Journal of Machine Learning Research 20 (2019) 1-34 Submitted 4/18; Revised 2/19; Published 6/19

Scalable Interpretable Multi-Response Regression via SEED

Zemin Zheng ZHENGZM@USTC.EDU.CN
School of Management and School of Data Science
International Institute of Finance
University of Science and Technololgy of China
Hefei, Anhui 230026, China

M. Taha Bahadori BAHADORI@GATECH.EDU
School of Computational Science and Engineering
Georgia Institute of Technology
Atlanta, GA 30332, USA

Yan Liu YANLIU.CS@USC.EDU
Computer Science Department
Viterbi School of Engineering
University of Southern California
Los Angeles, CA 90089, USA

Jinchi Lv JINCHILV@MARSHALL.USC.EDU
Data Sciences and Operations Department
Marshall School of Business
University of Southern California
Los Angeles, CA 90089, USA

Editor: Xiaotong Shen

Abstract

Sparse reduced-rank regression is an important tool for uncovering meaningful dependence structure between large numbers of predictors and responses in many big data applications such as genome-wide association studies and social media analysis. Despite the recent theoretical and algorithmic advances, scalable estimation of sparse reduced-rank regression remains largely unexplored. In this paper, we suggest a scalable procedure called sequential estimation with eigen-decomposition (SEED) which needs only a single top-r sparse singular value decomposition from a generalized eigenvalue problem to find the optimal low-rank and sparse matrix estimate. Our suggested method is not only scalable but also performs simultaneous dimensionality reduction and variable selection. Under some mild regularity conditions, we show that SEED enjoys nice sampling properties including consistency in estimation, rank selection, prediction, and model selection. Moreover, SEED employs only basic matrix operations that can be efficiently parallelized in high performance computing devices. Numerical studies on synthetic and real data sets show that SEED outperforms the state-of-the-art approaches for large-scale matrix estimation problem.

Keywords: reduced-rank Regression, scalability, high dimensionality, greedy algorithm, sparse eigenvector estimation

1. Introduction

Identifying complex dependence structures among predictors and responses is an important problem in statistics and machine learning, since these structures reveal hidden domain knowledge about the data. For example, in bioinformatics, identifying gene regulatory networks is crucial for understanding gene regulatory paths and gene functions, which helps disease prediction and diagnosis. Similarly, in social media analysis, inferring the influence networks from user activities, that is, *Diffusion Network Inference* problem (Leskovec et al., 2008; Zhou et al., 2013; Embar et al., 2014) is an important problem and it has found applications in social media marketing (Gomez-Rodriguez et al., 2012) and crisis management (Starbird and Palen, 2012). In these big data applications, inferring the dependence structures is challenging since the responses and predictors may be related through a few latent pathways and/or associated through only a subset of responses and predictors. Moreover, the curse of dimensionality and massive amounts of data, that is, scalability issues make the dependence structure discovery problem even harder to solve. To recover sparse response-predictor associations and latent predictors, regularization methods such as lasso (Tibshirani, 1996) and group lasso (Yuan and Lin, 2006), and reduced-rank regression approaches (Izenman, 1975; Velu and Reinsel, 2013) have been proposed, respectively. Chen et al. (2012) and Chen and Chan (2015) have proposed sparse reduced-rank regression approaches by combining the regularization and reduced-rank regression techniques to find the complex dependence structures between responses and predictors.

Sparse reduced-rank regression works by modeling the associations between the predictor and response variables via a sparse and low-rank representation of the coefficient matrix. It not only enhances the interpretability of the estimated matrix by eliminating irrelevant features (Chen et al., 2012), but also reduces the number of free parameters of the model and thus the number of observations required for desired estimation consistency (Yuan et al., 2007; Bunea et al., 2011; Candès and Plan, 2011; Negahban and Wainwright, 2011; Chen et al., 2013). Sparse reduced-rank regression has found applications in micro-array biclustering (Chen et al., 2012), subspace clustering (Wang et al., 2013), social network community discovery (Richard et al., 2012; Zhou et al., 2013), and motion segmentation (Feng et al., 2014). In these applications, joint sparsity and low-rankness has been used to enforce a clustered dependence structure among data points. In particular, the key idea is to estimate a similarity matrix among data points that is simultaneously sparse and low-rank and then permute the rows and columns of the matrix to yield approximately block-diagonal structures, which naturally lead to clustering of data points into several groups. Note that Chandrasekaran et al. (2010), Agarwal et al. (2012), and the references therein have considered estimating matrices with a low-rank plus sparse representation which is different from our work as we are interested in estimating a matrix that is jointly low-rank and sparse.

A natural approach to solving the sparse reduced-rank regression problem is to simultaneously penalize the parameter matrix using the L_1 and nuclear norm regularizers, as they are convex relaxations to sparsity and low-rankness of a matrix, respectively. The resulting optimization problem is convex and can be solved using the alternating direction method of multipliers (ADMM) (Boyd et al., 2010) as proposed by Richard et al. (2012) and Zhou et al. (2013). In Bunea et al. (2012), an alternative approach, called rank constrained group lasso (RCGL), was proposed which directly penalizes the rank and the number of nonzero rows of the parameter matrix. They showed oracle rates for the estimated matrix and also provided a practical algorithm which iteratively and jointly solves a L_1-regularization and low-rank estimation problem. To further improve the estimation accuracy, Chen et al. (2012) borrowed ideas from adaptive Lasso (Zou, 2006) and proposed the iterative exclusive extraction algorithm (IEEA) which finds a locally optimal solution in the neighborhood of the initial value. They also showed model selection consistency and asymptotic normality results along with the improved empirical performance of IEEA on microarray biclustering analysis data.

All the above approaches for sparse reduced-rank regression achieve both desirable theoretical properties and strong empirical results. However, they cannot scale to large matrix estimation problems in many big data applications. The ADMM algorithm of Richard et al. (2012) and Zhou et al. (2013) uses iterative singular value thresholding (Cai et al., 2010) for solving the joint L_1 and nuclear norm regularization. Iterative singular value thresholding is known to be computationally expensive since it performs a full singular value decomposition of the parameter matrix in each iteration. On the other hand, RCGL (Bunea et al., 2012) is computationally much faster than ADMM since it only performs top-r singular value decomposition for estimating a rank-r matrix in each iteration. Despite a lower computational cost per iteration, it is unclear how many iterations RCGL needs for convergence. IEEA (Chen et al., 2012) performs nested loops of alternating L_1-penalized regression for each singular vector which can be expensive, especially on parallel computing devices. The iterative nature of these three approaches makes them not scalable and renders them inefficient for large matrix estimation even on high performance computing devices.

To overcome the scalability issues of the previous approaches, we propose a simple and scalable sparse reduced-rank regression procedure called sequential estimation with eigen-decomposition (SEED). SEED is designed for high-performing computing platforms. It converts the sparse and low-rank regression problem to a sparse generalized eigenvalue problem and then solves the problem using the recent algorithms for sparse eigenvalue decomposition (Cai et al., 2013; Ma, 2013; Yuan and Zhang, 2013). As a pure learning algorithm, SEED is expected to perform only a single top-r sparse eigenvalue decomposition for estimating a rank-r matrix, which makes it truly scalable and efficient for large matrix estimation problems.

The main contributions of our paper are threefold. First, for the sparse reduced-rank regression problem, our proposed procedure SEED provides a scalable approach to uncovering the sparse predictor-response association network while simultaneously achieving dimension reduction and variable selection. Second, for the high-dimensional settings, our theoretical analysis shows that SEED can consistently estimate the singular vectors, latent factors as well as the regression coefficient matrix, identify the correct rank of the matrix, accurately predict the multivariate response vector, and recover the support of the singular vectors under mild conditions. Note that, compared with Chen et al. (2012), we do not make any assumption on the positive definiteness of the design matrix for proving our consistency results. Third, we empirically demonstrate that SEED can not only be efficiently implemented on both central processing units (CPU) and graphics processing units (GPU) for large-scale applications, but it also outperforms the state-of-the-art sparse reduced-rank regression approaches.

Recently, Mishra et al. (2017) proposed the sequential co-sparse factor regression in a similar sparse and low-rank model setting, where a unit rank sparse coefficient matrix was recovered at each step such that both the left and the right singular vectors could be estimated with co-sparse pattern (an earlier version of the current paper was cited in theirs). Compared with their method, our approach separates the estimation of the singular vectors at each step with the left ones obtained through a generalized sparse eigenvalue problem and the right ones recovered directly based on the left ones. Moreover, we provide comprehensive theoretical properties to justify the validity of sequential estimation for sparse and low-rank coefficient matrices in high dimensions.

The rest of this paper is organized as follows. Section 2 introduces our SEED method. We discuss the implementation details of SEED in Section 3 and present its asymptotic properties in Section 4. We demonstrate the advantages of SEED on both synthetic and real data sets in Section 5, and in Section 6 we discuss some extensions of our SEED method. The proofs of some main results are relegated to the Appendix. Additional technical details are provided in the Supplementary Material.

2. Sequential Estimation with Eigen-Decomposition

We will first introduce the model setup and then the proposed methodology.

2.1. Model and Problem Formulation

Denote by $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^n$ n observations in the fixed design setting, where $\mathbf{x}_i \in \mathbb{R}^p$ and $\mathbf{y}_i \in \mathbb{R}^q$ represent the ith predictor and the corresponding response vectors, respectively. Given a predictor vector $\mathbf{x}$, the corresponding response vector $\mathbf{y}$ is drawn from the following model

$$\mathbf{y} = \mathbf{C}^{*T}\mathbf{x} + \boldsymbol{\varepsilon},$$

where the noise vector $\boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$ is a q-dimensional multivariate Gaussian random vector with mean zero and covariance matrix $\boldsymbol{\Sigma}$, and $\mathbf{C}^* \in \mathbb{R}^{p\times q}$ is the regression coefficient matrix.[1] We can rewrite the model in the matrix form as follows

$$\mathbf{Y} = \mathbf{X}\mathbf{C}^* + \mathbf{E}, \tag{1}$$

where $\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_n]^T$, $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_n]^T$, and $\mathbf{E} = [\boldsymbol{\varepsilon}_1, \dots, \boldsymbol{\varepsilon}_n]^T$ denote the matrices of stacked response, predictor and noise vectors, respectively.

Let $\mathbf{P} = n^{-1}\mathbf{X}^T\mathbf{X}$ be the Gram matrix of the predictors. We consider model (1) from a latent factor point of view, where the true regression coefficient matrix $\mathbf{C}^*$ is jointly low-rank and sparse, similar to Bunea et al. (2012) and Chen et al. (2012). In particular, $\mathrm{rank}(\mathbf{C}^*) = r^*$ with the matrix rank $r^* \ll \min(p, q)$. Without loss of generality, we assume that $\mathrm{rank}(\mathbf{X}\mathbf{C}^*) = \mathrm{rank}(\mathbf{C}^*)$ since if $\mathrm{rank}(\mathbf{X}\mathbf{C}^*) < \mathrm{rank}(\mathbf{C}^*)$, the redundant part of $\mathbf{C}^*$ can be removed such that it reflects the true number of latent factors. Then $\mathbf{C}^*$ will have the following decomposition

$$\mathbf{C}^* = \sum_{k=1}^{r^*} \mathbf{u}_k^*\mathbf{v}_k^{*T} = \sum_{k=1}^{r^*} \mathbf{C}_k^*, \tag{2}$$

[1] Note that the Gaussianity of noise variables is not essential to either our procedure or the theoretical results and similar results would hold under the sub-Gaussian assumption (Bühlmann and van de Geer, 2011, Chapter 14).

where the left singular vectors $\mathbf{u}_k^* \in \mathbb{R}^p$ are $\mathbf{P}$-orthogonal with unit length, that is, $\mathbf{u}_k^{*T}\mathbf{P}\mathbf{u}_{k'}^* = 0$ if $k \neq k'$ and $\|\mathbf{u}_k^*\|_2 = 1$, while the right singular vectors $\mathbf{v}_k^* \in \mathbb{R}^q$ are orthogonal, that is, $\mathbf{v}_k^{*T}\mathbf{v}_{k'}^* = 0$ for $k \neq k'$, and $\mathbf{C}_k^*$ is the layer k unit rank matrix of $\mathbf{C}^*$. The singular vectors are sorted by the magnitudes of the singular values $\sigma_k = \frac{1}{\sqrt{nq}}\|\mathbf{X}\mathbf{C}_k^*\|_F$ in descending order, consistent with their contributions to the prediction of $\mathbf{Y}$.

We consider the left singular vectors (both the population and estimated ones) in the constraint space $\mathbf{u} \perp \mathrm{Ker}(\mathbf{P})$, where $\mathrm{Ker}(\mathbf{P})$ denotes the null space of $\mathbf{P}$, to guarantee the model identifiability. Otherwise, $\mathbf{u}$ would contain certain component $\widetilde{\mathbf{u}}$ such that $\mathbf{X}\widetilde{\mathbf{u}} = 0$, which does not contribute to the prediction of $\mathbf{Y}$. It is worth noticing that the decomposition in the form of $\mathbf{C}^* = \sum_{k=1}^{r^*}\mathbf{u}_k^*\mathbf{v}_k^{*T}$ is not unique without orthogonality constraints and the entrywise sparsity in the singular vectors is not invariant to rotations. The decomposition in (2) is a special one since the $\mathbf{P}$-orthogonality of $\mathbf{u}_k^*$ arises naturally from the power method and leads to a latent factor analysis interpretation of the reduced rank estimation in which the latent factors $\mathbf{X}\mathbf{u}_k^*$ are uncorrelated. Moreover, decomposition (2) coincides with the singular value decomposition of $\mathbf{X}\mathbf{C}^*$ except for different scalings on the singular vectors. We defer the discussion on the identifiability of decomposition (2) (existence and uniqueness up to simultaneous sign changes of $\mathbf{u}_k^*$ and $\mathbf{v}_k^*$) to Supplementary Material.

The aforementioned modeling of the regression coefficient matrix gives a latent factor model with r^* latent factors, where $\mathbf{X}\mathbf{u}_k^*$ is the kth latent factor and $\mathbf{v}_k^*$ describes the impacts of the kth factor on the response variables. As illustrated in Yuan et al. (2007), the low-rankness of $\mathbf{C}^*$ renders dimension reduction such that all responses can be predicted by a relatively small set of common factors. Furthermore, the left singular vectors $\mathbf{u}_k^*$ are assumed to be sparse, yielding the necessity of predictor selection. Similar sparsity assumptions were made in Bunea et al. (2012) and Chen et al. (2012) to enhance model interpretability by removing irrelevant features in high dimensions. Specifically, Chen et al. (2012) assumed that both the left and right singular vectors are sparse while Bunea et al. (2012) imposed restriction on the number of nonzero rows of the regression coefficient matrix. In this paper, we are interested in two cases: (i) when the right singular vectors are not required to be sparse and (ii) when it is desirable to have sparse right singular vectors, which entails the response selection. We will show that both cases are efficiently accommodated by our procedure.

Our goal is to accurately estimate the singular vectors $\mathbf{u}_k^*$ and $\mathbf{v}_k^*$, and the true rank r^* such that we can recover the latent factors, their impacts, and the underlying number of latent factors as well as the significant predictors. As a singular vector can have two opposite directions, we always assume that the estimated left singular vector takes the correct one, that is, the angles between estimated and population left singular vectors are no more than a right angle. Once the estimated rank and singular vectors are obtained, the estimate $\widehat{\mathbf{C}}$ of the true matrix $\mathbf{C}^*$ follows immediately from (2). Unlike most existing sparse and low-rank estimation methods which adopt the regularization framework of minimizing a loss function plus certain penalties, we will show that the proposed procedure SEED is indeed a pure learning algorithm that predicts $\mathbf{Y}$ using $\mathbf{X}\widehat{\mathbf{C}}$.

2.2. Description of SEED

The following proposition provides us insight into estimating the top-r^* left and right singular vectors of $\mathbf{C}^*$.

Proposition 1 *Consider the noiseless case* $\mathbf{Y}^* = \mathbf{X}\mathbf{C}^*$ *with* $\mathbf{C}^* = \sum_{k=1}^{r^*} \mathbf{u}_k^* \mathbf{v}_k^{*T}$ *defined in (2). Then* $\mathbf{u}_1^*, \ldots, \mathbf{u}_{r^*}^*$ *are the* r^* *non-degenerate left singular vectors of* $\mathbf{C}^*$ *if and only if they are eigenvectors of the following generalized eigenvalue problem*

$$\mathbf{X}^T\mathbf{Y}^*\mathbf{Y}^{*T}\mathbf{X}\mathbf{u} = \lambda \mathbf{X}^T\mathbf{X}\mathbf{u} \tag{3}$$

with respect to the nonzero eigenvalues $\lambda_1, \cdots, \lambda_{r^*}$*, where* $\lambda_k = nq\sigma_k^2$ *is the kth largest eigenvalue of* $\mathbf{Y}^*\mathbf{Y}^{*T}$ *with* $\sigma_k = \|\mathbf{X}\mathbf{C}_k^*\|_F/\sqrt{nq}$ *defined in Section 2.1. Furthermore, given the left singular vector* $\mathbf{u}_k^*$*, the corresponding right singular vector* $\mathbf{v}_k^*$ *can be written as*

$$\mathbf{v}_k^* = \frac{1}{\mathbf{u}_k^{*T}\mathbf{X}^T\mathbf{X}\mathbf{u}_k^*}\mathbf{Y}^{*T}\mathbf{X}\mathbf{u}_k^*. \tag{4}$$

Proposition 1 shows that the problem of estimating the singular vectors can be transformed into the generalized eigenvalue problem in (3), thanks to the **P**-orthogonality of the left singular vectors. It motivates us to estimate the left singular vectors in the noisy case $\mathbf{Y} = \mathbf{X}\mathbf{C}^* + \mathbf{E}$ by solving the following generalized eigenvalue problem

$$\mathbf{X}^T\mathbf{Y}\mathbf{Y}^T\mathbf{X}\mathbf{u} = \lambda \mathbf{X}^T\mathbf{X}\mathbf{u}. \tag{5}$$

The estimation consistency can be guaranteed by the matrix perturbation theory (see Section 4 for details). On the other hand, it is not difficult to see that the eigenvectors with respect to different eigenvalues of problem (5) are **P**-orthogonal, which further gives the orthogonality of the right singular vectors estimated according to (4). It implies that the right and left singular vectors obtained by solving the generalized eigenvalue problem will automatically be orthogonal and **P**-orthogonal, respectively.

Related results of principal component analysis in low dimensions can be found in Baldi and Hornik (1989), Diamantaras and Kung (1996), and De La Torre and Black (2003). Note that in the high-dimensional setting, the regime of interest for this paper, the Gram matrix **P** can not be invertible and the generalized eigenvalue problem is potentially challenging to solve. We will address the implementation challenges in Section 3.

Based on Proposition 1, our proposed procedure SEED performs a two-step estimation for the regression coefficient matrix. It first solves the generalized eigenvalue problem (5) to obtain the estimated left singular vectors $\widehat{\mathbf{u}}_1, \ldots, \widehat{\mathbf{u}}_r$ with unit length and then finds the estimated right singular vectors $\widehat{\mathbf{v}}_1, \ldots, \widehat{\mathbf{v}}_r$ according to (4) as follows

$$\widehat{\mathbf{v}}_k = \frac{1}{\widehat{\mathbf{u}}_k^T\mathbf{X}^T\mathbf{X}\widehat{\mathbf{u}}_k}\mathbf{Y}^T\mathbf{X}\widehat{\mathbf{u}}_k. \tag{6}$$

The maximum rank r depends on the magnitude of the estimated singular value $\widehat{\sigma}_k = \|\mathbf{X}\widehat{\mathbf{C}}_k\|_F/\sqrt{nq}$ with $\widehat{\mathbf{C}}_k = \widehat{\mathbf{u}}_k\widehat{\mathbf{v}}_k^T$ (whether it is larger than a threshold μ) and the optimal rank can be tuned by cross validation or the information criterion described in Section 4.

```
Algorithm 1: SEED
   Input: (1) Data $\mathbf{Y} \in \mathbb{R}^{n \times q}$ and $\mathbf{X} \in \mathbb{R}^{n \times p}$ (2) Termination accuracy $\mu$ and sparsity
          parameter $\theta$
1  Compute matrices: $\mathbf{P} \leftarrow \frac{1}{n}\mathbf{X}^T\mathbf{X},\ \mathbf{R} \leftarrow \frac{1}{n}\mathbf{Y}^T\mathbf{X},\ \mathbf{Q} \leftarrow \frac{1}{q}\mathbf{R}^\top\mathbf{R}$
2  $k \leftarrow 1$
3  repeat
4      $(\widehat{\mathbf{u}}_k, \widehat{\sigma}_k) \leftarrow k$th $\theta$-sparse eigenvector and eigenvalue of $\mathbf{Qu} = \lambda\mathbf{Pu}$
5      if $\widehat{\sigma}_k > \mu$ then
6          $\widehat{\mathbf{v}}_k \leftarrow \frac{1}{\widehat{\mathbf{u}}_k^\top\mathbf{P}\widehat{\mathbf{u}}_k}\mathbf{R}\widehat{\mathbf{u}}_k$        #Optional thresholding of $\widehat{\mathbf{v}}_k$ for sparsity
7          $\widehat{\mathbf{C}}_k \leftarrow \widehat{\mathbf{u}}_k\widehat{\mathbf{v}}_k^\top$
8          $k \leftarrow k + 1$
9      end
10 until $\widehat{\sigma}_k < \mu$
11 tune the optimal rank $\widehat{r}$
12 return $\widehat{\mathbf{C}} = \sum_{k=1}^{\widehat{r}} \widehat{\mathbf{C}}_k$
```

The details of the procedure are provided in Algorithm 1. To achieve a sparse solution, we need to find the rank-r sparse matrix via a sparse eigenvalue decomposition procedure in Line 4 of Algorithm 1. This can be done in a sequential way and practical methods will be provided in Section 3. The practical methods need a sparsity parameter θ, such as a threshold (Ma, 2013) or a sparsity size (Yuan and Zhang, 2013). We will show in Section 5 that SEED is robust to the choices of parameters θ and μ. If the right singular vectors are also required to be sparse, we perform a simple element-wise thresholding on $\widehat{\mathbf{v}}_k$ after we obtain it in Line 6.

3. Scalable Implementation of SEED

Algorithm 1 requires a sparse solution of (5) which is a generalized eigenvalue problem with a rank deficient matrix $\mathbf{P}$. In this section, we propose two different procedures to solve (5) and study multiple practical aspects of SEED.

3.1. Fast Approach

The bottleneck in speeding up Algorithm 1 is Line 4 where we need to solve a sparse generalized eigenvalue problem. To overcome this bottleneck, we propose a new solution to estimating the left singular vectors by rewriting equation (5) as

$$\mathbf{X}^T(\mathbf{Y}\mathbf{Y}^T - \lambda\mathbf{I})\mathbf{X}\mathbf{u} = \mathbf{0}.$$

Similar to Proposition 1, when $\mathbf{X}$ is of full row rank (which is easy to satisfy in the high-dimensional setting), the above equation shares the same nonzero eigenvalues with $\mathbf{Y}\mathbf{Y}^T$. Even if $\mathbf{X}$ is row rank deficient, we still have nonzero solution $\mathbf{u}$ when the perturbation is relatively small, ensured by the perturbation theory in Lemma 6. Thus, we propose the following two-step procedure for Line 4 of Algorithm 1:

(1) $\lambda \leftarrow \lambda_{\max}(\mathbf{Y}\mathbf{Y}^T)$.

(2) $\widehat{\mathbf{u}}_1 \leftarrow$ sparse eigenvector relative to the zero eigenvalue of $\mathbf{X}^T\mathbf{Y}\mathbf{Y}^T\mathbf{X} - \lambda\mathbf{X}^T\mathbf{X}$.

Based on $\widehat{\mathbf{u}}_1$, the corresponding right singular vector $\widehat{\mathbf{v}}_1$ and the unit rank matrix $\widehat{\mathbf{C}}_1$ will be obtained. To continue, we compute the residual $\mathbf{Y}_k = \mathbf{Y} - \mathbf{X}\sum_{j=1}^{k-1}\widehat{\mathbf{C}}_j$ in the kth step and replace $\mathbf{Y}$ with $\mathbf{Y}_k$ in the above two-step procedure to obtain the kth left singular vector $\widehat{\mathbf{u}}_k$. Overall, the first step requires calculation of the top-r eigenvalues for an $n \times n$ matrix (or $q \times q$ if $q < n$) while the second step finds the corresponding eigenvectors by solving a regular sparse eigenvalue problem.

The above procedure significantly accelerates the speed of SEED as it converts a degenerate sparse generalized eigenvalue problem to two simpler regular sparse eigenvalue problems. Existing procedures such as the iterative thresholding method (Ma, 2013) can be used to compute both eigenvalue problems efficiently. Generally speaking, SEED will be extremely efficient in applications with low rank structure such as image processing as it stops early after achieving a few important signals. Moreover, the speed of SEED can be greatly enhanced by parallel implementation on high performance computing devices such as GPU, due to the fact that it employs only basic matrix operations.

3.2. Alternative with Enhanced Stability

In cases where the perturbation can be large, for numerical stability purposes, we can also solve (5) by the following modified problem with a very small positive ρ:

$$\mathbf{X}^T\mathbf{Y}\mathbf{Y}^T\mathbf{X}\mathbf{u} = \lambda(\mathbf{X}^T\mathbf{X} + \rho\mathbf{I})\mathbf{u}. \tag{7}$$

Note that $\mathbf{X}^T\mathbf{X} + \rho\mathbf{I}$ is invertible since the eigenvalues of $\mathbf{X}^T\mathbf{X}$ are nonnegative. Denote by $\widetilde{\mathbf{X}} \in \mathbb{R}^{p\times p}$ the modified predictor matrix such that $\widetilde{\mathbf{X}}^T\widetilde{\mathbf{X}} = \mathbf{X}^T\mathbf{X} + \rho\mathbf{I}$, which can be obtained via the Cholesky decomposition, and $\widetilde{\mathbf{Y}} = (\widetilde{\mathbf{X}}^T)^{-1}\mathbf{X}^T\mathbf{Y}$ the modified response matrix. Then the above equation (7) can be rewritten as

$$\widetilde{\mathbf{X}}^T\widetilde{\mathbf{Y}}\widetilde{\mathbf{Y}}^T\widetilde{\mathbf{X}}\mathbf{u} = \lambda\widetilde{\mathbf{X}}^T\widetilde{\mathbf{X}}\mathbf{u}, \tag{8}$$

which adopts the same form as (5).

A computationally efficient technique for solving equation (8) is to solve the sparse eigenvalue problem $\widetilde{\mathbf{P}}^{-1}\widetilde{\mathbf{Q}}\mathbf{u} = \lambda\mathbf{u}$, where the modified Gram matrix $\widetilde{\mathbf{P}} = n^{-1}\widetilde{\mathbf{X}}^T\widetilde{\mathbf{X}}$ and $\widetilde{\mathbf{Q}}$ is defined accordingly as in Algorithm 1. We can compute $\widetilde{\mathbf{P}}^{-1}$ by the Sherman-Morrison-Woodbury formula as follows:

$$(\rho\mathbf{I}_p + \mathbf{X}^T\mathbf{X})^{-1} = \frac{1}{\rho}\mathbf{I}_p - \frac{1}{\rho^2}\mathbf{X}^T(\mathbf{I}_n + \frac{1}{\rho}\mathbf{X}\mathbf{X}^T)^{-1}\mathbf{X}.$$

The above equation requires inversion of an $n \times n$ matrix instead of a $p \times p$ matrix, which is significantly faster in the high-dimensional setting when $p \gg n$.

Remark. The formulation of $\widetilde{\mathbf{X}}$ and $\widetilde{\mathbf{Y}}$ can be regarded as a generalization of the ridge regression to the multivariate response setting. In fact, since $\mathbf{C}^*$ is the minimizer of $\|\mathbf{Y}^* - \mathbf{X}\mathbf{C}\|_F^2$, we can enhance the stability by adding a small Frobenius norm regularization as follows:

$$\widetilde{\mathbf{C}} = \operatorname{argmin}_{\mathbf{C}} \left\{ \|\mathbf{Y} - \mathbf{X}\mathbf{C}\|_F^2 + \rho \|\mathbf{C}\|_F^2 \right\},$$

where the Frobenius norm is defined as $\|\mathbf{C}\|_F^2 = \sum_{i,j} \mathbf{C}_{i,j}^2$ for any matrix $\mathbf{C}$. After completing the squares, we get

$$\widetilde{\mathbf{C}} = \operatorname{argmin}_{\mathbf{C}} \left\{ \|\widetilde{\mathbf{Y}} - \widetilde{\mathbf{X}}\mathbf{C}\|_F^2 \right\},$$

which means that $\widetilde{\mathbf{X}}$ and $\widetilde{\mathbf{Y}}$ are the corresponding predictor and response matrices that take into account the shrinkage effects (James and Stein, 1961; Zheng et al., 2014).

3.3. Practical Aspects

Now we analyze several practical aspects of SEED as follows.

Refitting. In Algorithm 1, a refitting can be performed during the eigenvalue decomposition in Line 4 to enhance the stability. The refitting procedure is as follows. In the kth step, we perform the top-k singular value decomposition $\widehat{\mathbf{C}} = \mathbf{U}\widehat{\mathbf{S}}\mathbf{V}^T$ for $\widehat{\mathbf{C}} = \sum_{j=1}^{k} \widehat{\mathbf{C}}_j$ and refit the solution by finding $\widetilde{\mathbf{S}} = \operatorname{argmin}_{\mathbf{S}} \left\|\mathbf{Y} - \mathbf{X}\mathbf{U}\mathbf{S}\mathbf{V}^T\right\|_F^2$. The estimate with refitting is defined as $\widetilde{\mathbf{C}} = \mathbf{U}\widetilde{\mathbf{S}}\mathbf{V}^T$. In practice, we find this approach more stable and report the results based on this variation of SEED in numerical studies.

Sparse eigenvector estimation. The previous approaches for solving the generalized eigenvalue decomposition problem indicate that we can solve the problem via regular sparse eigenvalue decomposition. This allows us to reuse the existing procedures for sparse eigenvalue decomposition such as Cai et al. (2013), Ma (2013), Yuan and Zhang (2013), and Lei and Vu (2015) to solve the problem in (5). In numerical studies, we use the iterative thresholding method (Ma, 2013), which is detailed in Algorithm 2 for estimating the sparse eigenvector relative to the largest eigenvalue of a given sparse eigenvalue problem $\mathbf{S}\mathbf{u} = \lambda\mathbf{u}$ with sparsity parameter θ (threshold level).

It is worth pointing out that although sparse eigenvalue decomposition is a nonconvex problem, Ma (2013) proved that the sparse eigenvectors obtained by the iterative thresholding algorithm will converge to their population counterparts with asymptotic probability one within $O(\log n)$ iterations. The initial estimate $\widehat{\mathbf{u}}^{(0)}$ is generated from the diagonal thresholding sparse PCA algorithm proposed in Johnstone and Lu (2009), which is a regular eigenvalue problem after constraining on the significant coordinates (variances). This iterative thresholding algorithm can also be applied to recover several sparse eigenvectors simultaneously.

Parallel implementation. In order to scale up the procedures, we often need to utilize the parallel computing tools. Given that SEED only uses basic matrix operations, we can employ parallel implementation of the large matrix operations in high performance computing devices to accelerate SEED. In Section 5, our experiments with GPU which contain thousands of processing units show that the matrix operations in SEED can be efficiently parallelized and it significantly enhances the speed of SEED. Whenever the data can not be loaded into the memory of a single device, efficient distributed algorithms can be used. See for example, Kang et al. (2011) and the references therein.

Algorithm 2: Iterative thresholding

```
Input: Matrix $\mathbf{S} \in \mathbb{R}^{p \times p}$, threshold level $\theta$, and initial estimate $\widehat{\mathbf{u}}^{(0)} \in \mathbb{R}^p$.
$k \leftarrow 1$
repeat
    Multiplication: $\mathbf{t}^{(k)} = (t_{ij}^{(k)}) = \mathbf{S}\widehat{\mathbf{u}}^{(k-1)}$
    Thresholding: $\widehat{\mathbf{t}}^{(k)} = (\widehat{t}_{ij}^{(k)})$ with $\widehat{t}_{ij}^{(k)} = t_{ij}^{(k)} 1_{(|t_{ij}| \geq \theta)}$
    QR factorization: $\widehat{\mathbf{u}}^{(k)} R^{(k)} = \widehat{\mathbf{t}}^{(k)}$ with $R^{(k)} = \|\widehat{\mathbf{t}}^{(k)}\|_2$
    $k \leftarrow k + 1$.
until convergence
return $\widehat{\mathbf{u}}^{(k)}$ at convergence
```

4. Asymptotic Properties of SEED

In this section, we will analyze the statistical properties of SEED. Define the maximum sparsity level of the left singular vectors as $s^* = \max_{k=1}^{r^*} \|\mathbf{u}_k^*\|_0 \ll p$, which is assumed mainly for theoretical analysis (see Condition 1 below) and unknown in our practical implementation. We consider the estimated left singular vectors with the number of nonzero elements less than certain sparsity level $s > s^*$, that is, $\|\widehat{\mathbf{u}}_k\|_0 \leq s$ for $k = 1, \cdots, r$, where $r > r^*$ is an upper bound of the estimated rank that can be controlled by the algorithm in practice. When the generalized eigenvalue problem (5) does not have an s-sparse solution, the estimated largest s-sparse eigenvalue can be reformulated as

$$\widehat{\lambda} = \max_{\mathbf{u} \neq \mathbf{0}, \|\mathbf{u}\|_0 \leq s} \frac{\mathbf{u}^T(\mathbf{X}^T\mathbf{Y}\mathbf{Y}^T\mathbf{X})\mathbf{u}}{\mathbf{u}^T(\mathbf{X}^T\mathbf{X})\mathbf{u}}$$

due to the extremal property of eigenvalues, and the corresponding maximizer $\widehat{\mathbf{u}}$ will be treated as the estimated left singular vector. Note that there is no sparsity constraint on either the population or the estimated right singular vectors since $\mathbf{v}_k^*$ can be recovered from (4) based on accurate estimation of $\mathbf{u}_k^*$.

4.1. Technical Conditions

Here we list a few technical conditions and discuss their relevance in detail.

Condition 1 (Restricted Isometry) *There exists a positive constant ϕ_s such that the Gram matrix $\mathbf{P}$ satisfies*

$$\phi_s \leq \min_{\mathbf{z} \in \mathbb{R}^p} \left\{ \frac{\|\mathbf{P}\mathbf{z}\|_2}{\|\mathbf{z}\|_2} : \|\mathbf{z}\|_0 \leq 2s \right\} \leq \max_{\mathbf{z} \in \mathbb{R}^p} \left\{ \frac{\|\mathbf{P}\mathbf{z}\|_2}{\|\mathbf{z}\|_2} : \|\mathbf{z}\|_0 \leq 2s \right\} \leq \phi_s^{-1}$$

for some $s > s^*$.

Condition 2 (Minimum Singular Value Separation) *The non-zero singular values σ_k satisfy $\sigma_k^2 - \sigma_{k+1}^2 \geq d_\sigma > 0$ for some positive constant d_σ and $k = 1, \ldots, r^*$.*

Condition 3 (Bounded Eigenvalues) *The eigenvalues of the population covariance matrix of the noise vector* $\boldsymbol{\varepsilon}$ *satisfy* $0 < \gamma_l^2 \leq \lambda_j(\boldsymbol{\Sigma}) \leq \gamma_u^2 < \infty$ *for* $j = 1, \ldots, q$*, where* γ_l *and* γ_u *are positive constants with* $\gamma_u \leq c_\gamma \sigma_{r^*}$ *for some positive constant* c_γ*.*

Condition 4 (Minimum Signal Strength) *There exists some positive constant* $\delta \in (0, 1)$ *such that the following lower bounds on the magnitudes of the non-zero elements of* $\mathbf{u}_k^*$ *and* $\mathbf{v}_k^*$ *hold for any* $1 \leq k \leq r^*$

$$\min_{i \in \mathrm{supp}(\mathbf{u}_k^*)} |u_i^*| \geq 3C_u \sqrt{\frac{s}{n} \log \frac{pq}{\delta}},$$
$$\min_{i \in \mathrm{supp}(\mathbf{v}_k^*)} \frac{1}{\sqrt{q}} |v_i^*| \geq 3C_v \sqrt{\frac{s}{n} \log \frac{pq}{\delta}},$$

where C_u *and* C_v *are constants defined in Theorem 1.*

Condition 1 imposes bounds on the $2s$-sparse eigenvalues of $\mathbf{P}$, which is weaker than the regular bounded eigenvalue assumption since the sparse eigenvalues do not grow as fast as the regular eigenvalues when the dimensionality p increases. As a typical condition in high dimensions, it restricts the correlations between small numbers of features and thus guarantees the identifiability of the true support. See, for instance, Candès and Tao (2005) and Zhang (2011) for more discussion on it.

Recall that σ_k is the kth largest singular value of $\mathbf{X}\mathbf{C}^*/\sqrt{nq}$. Condition 2 requires strict separation among the singular values such that the left singular vectors are distinguishable. For ease of presentation, we assume d_σ to be a constant and indicate the roles of d_σ clearly in the constants of the theoretical results. In fact, $\mathbf{X}\mathbf{C}^*$ adopts a spiked eigen-structure (Johnstone, 2001; Shen et al., 2016) with the spiked singular values allowed to diverge at the rate of $\sqrt{nq}$. This rate is reasonable for an n by q matrix as we do not impose any sparsity on the columns of $\mathbf{C}^*$.

The elements of the unobserved noise vector $\boldsymbol{\varepsilon}$ were assumed to be independent and identically distributed (i.i.d.) in Bunea et al. (2012). We relax it a bit in Condition 3 by imposing bounded eigenvalues for the noise covariance matrix for recovering the true rank in Theorem 2. Our technical argument still applies when either $\gamma_l \to 0$ or $\gamma_u \to \infty$ as long as their rates of convergence can be controlled within certain magnitudes.

The two inequalities in Condition 4 are imposed for the model selection consistency of the predictors and responses, respectively. The magnitude of the minimum signal strength is $O\left(\sqrt{s \log(pq)/n}\right)$, which is relatively mild as it converges to zero in our setting. Since $\mathbf{u}_k^*$ is assumed to have unit length, the singular values of $\mathbf{C}^*$ are absorbed into $\mathbf{v}_k^*$ in view of decomposition (2) so that there is an extra scaling factor $\frac{1}{\sqrt{q}}$ in the second inequality.

4.2. Main Results

Denote by $P^2 = \max_{j=1}^{p} \mathbf{P}_{jj}$ and $\gamma^2 = \max_{j=1}^{q} \boldsymbol{\Sigma}_{jj}$ the maximum diagonal components of the Gram matrix and noise covariance matrix, respectively. Without loss of generality, we assume that $V = \max_{k=1}^{r^*} \frac{1}{\sqrt{q}} \|\mathbf{v}_k^*\|_2$ is finite for the q-dimensional vectors $\mathbf{v}_k^*$. Moreover, it is clear that under Conditions 1 and 3, P and γ are also finite constants. The estimated regression coefficient matrix is given by $\widehat{\mathbf{C}} = \sum_{k=1}^{\tilde{r}} \widehat{\mathbf{C}}_k$, where $\tilde{r}$ is the optimal rank tuned by information criterion (9) and $\widehat{\mathbf{C}}_k = \widehat{\mathbf{u}}_k \widehat{\mathbf{v}}_k^T$. The following theorem bounds the estimation errors of SEED with the estimated left singular vectors $\widehat{\mathbf{u}}_k$ taking the correct signs as discussed before.

Theorem 1 (Consistency of Estimation and Prediction) *Suppose that Conditions 1 and 2 hold, γ is finite, and $s\log(pq) = o(n)$, then with probability at least $1-\delta$ for any $\delta \in (0,1)$ and uniformly over $k = 1, \ldots, r^*$, we have*

$$\begin{aligned}
\|\widehat{\mathbf{u}}_k - \mathbf{u}_k^*\|_2 &\le C_u \sqrt{\frac{s}{n}\log\frac{pq}{\delta}} + o\left(\sqrt{\frac{s}{n}\log\frac{pq}{\delta}}\right), \\
\frac{1}{\sqrt{q}}\|\widehat{\mathbf{v}}_k - \mathbf{v}_k^*\|_2 &\le C_v \sqrt{\frac{s}{n}\log\frac{pq}{\delta}} + o\left(\sqrt{\frac{s}{n}\log\frac{pq}{\delta}}\right), \\
\frac{\|\mathbf{X}(\widehat{\mathbf{u}}_k - \mathbf{u}_k^*)\|_2}{\sqrt{n}} &\le \frac{C_u}{\sqrt{\phi_s}} \sqrt{\frac{s}{n}\log\frac{pq}{\delta}} + o\left(\sqrt{\frac{s}{n}\log\frac{pq}{\delta}}\right), \\
\frac{1}{\sqrt{q}}\|\widehat{\mathbf{C}}_k - \mathbf{C}_k^*\|_F &\le (VC_u + C_v)\left(\sqrt{\frac{s}{n}\log\frac{pq}{\delta}}\right) + o\left(\sqrt{\frac{s}{n}\log\frac{pq}{\delta}}\right), \\
\frac{\|\mathbf{X}(\widehat{\mathbf{C}}_k - \mathbf{C}_k^*)\|_F}{\sqrt{nq}} &\le \frac{(VC_u + C_v)}{\sqrt{\phi_s}}\left(\sqrt{\frac{s}{n}\log\frac{pq}{\delta}}\right) + o\left(\sqrt{\frac{s}{n}\log\frac{pq}{\delta}}\right),
\end{aligned}$$

where the constants $C_u = \frac{4\gamma P\sigma_1}{d_\sigma \phi_s^{5/2}}$ and $C_v = 2\sqrt{2}\phi_s^{-3/2}(2V\phi_s^{-1/2} + \sigma_1)C_u + 2\sqrt{2}\phi_s^{-1}\gamma P$.

Theorem 1 is established based on the perturbation theory of sparse generalized eigenvalue problem (Lemma 6). It shows that the uniform estimation error bounds for both top-r^* singular vectors $\mathbf{u}_k^*$ and $\frac{1}{\sqrt{q}}\mathbf{v}_k^*$, top-$r^*$ latent factors $\frac{1}{\sqrt{n}}\mathbf{X}\mathbf{u}_k^*$ and unit rank matrices $\frac{1}{\sqrt{q}}\mathbf{C}_k^*$, and the uniform prediction error bounds of the top-r^* latent factors are all in the same order of $O\left(\sqrt{\frac{s}{n}\log\frac{pq}{\delta}}\right)$. The tail probability δ can decay to zero quickly as p and q grow with rates such as $\delta \propto (pq)^{-\alpha}$ for some positive constant $\alpha > 1$. This is due to the fact that when $\delta \propto (pq)^{-\alpha}$, we have $\sqrt{\frac{s}{n}\log\frac{pq}{\delta}} \to 0$ under the assumption that $s\log(pq) = o(n)$. Furthermore, the estimation and prediction accuracy would then be within the rate of $O\left(\sqrt{s\log(pq)/n}\right)$, where the factor $\log(pq)$ reflects the curse of dimensionality as there are pq parameters in total from the regression coefficient matrix $\mathbf{C}^*$.

If the true rank r^* can be correctly identified, it is not difficult to see that the estimation accuracy for $\frac{1}{\sqrt{q}}\mathbf{C}^*$ will be within the rate of $O\{\sqrt{r^* s\log(pq)/n}\}$ (see Corollary 3 below), which coincides with the minimax error bound for estimating the regression coefficient vector in the univariate response setting (Raskutti et al., 2011) with the dimensionality p and sparsity level s replaced by the overall dimensionality pq and the product r^*s, respectively.

Similarly, with the true rank r^*, the normalized prediction error $\|\mathbf{X}(\widehat{\mathbf{C}} - \mathbf{C}^*)\|_F/\sqrt{nq}$ will be within the rate of $O\left\{\sqrt{r^* s\log(pq)/n}\right\}$, which is similar to the prediction error bound established in Bunea et al. (2012). But we have an extra scaling factor of $\sqrt{q}$ since the singular values of $\mathbf{X}\mathbf{C}^*$ are allowed to diverge at the rate of $\sqrt{nq}$, larger than the rate $\sqrt{n} + \sqrt{q}$ in Bunea et al. (2012). Our theoretical results show that when the left singular vectors are consistently estimated, the normalized prediction error will converge to zero asymptotically. Overall speaking, the prediction error bound in Bunea et al. (2012) was derived from the regularization framework, while the prediction accuracy of SEED is obtained from the matrix perturbation theory (Lemmas 5 and 6).

Based on the discussion before, a desirable statistical property of any low-rank estimation procedure is accurately recovering the true rank of the parameter matrix. Similar to Lasso, the nuclear norm regularization needs to be enhanced by techniques such as adaptive regularization to accurately recover the rank of the matrix (Bach, 2008; Chen et al., 2013). In contrast, we can directly control the rank of the solution in SEED by limiting the number of steps. In particular, we propose a GIC-type (Fan and Tang, 2013) information criterion that guarantees rank recovery by SEED when the optimal rank is tuned according to it.

Theorem 2 (Consistency of Rank Recovery) *Suppose Conditions 1-3 hold,* $s\log(pq) = o(n)$*,* $r^* = o\left(\frac{1}{\sqrt{(\log\log n)\vee\sqrt{s}}} \cdot \left[\frac{n}{\log(pq)}\right]^{\frac{1}{4}}\right)$*, and* $r = o\left(\left[\frac{n}{s\log(pq)}\right]^{\frac{1}{4}}\right)$*. Define the following information criterion*

$$\mathcal{C}_n = \sqrt{n}\log\mathcal{L}_n(\mathbf{Y},\mathbf{X},\widehat{\mathbf{C}}) + \mathrm{rank}(\widehat{\mathbf{C}})\sqrt{\log(pq)}\log\log n, \tag{9}$$

where $\mathcal{L}_n(\mathbf{Y},\mathbf{X},\mathbf{C}) = \frac{1}{qn}\|\mathbf{Y}-\mathbf{X}\mathbf{C}\|_F^2$*. Under the above information criterion, with probability at least* $1-(pq)^{-\alpha}$ *for some positive constant* $\alpha > 1$ *and sufficiently large* n*, SEED will select the true rank, that is,* $\mathrm{rank}(\widehat{\mathbf{C}}) = \mathrm{rank}(\mathbf{C}^*)$*.*

In the high-dimensional setting where the number of predictors can increase exponentially with the sample size, it is demonstrated in Fan and Tang (2013) that we need some power of the logarithmic factor of dimensionality ($\sqrt{\log(pq)}$ for our setting) in the model complexity penalty of the information criterion to consistently identify the true model, and the slow diverging rate $\log\log n$ is set to prevent underfitting. The proof of Theorem 2 indeed shows that information criterion (9) will keep decreasing until the estimated rank reaches the true rank r^*, where in each step the amount of decrease in the objective function $\mathcal{L}_n(\mathbf{Y},\mathbf{X},\widehat{\mathbf{C}})$ equals to the squared singular value obtained by solving the generalized eigenvalue problem (5). After reaching the true rank, the estimated singular value becomes small such that the model complexity penalty will overweight the decrease and then information criterion (9) would start increasing. Therefore, in the sequence of solutions generated by SEED, the estimate $\widehat{\mathbf{C}}$ with rank r^* will be the minimizer of (9) such that the true rank can be correctly identified.

As discussed before, correct identification of the true rank will yield the estimation accuracy of $\mathbf{C}^*$ as well as the prediction accuracy of $\mathbf{X}\mathbf{C}^*$. Therefore, combined with Theorem 2, it is immediate that the results in Theorem 1 give the following corollary.

Corollary 3 (Consistency of Overall Estimation and Prediction) *Given Conditions 1-3,* $s\log(pq) = o(n)$*,* $r^* = o\left(\frac{1}{\sqrt{(\log\log n)\vee\sqrt{s}}} \cdot \left[\frac{n}{\log(pq)}\right]^{\frac{1}{4}}\right)$*, and* $r = o\left(\left[\frac{n}{s\log(pq)}\right]^{\frac{1}{4}}\right)$*, if the optimal rank is tuned by information criterion (9), then with probability at least* $1-(pq)^{-\alpha}$

for any constant $\alpha > 0$ and sufficiently large n, we have

$$\frac{1}{\sqrt{q}}\|\widehat{\mathbf{C}} - \mathbf{C}^*\|_F \le (1+\alpha)(VC_u + C_v)\left(\sqrt{\frac{r^* s \log(pq)}{n}}\right) + o\left(\sqrt{\frac{r^* s \log(pq)}{n}}\right),$$

$$\frac{\|\mathbf{X}(\widehat{\mathbf{C}} - \mathbf{C}^*)\|_F}{\sqrt{qn}} \le \frac{(1+\alpha)(VC_u + C_v)}{\sqrt{\phi_s}}\left(\sqrt{\frac{r^* s \log(pq)}{n}}\right) + o\left(\sqrt{\frac{r^* s \log(pq)}{n}}\right).$$

Besides estimation consistency and rank recovery, SEED is also able to find the true support of the singular vectors. To achieve this goal, after selecting the optimal rank, we need to further refine the model selection procedure by performing a hard-thresholding. See, for example, Fan and Lv (2013) for applications of thresholding in high-dimensional sparse modeling. Specifically, denote by $T_\theta(\mathbf{z})$ the estimator after the hard-thresholding operation on every element of $\mathbf{z} = (z_1, \cdots, z_p) \in \mathbb{R}^p$ such that

$$T_\theta(z_i) = \begin{cases} 0 & \text{if } |z_i| < \theta \\ z_i & \text{otherwise} \end{cases}, \quad i = 1, \ldots, p.$$

Based on the results of Theorems 1 and 2 and the signal strength assumption in Condition 4, we have the following properties for the estimator with a further thresholding.

Theorem 4 (Support Recovery of Singular Vectors) *Given Conditions 1-4, $s\log(pq) = o(n)$, $r^* = o\left(\frac{1}{\sqrt{(\log\log n) \vee \sqrt{s}}} \cdot \left[\frac{n}{\log(pq)}\right]^{\frac{1}{4}}\right)$, and $r = o\left(\left[\frac{n}{s\log(pq)}\right]^{\frac{1}{4}}\right)$, for every pair of singular vectors, $(\widehat{\mathbf{u}}_k, \widehat{\mathbf{v}}_k)$, $k = 1, \ldots, r^*$, the following results hold.*

a) If the threshold $\theta \in (\frac{5}{4}T_u, \frac{7}{4}T_u)$ with $T_u = C_u\sqrt{\frac{s}{n}\log\frac{pq}{\delta}}$, then with probability at least $1-\delta$, we have $\mathrm{supp}(T_\theta(\widehat{\mathbf{u}}_k)) = \mathrm{supp}(\mathbf{u}_k^*)$;

b) If the threshold $\theta \in (\frac{5}{4}T_v, \frac{7}{4}T_v)$ with $T_v = C_v\sqrt{\frac{qs}{n}\log\frac{pq}{\delta}}$, then with probability at least $1-\delta$, we have $\mathrm{supp}(T_\theta(\widehat{\mathbf{v}}_k)) = \mathrm{supp}(\mathbf{v}_k^*)$.

Theorem 4 shows that both supports of the left and right singular vectors can be accurately recovered with properly chosen tuning parameter θ. Together with the correctly identified true rank r^*, the above results indeed yield consistent selection of both predictors and responses. In practice, this threshold θ can be tuned by criteria such as cross-validation.

Besides the statistical properties established before, the proposed procedure SEED enjoys great flexibility in the sense that it does not rely on exact eigenvalue decomposition and the perturbation errors in the generalized eigenvalue problem (5) will be linearly incorporated into the estimated singular vectors $\widehat{\mathbf{u}}_k$ and $\widehat{\mathbf{v}}_k$. Furthermore, our analysis does not reply on the positive definiteness of the Gram matrix (Chen et al., 2012) in high dimensions.

5. Numerical Studies

In this section, we conduct experiments on three data sets, including two simulation data sets (one for a medium-scale experiment and one for a large-scale experiment) and one application data set in social media analysis, to examine the empirical performance of SEED.

5.1. Simulation Studies

5.1.1. Simulation Example 1

We generate a medium-scale synthetic data set as follows: the predictors $\mathbf{x}$ are drawn from a multivariate Gaussian distribution as $\mathbf{x} \sim \mathcal{N}(\mathbf{0}_{p\times 1}, \mathbf{\Sigma}_X)$, where $\mathbf{\Sigma}_X$ is the $p \times p$ covariance matrix with auto-regressive structure, that is, $\Sigma_{X,i,j} = \rho^{|i-j|}$ for some $0 < \rho < 1$ which will be specified later. The responses $\mathbf{y}$ are drawn according to conditional distribution $\mathbf{y}|\mathbf{x} \sim \mathcal{N}(\mathbf{C}^\top \mathbf{x}, \gamma\mathbf{\Sigma}_E)$ where the noise covariance matrix $\mathbf{\Sigma}_E$ is also selected to have the autoregressive structure with $\rho = 0.5$ and we set $\gamma = 0.3$. We generate the parameter matrix $\mathbf{C}$ as follows: first we generate a block-sparse matrix $\widetilde{\mathbf{C}}$ with 5% non-zero elements. Each non-zero element of $\widetilde{\mathbf{C}}$ is drawn from a $\mathcal{N}(0, 1)$. To achieve a low-rank structure, we find the top-r singular value decomposition of $\widetilde{\mathbf{C}}$ as $\widetilde{\mathbf{C}} = \mathbf{USV}^T$, and then set the elements of $\mathbf{U}$ and $\mathbf{V}$ whose magnitude is smaller than 0.01 to zero to obtain $\bar{\mathbf{U}}$ and $\bar{\mathbf{V}}$. The final parameter matrix is obtained as $\mathbf{C} = \bar{\mathbf{U}}\bar{\mathbf{S}}\bar{\mathbf{V}}^T$ where the first r diagonal elements of the diagonal matrix $\bar{\mathbf{S}}$ are set to $100, 99, \ldots, 101 - r$. Without loss of generality, we add a few vectors to the design matrix to ensure the orthogonality condition of Section 4.1. In all of the simulation experiments, we generate 100 data sets and report the mean and standard error of performance for different methods.

We compare the performance of SEED with two state-of-art methods: (1) RCGL (Bunea et al., 2012) and (2) Penalized regression with simultaneous L_1 and nuclear-norm penalization. The optimization problem is solved by the popular alternating direction method of multipliers (Boyd et al., 2010) and we will refer to this baseline as the "ADMM" algorithm. For a fair comparison, all model parameters are set based on a separate validation set with size $n_{\text{valid}} = 500$. To tune the parameters in SEED, we created a grid of sparsity thresholds θ and for each value of θ, the validation errors were recorded while increasing the rank of the solution matrices. The robustness of sparsity threshold θ and termination parameter μ will also be analyzed.

The quality of the estimator $\widehat{\mathbf{C}}$ is evaluated via four performance metrics listed as follows. (1) *Normalized Prediction Error* defined as

$$\text{Normalized Prediction Error} = \frac{\|\mathbf{Y}_{\text{test}} - \mathbf{X}_{\text{test}}\widehat{\mathbf{C}}\|_F}{\|\mathbf{Y}_{\text{test}}\|_F}.$$

(2) *Normalized Parameter Estimation Error* defined as

$$\text{Normalized Parameter Estimation Error} = \frac{\|\widehat{\mathbf{C}} - \mathbf{C}\|_F}{\|\mathbf{C}\|_F},$$

where $\mathbf{C}$ is the true parameter matrix.

(3) *Rank Recovery Error* defined as

$$\text{Rank Recovery Error} = |\text{rank}(\widehat{\mathbf{C}}) - \text{rank}(\mathbf{C})|.$$

Since the solution of the nuclear norm always leads to small non-zero singular values (which prevents $\widehat{\mathbf{C}}$ from being low-rank), we threshold the singular values of $\widehat{\mathbf{C}}$ that are more than 100 times smaller than its largest singular value to have a fair comparison.

(4) *Support Recovery AUC*, that is, the area under the receiver operating characteristic (ROC) curve of comparing support of $\widehat{\mathbf{C}}$ with the ground truth, which is always between 0 and 1. It is computed by varying the decision threshold and obtaining the false positive and true positive curve. Then the area under the false positive and true positive curve is reported as AUC. The value of AUC indicates the probability that a procedure assigns a higher value to a randomly chosen non-zero element than a randomly chosen zero element (Hanley and McNeil, 1982). It is an appropriate metric for measuring support recovery accuracy because in sparse support recovery we have more zeros than non-zeros which inflates the result of the simple 0-1 accuracy measure. In contrast, AUC is more robust to imbalanced positive/negative prediction labels.

Table 1 shows the results of all algorithms on a variety of regimes by varying the dimensionality p and the rank r. We can see that SEED is superior or comparable to the baseline algorithms across all four measures. As the results show and the theory predicts, in most high-dimensional cases, nuclear norm usually overestimates the true rank of the matrix. Furthermore, we find that the iterative SVD procedure in the RCGL algorithm often results in significant underestimation of the true rank, when the true rank is large. Note that in addition to accuracy, SEED also significantly reduces the variance of the estimation.

Figure 1a shows the solution path for SEED on one example data set ($p = 400$, $r = 5$, $q = 200$, $\rho = 0.5$, and $n = 100$). The corresponding singular values are set to $30, 27, 24, 21$, and 18. In the horizontal axis, we show the termination parameter μ normalized by $\|\mathbf{Y}\|_F^2/(nq)$. We can see that SEED can identify the correct rank with medium values of μ. Figure 1b shows the solution path for the top left singular vector $\mathbf{u}_1$ of $\mathbf{C}$ on an example data set ($p = 200$, $q = 100$, $n = 50$, $\rho = 0.5$, and $r = 1$). Both of the solution paths indicate that SEED is robust to the particular choice of parameters and in a large range of parameters SEED is able to successfully recover the true rank of the matrix and the support of the singular vectors.

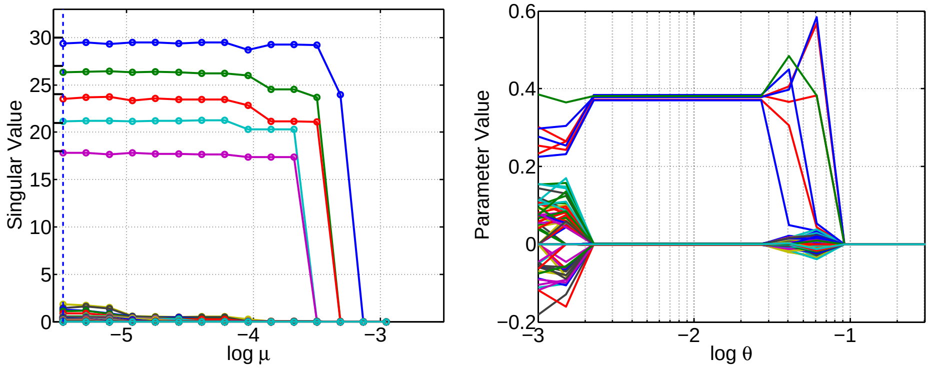
5.1.2. Simulation Example 2

In order to study scalability of SEED, we conduct the experiments on two computing environment, including: (1) an off-the-shelf personal computer (PC) and (2) a graphics processing unit (GPU), to demonstrate the runtime efficiency and the parallelization capability of SEED.

First, we run our experiments on an off-the-shelf PC with Intel i7 at 3.4GHz and 8GB of memory. The system runs MATLAB R2013b on the Windows operating system. We generate 5 data sets with $r = 1$, non-zero ratio of 10%, $q = 1000$, and $n = 1000$. Figure 2a shows the average CPU runtime of three algorithms as the dimension p increases. We can see that SEED can achieve a speed up of 10-100 times in runtime compared with baseline methods.

p	Algorithm	Normalized Prediction Error	Normalized Estimation Error	Rank Recovery Error	Support Recovery AUC
$n = 100$, $q = 200$, $\mathbf{r} = \mathbf{3}$, and $\rho = 0.5$					
100	SEED	**0.039 (0.013)**	**0.012 (0.006)**	**0.06 (0.445)**	**0.980 (0.024)**
	ADMM	0.041 (0.012)	0.015 (0.004)	0.01 (0.100)	0.975 (0.026)
	RCGL	0.053 (0.013)	0.040 (0.009)	0.03 (0.171)	0.979 (0.028)
400	SEED	**0.026 (0.005)**	**0.011 (0.002)**	**0 (0)**	**0.970 (0.015)**
	ADMM	0.035 (0.007)	0.035 (0.008)	0.06 (0.278)	0.961 (0.018)
	RCGL	0.158 (0.050)	0.193 (0.066)	0 (0)	0.934 (0.027)
800	SEED	**0.019 (0.003)**	**0.010 (0.002)**	**0 (0)**	**0.970 (0.014)**
	ADMM	0.076 (0.020)	0.127 (0.031)	4.14 (1.614)	0.954 (0.020)
	RCGL	0.482 (0.072)	0.603 (0.074)	0.01 (0.100)	0.834 (0.030)
1500	SEED	**0.015 (0.002)**	**0.010 (0.002)**	**0 (0)**	**0.971 (0.011)**
	ADMM	0.072 (0.015)	0.145 (0.024)	3.02 (0.141)	0.968 (0.010)
	RCGL	0.698 (0.054)	0.820 (0.022)	0.09 (0.379)	0.809 (0.047)
2000	SEED	**0.014 (0.001)**	**0.010 (0.002)**	**0 (0)**	**0.973 (0.011)**
	ADMM	0.066 (0.010)	0.136 (0.016)	3 (0)	0.970 (0.011)
	RCGL	0.746 (0.041)	0.857 (0.021)	0.2 (0.603)	0.789 (0.045)
$n = 100$, $q = 200$, $\mathbf{r} = \mathbf{30}$, and $\rho = 0.5$					
100	SEED	**0.010 (0.001)**	**0.043 (0.003)**	**0.29 (0.456)**	**0.989 (0.003)**
	ADMM	0.015 (0.002)	0.066 (0.006)	48.51 (0.882)	0.964 (0.027)
	RCGL	0.058 (0.010)	0.220 (0.017)	0.04 (0.243)	0.774 (0.064)
400	SEED	**0.035 (0.018)**	**0.155 (0.082)**	**1.17 (1.092)**	**0.770 (0.023)**
	ADMM	0.037 (0.002)	0.139 (0.007)	42.34 (3.085)	0.783 (0.018)
	RCGL	0.395 (0.029)	0.558 (0.023)	0.02 (0.200)	0.610 (0.010)
800	SEED	**0.003 (0.000)**	**0.005 (0.000)**	**0.02 (0.141)**	**0.999 (0.000)**
	ADMM	0.014 (0.003)	0.022 (0.004)	9.34 (2.388)	0.999 (0.001)
	RCGL	0.286 (0.037)	0.429 (0.020)	0.13 (0.733)	0.966 (0.004)
1500	SEED	**0.003 (0.000)**	**0.007 (0.000)**	**0 (0)**	**0.999 (0.000)**
	ADMM	0.021 (0.001)	0.088 (0.004)	23.61 (1.348)	0.999 (0.000)
	RCGL	0.315 (0.024)	0.457 (0.017)	2.31 (3.243)	0.970 (0.002)
2000	SEED	**0.002 (0.000)**	**0.007 (0.000)**	**0 (0)**	**0.999 (0.000)**
	ADMM	0.020 (0.001)	0.091 (0.003)	22.54 (1.275)	0.999 (0.000)
	RCGL	0.335 (0.019)	0.472 (0.012)	3.10 (3.538)	0.974 (0.002)

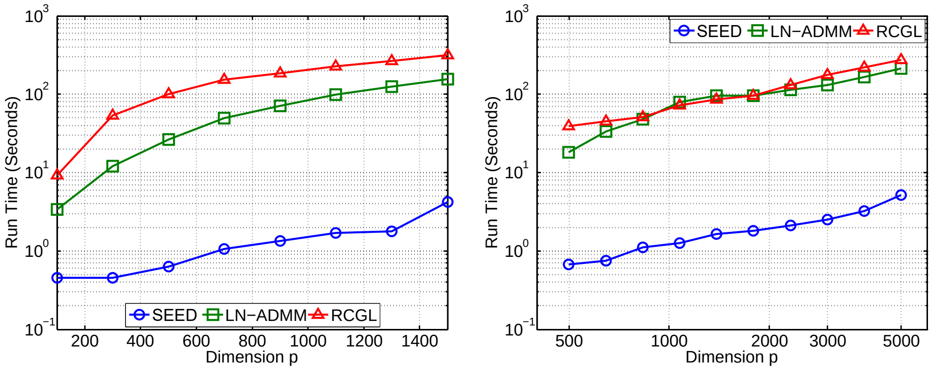
Table 1: Simulation Results



(a) Rank

(b) Support

Figure 1: (a) Solution path for the singular values of the estimated matrices. The plot show the value of top five singular values of the solution $\widehat{\mathbf{C}}$ as we change the stopping error μ. (b) Solution path for the top left singular vector $\mathbf{u}$ of the estimated matrices. Only seven coefficients are non-zero. The range of the parameters are generated as follows: $\mu = \text{logspace}(-5, -1, 5)$ and $\theta = \text{logspace}(-1, \log_{10}(20), 10)$, where $\text{logspace}(a, b, n)$ indicates the minimum value 10^a, maximum value 10^b, and total number n.



(a) CPU

(b) GPU

Figure 2: Speedup by SEED on (a) CPU and (b) GPU devices. Note that the vertical axis is in logarithmic scale.

Next, in order to test scalability of SEED in extremely large data sets, we use a machine that is equipped with a Tesla K40 GPU which has 2880 processing cores at 745MHz and 12GB of memory. We perform our experiments with MATLAB R2013b on a Debian Linux operating system. GPUs are built to have many less-powerful processing units which makes them ideal for parallel implementation (Bekkerman et al., 2012, Chapter 5). Therefore we apply the two-step fast eigenvalue decomposition described in Section 3, which involves only simple matrix operation and can be paralleled easily. The experiment results shown in

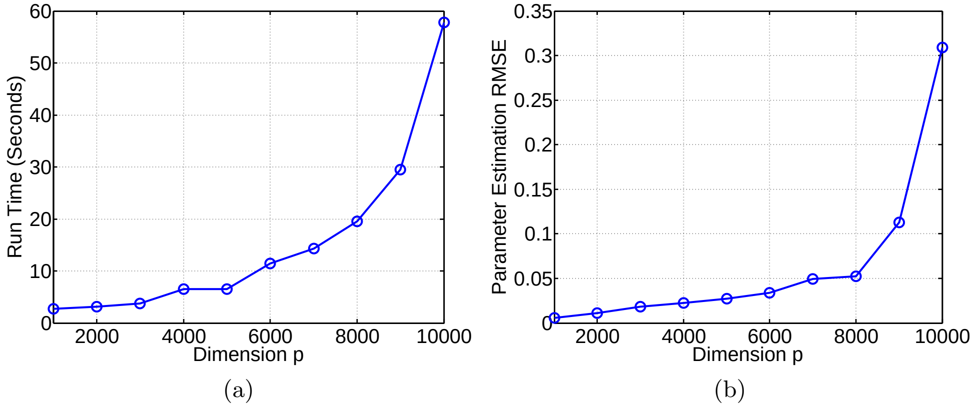


Figure 3: Scalability experiments on very large data sets on GPU using the fast approach. Accuracy results are normalized.

Figures 2b and 3 are obtained under the setting of $q = 10000$, $n = 5000$, $r = 1$ and non-zero ratio of 10%. The results indicate that while SEED is fast on the GPU, it also achieves reasonable accuracy. Note that the results show that SEED is able to estimate a sparse and low-rank matrix with 10^8 elements in less than a minute, confirming its extreme scalability.

5.2. Real Data Analysis

Diffusion Network Inference, that is, the task of inferring influence networks from user activities, is one of the central tasks in social networks analysis (Gomez-Rodriguez et al., 2012) because it helps improve social marketing by finding the influential users in a network. It is a challenging problem because: (i) in many social networks the influence is expressed implicitly (Gomez-Rodriguez et al., 2012) and (ii) empirical studies show that common metrics such as number of friends or followers fail to accurately measure the social influence of the users (Cha et al., 2010).

A popular computational approach in estimating social influence among users is to count the number of users' activities over a time span (in regularly or irregularly spaced intervals) and analyze the resulting time series data (Truccolo et al., 2005). Many different models have been developed, among which the vector auto-regressive model arises as a simple and robust solution (Trusov et al., 2009; Bahadori and Liu, 2013). That is, every user i is described by a time series $x_i(t)$ for $t = 1, \ldots, T$ and $i = 1, \ldots, q$, such that

$$x_i(t) = \sum_{j=1}^{q} \boldsymbol{\beta}_{i,j}^T \mathbf{x}_j^{t,Lagged} + \varepsilon_i(t),$$

where $\boldsymbol{\beta}_{i,j}$ is the vector of coefficients modeling the effects of time series x_j, $\mathbf{x}_j^{t,Lagged} = [x_j(t-L), \ldots, x_j(t-1)]^T$ is the history of x_j up to time t with L the maximal time lag, and $\varepsilon_i(t)$ is the random noise at time t. Denoting by $\mathbf{x}(t) = [x_1(t), \ldots, x_q(t)]^T$, we have the following multi-response regression model:

$$\mathbf{x}(t) = \mathbf{B}^T \mathbf{x}^{t,Lagged} + \boldsymbol{\varepsilon}(t),$$

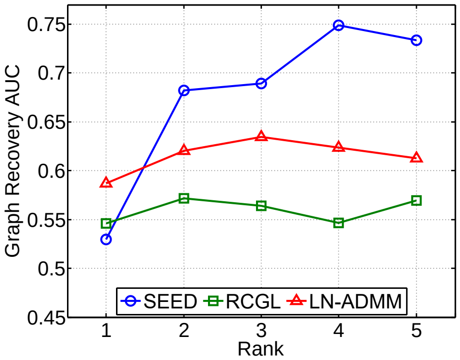


Figure 4: The graph recovery accuracy of the algorithms as the rank of solution varies.

where the predictor vector $\mathbf{x}^{t,Lagged} = [(\mathbf{x}_1^{t,Lagged})^T, \dots, (\mathbf{x}_q^{t,Lagged})^T]^T$, $\mathbf{B}^T = \big(\boldsymbol{\beta}_{i,j}^T\big)_{1\le i,j\le q}$ is the evolution matrix, and $\boldsymbol{\varepsilon}(t) = [\varepsilon_1(t), \dots, \varepsilon_q(t)]^T$. The influence network can be built from the evolution matrix by establishing an edge from node j to node i if $\|\boldsymbol{\beta}_{i,j}\|_1$ is significantly larger than zero.

In this experiment, we gather a Twitter data set with tweets on the "Haiti earthquake" and apply vector auto-regressive model to identify the potential top influencers on this topic (that is, those Twitter accounts with the largest impact on the others). We divide the 17 days after the Haiti Earthquake on Jan. 12, 2010 into 1000 uniformly spaced intervals and generate a multivariate time series data set by counting the number of tweets on this topic for the top 1000 users who tweeted most about it. For accurate modeling, we remove the users that were highly correlated with each other, most of which were operated by the same users and tweeted exactly the same content. We also remove robot-like user-accounts which tweeted on very regular intervals, which in total led to a subset of 270 users. We analyze this data by a VAR model with the maximal time lag $L = 5$ based on the intuition about the maximum retweeting delay, which requires estimation of a $q = 270$ dimensional response vector using $p = 1350$ predictors while we have only $n = 995$ observations.

Since we do not have access to the true influence network, we use the retweet network as a surrogate of the ground truth following the evaluation convention in the social networks community. The retweet network is constructed by adding an edge from user i to user j if user j has retweeted at least 4 of the tweets of user i. Clearly, the retweet network is not the actual underlying temporal dependency graph, mainly because there are possible implicit influence patterns as well. However, it is the best possible metric that we could obtain for graph estimation accuracy evaluation in our data set (Cha et al., 2010). The retweet network for the 270 selected users is sparse; it has only 0.11% of possible edges.

We apply SEED, ADMM, and RCGL algorithms to uncover the influence network in our twitter data set. Figure 4 shows the accuracy of the procedures in uncovering the true influence network in terms of AUC. For every value of the rank parameter, we tune the sparsity by 5-fold cross-validation. Given the fact that exact rank constraint cannot be enforced directly in the ADMM algorithm, we find the best value of the nuclear norm regularization parameter λ_L by 5-fold cross-validation. Then, we compute the low-rank approximations of the parameter matrix and evaluate the accuracy at each rank.

SEED	ADMM	RCGL
2.83	127.34	256.87

Table 2: Run time (in seconds) of the algorithms on the application data set.

The results in Figure 4 show that SEED significantly outperforms the baseline procedures. They also indicate that, in all of the algorithms, as we increase the rank of the solution matrix, the accuracy is improved initially and then quickly saturates. SEED achieves the highest accuracy when the rank is 4. Note that this result also confirms other studies that the social network connections may be strongly influenced by a few unobserved exogenous variables (Myers et al., 2012). The results in Table 2 demonstrate the significant speedup achieved by SEED compared to the baselines.

6. Discussion

In this paper, we propose to convert the problem of sparse reduced-rank regression into a sparse generalized eigenvalue problem, which allows us to efficiently employ the recently developed sparse eigenvalue decomposition techniques. After this transformation, the left singular vectors can be estimated in simple steps and the estimation of both sparse and dense right singular vectors is unified in a single framework. As a pure learning algorithm, SEED deviates from traditional regularization frameworks (that is, a loss function plus certain penalties), leading to computational efficiency and scalability. Furthermore, we prove that SEED achieves nice estimation and prediction accuracy similar to the minimax error bound in the univariate regression setting (Raskutti et al., 2011).

Some interesting problems for future research include extending the current formulation of the regression coefficient matrix in (2) to the case where the singular values can be repeated such that the left singular vectors (which correspond to latent factors) are not identifiable. Then we will need to estimate the eigenspaces spanned by important singular vectors and characterize the estimation accuracy by some new criterion, such as the one in Cai et al. (2013) and Ma (2013). Another research direction is to explore the theory of random design matrices and this can be addressed by using an extended version of perturbation theory (Lemma 6), where the perturbation in $\mathbf{P}$ is also included in the analysis.

Moreover, it is computationally straightforward to extend SEED to the generalized linear models by adapting the sequential quadratic programming framework. For this extension, we first approximate the loss function by the quadratic loss function and find the optimal unit rank matrix. Then we can add the unit rank matrix to the solution and re-approximate the loss function with another quadratic function around this new solution. By performing these three steps sequentially, we can efficiently estimate the low-rank coefficient matrix. Statistical properties of such estimator can be analyzed by extending the results in Lozano et al. (2011) for greedy sparse procedures to reduced-rank regression.

Acknowledgments

We would like to acknowledge support for this project from NSF CAREER Awards DMS-0955316 and IIS-1254206, NSFC-11601501, 11671374 and 71731010, Anhui Provincial Natural Science Foundation-1708085QA02, Fundamental Research Funds for Central Universities-WK2040160028, Okawa Foundation Research Award, a grant from the Simons Foundation, and Adobe Data Science Research Award. The authors would like to thank Emre Demirkaya and Sanjay Purushotham for helpful discussions. The authors also sincerely thank the Action Editor and referees for their valuable comments that helped improve the article substantially.

Appendix A. Proofs of Theorems 1 and 2

We need the following two lemmas in the proofs of Theorems 1 and 2.

Lemma 5 *Suppose $\boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}_q, \boldsymbol{\Sigma})$ and $\gamma^2 = \max_j \boldsymbol{\Sigma}_{jj}$. We have stacked n realization of these random vectors in the rows of $n \times q$ matrix $\mathbf{E}$. Denote by $P^2 = \max_j [\frac{1}{n}\mathbf{X}^T\mathbf{X}]_{jj}$ given a deterministic $n \times p$ matrix $\mathbf{X}$. Then for any $\delta \in (0,1)$, with probability at least $1-\delta$,*

$$\frac{1}{n}\|\mathbf{E}^T\mathbf{X}\|_{2,s} \leq \sqrt{\frac{2qs(\gamma P)^2}{n}\log\frac{2pq}{\delta}}.$$

Lemma 6 *(Perturbation Theory for Sparse Symmetric Generalized Eigenvalue Problems) Suppose $\mathbf{Q}, \mathbf{P} \in \mathbb{S}_p^+$ are two $p \times p$ semi-definite matrices. For the following generalized eigenvalue problem and its perturbed variant with sparse eigenvectors*

$$\mathbf{Q}\mathbf{u} = \lambda\mathbf{P}\mathbf{u}, \tag{10}$$

$$(\mathbf{Q} + \boldsymbol{\delta}_Q)\widehat{\mathbf{u}} = \widehat{\lambda}\mathbf{P}\widehat{\mathbf{u}}, \tag{11}$$

where $\mathbf{u}, \widehat{\mathbf{u}} \perp \mathrm{Ker}(\mathbf{P})$ with $\|\mathbf{u}\|_0 \leq s^*$, $\|\widehat{\mathbf{u}}\|_0 \leq s$ and $s^* < s$, under Condition 1, we have uniformly over $k = 1, \cdots, p - \dim\{\mathrm{Ker}(\mathbf{P})\}$,

$$|\widehat{\lambda}_k - \lambda_k| \leq \phi_s^{-1}\|\boldsymbol{\delta}_Q\|_{2,s}. \tag{12}$$

Here λ_k and $\widehat{\lambda}_k$ are the kth largest eigenvalues of equations (10) and (11), respectively, $\|\boldsymbol{\delta}_Q\|_{2,s}$ denotes the s-sparse largest singular value of $\boldsymbol{\delta}_Q$, and ϕ_s is defined in Condition 1.

Furthermore, denote by $\mathbf{u}_k$ and $\widehat{\mathbf{u}}_k$ the unit length sparse eigenvectors correspond to λ_k and $\widehat{\lambda}_k$, respectively, with $\widehat{\mathbf{u}}_k$ taking the correct directions. When there exists some positive eigengap d_λ which is the minimum difference between non-zero eigenvalues of equation (10) and the perturbation of $\mathbf{Q}$ satisfies $\|\boldsymbol{\delta}_Q\|_{2,s} = o(d_\lambda)$, then uniformly over k such that $\lambda_k \neq 0$,

$$\|\widehat{\mathbf{u}}_k - \mathbf{u}_k\|_2 \leq \sqrt{2}\phi_s^{-2}d_\lambda^{-1}\|\boldsymbol{\delta}_Q\|_{2,s} + o(d_\lambda^{-1}\|\boldsymbol{\delta}_Q\|_{2,s}). \tag{13}$$

A.1. Proof of Theorem 1

The proof is established based on analyzing the impact of perturbation of the matrices in the solution of equation (3). To do so, in the first step, we bound the amount of perturbation in the matrix of the generalized eigenvalue problem in Proposition 1. In the second step, using the eigenvalue perturbation theory (Lemma 6), we derive the error bound for $\widehat{\mathbf{u}}_k$. Using the bound for $\widehat{\mathbf{u}}_k$, in the third step, the error bound for $\widehat{\mathbf{v}}_k$ is calculated.

Step 1: Bounding the perturbation of matrix $\mathbf{Q}$. For notational simplicity, define the noise free response variables as $\mathbf{Y}^* = \mathbf{X}\mathbf{C}^*$, $\mathbf{Q} = \frac{1}{qn^2}\mathbf{X}^T\mathbf{Y}^*\mathbf{Y}^{*T}\mathbf{X}$ and its noisy version $\widehat{\mathbf{Q}} = \frac{1}{qn^2}\mathbf{X}^T\mathbf{Y}\mathbf{Y}^T\mathbf{X}$. Denote by $\|\widehat{\mathbf{Q}} - \mathbf{Q}\|_{2,s}$ the s-sparse largest singular value of $\widehat{\mathbf{Q}} - \mathbf{Q}$. Since $\mathbf{Y} = \mathbf{Y}^* + \mathbf{E}$, we can derive the following bound,

$$\|\widehat{\mathbf{Q}} - \mathbf{Q}\|_{2,s} = \frac{1}{qn^2}\left\|\mathbf{X}^T\mathbf{Y}\mathbf{Y}^T\mathbf{X} - \mathbf{X}^T\mathbf{Y}^*\mathbf{Y}^{*T}\mathbf{X}\right\|_{2,s} \le \frac{2}{n^2q}\left\|\mathbf{X}^T\mathbf{Y}^*\mathbf{E}^T\mathbf{X}\right\|_{2,s} + \frac{1}{n^2q}\|\mathbf{E}^T\mathbf{X}\|_{2,s}^2,$$

where the last inequality is due to the expansion of $\mathbf{X}^T\mathbf{Y}\mathbf{Y}^T\mathbf{X}$ and an application of the triangular inequality.

By Condition 1 and Proposition 1, we have $\|\mathbf{X}\|_{2,s} \le \sqrt{n}\phi_s^{-1/2}$ and

$$\|\mathbf{X}^T\mathbf{Y}^*\|_2 \le \|\mathbf{X}\|_{2,s} \cdot \|\mathbf{Y}^*\|_2 = \|\mathbf{X}\|_{2,s} \cdot \sqrt{nq}\sigma_1 \le n\sqrt{q}\sigma_1\phi_s^{-1/2}, \tag{14}$$

where the first inequality holds since the unit length vector $\mathbf{v}$ satisfying $\|\mathbf{X}^T\mathbf{Y}^*\mathbf{v}\|_2 = \|\mathbf{X}^T\mathbf{Y}^*\|_2$ must be one of the $\mathbf{v}_k^*$ in (2) due to the strict separation of the singular values by Condition 2, so that $\mathbf{X}^T\mathbf{Y}^*\mathbf{v}_k^* = \mathbf{X}^T\mathbf{X}\mathbf{C}^*\mathbf{v}_k^*$ with $\mathbf{C}^*\mathbf{v}_k^*$ yielding an s-sparse vector for $\mathbf{X}^T\mathbf{X}$. Therefore, we get

$$\frac{2}{n^2q}\left\|\mathbf{X}^T\mathbf{Y}^*\mathbf{E}^T\mathbf{X}\right\|_{2,s} \le \frac{2}{n^2q}\left\|\mathbf{X}^T\mathbf{Y}^*\right\|_2 \cdot \left\|\mathbf{E}^T\mathbf{X}\right\|_{2,s} \le \frac{2\sigma_1\phi_s^{-1/2}}{n\sqrt{q}}\left\|\mathbf{E}^T\mathbf{X}\right\|_{2,s}.$$

Let $a^* = \sigma_1\phi_s^{-1/2}$. It follows that

$$\|\widehat{\mathbf{Q}} - \mathbf{Q}\|_{2,s} \le \frac{2a^*}{n\sqrt{q}}\|\mathbf{E}^T\mathbf{X}\|_{2,s} + \frac{1}{n^2q}\|\mathbf{E}^T\mathbf{X}\|_{2,s}^2. \tag{15}$$

Step 2: Error bounds for $\widehat{\mathbf{u}}_k$ *and* $\mathbf{X}\widehat{\mathbf{u}}_k$. Using Lemma 5, for any $\delta \in (0,1)$ with probability at least $1-\delta$, we have

$$\frac{1}{n\sqrt{q}}\|\mathbf{E}^T\mathbf{X}\|_{2,s} \le \gamma P\sqrt{\frac{2s}{n}\log\frac{2pq}{\delta}}.$$

Since $\frac{s}{n}\log\frac{pq}{\delta} \to 0$, substituting the above bound in equation (15) yields

$$\|\widehat{\mathbf{Q}} - \mathbf{Q}\|_{2,s} \le \gamma P a^*\sqrt{\frac{8s}{n}\log\frac{2pq}{\delta}} + o\left(\sqrt{\frac{s}{n}\log\frac{pq}{\delta}}\right). \tag{16}$$

Therefore, under Conditions 1 and 2, applying Lemma 6 with $\widehat{\mathbf{Q}} = \frac{1}{qn^2}\mathbf{X}^T\mathbf{Y}\mathbf{Y}^T\mathbf{X}$, $\mathbf{Q} = \frac{1}{qn^2}\mathbf{X}^T\mathbf{Y}^*\mathbf{Y}^{*T}\mathbf{X}$, and $\mathbf{P} = \frac{1}{n}\mathbf{X}^T\mathbf{X}$ gives the following bound for $\widehat{\mathbf{u}}_k$,

$$\|\widehat{\mathbf{u}}_k - \mathbf{u}_k^*\|_2 \le \frac{4\gamma P a^*}{d_\sigma\phi_s^2}\sqrt{\frac{s}{n}\log\frac{pq}{\delta}} + o\left(\sqrt{\frac{s}{n}\log\frac{pq}{\delta}}\right) = C_u\sqrt{\frac{s}{n}\log\frac{pq}{\delta}} + o\left(\sqrt{\frac{s}{n}\log\frac{pq}{\delta}}\right),$$

where $C_u = \frac{4\gamma P a^*}{d_\sigma \phi_s^2} = \frac{4\gamma P \sigma_1}{d_\sigma \phi_s^{5/2}}$ is a constant. Since $\|\widehat{\mathbf{u}}_k - \mathbf{u}_k\|_0 \leq 2s$, further applying Condition 1 yields

$$\frac{1}{\sqrt{n}} \|\mathbf{X}(\widehat{\mathbf{u}}_k - \mathbf{u}_k^*)\|_2 \leq \phi_s^{-1/2} C_u \sqrt{\frac{s}{n} \log \frac{pq}{\delta}} + o\left(\sqrt{\frac{s}{n} \log \frac{pq}{\delta}}\right).$$

Step 3: Error bound for $\widehat{\mathbf{v}}_k$*.* With the estimated left singular vector $\widehat{\mathbf{u}}_k$, we can further estimate the corresponding right singular vector as

$$\widehat{\mathbf{v}}_k = \frac{1}{\widehat{\mathbf{u}}_k^\top \mathbf{X}^T \mathbf{X} \widehat{\mathbf{u}}_k} \mathbf{Y}^T \mathbf{X} \widehat{\mathbf{u}}_k,$$

and compare it to $\mathbf{v}_k^*$, which by Proposition 1 can be expressed as

$$\mathbf{v}_k^* = \frac{1}{\mathbf{u}_k^{*\top} \mathbf{X}^T \mathbf{X} \mathbf{u}_k^*} \mathbf{Y}^{*T} \mathbf{X} \mathbf{u}_k^*.$$

For simplicity of notation, we drop the index k of $\mathbf{v}_k$ and $\mathbf{u}_k$ and write

$$\|\widehat{\mathbf{v}} - \mathbf{v}^*\|_2 = \left\| \frac{1}{\mathbf{u}^{*\top} \mathbf{X}^T \mathbf{X} \mathbf{u}^* - e_1} \left(\mathbf{Y}^{*T} \mathbf{X} \mathbf{u}^* - \mathbf{e}_2 \right) - \frac{1}{\mathbf{u}^{*\top} \mathbf{X}^T \mathbf{X} \mathbf{u}^*} \mathbf{Y}^{*T} \mathbf{X} \mathbf{u}^* \right\|_2,$$

where $e_1 \triangleq \mathbf{u}^{*\top} \mathbf{X}^T \mathbf{X} \mathbf{u}^* - \widehat{\mathbf{u}}^\top \mathbf{X}^T \mathbf{X} \widehat{\mathbf{u}}$ and $\mathbf{e}_2 \triangleq \mathbf{Y}^{*T} \mathbf{X} \mathbf{u}^* - \mathbf{Y}^T \mathbf{X} \widehat{\mathbf{u}}$. Using the Taylor expansion of $\frac{1}{1-x}$ at $x = 0$, we can write

$$\begin{aligned}
\|\widehat{\mathbf{v}} - \mathbf{v}^*\|_2 &\leq \left\| \frac{\mathbf{Y}^{*T} \mathbf{X} \mathbf{u}^*}{\mathbf{u}^{*\top} \mathbf{X}^T \mathbf{X} \mathbf{u}^*} \frac{e_1}{\mathbf{u}^{*\top} \mathbf{X}^T \mathbf{X} \mathbf{u}^*} \right\|_2 + \left\| \frac{\mathbf{e}_2}{\mathbf{u}^{*\top} \mathbf{X}^T \mathbf{X} \mathbf{u}^*} \right\|_2 + T_e \\
&= (\mathbf{u}^{*\top} \mathbf{X}^T \mathbf{X} \mathbf{u}^* / n)^{-1} \left(\|\mathbf{v}^*\|_2 |e_1|/n + \|\mathbf{e}_2\|_2/n \right) + T_e \\
&\leq \phi_s^{-1} (\|\mathbf{v}^*\|_2 |e_1|/n + \|\mathbf{e}_2\|_2/n) + T_e, \qquad (17)
\end{aligned}$$

where $T_e = \left(\frac{\|\mathbf{v}^*\|_2 |e_1|/n}{\mathbf{u}^{*\top} \mathbf{P} \mathbf{u}^*} + \frac{\|\mathbf{e}_2\|_2/n}{\mathbf{u}^{*\top} \mathbf{P} \mathbf{u}^*} \right) \sum_{\ell=1}^\infty \left(\frac{|e_1|/n}{\mathbf{u}^{*\top} \mathbf{P} \mathbf{u}^*} \right)^\ell$ denotes the higher order terms in the Taylor expansion and the last step is by Condition 1.

Bounding $|e_1|$*:* Let $\widehat{\mathbf{u}} = \mathbf{u}^* + \boldsymbol{\delta}_{\mathbf{u}}$. Since $\|\mathbf{u}^*\|_0 \leq s^*$, $\|\widehat{\mathbf{u}}\|_0 \leq s$ and $s^* < s$, we have $\|\boldsymbol{\delta}_{\mathbf{u}}\|_0 \leq 2s$. As $\mathbf{u}^*$ is a unit length vector, it yields from Condition 1 that

$$\begin{aligned}
|e_1|/n &= \left| \frac{1}{n} (\mathbf{u}^* + \boldsymbol{\delta}_{\mathbf{u}})^\top \mathbf{X}^T \mathbf{X} (\mathbf{u}^* + \boldsymbol{\delta}_{\mathbf{u}}) - \frac{1}{n} \mathbf{u}^{*\top} \mathbf{X}^T \mathbf{X} \mathbf{u}^* \right| \\
&= |2\mathbf{u}^{*\top} \mathbf{P} \boldsymbol{\delta}_{\mathbf{u}} + \boldsymbol{\delta}_{\mathbf{u}}^T \mathbf{P} \boldsymbol{\delta}_{\mathbf{u}}| \leq \phi_s^{-1} (2\|\boldsymbol{\delta}_{\mathbf{u}}\|_2 + \|\boldsymbol{\delta}_{\mathbf{u}}\|_2^2). \qquad (18)
\end{aligned}$$

Bounding $\|\mathbf{e}_2\|_2$*:* Similarly, let $\mathbf{Y}^T \mathbf{X} = \mathbf{Y}^{*T} \mathbf{X} + \mathbf{E}^T \mathbf{X}$. We obtain

$$\begin{aligned}
\|\mathbf{e}_2\|_2/n &= \|(\mathbf{Y}^{*T} \mathbf{X} + \mathbf{E}^T \mathbf{X})(\mathbf{u}^* + \boldsymbol{\delta}_{\mathbf{u}}) - \mathbf{Y}^{*T} \mathbf{X} \mathbf{u}^*\|_2/n \\
&\leq \|\mathbf{Y}^{*T} \mathbf{X} \boldsymbol{\delta}_{\mathbf{u}}\|_2/n + \|\mathbf{E}^T \mathbf{X} \mathbf{u}^*\|_2/n + \|\mathbf{E}^T \mathbf{X} \boldsymbol{\delta}_{\mathbf{u}}\|_2/n \\
&\leq a^* \sqrt{q} \|\boldsymbol{\delta}_{\mathbf{u}}\|_2 + \|\mathbf{E}^T \mathbf{X}\|_{2,s}/n + \|\mathbf{E}^T \mathbf{X}\|_{2,s} \|\boldsymbol{\delta}_{\mathbf{u}}\|_2/n, \qquad (19)
\end{aligned}$$

where in the last step we used $\|\mathbf{Y}^{*T}\mathbf{X}\boldsymbol{\delta}_{\mathbf{u}}\|_2/n \le a^*\sqrt{q}\|\boldsymbol{\delta}_{\mathbf{u}}\|_2$ similarly as in (14).

Since $\frac{1}{\sqrt{q}}\|\mathbf{v}_k^*\|_2 \le V$, substituting inequalities (18) and (19) in inequality (17) with reorganization yields

$$\begin{aligned}\|\widehat{\mathbf{v}}_k - \mathbf{v}_k^*\|_2 \le & \frac{V\sqrt{q}\|\boldsymbol{\delta}_{\mathbf{u}}\|_2}{\phi_s^2}\big(2+\|\boldsymbol{\delta}_{\mathbf{u}}\|_2\big) + \frac{1}{n\phi_s}\|\mathbf{E}^T\mathbf{X}\|_{2,s}(1+\|\boldsymbol{\delta}_{\mathbf{u}}\|_2)\\ & + \phi_s^{-1}a^*\sqrt{q}\|\boldsymbol{\delta}_{\mathbf{u}}\|_2 + T_e.\end{aligned}$$

When $s\log(pq/\delta) = o(n)$, we can see that both the events $\mathcal{E}_1 = \Big\{\|\boldsymbol{\delta}_{\mathbf{u}}\|_2 = O\big(\sqrt{\frac{s}{n}\log\frac{pq}{\delta}}\big)\Big\}$ and $\mathcal{E}_2 = \Big\{T_e = O\big(\sqrt{\frac{qs}{n}\log\frac{pq}{\delta}}\big)\Big\}$ occur, if the event $\mathcal{E}_0 = \Big\{\frac{1}{n\sqrt{q}}\|\mathbf{E}^T\mathbf{X}\|_{2,s} = O\big(\sqrt{\frac{s}{n}\log\frac{pq}{\delta}}\big)\Big\}$ holds for some $\delta \in (0,1)$, where in the second event we have used the results in inequalities (18) and (19) together with an application of geometric series sum. Note that the first term of T_e is $O\big(\sqrt{\frac{qs}{n}\log\frac{pq}{\delta}}\big)$ and the common ratio is $O\left(\sqrt{\frac{s}{n}\log\frac{pq}{\delta}}\right)$.

Therefore, by applying the result in Lemma 5 for bounding $\|\mathbf{E}^T\mathbf{X}\|_{2,s}/n$ and the result for the estimation error bound of $\|\widehat{\mathbf{u}}_k - \mathbf{u}_k^*\|_2$ in **Step 2**, we can conclude with probability at least $1-\delta$ that

$$\begin{aligned}\|\widehat{\mathbf{v}}_k - \mathbf{v}_k^*\|_2 &\le \frac{(2V\phi_s^{-1}+a^*)C_u + \gamma P}{\phi_s}\sqrt{\frac{2qs}{n}\log\frac{pq}{\delta}} + o\left(\sqrt{\frac{qs}{n}\log\frac{pq}{\delta}}\right) + T_e\\ &\le \frac{2\{(2V\phi_s^{-1}+a^*)C_u + \gamma P\}}{\phi_s}\sqrt{\frac{2qs}{n}\log\frac{pq}{\delta}} + o\left(\sqrt{\frac{qs}{n}\log\frac{pq}{\delta}}\right)\\ &= C_v\sqrt{\frac{qs}{n}\log\frac{pq}{\delta}} + o\left(\sqrt{\frac{qs}{n}\log\frac{pq}{\delta}}\right),\end{aligned}$$

where $C_v = 2\sqrt{2}\phi_s^{-1}\left\{(2V\phi_s^{-1/2}+\sigma_1)\phi_s^{-1/2}C_u + \gamma P\right\}$.

Step 4: Error bounds for $\frac{1}{\sqrt{q}}\|\widehat{\mathbf{C}}_k - \mathbf{C}_k^*\|_F$ *and* $\frac{1}{\sqrt{qn}}\|\mathbf{X}(\widehat{\mathbf{C}}_k - \mathbf{C}_k^*)\|_F$*.* With the estimation error bounds of $\widehat{\mathbf{u}}_k$ and $\widehat{\mathbf{v}}_k$, we write

$$\frac{1}{qn}\|\mathbf{X}(\mathbf{C}_k^* - \widehat{\mathbf{C}}_k)\|_F^2 = \frac{1}{q}\mathrm{trace}\{(\mathbf{C}_k^* - \widehat{\mathbf{C}}_k)^T\mathbf{P}(\mathbf{C}_k^* - \widehat{\mathbf{C}}_k)\} = \tilde{e}_1 + \tilde{e}_2 + \tilde{e}_3, \qquad (20)$$

where the last equality follows from the decomposition

$$\mathbf{C}_k^* - \widehat{\mathbf{C}}_k = \mathbf{u}_k^*\mathbf{v}_k^{*T} - \widehat{\mathbf{u}}_k\widehat{\mathbf{v}}_k^T = (\mathbf{u}_k^* - \widehat{\mathbf{u}}_k)\mathbf{v}_k^{*T} + \widehat{\mathbf{u}}_k(\mathbf{v}_k^* - \widehat{\mathbf{v}}_k)^T \qquad (21)$$

such that

$$\begin{aligned}\tilde{e}_1 &= \frac{1}{q}\mathrm{trace}\{\mathbf{v}_k^*(\mathbf{u}_k^* - \widehat{\mathbf{u}}_k)^T\mathbf{P}(\mathbf{u}_k^* - \widehat{\mathbf{u}}_k)\mathbf{v}_k^{*T}\},\\ \tilde{e}_2 &= \frac{2}{q}\mathrm{trace}\{(\mathbf{v}_k^* - \widehat{\mathbf{v}}_k)\widehat{\mathbf{u}}_j^T\mathbf{P}(\mathbf{u}_k^* - \widehat{\mathbf{u}}_k)\mathbf{v}_k^{*T}\},\\ \tilde{e}_3 &= \frac{1}{q}\mathrm{trace}\{(\mathbf{v}_k^* - \widehat{\mathbf{v}}_k)\widehat{\mathbf{u}}_k^T\mathbf{P}\widehat{\mathbf{u}}_k(\mathbf{v}_k^* - \widehat{\mathbf{v}}_k)^T\}.\end{aligned}$$

We will bound them separately. Since the estimation error bounds of $\widehat{\mathbf{u}}_k$ and $\widehat{\mathbf{v}}_k$, $k = 1, \cdots, r^*$, hold with probability at least $1-\delta$. It yields that

$$\begin{aligned}\tilde{e}_1 &= \frac{1}{q}\text{trace}\{(\mathbf{u}_k^* - \widehat{\mathbf{u}}_k)^T\mathbf{P}(\mathbf{u}_k^* - \widehat{\mathbf{u}}_k)\mathbf{v}_k^{*T}\mathbf{v}_k^*\} = V^2 \cdot (\mathbf{u}_k^* - \widehat{\mathbf{u}}_k)^T\mathbf{P}(\mathbf{u}_k^* - \widehat{\mathbf{u}}_k)\\ &\leq V^2\phi_s^{-1}\|\mathbf{u}_k^* - \widehat{\mathbf{u}}_k\|_2^2 \leq V^2\phi_s^{-1}C_u^2\left(\frac{s}{n}\log\frac{pq}{\delta}\right) + o\left(\frac{s}{n}\log\frac{pq}{\delta}\right),\end{aligned}$$

where the first inequality follows from Condition 1 and the fact $\|\mathbf{u}_k^* - \widehat{\mathbf{u}}_k\|_0 \leq 2s$. Similarly, we have

$$\begin{aligned}\tilde{e}_2 &\leq 2V\phi_s^{-1}C_uC_v\left(\frac{s}{n}\log\frac{pq}{\delta}\right) + o\left(\frac{s}{n}\log\frac{pq}{\delta}\right),\\ \tilde{e}_3 &\leq \phi_s^{-1}C_v^2\left(\frac{s}{n}\log\frac{pq}{\delta}\right) + o\left(\frac{s}{n}\log\frac{pq}{\delta}\right).\end{aligned}$$

In view of (20), the above bounds together give

$$\frac{1}{qn}\|\mathbf{X}(\widehat{\mathbf{C}}_k - \mathbf{C}_k^*)\|_F^2 \leq \phi_s^{-1}(VC_u + C_v)^2\left(\frac{s}{n}\log\frac{pq}{\delta}\right) + o\left(\frac{s}{n}\log\frac{pq}{\delta}\right).$$

Thus we have

$$\frac{1}{\sqrt{qn}}\|\mathbf{X}(\widehat{\mathbf{C}}_k - \mathbf{C}_k^*)\|_F \leq \frac{(VC_u + C_v)}{\phi_s^{1/2}}\left(\sqrt{\frac{s}{n}\log\frac{pq}{\delta}}\right) + o\left(\sqrt{\frac{s}{n}\log\frac{pq}{\delta}}\right).$$

By a similar but simpler argument, it follows from the decomposition (21) that

$$\begin{aligned}\frac{1}{\sqrt{q}}\|\mathbf{C}_k^* - \widehat{\mathbf{C}}_k\|_F &\leq \frac{1}{\sqrt{q}}\|(\mathbf{u}_k^* - \widehat{\mathbf{u}}_k)\mathbf{v}_k^{*T}\|_F + \frac{1}{\sqrt{q}}\|\widehat{\mathbf{u}}_k(\mathbf{v}_k^* - \widehat{\mathbf{v}}_k)^T\|_F\\ &= \frac{1}{\sqrt{q}}\|\mathbf{u}_k^* - \widehat{\mathbf{u}}_k\|_2 \cdot \|\mathbf{v}_k^*\|_2 + \frac{1}{\sqrt{q}}\|\widehat{\mathbf{u}}_k\|_2 \cdot \|\mathbf{v}_k^* - \widehat{\mathbf{v}}_k\|_2\\ &\leq (VC_u + C_v)\left(\sqrt{\frac{s}{n}\log\frac{pq}{\delta}}\right) + o\left(\sqrt{\frac{s}{n}\log\frac{pq}{\delta}}\right).\end{aligned}$$

It concludes the proof.

A.2. Proof of Theorem 2

The main idea for proving the rank consistency result is noting that $\widehat{\sigma}_k^2 = (\widehat{\mathbf{u}}_k^\top\widehat{\mathbf{Q}}\widehat{\mathbf{u}}_k)/(\widehat{\mathbf{u}}_k^\top\mathbf{P}\widehat{\mathbf{u}}_k)$ which relates to the termination criteria is in fact the kth largest eigenvalue of the problem $\widehat{\mathbf{Q}}\mathbf{u} = \lambda\mathbf{P}\mathbf{u}$. Thus, we will show that the amount of decrease in the objective function in the kth step equals to the the kth largest eigenvalue of the problem $\widehat{\mathbf{Q}}\mathbf{u} = \lambda\mathbf{P}\mathbf{u}$. Given the perturbation bounds in Lemma 6, we can bound its difference from the true eigenvalue and show that after r^* greedy steps, the eigenvalues become almost zero.

To prove the result in Theorem 2, we need to bound $\log\mathcal{L}_{k-1} - \log\mathcal{L}_k$, where $\mathcal{L}_k = \frac{1}{nq}\|\mathbf{Y} - \mathbf{X}\mathbf{C}_k\|_F^2$ is the objective function with $\mathbf{C}_k = \sum_{j=1}^k\widehat{\mathbf{C}}_j$ the estimated coefficient matrix up to the kth step. By using the fact that $1 - \frac{1}{x} \leq \log(x) \leq x - 1$ for $x > 0$, we can write

$$\frac{\mathcal{L}_{k-1} - \mathcal{L}_k}{\mathcal{L}_{k-1}} \leq \log\left(\frac{\mathcal{L}_{k-1}}{\mathcal{L}_k}\right) \leq \frac{\mathcal{L}_{k-1} - \mathcal{L}_k}{\mathcal{L}_k}.$$

Next, we show that the amount of the decrease in the objective function in the kth step is equal to the the kth largest eigenvalue of the problem $\widehat{\mathbf{Q}}\mathbf{u} = \lambda \mathbf{P}\mathbf{u}$, which satisfies the following equality

$$\begin{aligned}\mathcal{L}_{k-1} - \mathcal{L}_k &= \frac{1}{nq}\|\mathbf{Y} - \mathbf{X}\mathbf{C}_{k-1}\|_F^2 - \frac{1}{nq}\|\mathbf{Y} - \mathbf{X}\mathbf{C}_k\|_F^2 \\ &= \frac{1}{nq}\|\mathbf{Y} - \mathbf{X}\mathbf{C}_{k-1}\|_F^2 - \frac{1}{nq}\|\mathbf{Y} - \mathbf{X}\mathbf{C}_{k-1} - \mathbf{X}\widehat{\mathbf{u}}_k\widehat{\mathbf{v}}_k^\top\|_F^2 \\ &= \frac{1}{nq}\left(2\left\langle \mathbf{Y} - \mathbf{X}\mathbf{C}_{k-1}, \mathbf{X}\widehat{\mathbf{u}}_k\widehat{\mathbf{v}}_k^\top\right\rangle - \|\mathbf{X}\widehat{\mathbf{u}}_k\widehat{\mathbf{v}}_k^\top\|_F^2\right).\end{aligned}$$

Now, using the $\mathbf{P}$-orthogonality of $\widehat{\mathbf{u}}_k$'s, we have

$$\begin{aligned}\mathcal{L}_{k-1} - \mathcal{L}_k &= \frac{1}{nq}\left(2\left\langle \mathbf{Y}, \mathbf{X}\widehat{\mathbf{u}}_k\widehat{\mathbf{v}}_k^\top\right\rangle - \|\mathbf{X}\widehat{\mathbf{u}}_k\widehat{\mathbf{v}}_k^\top\|_F^2\right) \\ &= \frac{1}{nq}\left(2\frac{\widehat{\mathbf{u}}_k^\top\mathbf{X}^T\mathbf{Y}\mathbf{Y}^T\mathbf{X}\widehat{\mathbf{u}}_k}{\widehat{\mathbf{u}}_k^\top\mathbf{X}^T\mathbf{X}\widehat{\mathbf{u}}_k} - \frac{\widehat{\mathbf{u}}_k^\top\mathbf{X}^T\mathbf{Y}\mathbf{Y}^T\mathbf{X}\widehat{\mathbf{u}}_k}{\widehat{\mathbf{u}}_k^\top\mathbf{X}^T\mathbf{X}\widehat{\mathbf{u}}_k}\right) \\ &= \frac{\widehat{\mathbf{u}}_k^\top\widehat{\mathbf{Q}}\widehat{\mathbf{u}}_k}{\widehat{\mathbf{u}}_k^\top\mathbf{P}\widehat{\mathbf{u}}_k} = \widehat{\sigma}_k^2,\end{aligned}$$

where the second step is due to the substitution of the solution for $\widehat{\mathbf{v}}_k$ in (6) and a few steps of algebraic rearrangement.

Underfitted regime $k \leq r^*$. Under Condition 1, using the perturbation bound in Lemma 6 for the eigenvalues σ_k^2's with $k \leq r^*$, we can write

$$\widehat{\sigma}_k^2 \geq \sigma_k^2 - \phi_s^{-1}\|\boldsymbol{\Delta}_Q\|_{2,s}.$$

Further applying inequality (16), we know that there exists some positive constant C such that

$$\phi_s^{-1}\|\boldsymbol{\Delta}_Q\|_{2,s} \leq 2a^*\phi_s^{-1}\gamma P\sqrt{\frac{2s}{n}\log\frac{pq}{\delta}} + o\left(\sqrt{\frac{s}{n}\log\frac{pq}{\delta}}\right) \leq C\sqrt{\frac{s}{n}\log\frac{pq}{\delta}}.$$

Therefore, it follows that with probability at least $1-\delta$,

$$\widehat{\sigma}_k^2 \geq \sigma_k^2 - C\sqrt{\frac{s}{n}\log\frac{pq}{\delta}}. \tag{22}$$

Then we can derive the following lower bound

$$\sqrt{\frac{\mathcal{L}_{k-1} - \mathcal{L}_k}{\mathcal{L}_{k-1}}} \geq \frac{\sqrt{\sigma_k^2 - C\sqrt{\frac{s}{n}\log\frac{pq}{\delta}}}}{\frac{1}{\sqrt{nq}}\left(\sum_{j=k}^{r^*}\|\mathbf{X}\mathbf{C}_j^*\|_F + \sum_{j=1}^{k-1}\|\mathbf{X}(\mathbf{C}_j^* - \widehat{\mathbf{C}}_j)\|_F + \|\mathbf{E}\|_F\right)}, \tag{23}$$

where $\mathbf{C}_j^* = \mathbf{u}_j^*\mathbf{v}_j^{*T}$, $\widehat{\mathbf{C}}_j = \widehat{\mathbf{u}}_j\widehat{\mathbf{v}}_j$, in the nominator we have used the result obtained in (22) and the denominator is the result of the triangular inequality. We will then bound the three terms in the denominator successively.

For the first term, given the definition of $\mathbf{C}_j^* = \mathbf{u}_j^* \mathbf{v}_j^{*T}$, $\mathbf{v}_j^{*T}\mathbf{v}_j^* = qa_j^2$ and $\mathbf{u}_j^{*T}\mathbf{P}\mathbf{u}_j^* = c_j^2$, we have

$$\frac{\|\mathbf{X}\mathbf{C}_j^*\|_F}{\sqrt{nq}} = \sqrt{\frac{\text{trace}(\mathbf{X}\mathbf{u}_j^*\mathbf{v}_j^{*T}\mathbf{v}_j^*\mathbf{u}_j^{*T}\mathbf{X}^T)}{nq}} = a_j \sqrt{\frac{\mathbf{u}_j^{*T}\mathbf{X}^T\mathbf{X}\mathbf{u}_j^*}{n}} = a_j c_j = \sigma_j.$$

It follows that $\frac{1}{\sqrt{nq}}\sum_{j=k}^{r^*}\|\mathbf{X}\mathbf{C}_j^*\|_F = \sum_{j=k}^{r^*}\sigma_j$.

To bound the second term, by Theorem 1, we have

$$\frac{1}{nq}\|\mathbf{X}(\mathbf{C}_j^* - \widehat{\mathbf{C}}_j)\|_F^2 \le \phi_s^{-1}(VC_u + C_v)^2\left(\frac{s}{n}\log\frac{pq}{\delta}\right) + o\left(\frac{s}{n}\log\frac{pq}{\delta}\right),$$

which gives

$$\sum_{j=1}^{k-1}\frac{\|\mathbf{X}(\mathbf{C}_j^* - \widehat{\mathbf{C}}_j)\|_F}{\sqrt{nq}} \le \frac{(k-1)(VC_u + C_v)}{\phi_s^{1/2}}\sqrt{\frac{s}{n}\log\frac{pq}{\delta}} + o\left(\sqrt{\frac{s}{n}\log\frac{pq}{\delta}}\right). \tag{24}$$

To bound the last term, as the components of $\mathbf{E}\mathbf{\Sigma}^{-1/2}$ are independent and identically distributed with the standard Gaussian distribution, given the tail bound for the χ^2 distribution in (Laurent and Massart, 2000, Lemma 1), we can see that with probability at least $1-\delta$,

$$\|\mathbf{E}\mathbf{\Sigma}^{-1/2}\|_F^2/nq \le 1 + 2\sqrt{\frac{1}{nq}\log\frac{1}{\delta}} + \frac{2}{nq}\log\frac{1}{\delta}.$$

Moreover, by Condition 3, we have

$$\|\mathbf{E}\mathbf{\Sigma}^{-1/2}\|_F^2 = \sum_{i=1}^{n}\|\mathbf{E}_{i:}\mathbf{\Sigma}^{-1/2}\|_2^2 \ge \sum_{i=1}^{n}\|\mathbf{E}_{i:}\|_2^2/\gamma_u^2 = \|\mathbf{E}\|_F^2/\gamma_u^2.$$

It gives

$$\|\mathbf{E}\|_F/\sqrt{nq} \le \gamma_u\left(1 + \sqrt{\frac{2}{nq}\log\frac{1}{\delta}}\ \right). \tag{25}$$

Thus, applying the union bound, we can see that with probability at least $1-2\delta$, the results in Theorem 1 $\big($including inequality (16)$\big)$, inequalities (24) and (25) hold simultaneously. In view of (23), it yields that

$$\sqrt{\frac{\mathcal{L}_{k-1} - \mathcal{L}_k}{\mathcal{L}_{k-1}}} \ge \frac{\sigma_k + O\left(\sqrt{\frac{s}{n}\log\frac{pq}{\delta}}\ \right)}{\sum_{j=k}^{r^*}\sigma_j + \gamma_u + O\left(r^*\sqrt{\frac{s}{n}\log\frac{pq}{\delta}}\ \right)}, \tag{26}$$

where we used the inequality $\sqrt{b-x} \ge \sqrt{b} - x/\sqrt{b}$ for $0 \le x \le b$ in the numerator. Let $\delta^{-1} = O\{(pq)^\alpha\}$ for some positive constant $\alpha > 1$. Since $\sum_{j=k}^{r^*}\sigma_j \le (r^* - k + 1)\sigma_k$,

$\gamma_u \leq c_\gamma \sigma_{r^*} \leq c_\gamma \sigma_k$ by Condition 3 and $r^* \left\{\frac{s\log(pq)}{n}\right\}^{1/4} = o(1)$, we get

$$\sqrt{\frac{\mathcal{L}_{k-1} - \mathcal{L}_k}{\mathcal{L}_{k-1}}} \geq \frac{1}{r^* - k + 1 + c_\gamma} + O\left(r^* \sqrt{\frac{s}{n} \log \frac{pq}{\delta}}\right) \tag{27}$$
$$= \frac{1}{r^* - k + 1 + c_\gamma} + o\left(\frac{1}{r^*}\right).$$

Overfitted regime $k > r^*$. Similar to the previous argument, by Condition 1 and Lemma 6, with probability at least $1 - \delta$, for all $k > r^*$ we have

$$\widehat{\sigma}_k^2 \leq \phi_s^{-1} \|\mathbf{\Delta}_Q\|_{2,s} \leq C\sqrt{\frac{s}{n} \log \frac{pq}{\delta}}. \tag{28}$$

Then we derive the following upper bound

$$\sqrt{\frac{\mathcal{L}_{k-1} - \mathcal{L}_k}{\mathcal{L}_k}} \leq \frac{O\left\{\left(\frac{s}{n} \log \frac{pq}{\delta}\right)^{1/4}\right\}}{\frac{1}{\sqrt{nq}}\left(\|\mathbf{E}\|_F - \sum_{j=1}^{r^*} \|\mathbf{X}(\mathbf{C}_j^* - \widehat{\mathbf{C}}_j)\|_F - \sum_{j=r^*+1}^{k} \|\mathbf{X}\widehat{\mathbf{C}}_j\|_F\right)}. \tag{29}$$

Bounds on the three terms in the denominator will be derived successively. First, another application of the tail bound for the χ^2 distribution in (Laurent and Massart, 2000, Lemma 1) gives that with probability at least $1 - \delta$,

$$\|\mathbf{E}\mathbf{\Sigma}^{-1/2}\|_F^2 / nq \geq 1 - 2\sqrt{\frac{1}{nq} \log \frac{1}{\delta}}.$$

Similarly, together with Condition 3, we have

$$\|\mathbf{E}\|_F / \sqrt{nq} \geq \gamma_l \left(1 - 2\sqrt{\frac{1}{nq} \log \frac{1}{\delta}}\right).$$

Moreover, by inequality (24), we get

$$\sum_{j=1}^{r^*} \frac{1}{\sqrt{nq}} \left\|\mathbf{X}(\mathbf{C}_j^* - \widehat{\mathbf{C}}_j)\right\|_F \leq O\left(r^* \sqrt{\frac{s}{n} \log \frac{pq}{\delta}}\right).$$

For the last term, by the estimation procedure of $\widehat{\mathbf{v}}_j$ in equation (6) and the fact that $\widehat{\mathbf{u}}_j$ is the eigenvector of the generalized eigenvalue problem $\widehat{\mathbf{Q}}\mathbf{u} = \lambda \mathbf{P}\mathbf{u}$ with respect to the jth largest eigenvalue $\widehat{\sigma}_j^2$, we have

$$\frac{1}{nq}\|\mathbf{X}\widehat{\mathbf{C}}_j\|_F^2 = \frac{1}{nq}\mathrm{trace}(\widehat{\mathbf{v}}_j \widehat{\mathbf{u}}_j^T \mathbf{X}^T \mathbf{X} \widehat{\mathbf{u}}_j \widehat{\mathbf{v}}_j^T) = \frac{1}{nq}(\widehat{\mathbf{u}}_j^T \mathbf{X}^T \mathbf{X} \widehat{\mathbf{u}}_j) \widehat{\mathbf{v}}_j^T \widehat{\mathbf{v}}_j$$
$$= \frac{1}{nq}\left(\frac{\widehat{\mathbf{u}}_j^T \mathbf{X}^T \mathbf{Y} \mathbf{Y}^T \mathbf{X} \widehat{\mathbf{u}}_j}{\widehat{\mathbf{u}}_j^T \mathbf{X}^T \mathbf{X} \widehat{\mathbf{u}}_j}\right) = \frac{\widehat{\mathbf{u}}_j^T \widehat{\mathbf{Q}} \widehat{\mathbf{u}}_j}{\widehat{\mathbf{u}}_j^T \mathbf{P} \widehat{\mathbf{u}}_j} = \widehat{\sigma}_j^2.$$

Thus, it follows from inequality (28) that

$$\sum_{j=r^*+1}^{k} \frac{1}{\sqrt{nq}} \|\mathbf{X}\widehat{\mathbf{C}}_j\|_F = \sum_{j=r^*+1}^{k} \widehat{\sigma}_j \leq O\left\{(r-r^*)\left(\frac{s}{n}\log\frac{pq}{\delta}\right)^{1/4}\right\}.$$

In view of inequality (29), using the same argument as in (26), the above bounds yield

$$\sqrt{\frac{\mathcal{L}_{k-1}-\mathcal{L}_k}{\mathcal{L}_k}} \leq \frac{O\left\{\left(\frac{s}{n}\log\frac{pq}{\delta}\right)^{1/4}\right\}}{\gamma_l + O\left(r^*\sqrt{\frac{s}{n}\log\frac{pq}{\delta}}\right) + O\left\{(r-r^*)\left(\frac{s}{n}\log\frac{pq}{\delta}\right)^{1/4}\right\}}.$$

Since $\delta^{-1} = O\{(pq)^\alpha\}$, $r^*\left\{\frac{s\log(pq)}{n}\right\}^{1/4} = o(1)$ and $r\left\{\frac{s\log(pq)}{n}\right\}^{1/4} = o(1)$, we conclude that

$$\sqrt{\frac{\mathcal{L}_{k-1}-\mathcal{L}_k}{\mathcal{L}_k}} \leq O\left\{\left(\frac{s}{n}\log\frac{pq}{\delta}\right)^{1/4}\right\}. \tag{30}$$

In view of the bounds in (27) and (30), using the union bound, it is not difficult to see that with probability at least $1-3\delta$ (both bounds are based on the event in Lemma 5 such that the results in Theorem 1 hold), the following bounds hold,

$$\begin{aligned}
&\text{For } k \leq r^*: \quad \log\left(\frac{\mathcal{L}_{k-1}}{\mathcal{L}_k}\right) \geq \frac{1}{(r^*-k+1+c_\gamma)^2} + o\left\{(r^*)^{-2}\right\},\\
&\text{For } k > r^*: \quad \log\left(\frac{\mathcal{L}_{k-1}}{\mathcal{L}_k}\right) \leq O\left\{\left(\frac{s}{n}\log\frac{pq}{\delta}\right)^{1/2}\right\}.
\end{aligned}$$

Thus, given the information criterion $\mathcal{C}_n = \mathrm{rank}(\widehat{\mathbf{C}})\sqrt{\log(pq)}\log\log n + \sqrt{n}\log\mathcal{L}_n$ and the assumption $(r^*)^2\sqrt{\log(pq)} = o(\sqrt{n}/\log\log n)$, by setting $\delta = \frac{1}{3}(pq)^{-\alpha}$, we can show that the following statements hold with probability at least $1-(pq)^{-\alpha}$:

$$\begin{aligned}
&\mathcal{C}_n(\mathrm{rank}(\widehat{\mathbf{C}}) = k-1) - \mathcal{C}_n(\mathrm{rank}(\widehat{\mathbf{C}}) = k) > 0, \qquad \text{if } k \leq r^*;\\
&\mathcal{C}_n(\mathrm{rank}(\widehat{\mathbf{C}}) = k-1) - \mathcal{C}_n(\mathrm{rank}(\widehat{\mathbf{C}}) = k) < 0, \qquad \text{if } k > r^*.
\end{aligned}$$

The above equations indicate that $\mathcal{C}_n$ will attain its minimum value when $\mathrm{rank}(\widehat{\mathbf{C}}) = r^*$, which means the algorithm will stop at $\mathrm{rank}(\widehat{\mathbf{C}}) = r^*$ with probability at least $1-(pq)^{-\alpha}$ for sufficiently large n, which concludes the proof of Theorem 2.

References

Alekh Agarwal, Sahand Negahban, and Martin J. Wainwright. Noisy matrix decomposition via convex relaxation: Optimal rates in high dimensions. *The Annals of Statistics*, 40(2): 1171–1197, 2012.

Francis R. Bach. Consistency of trace norm minimization. *Journal of Machine Learning Research*, 9(2):1019–1048, 2008.

M. Taha Bahadori and Yan Liu. An examination of practical granger causality inference. *SIAM International Conference on Data Mining*, pages 467–475, 2013.

Pierre Baldi and Kurt Hornik. Neural networks and principal component analysis: Learning from examples without local minima. *Neural Networks*, 2(1):53–58, 1989.

Ron Bekkerman, Mikhail Bilenko, and John Langford. *Scaling Up Machine Learning: Parallel and Distributed Approaches*. Cambridge University Press, 2012.

Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, 2010.

Peter Bühlmann and Sara van de Geer. *Statistics for High-Dimensional Data: Methods, Theory and Applications*. Springer Science & Business Media, 2011.

Florentina Bunea, Yiyuan She, and Marten H. Wegkamp. Optimal selection of reduced rank estimators of high-dimensional matrices. *The Annals of Statistics*, 39(2):1282–1309, 2011.

Florentina Bunea, Yiyuan She, and Marten H. Wegkamp. Joint variable and rank selection for parsimonious estimation of high-dimensional matrices. *The Annals of Statistics*, 40 (5):2359–2388, 2012.

Jian-Feng Cai, Emmanuel J. Candès, and Zuowei Shen. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 20(4):1956–1982, 2010.

Tony T. Cai, Zongming Ma, and Yihong Wu. Sparse PCA: Optimal rates and adaptive estimation. *The Annals of Statistics*, 41(6):3074–3110, 2013.

Emmanuel J. Candès and Yaniv Plan. Tight oracle inequalities for low-rank matrix recovery from a minimal number of noisy random measurements. *IEEE Transactions on Information Theory*, 57(4):2342–2359, 2011.

Emmanuel J. Candès and Terence Tao. Decoding by linear programming. *IEEE Transactions on Information Theory*, 51(12):4203–4215, 2005.

Meeyoung Cha, Hamed Haddadi, Fabricio Benevenuto, and Krishna P. Gummadi. Measuring user influence in twitter: The million follower fallacy. *International AAAI Conference on Weblogs and Social*, 10:10–17, 2010.

Venkat Chandrasekaran, Pablo Parrilo, and Alan S. Willsky. Latent variable graphical model selection via convex optimization. *Annual Allerton Conference on Communication, Control, and Computing*, pages 1610–1613, 2010.

Kun Chen and Kung-Sik Chan. A note on rank reduction in sparse multivariate regression. *Journal of Statistical Theory and Practice*, 10(1):100–120, 2015.

Kun Chen, Kung-Sik Chan, and Nils Chr. Stenseth. Reduced rank stochastic regression with a sparse singular value decomposition. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 74(2):203–221, 2012.

Kun Chen, Hongbo Dong, and Kung-Sik Chan. Reduced rank regression via adaptive nuclear norm penalization. *Biometrika*, 100(4):901–920, 2013.

Fernando De La Torre and Michael J. Black. A framework for robust subspace learning. *International Journal of Computer Vision*, 54(1-3):117–142, 2003.

Konstantinos I. Diamantaras and Sun Y. Kung. *Principal Component Neural Networks: Theory and Applications*. John Wiley & Sons, 1996.

Varun R. Embar, Rama K. Pasumarthi, and Indrajit Bhattacharya. A Bayesian framework for estimating properties of network diffusions. *International Conference on Knowledge Discovery and Data Mining*, pages 1216–1225, 2014.

Yingying Fan and Jinchi Lv. Asymptotic equivalence of regularization methods in thresholded parameter space. *Journal of the American Statistical Association*, 108(503):1044–1061, 2013.

Yingying Fan and Chengyong Tang. Tuning parameter selection in high dimensional penalized likelihood. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 75(3):531–552, 2013.

Jiashi Feng, Zhouchen Lin, Huan Xu, and Shuicheng Yan. Robust subspace segmentation with block-diagonal prior. *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3818–3825, 2014.

Manuel Gomez-Rodriguez, Jure Leskovec, and Andreas Krause. Inferring networks of diffusion and influence. *ACM Transactions on Knowledge Discovery from Data*, 5(4):21, 2012.

James A. Hanley and Barbara J. McNeil. The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology*, 143(1):29–36, 1982.

Alan J. Izenman. Reduced-rank regression for the multivariate linear model. *Journal of Multivariate Analysis*, 5(2):248–264, 1975.

Willard James and Charles Stein. Estimation with quadratic loss. *Berkeley Symposium on Mathematical Statistics and Probability*, 1:361–379, 1961.

Iain M. Johnstone. On the distribution of the largest eigenvalue in principal components analysis. *The Annals of Statistics*, 29(2):295–327, 2001.

Iain M. Johnstone and Arthur Y. Lu. On consistency and sparsity for principal components analysis in high dimensions. *Journal of the American Statistical Association*, 104(486): 682–693, 2009.

U Kang, Brendan Meeder, and Christos Faloutsos. Spectral analysis for billion-scale graphs: discoveries and implementation. *Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining*, pages 13–25, 2011.

Béatrice Laurent and Pascal Massart. Adaptive estimation of a quadratic functional by model selection. *The Annals of Statistics*, 28(5):1302–1338, 2000.

Jing Lei and Vincent Q. Vu. Sparsistency and agnostic inference in sparse PCA. *The Annals of Statistics*, 43(1):299–322, 2015.

Jure Leskovec, Deepayan Chakrabarti, Jon Kleinberg, Christos Faloutsos, and Zoubin Ghahramani. Kronecker graphs: an approach to modeling networks. *Journal of Machine Learning Research*, 11(3):985–1042, 2008.

Aurélie C. Lozano, Grzegorz Swirszcz, and Naoki Abe. Group orthogonal matching pursuit for logistic regression. *International Conference on Artificial Intelligence and Statistics*, pages 452–460, 2011.

Zongming Ma. Sparse principal component analysis and iterative thresholding. *The Annals of Statistics*, 41(2):772–801, 2013.

Aditya Mishra, Dipak K. Dey, and Kun Chen. Sequential co-sparse factor regression. *Journal of Computational and Graphical Statistics*, 26(4):814–825, 2017.

Seth A. Myers, Chenguang Zhu, and Jure Leskovec. Information diffusion and external influence in networks. *International Conference on Knowledge Discovery and Data Mining*, pages 33–41, 2012.

Sahand Negahban and Martin J. Wainwright. Estimation of (near) low-rank matrices with noise and high-dimensional scaling. *The Annals of Statistics*, 39(2):1069–1097, 2011.

Garvesh Raskutti, Martin J. Wainwright, and Bin Yu. Minimax rates of estimation for high-dimensional linear regression over ℓ_q balls. *IEEE Transactions on Information Theory*, 57(10):6976–6994, 2011.

Emile Richard, Stéphane Gaïffas, and Nicolas Vayatis. Link prediction in graphs with autoregressive features. *Journal of Machine Learning Research*, 15(1):489–517, 2012.

Dan Shen, Haipeng Shen, and J. S. Marron. A general framework for consistency of principal component analysis. *Journal of Machine Learning Research*, 17(1):5218–5251, 2016.

Kate Starbird and Leysia Palen. (How) will the revolution be retweeted?: information diffusion and the 2011 Egyptian uprising. *ACM Conference on Computer Supported Cooperative Work*, pages 7–16, 2012.

Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 58(1):267–288, 1996.

Wilson Truccolo, Uri T. Eden, Matthew R. Fellows, John P. Donoghue, and Emery N. Brown. A point process framework for relating neural spiking activity to spiking history, neural ensemble, and extrinsic covariate effects. *Journal of Neurophysiology*, 93(2):1074–89, 2005.

Michael Trusov, Randolph E. Bucklin, and Koen Pauwels. Effects of word-of-mouth versus traditional marketing: findings from an internet social networking site. *Journal of Marketing*, 73(5):90–102, 2009.

Raja Velu and Gregory C. Reinsel. *Multivariate Reduced-Rank Regression: Theory and Applications.* Springer Science & Business Media, 2013.

Yu-Xiang Wang, Huan Xu, and Chenlei Leng. Provable subspace clustering: when LRR meets SSC. *International Conference on Neural Information Processing Systems*, pages 64–72, 2013.

Ming Yuan and Yi Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67, 2006.

Ming Yuan, Ali Ekici, Zhaosong Lu, and Renato Monteiro. Dimension reduction and coefficient estimation in multivariate linear regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 69(3):329–346, 2007.

Xiao-Tong Yuan and Tong Zhang. Truncated power method for sparse eigenvalue problems. *Journal of Machine Learning Research*, 14(1):899–925, 2013.

Tong Zhang. Adaptive forward-backward greedy algorithm for learning sparse representations. *IEEE Transactions on Information Theory*, 57(7):4689–4708, 2011.

Zemin Zheng, Yingying Fan, and Jinchi Lv. High dimensional thresholded regression and shrinkage effect. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 76(3):627–649, 2014.

Ke Zhou, Hongyuan Zha, and Le Song. Learning social infectivity in sparse low-rank networks using multi-dimensional Hawkes processes. *International Conference on Artificial Intelligence and Statistics*, 31:641–649, 2013.

Hui Zou. The adaptive lasso and its oracle properties. *Journal of the American statistical association*, 101(476):1418–1429, 2006.