)$, $k = 1, \dots, r^*$. We evaluate rank estimation performance using the (relative) percentage of signal left in the $(r^* + 1)$ th estimated component, i.e., $R\% = 100(\hat{d}_{r^*+1}^2 / \sum_{k=1}^{r^*+1} \hat{d}_k^2)$. Finally, the orthogonality of the SeCURE estimates is measured by $ORT = \|\hat{\mathbf{U}}^T \mathbf{X}^T \mathbf{X} \hat{\mathbf{U}} / n\|_1 + \|\hat{\mathbf{V}}^T \hat{\mathbf{V}}\|_1 - 2\hat{r}$.

6.2 Simulation Results: Tables 1 and 2 reports the simulation results for Models I and II, respectively, under the setting that $\text{SNR} = 0.25$ and $\rho = 0.3$. Consider first the results in case of complete data. SeCURE methods outperform all other methods in both predictive accuracy and sparse recovery. As expected, LS performs the worst; RRR improves LS greatly but is outperformed by other sparse and low-rank approaches. Both RSSVD and SRRR do not perform well in sparsity recovery, because the former searches for the sparsity pattern according to the SVD of $\mathbf{C}^*$ rather than that of $\mathbf{XC}^*$, and the latter only considers rowwise sparsity in $\mathbf{C}^*$. All the regularized methods perform well in reducing the rank, as $R\%$ is mostly close to zero.

Neither RSSVD nor SRRR can handle incomplete data, so they are dropped from the comparison when $M\% \in \{15\%, 30\%\}$. As a benchmark, we implemented an algorithm for fitting RRR with incomplete data using the idea of matrix completion, which can be regarded as a special case of SeCURE for setting $\lambda = 0$. As expected, the performance of all methods deteriorates when the percentage of missing increases, but SeCURE still performs well and always greatly outperforms RRR.

In general, the performance of SeCURE using adaptive lasso is comparable to that of using adaptive elastic net; the former tends to perform slightly better in estimation and prediction. We still recommend to use elastic net in practice, because it better ensures the stability in optimization as shown in our convergence analysis. Also, when computing power is adequate, tuning α may further improve the elastic net.

Figure 1 compares different versions of SeCURE in terms of the prediction and the orthogonality measures using Model II. Here we consider SeCURE using either non-adaptive elastic net or adaptive elastic net, and with or without orthogonality constraints. (The lasso versions are omitted). The adaptive penalization in general improves model estimation. The exact orthogonality, when desired, can be efficiently achieved by SeCURE. However, as seen from Tables 1–2, enforcing orthogonality of the sample factors, in general, may hurt both model estimation and sparsity recovery. This clearly demonstrates the advantage of SeCURE, as it is able to bypass the orthogonality in the recovery of the co-sparsity pattern.

SeCURE is more computationally efficient than its closest competitor RSSVD, and the efficiency gain increases with the model dimension. We have also experimented with different SNR values, error correlations and missing proportions, and the results are all consistent with those reported herein. As expected, the performance of the methods improves when SNR becomes higher, the missing proportions decreases or the error correlation weakens; the error correlation impacts both the accuracy and the variability in model estimation slightly. To save space, all the additional simulation results are reported in Section 1 of the Supplementary Materials.

7 Yeast Cell Cycle Data—When $\mathbf{X} = \mathbf{I}_n$, SeCURE performs sparse and unit-rank approximation of $\mathbf{Y}$, which can serve as an unsupervised learning tool for biclustering. In Section 2 of the Supplementary Materials, we provide an example using a gene expression dataset from patients with diffuse large-B-cell lymphoma after chemotherapy (Rosenwald et al., 2002).

Here we consider a yeast cell cycle analysis, for identifying transcription factors regulating the RNA transcript levels of yeast genes within the eukaryotic cell cycle. The Eukaryotic cell cycle data were generated using α factor arrest method, consisting of RNA levels measured every 7 minutes for 119 minutes with a total of 18 time points covering two cell cycle of 6178 genes. The chromatin immunoprecipitation (ChIP) data ($\mathbf{X}$) (Lee et al., 2002) contain complete binding information of a subset of 1790 genes for a total of 113 transcription factors. However, the RNA data corresponding to these genes still contain about 2% missing values. Spellman et al. (1998) identified 800 genes responsible for cell cycle regulation. Chun and Kelecs (2010) and Chen and Huang (2012) analyzed 524 of the 800 genes after excluding the genes with missing RNA levels and/or binding information. Since SeCURE can directly handle missing values in the response, we use all the $n = 1790$ genes, to examine the association between the RNA levels along the $q = 18$ time points and the binding information of $p = 113$ transcription factors (TF).

The estimated rank by SeCURE is $\hat{r} = 4$. We mainly discuss the first three factors as they explain 93.7% of the variation of the four factors. Our SeCURE approach selected 51, 57, 13 TFs in the three factors, respectively, with 83 distinct TFs in total. Of these, 17 are among the 21 experimentally confirmed TFs that relate to cell cycle (Wang et al., 2007). The SRRR and the sparse partial least squares (SPLS) selected 60 and 12 TFs, of which 16 and 7 are among the confirmed TFs, respectively (Chen and Huang, 2012). We have also fitted the RSSVD method with imputed data by filling the blanks using column means, and the method selected 24 TFs, of which 11 are among the confirmed ones. Figure 2 shows the estimated effects (rows in $\hat{\mathbf{C}}$) of a few experimentally confirmed TFs identified by SeCURE. As expected, the effects are mostly periodic and the two cycles are clearly seen.

The SeCURE analysis identified a few latent factors from the 113 TFs, given by the columns in $\mathbf{X}\hat{\mathbf{U}}$. It is interesting to check how the RNA levels at 18 time points load on these factors, which is revealed by examining the columns of $\mathbf{V}\mathbf{D}$. Figure 3 plots the first three columns of $\mathbf{V}\mathbf{D}$ together with fitted curves using kernel smoothing. The periodic effects are very clear. Interestingly, the three curves appear to be at slightly different phases: the first curve picks around 15 and 75 minutes, the second picks around 20 and 80 minutes, and the third picks

pick around 30 and 90 minutes. In either of the first two curves, the two cycles are similar to each other in shape and magnitude. In contrast, in the third curve the two cycles are apparently different in magnitude and the second cycle almost disappears. Further examining the TFs involved in these factors may shed more light on the differential roles played by the TFs.

8 Discussion—We have developed a sequential estimation procedure to recover an appealing co-sparse and low-rank factor structure in multivariate regression. The model allows each latent factor to be constructed from a subset of predictors, and may potentially influence only a subset of responses. Importantly, we show that the orthogonality constraints that are required in joint estimation can be avoided in sequential estimation, and without them computational efficiency and estimation consistency can both be achieved.

There are several directions for future research. A priority is to perform a comprehensive non-asymptotic analysis to fully understand the finite sample behaviors of the SeCURE estimators. In our sequential procedure the residual matrix is used in each step to deflate the information found from the previous steps; it would be interesting to investigate other deflation methods. The sequential estimation procedure can be extended to vector generalized linear models, in which the response variables may be non-Gaussian, e.g., binary or Poisson. More generally, it is pressing to understand how the sequential estimation behaves in non-linear models. Many multivariate learning methods can be unified as generalized eigen-decomposition problems. It is thus interesting to develop a unified sequential estimation procedure that performs co-sparse generalized eigen-decomposition (Taha Bahadori *et al.*, 2016).

Supplementary Material

Supplementary Material Supp1.

Acknowledgments

Chen's research was partially supported by the National Science Foundation grant DMS- 1613295 and the National Institutes of Health (NIH) grant U01-HL114494. The authors are grateful to the Editor, the Associate Editor, and the two referees for their valuable comments and suggestions, which have led to significant improvement of the paper.

References

- Anderson TW (1951) Estimating linear restrictions on regression coefficients for multivariate normal distributions. *Annals of Mathematical Statistics*, 22, 327–351.
- Anderson TW and Rubin H (1956) Statistical inference in factor analysis In *Proceedings of the Third Berkeley Symposium on Mathematical Statistics and Probability*, vol. 5, 111–150. Berkeley, CA: University of California Press.
- Bunea F, She Y and Wegkamp M (2011) Optimal selection of reduced rank estimators of high-dimensional matrices. *Annals of Statistics*, 39, 1282–1309.
- Bunea F, She Y and Wegkamp M (2012) Joint variable and rank selection for parsimonious estimation of high dimensional matrices. *Annals of Statistics*, 40, 2359–2388.
- Candès EJ and Recht B (2009) Exact matrix completion via convex optimization. *Found. Comput. Math*, 9, 717–772.

- Chen K, Chan K-S and Stenseth NC (2012) Reduced rank stochastic regression with a sparse singular value decomposition. *Journal of the Royal Statistical Society: Series B*, 74, 203–221.
- Chen K, Dong H and Chan K-S (2013) Reduced rank regression via adaptive nuclear norm penalization. *Biometrika*, 100, 901–920. [PubMed: 25045172]
- Chen L and Huang JZ (2012) Sparse reduced-rank regression for simultaneous dimension reduction and variable selection. *Journal of the American Statistical Association*, 107, 1533–1545.
- Chun H and Kelecs S (2010) Sparse partial least squares regression for simultaneous dimension reduction and variable selection. *Journal of the Royal Statistical Society: Series B*, 72, 3–25.
- Fan Y and Tang CY (2013) Tuning parameter selection in high dimensional penalized likelihood. *Journal of the Royal Statistical Society: Series B*, 75, 531–552.
- Gorski J, Pfeuffer F and Klamroth K (2007) Biconvex sets and optimization with biconvex functions: a survey and extensions. *Mathematical Methods of Operations Research*, 66(3), 373–407.
- James GM, Paulson C and Rusmevichientong P (2013) Penalized and constrained regression Technical Report, University of Southern California.
- Journée M, Nesterov Y, Richtárik P and Sepulchre R (2010) Generalized power method for sparse principal component analysis. *J. Mach. Learn. Res*, 11, 517–553.
- Lee TI, Rinaldi NJ, Robert F, Odom DT, Bar-Joseph Z, Gerber GK, Han- nett NM, Harbison CT, Thompson CM, Simon I et al. (2002) Transcriptional regulatory networks in *saccharomyces cerevisiae*. *Science*, 298, 799–804. [PubMed: 12399584]
- Lin W, Shi P, Feng R and Li H (2014) Variable selection in regression with compositional covariates. *Biometrika*, 101, 785–797.
- Lu Z, Monteiro RDC and Yuan M (2012) Convex optimization methods for dimension reduction and coefficient estimation in multivariate linear regression. *Math. Program*, 131, 163–194.
- Luenberger D and Ye Y (2008) *Linear and Nonlinear Programming*. New York: Springer.
- Ma, Z. and Sun, T. (2014) Adaptive sparse reduced-rank regression. *ArXiv e-prints*. URL <http://arxiv.org/abs/1403.1922>.
- Mackey LW (2009) Deflation methods for sparse PCA In *Advances in Neural Information Processing Systems 21* (eds. Koller D, Schuurmans D, Bengio Y and Bottou L), 1017–1024. Curran Associates, Inc.
- Mukherjee A and Zhu J (2011) Reduced rank ridge regression and its kernel extensions. *Statistical Analysis and Data Mining*, 4, 612–622. [PubMed: 22993641]
- Negahban S and Wainwright MJ (2011) Estimation of (near) low-rank matrices with noise and high-dimensional scaling. *Annals of Statistics*, 39, 1069–1097.
- R Development Core Team (2017) *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria ISBN 3–900051–07–0.
- Reinsel GC and Velu P (1998) *Multivariate reduced-rank regression: theory and applications*. New York: Springer.
- Rosenwald A, Wright G, Chan WC, Connors JM, Campo E, Fisher RI, Gascoyne RD, Muller-Hermelink HK, Smeland EB, Giltmane JM et al. (2002) The use of molecular profiling to predict survival after chemotherapy for diffuse large-b-cell lymphoma. *New England Journal of Medicine*, 346, 1937–1947. [PubMed: 12075054]
- Spellman PT, Sherlock G, Zhang MQ, Iyer VR, Anders K, Eisen MB, Brown PO, Botstein D and Futcher B (1998) Comprehensive identification of cell cycle-regulated genes of the yeast *saccharomyces cerevisiae* by microarray hybridization. *Molecular Biology of the Cell*, 9, 3273–3297. [PubMed: 9843569]
- Stone M (1974) Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society: Series B*, 111–147.
- Taha Bahadori M, Zheng Z, Liu Y and Lv J (2016) Scalable interpretable multi-response regression via SEED. *ArXiv e-prints*.
- Tibshirani RJ (1996) Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B*, 58, 267–288.
- Wang L, Chen G and Li H (2007) Group scad regression analysis for microarray time course gene expression data. *Bioinformatics*, 23, 1486–1494. [PubMed: 17463025]

- Yuan M, Ekici A, Lu Z and Monteiro R (2007) Dimension reduction and coefficient estimation in multivariate linear regression. *Journal of the Royal Statistical Society: Series B*, 69, 329–346.
- Zhou H and Li L (2014) Regularized matrix regression. *Journal of the Royal Statistical Society: Series B*, 76, 463–483.
- Zou H (2006) The adaptive lasso and its oracle properties. *Journal of the American Statistical Association*, 101, 1418–1429.
- Zou H and Hastie TJ (2005) Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B*, 67, 301–320.
- Zou H and Zhang HH (2009) On the adaptive elastic-net with a diverging number of parameters. *Annals of Statistics*, 37, 1733–1751. [PubMed: 20445770]

Figure 1:

Simulation: Boxplots of scaled $\text{Er}(\mathbf{XC})$ (left panel) and ORT (right panel) in Model II and $\rho = 0.3$. SeCURE(E*) and SeCURE(E) denote SeCURE using non-adaptive elastic net penalty with and without orthogonality constraints, respectively.

Figure 2:

Yeast cell cycle data: the estimated transcriptional effects of 3 experimentally confirmed TFs identified by SeCURE.

Figure 3:

Yeast cell cycle data: the estimated loadings of the RNAs of 18 time points on the three identified latent factors from the TFs. The fitted curves using kernel smoothing are added. The two vertical lines are drawn at 15 and 75 in the first panel, at 20 and 80 in the second panel and at 30 and 90 in the third panel.

Initialization: set $k = 1$, and set a desired rank $r \geq 1$.

repeat

(1) Obtain $(\tilde{d}_k, \tilde{\mathbf{u}}_k, \tilde{\mathbf{v}}_k)$ from RRR, and construct $\mathbf{W}_k$ as in (11).

(2) Compute the current residual matrix $\mathbf{Y}_k$ as in (13).

(3) Perform the CURE($\mathbf{C}; \mathbf{Y}_k, \mathbf{X}, \mathbf{W}_k$) analysis via (12) (including the tuning process), and obtain $\hat{\mathbf{C}}_k$ or equivalently $(\hat{d}_k, \hat{\mathbf{u}}_k, \hat{\mathbf{v}}_k)$.

if $\hat{d}_k = 0$ **then**

Set $\hat{d}_h = 0$ for any $k \leq h \leq r$; $k \leftarrow r + 1$.

else

$k \leftarrow k + 1$.

end if

until $k = r + 1$.

return $\hat{\mathbf{C}}_k$ and $(\hat{d}_k, \hat{\mathbf{u}}_k, \hat{\mathbf{v}}_k)$ for all $k = 1, \dots, r$ with $\hat{d}_k \neq 0$.

Algorithm 1.

Sequential Factor Extraction via Co-Sparse Unit-Rank Estimation (SeCURE)

Initialization: Set $\lambda > 0$ and $0 < \alpha < 1$. Obtain initial value $\mathbf{v}_\lambda^0$. Set $s = 0$.

repeat

(1) **U-step:** Given $\mathbf{v} = \mathbf{v}_\lambda^s$, minimize objective function (16) to obtain $\check{\mathbf{u}}_\lambda^{s+1}$. Update $\tilde{d}_\lambda^{s+1} = \|\mathbf{X}\check{\mathbf{u}}_\lambda^{s+1}\|_2/\sqrt{n}$ and $\mathbf{u}_\lambda^{s+1} = \check{\mathbf{u}}_\lambda^{s+1}/\tilde{d}_\lambda^{s+1}$ when $\check{\mathbf{u}}_\lambda^{s+1} \neq \mathbf{0}$; otherwise return $d_\lambda^{s+1} = \tilde{d}_\lambda^{s+1} = 0$ and terminate the algorithm.

(2) **V-step:** Given $\mathbf{u} = \mathbf{u}_\lambda^{s+1}$, minimize the objective function (17) to obtain $\check{\mathbf{v}}_\lambda^{s+1}$. Update $d_\lambda^{s+1} = \|\check{\mathbf{v}}_\lambda^{s+1}\|_2$ and $\mathbf{v}_\lambda^{s+1} = \check{\mathbf{v}}_\lambda^{s+1}/d_\lambda^{s+1}$ when $\check{\mathbf{v}}_\lambda^{s+1} \neq \mathbf{0}$; otherwise return $d_\lambda^{s+1} = 0$ and terminate the algorithm.

$s \leftarrow s + 1$.

until convergence, e.g., $\|\mathbf{C}_\lambda^{s+1} - \mathbf{C}_\lambda^s\|_F/\|\mathbf{C}_\lambda^s\|_F < \epsilon$, where $\mathbf{C}_\lambda^s = d_\lambda^s \mathbf{u}_\lambda^s \mathbf{v}_\lambda^{sT}$ and $\epsilon = 10^{-4}$.

return $(\hat{\mathbf{d}}^{(\lambda)}, \hat{\mathbf{u}}^{(\lambda)}, \hat{\mathbf{v}}^{(\lambda)})$ and $\hat{\mathbf{C}}^{(\lambda)} = \hat{\mathbf{d}}^{(\lambda)} \hat{\mathbf{u}}^{(\lambda)} \hat{\mathbf{v}}^{(\lambda)T}$.

Algorithm 2.

Co-Sparse Unit-Rank Estimation Algorithm (CURE)

Table 1:

Simulation: results of Model I with SNR = 0.25 and $\rho = 0.3$. For better presentation, $\text{Er}(\mathbf{C})$ and $\text{Er}(\mathbf{XC})$ are scaled by multiplying 10^4 .

Method	Er(C)	Er(XC)	FPR	FNR	R%		
		Missing=	0%, SNR = 0.25				
LS	2994.4 (208.0)	227.7 (14.2)	100.0 (0.0)	0.0 (0.0)	-		
RRR	810.7 (96.4)	65.6 (7.1)	100.0 (0.0)	0.0 (0.0)	5.1 (0.8)		
SeCURE(AL*)	117.7 (38.7)	10.5 (3.2)	1.2 (0.8)	0.0 (0.0)	0.0 (0.0)		
SeCURE(AL)	92.9 (29.8)	8.1 (2.5)	0.3 (0.4)	0.0 (0.0)	0.0 (0.0)		
SeCURE(AE*)	121.1 (39.1)	10.8 (3.3)	1.5 (1.0)	0.0 (0.0)	0.0 (0.0)		
SeCURE(AE)	96.4 (30.4)	8.4 (2.5)	0.5 (0.6)	0.0 (0.0)	0.0 (0.0)		
RSSVD	317.8 (113.6)	28.6 (10.4)	8.2 (5.7)	3.4 (4.3)	0.0 (0.0)		
SRRR	388.2 (55.5)	34.2 (4.6)	55.4 (3.0)	0.0 (0.0)	2.5 (0.5)		
		Missing=	15%, SNR =	0.25			
RRR	951.8 (113.3)	76.8 (8.5)	100.0 (0.0)	0.0 (0.0)	3.3 (0.4)		
SeCURE(AL*)	137.9 (43.1)	12.7 (4.1)	1.5 (1.0)	0.0 (0.0)	0.0 (0.0)		
SeCURE(AL)	113.3 (36.4)	10.2 (3.3)	0.6 (0.7)	0.0 (0.0)	0.0 (0.0)		
SeCURE(AE*)	140.0 (42.3)	12.8 (3.9)	1.7 (1.0)	0.0 (0.0)	0.0 (0.0)		
SeCURE(AE)	118.3 (37.0)	10.7 (3.4)	0.7 (0.7)	0.0 (0.0)	0.0 (0.0)		
		Missing=	30%, SNR =	0.25			
RRR	1151.7 (118.3)	93.8 (9.7)	100.0 (0.0)	0.0 (0.0)	3.8 (0.4)		
SeCURE(AL*)	173.1 (51.4)	15.9 (4.8)	2.0 (1.3)	0.0 (0.0)	0.0 (0.0)		
SeCURE(AL)	153.4 (50.7)	13.6 (4.3)	0.9 (0.8)	0.0 (0.0)	0.0 (0.0)		
SeCURE(AE*)	176.9 (52.0)	16.1 (4.8)	2.3 (1.3)	0.0 (0.0)	0.0 (0.0)		
SeCURE(AE)	154.9 (47.4)	14.0 (4.3)	1.3 (1.0)	0.0 (0.0)	0.0 (0.0)		

Table 2:

Simulation: Results of Model II with SNR = 0.25 and $\rho = 0.3$. For better presentation, Er(C) and Er(XC) are scaled by multiplying 10^4 .

Method	Er(C)	Er(XC)	FPR	FNR	R%
		Missing=	0%, SNR =	0.25	
LS	1015.8 (40.2)	882.6 (34.7)	100.0 (0.0)	0.0 (0.0)	-
RRR	48.1 (4.1)	41.8 (3.5)	100.0 (0.0)	0.0 (0.0)	13.7 (1.4)
SeCURE(AL*)	0.4 (0.2)	0.4 (0.2)	0.1 (0.1)	0.0 (0.0)	0.0 (0.0)
SeCURE(AL)	0.1 (0.1)	0.1 (0.0)	0.0 (0.0)	0.0 (0.0)	0.0 (0.0)
SeCURE(AE*)	0.4 (0.2)	0.4 (0.2)	0.1 (0.1)	0.0 (0.0)	0.0 (0.0)
SeCURE(AE)	0.2 (0.1)	0.1 (0.0)	0.0 (0.0)	0.0 (0.0)	0.0 (0.0)
RSSVD	1.9 (0.9)	1.7 (0.8)	1.3 (0.6)	0.6 (1.4)	0.2 (0.4)
SRRR	2.5 (0.3)	2.5 (0.3)	30.1 (0.0)	0.0 (0.0)	11.0 (1.5)
		Missing=	15%, SNR =	0.25	
RRR	55.2 (4.2)	48.2 (3.6)	100.0 (0.0)	0.0 (0.0)	3.9 (0.2)
SeCURE(AL*)	0.5 (0.2)	0.5 (0.2)	0.1 (0.1)	0.0 (0.0)	0.0 (0.0)
SeCURE(AL)	0.2 (0.1)	0.2 (0.1)	0.0 (0.0)	0.0 (0.0)	0.0 (0.0)
SeCURE(AE*)	0.5 (0.2)	0.5 (0.2)	0.1 (0.1)	0.0 (0.0)	0.0 (0.0)
SeCURE(AE)	0.2 (0.1)	0.2 (0.1)	0.0 (0.0)	0.0 (0.0)	0.0 (0.0)
		Missing=	30%, SNR =	0.25	
RRR	75.1 (9.1)	65.4 (7.8)	100.0 (0.0)	0.0 (0.0)	4.5 (0.2)
SeCURE(AL*)	0.6 (0.3)	0.6 (0.2)	0.2 (0.1)	0.0 (0.0)	0.0 (0.0)
SeCURE(AL)	0.3 (0.1)	0.3 (0.1)	0.0 (0.0)	0.0 (0.0)	0.0 (0.0)
SeCURE(AE*)	0.7 (0.3)	0.6 (0.2)	0.2 (0.1)	0.0 (0.0)	0.0 (0.0)
SeCURE(AE)	0.4 (0.1)	0.3 (0.1)	0.0 (0.0)	(0.0)	0.0 (0.0)