Journal of Machine Learning Research 25 (2024) 1-59 Submitted 7/23; Revised 6/24; Published 7/24

Scalable High-Dimensional Multivariate Linear Regression for Feature-Distributed Data

Shuo-Chieh Huang SHUOCHIEH@CHICAGOBOOTH.EDU
Ruey S. Tsay RUEY.TSAY@CHICAGOBOOTH.EDU
Booth School of Business
University of Chicago
Chicago, IL 60637, USA

Editor: Jie Peng

Abstract

Feature-distributed data, referred to data partitioned by features and stored across multiple computing nodes, are increasingly common in applications with a large number of features. This paper proposes a two-stage relaxed greedy algorithm (TSRGA) for applying multivariate linear regression to such data. The main advantage of TSRGA is that its communication complexity does not depend on the feature dimension, making it highly scalable to very large data sets. In addition, for multivariate response variables, TSRGA can be used to yield low-rank coefficient estimates. The fast convergence of TSRGA is validated by simulation experiments. Finally, we apply the proposed TSRGA in a financial application that leverages unstructured data from the 10-K reports, demonstrating its usefulness in applications with many dense large-dimensional matrices.

Keywords: Frank-Wolfe algorithm, Distributed computing, Reduced-rank regression, Feature selection, Multi-view and multi-modal data

1. Introduction

A computational strategy often adopted for tackling high-dimensional big data is to employ feature-distributed analysis: to partition the data by features and to store them across multiple computing nodes. For instance, when the data have an extremely large number of features that do not fit in a single computer, this strategy is used to circumvent storage constraints or to accelerate computation (Heinze et al., 2016; Wang et al., 2017; Richtárik and Takáč, 2016; Gao and Tsay, 2023). In addition, feature-distributed data may be inevitable when the data are collected and maintained by multiple parties. Because of bandwidth or administrative reasons, merging them in a central computing node from those sources might not be feasible (Hu et al., 2019). In some applications, data come naturally feature-distributed, such as the wireless sensor networks (Bertrand and Moonen, 2010, 2014, 2015).

A challenge in estimating statistical models with feature-distributed data is to avoid the high *communication complexity*, which is the amount of data that are transmitted across the nodes. Indeed, because distributed computing systems typically operate under limited bandwidth, sending voluminous data significantly slows down the algorithm. Unfortunately, data transmission is often a necessary evil with feature-distributed data: each node by itself

is unable to learn about the parameters associated with the features it does not own. Thus, algorithms that have lower communication complexities are preferred in practice.

Based on the rationale that the empirical minimizers of certain optimization problems are desirable statistical estimators, prior works have proposed various optimization algorithms with feature-distributed data. Richtárik and Takáč (2016) and Fercoq et al. (2014) employed randomized coordinate descent to solve ℓ_1-regularized problems and to exploit parallel computation from the distributed computing system. In addition, random projection techniques were used in Wang et al. (2017) and Heinze et al. (2016) for ℓ_2-regularized convex problems. However, for estimating linear models, the existing approaches usually incur a high communication complexity for very large data sets. To illustrate, consider the Lasso problem. The Hydra algorithm of Richtárik and Takáč (2016) requires $O(np\log(1/\epsilon))$ bytes of communication to reach ϵ-close to the optimal loss, where n is the sample size and p is the number of features. For data with extremely large p and n that do not fit in a modern computer, such communication complexity appears prohibitively expensive. Similarly, the distributed iterative dual random projection (DIDRP) algorithm of Wang et al. (2017) needs $O(n^2 + n\log(1/\epsilon))$ bytes of total communication for estimating the ridge regression, where the dominating n^2 factor comes from each node sending the sketched data matrix to a coordinator node. Thus it incurs not only a high communication cost but also a storage bottleneck.

This paper proposes a two-stage relaxed greedy algorithm (TSRGA) for feature-distributed data to mitigate the high communication complexity. TSRGA first applies the conventional relaxed greedy algorithm (RGA) to feature-distributed data. But we terminate the RGA with the help of a just-in-time stopping criterion, which aims to save excessive communication via reducing RGA iterations. In the second stage, we employ a modification of RGA to estimate the coefficient matrices associated with the selected predictors from the first stage. The modified second-stage RGA yields low-rank coefficient matrices which exploit information across tasks and improve statistical performance.

Instead of treating TSRGA as merely an optimization means, we directly analyze the convergence of TSRGA to the unknown parameters, which in turn implies the communication costs of TSRGA. The key insight of the proposed method is that the conventional RGA often incurs a high communication cost because it takes many iterations to minimize its loss function, but it tends to select relevant predictors in its early iterations. Therefore, one should decide when the RGA has done screening the predictors *before* it iterates too many steps. To this end, the just-in-time stopping criterion tracks the reduction in training error in each step, and calls for halting the RGA as soon as the reduction becomes smaller than some threshold. With the potential predictors narrowed down in the first stage, the second-stage employs a modified RGA and focuses on the more amenable problem of estimating the coefficient matrices of the screened predictors. The two-stage design enables TSRGA to substantially cut down the communication costs and produce even more accurate estimates than the original RGA.

Our theoretical results show that the proposed TSRGA enjoys a communication complexity of $O_p(\mathfrak{s}_n(n + d_n))$ bytes, up to a multiplicative term depending logarithmically on the problem dimensions, where d_n is the dimension of the response vector (or the number of tasks), and $\mathfrak{s}_n$ is a sparsity parameter defined later. This communication complexity improves that of Hydra by a factor of $p/\mathfrak{s}_n$, and is much smaller than that of DIDRP and

other one-shot algorithms (for example, Wang et al. 2016 and Heinze et al. 2016) if $\mathfrak{s}_n \ll n$. The RGA was also employed by Bellet et al. (2015) as a solver for ℓ_1-constrained problems, but it requires $O(n/\epsilon)$ communication since it only converges at a sub-linear rate (see also Jaggi, 2013 and Garber, 2020), where ϵ is again the optimization tolerance. Hence TSRGA offers a substantial speedup for estimating sparse models compared to the conventional RGA.

To validate the performance of TSRGA, we apply it to both synthetic and real-world data sets and show that TSRGA converges much faster than other existing methods. In the simulation experiments, TSRGA achieved the smallest estimation error using the least number of iterations. It also outperforms other centralized iterative algorithms both in speed and statistical accuracy. In a large-scale simulation experiment, TSRGA can effectively estimate the high-dimensional multivariate linear regression model with more than 16 GB data in less than 5 minutes. For an empirical application, we apply TSRGA to predict simultaneously some financial outcomes (volatility, trading volume, market beta, and returns) of the S&P 500 component companies using textual features extracted from their 10-K reports. The results show that TSRGA efficiently utilizes the information provided by the texts and works well with high dimensional feature matrices.

Furthermore, we propose some extensions of TSRGA. First, we extend TSRGA to big feature-distributed data which have not only many features but also a large number of observations. Thus, in addition to separately storing each predictors in different computing nodes, it is also necessary to partition the observations of each feature into chunks that could fit in one node. In this case, the computing nodes shall coordinate both horizontally and vertically, and we show that the communication cost to carry out TSRGA in this setting is still free of the feature dimension p, but could be larger than that of the purely feature-distributed case. Second, the idea of TSRGA can be extended beyond linear regression models. In Appendix D, we show how TSRGA can be applied to the generalized linear models.

For ease in reading, we collect the notations used throughout the paper here. The transpose of a matrix $\mathbf{A}$ is denoted by $\mathbf{A}^\top$ and that of a vector $\mathbf{v}$ is $\mathbf{v}^\top$. The inner product between two vectors $\mathbf{u}$ and $\mathbf{v}$ is denoted interchangeably as $\langle \mathbf{u}, \mathbf{v} \rangle = \mathbf{u}^\top \mathbf{v}$. If $\mathbf{A}, \mathbf{B}$ are $\mathbb{R}^{m \times n}$, $\langle \mathbf{A}, \mathbf{B} \rangle = \mathrm{tr}(\mathbf{A}^\top \mathbf{B})$ denotes their trace inner product. The minimum and maximum eigenvalues of a matrix $\mathbf{A}$ are denoted by $\lambda_{\min}(\mathbf{A})$ and $\lambda_{\max}(\mathbf{A})$, respectively. We also denote by $\sigma_l(\mathbf{A})$ the l-th singular value of $\mathbf{A}$, in descending order. When the argument is a vector, $\|\cdot\|$ denotes the usual Euclidean norm and $\|\cdot\|_p$ the ℓ_p norm. If the argument is a matrix, $\|\cdot\|_F$ denotes the Frobenius norm, $\|\cdot\|_{op}$ the operator norm, and $\|\cdot\|_*$ the nuclear norm. For a set J, $\sharp(J)$ denotes its cardinality. For an event $\mathcal{E}$, its complement is denoted as $\mathcal{E}^c$ and its associated indicator function is denoted as $\mathbf{1}\{\mathcal{E}\}$. For two positive (random) sequences $\{x_n\}$ and $\{y_n\}$, we write $x_n = o_p(y_n)$ if $\lim_{n\to\infty} \mathbb{P}(x_n/y_n < \epsilon) = 1$ for any $\epsilon > 0$ and write $x_n = O_p(y_n)$ if for any $\epsilon > 0$ there exists some $M_\epsilon < \infty$ such that $\limsup_{n\to\infty} \mathbb{P}(x_n/y_n > M_\epsilon) < \epsilon$.

2. Distributed framework and two-stage relaxed greedy algorithm

In this section, we first introduce the multivariate linear regression model considered in the paper and show how the data are distributed across the nodes. Then we lay out the imple-

mentation details of the proposed TSRGA, which consists of two different implementations of the conventional RGA and a just-in-time stopping criterion to guide the termination of the first-stage RGA. The case of needing horizontal partition will be discussed in Section 6.

2.1 Model and distributed framework

Consider the following multivariate linear regression model:

$$\mathbf{y}_t = \sum_{j=1}^{p_n} \mathbf{B}_j^{*\top} \mathbf{x}_{t,j} + \boldsymbol{\epsilon}_t, \quad t = 1, \ldots, n, \tag{1}$$

where $\mathbf{y}_t \in \mathbb{R}^{d_n}$ is the response vector, $\mathbf{x}_{t,j} \in \mathbb{R}^{q_{n,j}}$ a multivariate predictor, for $j = 1, 2, \ldots, p_n$, and $\mathbf{B}_j^*$ is the $(q_{n,j} \times d_n)$ unknown coefficient matrix, for $j = 1, \ldots, p_n$. In particular, we are most interested in the case $p_n \gg n$ and $q_{n,j} < n$. Clearly, when $d_n = q_{n,1} = \ldots = q_{n,p_n} = 1$, (1) reduces to the usual multiple linear regression model. Without loss of generality, we assume $\mathbf{y}_t$, $\mathbf{x}_{t,j}$ and $\boldsymbol{\epsilon}_t$ are mean zero.

There are several motivations for considering general d_n and $q_{n,j}$'s. First, imposing group-sparsity can be advantageous when the predictors display a natural grouping structure (e.g. Lounici et al. 2011). This advantage is inherited by (1) when only a limited number of $\mathbf{B}_j^*$'s are non-zero. Second, it is not uncommon that we are interested in modeling more than one response variable ($d_n > 1$). In this case, one can gain statistical accuracy if the prediction tasks are related, which is often embodied by the assumption that $\mathbf{B}_j^*$'s are of low rank (see, e.g., Reinsel et al. 2022). In modern machine learning, some predictors may be constructed from unstructured data sources. For instance, for functional data, $\mathbf{x}_{t,j}$'s may be the first few Fourier coefficients (Fan et al., 2015). On the other hand, for textual data, $\mathbf{x}_{t,j}$'s may be topic loading or outputs from some pre-trained neural networks (Kogan et al., 2009; Yeh et al., 2020; Bybee et al., 2021). Finally, model (1) can also accommodate the so-called multi-view of multi-modal data, which have also received considerable attention in recent years.

Next, we specify how the data are distributed across computing nodes. In matrix notations, we can write (1) as

$$\mathbf{Y} = \sum_{j=1}^{p_n} \mathbf{X}_j \mathbf{B}_j^* + \mathbf{E}, \tag{2}$$

where $\mathbf{Y} = (\mathbf{y}_1, \ldots, \mathbf{y}_n)^\top$, $\mathbf{X}_j = (\mathbf{x}_{1,j}, \ldots, \mathbf{x}_{n,j})^\top \in \mathbb{R}^{n \times q_{n,j}}$, for $j = 1, 2, \ldots, p_n$, and $\mathbf{E} = (\boldsymbol{\epsilon}_1, \ldots, \boldsymbol{\epsilon}_n)^\top$. As discussed in the Introduction, since pooling the large matrices $\mathbf{X}_1, \ldots, \mathbf{X}_{p_n}$ in a central node may not be feasible, a common strategy is to store them across nodes. In the following, we suppose that M nodes are available. Furthermore, the i-th node contains the data $\{\mathbf{Y}, \mathbf{X}_j : j \in \mathcal{I}_i\}$, for $i = 1, 2, \ldots, M$, where $\cup_{i=1}^M \mathcal{I}_i = \{1, 2, \ldots, p_n\} := [p_n]$. For ease in exposition, we assume a master node coordinates the other computing nodes. In particular, each worker node is able to send and receive data from the master node.

2.2 First-stage relaxed greedy algorithm and a just-in-time stopping criterion

We now introduce the first-stage RGA and describe how it can be applied to feature-distributed data. First, initialize $\hat{\mathbf{G}}^{(0)} = \mathbf{0}$ and $\hat{\mathbf{U}}^{(0)} = \mathbf{Y}$. For iteration $k = 1, 2, \ldots$, RGA

finds $(\hat{j}_k, \tilde{\mathbf{B}}_{\hat{j}_k})$ such that

$$(\hat{j}_k, \tilde{\mathbf{B}}_{\hat{j}_k}) \in \arg \max_{\substack{1\le j\le p_n \\ \|\mathbf{B}_j\|_*\le L_n}} \langle \hat{\mathbf{U}}^{(k-1)}, \mathbf{X}_j \mathbf{B}_j \rangle, \tag{3}$$

where $L_n = d_n^{1/2} L_0$ for some large constant $L_0 > 0$. Then RGA constructs updates by

$$\begin{aligned} \hat{\mathbf{G}}^{(k)} =& (1-\hat{\lambda}_k)\hat{\mathbf{G}}^{(k-1)} + \hat{\lambda}_k \mathbf{X}_{\hat{j}_k} \tilde{\mathbf{B}}_{\hat{j}_k}, \\ \hat{\mathbf{U}}^{(k)} =& \mathbf{Y} - \hat{\mathbf{G}}^{(k)}, \end{aligned} \tag{4}$$

where $\hat{\lambda}_k$ is determined by

$$\hat{\lambda}_k \in \arg \min_{0\le\lambda\le 1} \|\mathbf{Y} - (1-\lambda)\hat{\mathbf{G}}^{(k-1)} - \lambda \mathbf{X}_{\hat{j}_k} \tilde{\mathbf{B}}_{\hat{j}_k}\|_F. \tag{5}$$

RGA has important computational advantages that are attractive for big data computation. First, for a fixed j, the maximum in (3) is achieved at $\mathbf{B}_j = L_n \mathbf{u}\mathbf{v}^\top$, where $(\mathbf{u}, \mathbf{v})$ is the leading pair of singular vectors (i.e., corresponding to the largest singular value) of $\mathbf{X}_j^\top \hat{\mathbf{U}}^{(k-1)}$. Since computing the leading singular vectors is much cheaper than full SVD, RGA is computationally lighter than algorithms using singular value soft-thresholding, such as the alternating direction method of multipliers (ADMM). This feature has already been exploited in Zheng et al. (2018) and Zhuo et al. (2020) for nuclear-norm constrained optimization. Second, $\hat{\lambda}_k$ is easy to compute and has the closed-form $\hat{\lambda}_k = \max\{\min\{\hat{\lambda}_{k,uc}, 1\}, 0\}$, where

$$\hat{\lambda}_{k,uc} = \frac{\langle \hat{\mathbf{U}}^{(k-1)}, \mathbf{X}_{\hat{j}_k} \tilde{\mathbf{B}}_{\hat{j}_k} - \hat{\mathbf{G}}^{(k-1)} \rangle}{\|\mathbf{X}_{\hat{j}_k} \tilde{\mathbf{B}}_{\hat{j}_k} - \hat{\mathbf{G}}^{(k-1)}\|_F^2}$$

is the unconstrained minimizer of (5).

When applied to feature-distributed data, we can leverage these advantages. Observe from (3)-(5) that the history of RGA is encoded in $\hat{\mathbf{G}}^{(k)}$. That is, to construct $\hat{\mathbf{G}}^{(k+1)}$, which predictors were chosen and the order in which they were chosen are irrelevant, provided $\hat{\mathbf{G}}^{(k)}$ is known. In particular, each node only needs $\hat{\lambda}_{k+1}$ and $\mathbf{X}_{\hat{j}_{k+1}} \tilde{\mathbf{B}}_{\hat{j}_{k+1}}$ to construct $\hat{\mathbf{G}}^{(k+1)}$. As argued in the previous paragraph, $\mathbf{X}_{\hat{j}_{k+1}} \tilde{\mathbf{B}}_{\hat{j}_{k+1}}$ is a rank-one matrix. Thus transmitting this matrix only requires $O(n+d_n)$ bytes of communication, which are much lighter than that of the full matrix with $O(nd_n)$ bytes. In addition, each node requires only the extra memory to store $\hat{\mathbf{G}}^{(k)}$ throughout the training. This is less burdensome than random projection techniques, which require at least one node to make extra room to store the sketched matrix of size $O(n^2)$.

The above discussions are summarized in Algorithm 1, detailing how workers and the master node communicate to implement RGA with feature-distributed data. Clearly, each node sends and receives data of size $O(n + d_n)$ bytes (line 4 and 15) in each iteration. We remark that Algorithm 1 asks each node to send the potential updates to the master (line 15). This is for reducing rounds of communications, which can be a bottleneck in practice. If bandwidth limit is more stringent, one can instead first ask the workers to send ρ_c to the

Algorithm 1: Feature-distributed relaxed greedy algorithm (RGA)

Input: Number of maximum iterations K_n; $L_n > 0$.

Output: Each worker $1 \le c \le M$ obtains the coefficient matrices $\{\hat{\mathbf{B}}_j : j \in \mathcal{I}_c\}$.

Initialization: $\hat{\mathbf{B}}_j = \mathbf{0}$ for all j and $\hat{\mathbf{G}}^{(0)} = \mathbf{0}$

for $k = 1, 2, \ldots, K_n$ **do**

  Workers $c = 1, 2, \ldots, M$ **in parallel do**

    if $k > 1$ **then**

      Receive $(c^*, \hat{\lambda}_{k-1}, \sigma_{\hat{j}_{k-1}}, \mathbf{u}_{\hat{j}_{k-1}}, \mathbf{v}_{\hat{j}_{k-1}})$ from the master.

      $\hat{\mathbf{G}}^{(k-1)} = (1 - \hat{\lambda}_{k-1})\hat{\mathbf{G}}^{(k-2)} + \hat{\lambda}_{k-1}\sigma_{\hat{j}_{k-1}}\mathbf{u}_{\hat{j}_{k-1}}\mathbf{v}_{\hat{j}_{k-1}}^\top$.

      $\hat{\mathbf{B}}_j = (1 - \hat{\lambda}_{k-1})\hat{\mathbf{B}}_j$ for $j \in \mathcal{I}_c$.

      if $c = c^*$ **then**

        $\hat{\mathbf{B}}_{\hat{j}_{k-1}^{(c)}} = \hat{\mathbf{B}}_{\hat{j}_{k-1}^{(c)}} + \hat{\lambda}_{k-1}\tilde{\mathbf{B}}_{\hat{j}_{k-1}^{(c)}}$

      end

    end

    $\hat{\mathbf{U}}^{(k-1)} = \mathbf{Y} - \hat{\mathbf{G}}^{(k-1)}$

    $(\hat{j}_k^{(c)}, \tilde{\mathbf{B}}_{\hat{j}_k^{(c)}}) \in \arg\max_{\substack{j \in \mathcal{I}_c \\ \|\mathbf{B}_j\|_* \le L_n}} |\langle \hat{\mathbf{U}}^{(k-1)}, \mathbf{X}_j \mathbf{B}_j \rangle|$

    $\rho_c = |\langle \hat{\mathbf{U}}^{(k-1)}, \mathbf{X}_{\hat{j}_k^{(c)}} \tilde{\mathbf{B}}_{\hat{j}_k^{(c)}} \rangle|$

    Find the leading singular value decomposition: $\mathbf{X}_{\hat{j}_k^{(c)}} \tilde{\mathbf{B}}_{\hat{j}_k^{(c)}} = \sigma_{\hat{j}_k^{(c)}} \mathbf{u}_{\hat{j}_k^{(c)}} \mathbf{v}_{\hat{j}_k^{(c)}}^\top$

    Send $(\sigma_{\hat{j}_k^{(c)}}, \mathbf{u}_{\hat{j}_k^{(c)}}, \mathbf{v}_{\hat{j}_k^{(c)}}, \rho_c)$ to the master.

  end

  Master do

    Receives $\{(\sigma_{\hat{j}_k^{(c)}}, \mathbf{u}_{\hat{j}_k^{(c)}}, \mathbf{v}_{\hat{j}_k^{(c)}}, \rho_c) : c = 1, 2, \ldots, M\}$ from the workers.

    $c^* = \arg\max_{1 \le c \le N} \rho_c$

    $\sigma_{\hat{j}_k} = \sigma_{\hat{j}_k^{(c^*)}}, \mathbf{u}_{\hat{j}_k} = \mathbf{u}_{\hat{j}_k^{(c^*)}}, \mathbf{v}_{\hat{j}_k} = \mathbf{v}_{\hat{j}_k^{(c^*)}}$

    $\hat{\mathbf{G}}^{(k)} = (1 - \hat{\lambda}_k)\hat{\mathbf{G}}^{(k-1)} + \hat{\lambda}_k \sigma_{\hat{j}_k} \mathbf{u}_{\hat{j}_k} \mathbf{v}_{\hat{j}_k}^\top$, where $\hat{\lambda}_k$ is determined by

$$\hat{\lambda}_k \in \arg\min_{0 \le \lambda \le 1} \|\mathbf{Y} - (1 - \lambda)\hat{\mathbf{G}}^{(k-1)} - \lambda \sigma_{\hat{j}_k} \mathbf{u}_{\hat{j}_k} \mathbf{v}_{\hat{j}_k}^\top\|_F^2.$$

    Broadcasts $(c^*, \hat{\lambda}_k, \sigma_{\hat{j}_k}, \mathbf{u}_{\hat{j}_k}, \mathbf{v}_{\hat{j}_k})$ to all workers.

  end

end

master. After master decides c^*, it only asks the c^*-th node to send the update, so that only one node is transmitting the data.

Although the per-iteration communication complexity is low for RGA, the total communication can still be costly if the required number of iteration is high. Indeed, RGA converges to $\arg\min_{\sum_{j=1}^{p_n} \|\mathbf{B}_j\|_* \le L_n} \|\mathbf{Y} - \sum_{j=1}^{p_n} \mathbf{X}_j \mathbf{B}_j\|_F^2$ at the rate $O(k^{-1})$, where k is the number of iterations (Jaggi, 2013; Temlyakov, 2015). There are many attempts to design

variants of RGA that converge faster (see Jaggi and Lacoste-Julien, 2015; Lei et al., 2019; Garber, 2020 and references therein). Instead of adapting these increasingly sophisticated optimization schemes with feature-distributed data, we propose to terminate RGA early with the help of a just-in-time stopping criterion. The key insight, as to be shown in Theorem 1, is that RGA is capable of screening relevant predictors in the early iterations. The stopping criterion is defined as follows. Let $\hat{\sigma}_k^2 = (nd_n)^{-1}\|\mathbf{Y} - \hat{\mathbf{G}}^{(k)}\|_F^2$. We terminate the first-stage RGA at step $\hat{k}$, defined as

$$\hat{k} = \min\left\{1 \le k \le K_n : \frac{\hat{\sigma}_k^2}{\hat{\sigma}_{k-1}^2} \ge 1 - t_n\right\}, \tag{6}$$

and $\hat{k} = K_n$ if $\hat{\sigma}_k^2/\hat{\sigma}_{k-1}^2 < 1-t_n$, for all $1 \le k \le K_n$, where t_n is some threshold specified later and K_n is a prescribed maximum number of iterations. Intuitively, $\hat{k}$ is determined based on whether the current iteration provides sufficient improvement in reducing the training error. Note that $\hat{k}$ is determined just-in-time without fully iterating K_n steps. The algorithm is halted once the criterion is triggered, thereby saving excessive communication costs. This is in sharp contrast to the model selection criteria used in prior works to terminate greedy-type algorithms that compare all K_n models, such as the information criteria (Ing and Lai, 2011; Ing, 2020).

2.3 Second-stage relaxed greedy algorithm

After the first-stage RGA is terminated, the second-stage RGA focuses on estimation of the coefficient matrices. In this stage, we implement a modified version of RGA so that the coefficient estimates are of low rank.

For predictors with "large" coefficient matrices, failing to account for their low-rank structure may result in statistical inefficiency. To see this, let $\hat{J} := \hat{J}_{\hat{k}}$ be the predictors selected by the first-stage RGA, and let $\hat{\mathbf{B}}_j$, $j \in \hat{J}$, be the corresponding coefficient estimates produced by the first-stage RGA. Assume for now $q_{n,j} = q_n$. If $\min\{q_n, d_n\} > \hat{r} = \sum_{j\in\hat{J}} \hat{r}_j$, where $\hat{r}_j = \mathrm{rank}(\hat{\mathbf{B}}_j)$, then estimating this coefficient matrix alone without regularization amounts to estimating $d_n q_n$ parameters. It will be shown later in Theorem 1 that $\hat{r}_j \ge \mathrm{rank}(\mathbf{B}_j^*)$ with probability tending to one. Since $d_n q_n \asymp \min\{d_n, q_n\}(q_n + d_n) > \hat{r}(q_n + d_n)$, estimating this coefficient matrix would cost us more than the best achievable degrees of freedom (Reinsel et al., 2022).

To avoid loss in efficiency for these large coefficient estimators, we impose a constraint on the space in which our final estimators reside. Suppose the j-th predictor, $j \in \hat{J}$, satisfies $\min\{q_{n,j}, d_n\} > \hat{r}$. We require its coefficient estimator to be of the form $\hat{\mathbf{\Sigma}}_j^{-1}\mathbf{U}_j\mathbf{S}\mathbf{V}_j^\top$, where $\hat{\mathbf{\Sigma}}_j = n^{-1}\mathbf{X}_j^\top\mathbf{X}_j$; $\mathbf{U}_j = (\mathbf{u}_{1,j}, \ldots, \mathbf{u}_{\hat{r},j})$ and $\mathbf{V}_j = (\mathbf{v}_{1,j}, \ldots, \mathbf{v}_{\hat{r},j})$ form the leading $\hat{r}$ pairs of singular vectors of $\mathbf{X}_j^\top\mathbf{Y}$, and $\mathbf{S}$ is an $\hat{r} \times \hat{r}$ matrix to be optimized.

The second-stage RGA proceeds as follows. Initialize again $\hat{\mathbf{G}}^{(0)} = \mathbf{0}$ and $\hat{\mathbf{U}}^{(0)} = \mathbf{Y}$. For $k = 1, 2, \ldots$, choose

$$(\hat{j}_k, \hat{\mathbf{S}}_k) \in \arg\max_{\substack{j\in\hat{J} \\ \|\mathbf{S}\|_* \le L_n}} \langle \hat{\mathbf{U}}^{(k-1)}, \mathbf{X}_j\hat{\mathbf{\Sigma}}_j^{-1}\mathbf{U}_j\mathbf{S}\mathbf{V}_j^\top\rangle, \tag{7}$$

where the maximum is searching over $\mathbf{S} \in \mathbb{R}^{\hat{r}\times\hat{r}}$ if $\hat{r} < \min\{q_{n,j}, d_n\}$. For j such that $\hat{r} \geq \min\{q_{n,j}, d_n\}$, we define $\mathbf{U}_j$ and $\mathbf{V}_j$ to be the full set of singular vectors and the maximum is searching over $\mathbf{S} \in \mathbb{R}^{q_{n,j}\times d_n}$. Next, we construct the update by

$$\begin{aligned}\hat{\mathbf{G}}^{(k)} =&(1-\hat{\lambda}_k)\hat{\mathbf{G}}^{(k-1)} + \hat{\lambda}_k \mathbf{X}_{\hat{j}_k}\hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1}\mathbf{U}_{\hat{j}_k}\hat{\mathbf{S}}_k\mathbf{V}_{\hat{j}_k}^\top,\\ \hat{\mathbf{U}}^{(k)} =&\mathbf{Y} - \hat{\mathbf{G}}^{(k)},\end{aligned} \tag{8}$$

where $\hat{\lambda}_k$ is, again, determined by

$$\hat{\lambda}_k \in \arg\min_{0\leq\lambda\leq 1} \|\mathbf{Y} - (1-\lambda)\hat{\mathbf{G}}^{(k-1)} - \lambda \mathbf{X}_{\hat{j}_k}\hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1}\mathbf{U}_{\hat{j}_k}\hat{\mathbf{S}}_k\mathbf{V}_{\hat{j}_k}^\top\|_F^2. \tag{9}$$

At first glance, the updating scheme (7)-(9) may appear similar to those proposed by Ding et al. (2021) or Ding et al. (2020), but we note one important difference here: the matrices $\mathbf{U}_j$ and $\mathbf{V}_j$ are fixed at the onset of the second stage. Thus our estimators' ranks remain controlled, which is not the case in the aforementioned works. More comparisons between TSRGA and these works will be made in Section 3.2.

We briefly comment on the computational aspects of the second-stage RGA. First, similarly to the first-stage, for a fixed j the maximum in (7) is attained at $\mathbf{S} = L_n\mathbf{u}\mathbf{v}^\top$, where $(\mathbf{u}, \mathbf{v})$ is the leading pair of singular vectors of $\mathbf{U}_j^\top\hat{\mathbf{\Sigma}}_j^{-1}\mathbf{X}_j^\top\hat{\mathbf{U}}^{(k-1)}\mathbf{V}_j$, which can be computed locally by each node. As a result, the per-iteration communication is still $O(n+d_n)$ for each node. For $j \in \hat{J}$ with $\hat{r} \geq \min\{q_{n,j}, d_n\}$, since $\mathbf{U}_j$ and $\mathbf{V}_j$ are non-singular, the parameter space is not limited except for the bounded nuclear norm constraint. Indeed, it is not difficult to see that for such j,

$$\max_{\|\mathbf{S}\|_*\leq L_n} \langle \hat{\mathbf{U}}^{(k-1)}, \mathbf{X}_j\hat{\mathbf{\Sigma}}_j^{-1}\mathbf{U}_j\mathbf{S}\mathbf{V}_j^\top\rangle$$

is equivalent to

$$\max_{\|\mathbf{B}\|_*\leq L_n} \langle \hat{\mathbf{U}}^{(k-1)}, \mathbf{X}_j\hat{\mathbf{\Sigma}}_j^{-1}\mathbf{B}\rangle \tag{10}$$

with the correspondence $\mathbf{B} = \mathbf{U}_j\mathbf{S}\mathbf{V}_j^\top$. Thus, for such j, it is not necessary to compute the singular vectors $\mathbf{U}_j$ and $\mathbf{V}_j$. Instead, one can directly solve (10). Finally, it is straightforward to modify Algorithm 1 to implement the second-stage RGA with feature-distributed data. We defer the details to Appendix A.

It is worth mentioning that the idea of two-stage RGA can be employed beyond the linear regression setup. For example, by replacing the squared loss with log likelihood function, we can use TSRGA to estimate generalized linear models, which include logistic regression for classification tasks and Poisson regression for modeling count data. The details of the modified algorithm are deferred to Appendix D, where we also examine its performance through simulations.

2.4 Related algorithms

In this subsection, we consider TSRGA in several contexts and compare it with related algorithms. By viewing TSRGA as either a novel feature-distributed algorithm, an improvement

over the Frank-Wolfe algorithm, a new method to estimate the integrative multi-view regression (Li et al., 2019), or a close relative of the greedy-type algorithms (Temlyakov, 2000), we highlight both its computational ease in applying to feature-distributed data and its theoretical applicability in estimating high-dimensional linear models.

Over the last decade, a few methods for estimating linear regression with feature-distributed data have been proposed. For instance, Richtárik and Takáč (2016) and Fercoq et al. (2014) use randomized coordinate descent to solve ℓ_1-regularized optimization problem, and Hu et al. (2019) proposes an asynchronous stochastic gradient descent algorithm, to name just a few. These methods either require a communication complexity that scales with p_n, or converge only at sub-linear rates, both of which translate to high communication costs. The screen-and-clean approach of Yang et al. (2016), similar in spirit to TSRGA, first applies sure independence screening (SIS, Fan and Lv, 2008) to identify a subset of potentially relevant predictors. Then it uses an iterative procedure similar to the iterative Hessian sketch (Pilanci and Wainwright, 2016) to estimate the associated coefficients. While SIS does not require communication, it imposes stronger assumptions on the predictors and the error term. In contrast, the proposed TSRGA can be applied at low communication complexity without succumbing to those assumptions.

TSRGA also adds to the line of studies that attempt to modify the conventional Frank-Wolfe algorithm (Frank and Wolfe, 1956). RGA, more often called the Frank-Wolfe algorithm in the optimization literature, has been widely adopted in big data applications for its computational simplicity. Recently, various modifications of the Frank-Wolfe algorithm have been proposed to attain a linear convergence rate that does not depend on the feature dimension p_n (Lei et al., 2019; Garber, 2020; Ding et al., 2021, 2020). However, strong convexity or quadratic growth of the loss function is typically assumed in these works, which precludes high-dimensional data ($n \ll p_n$). Frank-Wolfe algorithm has also been found useful in distributed systems, though most prior works employed the horizontally-partitioned data (Zheng et al., 2018; Zhuo et al., 2020). That is, data are partitioned and stored across nodes by observations instead of by features. A notable exception is Bellet et al. (2015), who found that Frank-Wolfe outperforms ADMM in communication and wall-clock time for sparse scalar regression with feature-distributed data, despite that Frank-Wolfe still suffers from sub-linear convergence. In this paper, we neither assume strong convexity (or quadratic growth) nor limit ourselves to scalar regression, and TSRGA demands much less computation than the usual Frank-Wolfe algorithm.

Model (1) was also employed by Li et al. (2019), and they termed it the integrative multi-view regression. They propose an ADMM-based algorithm, integrative reduced-rank regression (iRRR), for optimization in a centralized computing framework. The major drawback, as discussed earlier, is a computationally-expensive step of singular value soft-thresholding. Thus, TSRGA can serve as a computationally attractive alternative. In Section 4, we compare their empirical performance and find that TSRGA is much more efficient.

Other closely related greedy algorithms such as the orthogonal greedy algorithm (OGA) have also been applied to high-dimensional linear regression. OGA, when used in conjunction with an information criterion, attains the optimal prediction error (Ing, 2020) under various sparsity assumptions. However, it is computationally less adaptable to feature-distributed data. To keep the per-iteration communication low, the sequential orthogonal-

ization scheme of Ing and Lai (2011) can be used with feature-distributed data, but the individual nodes would not have the correct coefficients to use at the prediction time when new data, possibly not orthogonalized, become available. Alternatively, one needs to allocate extra memory in each node to store the history of the OGA path to compute the projection in each iteration.

3. Communication complexity of TSRGA

In this section, we derive theoretical guarantees on the communication complexity of TSRGA. Specifically, we show that the communication complexity of TSRGA does not scale with the feature dimension p_n, but instead depends on the sparsity of the underlying problem.

3.1 Assumptions

For the theoretical analysis, we maintain the following mild assumptions of model (1).

(C1) There exists some $0 < \mu < \infty$ such that with probability approaching one,

$$\mu^{-1} \leq \min_{1\leq j\leq p_n} \lambda_{\min}(\hat{\mathbf{\Sigma}}_j) \leq \max_{1\leq j\leq p_n} \lambda_{\max}(\hat{\mathbf{\Sigma}}_j) \leq \mu,$$

where $\hat{\mathbf{\Sigma}}_j = n^{-1}\mathbf{X}_j^\top \mathbf{X}_j$ with $\mathbf{X}_j$ being defined in (2).

(C2) Let $\xi_E = \max_{1\leq j\leq p_n} \|\mathbf{X}_j^\top \mathbf{E}\|_{op}$. There exists a sequence of $K_n \to \infty$ such that $K_n \xi_E = O_p(n d_n^{1/2})$.

(C3)

$$\lim_{n\to\infty} \mathbb{P}\left(\min_{\sharp(J)\leq 2K_n} \lambda_{\min}(n^{-1}\mathbf{X}(J)^\top \mathbf{X}(J)) > \mu^{-1} \right) = 1,$$

where $\mathbf{X}(J) = (\mathbf{X}_j : j \in J) \in \mathbb{R}^{n\times(\sum_{j\in J} q_{n,j})}$.

(C4) There exists some large $L < \infty$ such that $d_n^{-1/2} \sum_{j=1}^{p_n} \|\mathbf{B}_j^*\|_* \leq L$. Moreover, there exists a non-decreasing sequence $\{s_n\}$ such that $s_n^2 = o(K_n)$ and

$$\min_{j\in J_n} \sigma_{r_j^*}^2 \left(d_n^{-1/2} \mathbf{B}_j^* \right) \geq s_n^{-1},$$

where $J_n = \{1 \leq j \leq p_n : \mathbf{B}_j^* \neq \mathbf{0}\}$ is the set of indices corresponding the relevant predictors, and $r_j^* = \mathrm{rank}(\mathbf{B}_j^*)$.

These assumptions are quite standard. (C1) requires the variances of the predictors to be on the same order of magnitude, which is often the case if the predictors are normalized. ξ_E in (C2) is typically regarded as the effect size of the noise. Through auxiliary concentration inequalities in the literature, we will verify (C2) in the examples following the main result. (C3) assumes a lower bound on the minimum eigenvalue of the covariance matrices formed by small subsets of predictors. Note that (C3) could hold even when $p_n \gg n$ and the

observations are dependent; we refer to Ing and Lai (2011) and Ing (2020) for related discussions on (C3). s_n in (C4) imposes a lower bound on the minimum non-zero singular value of the (normalized) coefficient matrices $d_n^{-1/2}\mathbf{B}_j^*$. Since (C4) implies $\sharp(J_n) \leq s_n^{1/2}L$, it can be interpreted as a measure of sparsity of the underlying model.

Next, we introduce two assumptions that are important to the feature-distributed problem. Let $\tilde{\mathbf{Y}} = \sum_{j=1}^{p_n} \mathbf{X}_j\mathbf{B}_j^*$ be the noiseless part of $\mathbf{Y}$.

(C5) Let $\bar{r}_j = \text{rank}(\mathbf{X}_j^\top\tilde{\mathbf{Y}})$ and $J_o = J_n \cap \{j : \min\{q_{n,j}, d_n\} > \bar{r}_j\}$. There exists $\delta_n > 0$ such that $\xi_E = o_p(n\delta_n)$ and with probability approaching one,

$$\min_{j\in J_o} \sigma_{\bar{r}_j}(\mathbf{X}_j^\top\tilde{\mathbf{Y}}) \geq n\delta_n.$$

(C6) (Local revelation) If the column vectors of $\tilde{\mathbf{U}}_j \in \mathbb{R}^{q_{n,j}\times\bar{r}_j}$ and $\tilde{\mathbf{V}}_j \in \mathbb{R}^{d_n\times\bar{r}_j}$ are the leading pairs of singular vectors corresponding to the non-zero singular values of $\mathbf{X}_j^\top\tilde{\mathbf{Y}}$, then with probability approaching one, there exists an $\bar{r}_j \times \bar{r}_j$ matrix $\mathbf{\Lambda}_j$ such that

$$\hat{\mathbf{\Sigma}}_j\mathbf{B}_j^* = \tilde{\mathbf{U}}_j\mathbf{\Lambda}_j\tilde{\mathbf{V}}_j^\top \tag{11}$$

for all $j \in J_o$.

(C5) and (C6) are assumptions that endow the local nodes sufficient information in the feature-distributed setting. Both assumptions concern relevant predictors that are "large" such that their dimensions $q_{n,j} \times d_n$ satisfy $\min\{q_{n,j}, d_n\} > \bar{r}_j$. Intuitively, (C5) requires, for relevant predictors which are of large dimension, the marginal correlations between these predictors and $\tilde{\mathbf{Y}}$ are sufficiently large. The local revelation condition (C6) assumes each node could use its local data to re-construct $\hat{\mathbf{\Sigma}}_j\mathbf{B}_j^*$ for $j \in J_o$. This would simplify information sharing between the nodes. Although they are key assumptions used to derive a fast convergence rate for the second-stage RGA, they are not needed for establishing the sure-screening property of the just-in-time stopping criterion (see Theorem 1). In addition, these two assumptions are vacuous when all predictors are of small dimensions. For instance, for scalar group-sparse linear regression, $\min\{d_n, q_{n,j}\} = \min\{1, q_{n,j}\} = 1 \leq \bar{r}_j$. Hence $J_o = \emptyset$ and the two assumptions are immaterial.

To better understand (11), consider the following example.

$$\mathbf{y}_t = \mathbf{B}_1^{*\top}\mathbf{x}_{t,1} + \mathbf{B}_2^{*\top}\mathbf{x}_{t,2} + \boldsymbol{\epsilon}_t,$$

where $\mathbf{B}_1^*, \mathbf{B}_2^*$ are rank-1 matrices such that $\mathbf{B}_1^* = \mathbf{u}_1^*\mathbf{v}_1^{*\top}$ and $\mathbf{B}_2^* = \mathbf{u}_2^*\mathbf{v}_2^{*\top}$. In matrix notation, we write $\mathbf{Y} = \mathbf{X}_1\mathbf{B}_1^* + \mathbf{X}_2\mathbf{B}_2^* + \mathbf{E}$. Suppose $q_{n,1} = q_{n,2} > 2$, and consider

$$\mathbf{X}_1^\top\tilde{\mathbf{Y}} = \underbrace{\begin{pmatrix}\mathbf{X}_1^\top\mathbf{X}_1\mathbf{u}_1^* & \mathbf{X}_1^\top\mathbf{X}_2\mathbf{u}_2^*\end{pmatrix}}_{\mathbf{A}}\underbrace{\begin{pmatrix}\mathbf{v}_1^{*\top}\\ \mathbf{v}_2^{*\top}\end{pmatrix}}_{\mathbf{B}}.$$

It is not difficult to show that (11) holds (for $j = 1$) if $\mathbf{A}$ and $\mathbf{B}$ are of full rank. Since $\mathbf{y}_t = (\mathbf{x}_{t,1}^\top\mathbf{u}_1^*)\mathbf{v}_1^* + (\mathbf{x}_{t,2}^\top\mathbf{u}_2^*)\mathbf{v}_2^*$, one can interpret $f_{t,j} = \mathbf{x}_{t,j}^\top\mathbf{u}_j^*$ as the predictive factor associated with predictor j, for $j = 1, 2$. $f_{t,j}$ has differential effects on each element of $\mathbf{y}_t$, which

are determined by $\mathbf{v}_j^*$. Hence, that $\mathbf{B}$ has full rank translates to that the two factors $f_{t,1}$ and $f_{t,2}$ have distinct impacts on $\mathbf{y}_t$. On the other hand, $\mathbf{A}$ has full rank if and only if $\mathbf{u}_1^* \neq \alpha(\mathbf{X}_1^\top \mathbf{X}_1)^{-1}\mathbf{X}_1^\top \mathbf{X}_2 \mathbf{u}_2^*$ for any $\alpha \neq 0$. This implies the factor $f_{t,1}$ must not be equal to the projection of $f_{t,2}$ onto the space spanned by $\mathbf{X}_1$. Therefore, (11) can be interpreted as requiring the factors $f_{t,1}$ and $f_{t,2}$ are truly distinct and make distinguishable contributions to the response vector. Moreover, if (11) fails, the marginal product $\mathbf{X}_1^\top \tilde{\mathbf{Y}}$ may no longer be useful, because the signals are contaminated by possible collinearity.

3.2 Main results

We now present some theoretical properties of TSRGA, with proofs relegated to Appendix B. In the following, we assume L_n, the hyperparameter input to the TSRGA algorithm, is chosen to be $L_n = d_n^{1/2} L_0$ with $L_0 \geq L/(1-\epsilon_L)$, where $1 - \epsilon_L \leq \mu^{-2}/4$.

Our first result proves that RGA, coupled with the just-in-time stopping criterion, can screen the relevant predictors. Moreover, it provides an upper bound on the rank of the corresponding coefficient matrices.

Theorem 1 *Assume (C1)-(C4) hold. Suppose there exists an $M_o < \infty$ such that $M_o^{-1} \leq (nd_n)^{-1}\|\mathbf{E}\|_F^2 \leq M_o$ with probability tending to one. Write $\hat{\mathbf{G}}^{(k)} = \sum_{j=1}^{p_n} \mathbf{X}_j \hat{\mathbf{B}}_j^{(k)}$, $k = 1, 2, \ldots, K_n$, for the iterates of the first-stage RGA. If $\hat{k}$ is defined by (6) with $t_n = Cs_n^{-2}$ for some sufficiently small $C > 0$, then*

$$\lim_{n\to\infty} \mathbb{P}\left(\operatorname{rank}(\mathbf{B}_j^*) \leq \operatorname{rank}(\hat{\mathbf{B}}_j^{(\hat{k})}) \text{ for all } j\right) = 1. \tag{12}$$

Although Theorem 1 only provides an upper bound for the ranks of $\mathbf{B}_j^*$'s, it renders a useful diagnosis for the rank of the coefficient matrices for model (1). When $p_n = 1$, Bunea et al. (2011) proposed a rank selection criterion (RSC) to select the optimal reduced rank estimator, which is shown to be a consistent estimator of the effective rank. However, rank selection for model (1) with $p_n > 1$ is less investigated. Moreover, we can bound $\hat{k}$ by the following lemma.

Lemma 2 *Under the assumptions of Theorem 1, $\hat{k} = O_p(s_n^2)$.*

Lemma 2 ensures the just-in-time stopping criterion is triggered in no more than $O(s_n^2)$ iterations, which is much smaller than $O(K_n)$ by (C4). Thus compared to the model selection rules using information criteria that iterate K_n steps in full, the proposed method greatly reduces communication costs.

Next, we derive the required number of iterations for TSRGA to converge near the unknown parameters, which translates to its communication costs. With a slight abuse of notation, we also write the second-stage RGA iterates as $\hat{\mathbf{G}}^{(k)} = \sum_{j\in\hat{J}} \mathbf{X}_j \hat{\mathbf{B}}_j^{(k)}$.

Theorem 3 *Assume the assumptions of Theorem 1 hold, and additionally (C5) and (C6) also hold. If $\xi_E = O_p(\xi_n)$ and $m_n = \lceil \rho\kappa_n \log(n^2 d_n/\xi_n^2) \rceil$ for some sequence $\{\xi_n\}$ of positive numbers, where $\rho = 64\mu^5/\tau^2$ with $0 < \tau < 1$ being arbitrary, and*

$$\kappa_n = \sharp(\hat{J}) \max\left\{ \max_{j\in\hat{J}-\hat{J}_o} (q_{n,j} \wedge d_n), \hat{r}\mathbf{1}\{\hat{J}_o \neq \emptyset\} \right\},$$

with $a \wedge b = \min\{a, b\}$ *and* $\hat{J}_o = \{j \in \hat{J} : \hat{r} < \min\{q_{n,j}, d_n\}\}$*, then the proposed second-stage RGA satisfies*

$$\sup_{m \geq m_n} \frac{1}{d_n} \sum_{j=1}^{p_n} \|\mathbf{B}_j^* - \hat{\mathbf{B}}_j^{(m)}\|_F^2 = O_p\left(\frac{\kappa_n \xi_n^2}{n^2 d_n} \log \frac{n^2 d_n}{\xi_n^2} + \frac{\xi_n^2}{n^2 \delta_n^2} \mathbf{1}\{J_o \neq \emptyset\}\right).$$

Since the per-iteration communication cost of TSRGA is $O(n+d_n)$, Theorem 3, together with Lemma 2, directly imples the communication complexity of TSRGA, which we state as the following corollary.

Corollary 4 *If* $\kappa_n = O_p(\mathfrak{s}_n)$ *for some sequence* $\{\mathfrak{s}_n\}$ *of positive numbers, then TSRGA achieves an error of order*

$$O_p\left(\frac{\mathfrak{s}_n \xi_n^2}{n^2 d_n} \log \frac{n^2 d_n}{\xi_n^2} + \frac{\xi_n^2}{n^2 \delta_n^2} \mathbf{1}\{J_o \neq \emptyset\}\right),$$

with a communication complexity of order

$$O_p\left((n + d_n)\mathfrak{s}_n \log \frac{n^2 d_n}{\xi_n^2}\right).$$

Thus, the communication complexity, up to a logarithmic factor, scales mainly with $\mathfrak{s}_n$. In general, Lemma 2 implies $\mathfrak{s}_n = O_p(s_n^4)$. Thus $\mathfrak{s}_n$ is also a measure of the sparsity of the underlying model. Moreover, in the important special case when the response is a scalar, $\mathfrak{s}_n = O_p(s_n^2)$ since $d_n = 1$ and $\hat{J}_o = \emptyset$. To demonstrate this result more concretely, we discuss the communication complexity of TSRGA when applied to several well-known models below.

Example 1 (High-dimensional sparse linear regression) *Consider the model* $y_t = \sum_{j=1}^{p_n} \beta_j x_{t,j} + \epsilon_t$*. Under suitable conditions, such as* $\{\epsilon_t\}$ *being i.i.d. sub-Gaussian random variables, it can be shown that* $\xi_E = O_p(\sqrt{n \log p_n})$ *(see, for example, Ing and Lai, 2011 and Ing, 2020). Then TSRGA achieves an error of order*

$$\sum_{j=1}^{p_n} |\beta_j - \hat{\beta}_j|^2 = O_p\left(\frac{s_n^2 \log p_n}{n}\right) \tag{13}$$

with a communication complexity of

$$O_p\left(n s_n^2 \log \frac{n}{\log p_n}\right).$$

To reach ϵ-close to the minimizer of the Lasso problem, the communication complexity of the Hydra algorithm (Richtárik and Takáč, 2016) is

$$O\left(\frac{n p_n}{M \tau} \log \frac{1}{\epsilon}\right),$$

where M is the number of nodes and τ is the number of coordinates to update in each iteration. Given limited computational resources, τM may still be of order smaller than p_n. Thus the communication complexity of TSRGA, which does not scale with p_n, is more favorable for large data sets with huge p_n. In our simulation studies, we also observe that TSRGA converges near $(\beta_1, \ldots, \beta_{p_n})$ much faster than Hydra-type algorithms.

Example 2 (Multi-task linear regression with common relevant predictors) *Suppose we are interested in modeling T tasks simultaneously. Let $\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_T$ be the vectors of n observations of the T responses, and $\mathbf{X}$ be the $n \times p$ design matrix consisting of p predictors. Consider the system of linear regressions*

$$\mathbf{y}_t = \mathbf{X}\mathbf{b}_t + \mathbf{e}_t, \quad t = 1, \ldots, T, \tag{14}$$

where $\mathbf{b}_i = (\beta_{i,1}, \beta_{i,2}, \ldots, \beta_{i,p})^T$, for $i = 1, 2, \ldots, T$, and $\mathbf{e}_i$, for $1 \le i \le T$, are independent standard Gaussian random vectors. Let $\mathbf{x}_j$ be the j-th column vector of $\mathbf{X}$. Then we may rearrange (14) *as*

$$\begin{pmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \vdots \\ \mathbf{y}_T \end{pmatrix} = \sum_{j=1}^{p} \mathbf{X}_j \mathbf{B}_j + \begin{pmatrix} \mathbf{e}_1 \\ \mathbf{e}_2 \\ \vdots \\ \mathbf{e}_T \end{pmatrix}, \tag{15}$$

where $\mathbf{B}_j = (\beta_{1,j}, \beta_{2,j}, \ldots, \beta_{T,j})^T$ and $\mathbf{X}_j = \mathbf{I}_T \otimes \mathbf{x}_j$, where $\mathbf{I}_T$ is the $T \times T$ identity matrix and $\mathbf{A} \otimes \mathbf{B}$ denotes the Kronecker product of $\mathbf{A}$ and $\mathbf{B}$. Now (15) *falls under our general model* (1)*. Sparsity of the $\mathbf{B}_j$'s promotes that each task is driven by the same small set of predictors, or equivalently, $\mathbf{b}_j$'s in* (14) *have a common support. By a similar argument used in Lemma 3.1 of Lounici et al. (2011), it can be shown that $\xi_E = O_p(\sqrt{nT(1 + T^{-1}\log p)})$. Hence Corollary 4 implies that TSRGA applied to* (15) *achieves an error of order*

$$\sum_{j=1}^{p} \|\mathbf{B}_j - \hat{\mathbf{B}}_j\|^2 = O_p\left(\frac{s_n^2}{nT}(1 + \frac{\log p}{T})\right) \tag{16}$$

with the communication complexity

$$O_p\left(nTs_n^2 \log \frac{nT}{1 + T^{-1}\log p}\right).$$

Notice again that the iteration complexity scales primarily with the strong sparsity parameter s_n, not with p. As illustrated by Lounici et al. (2011), (14) can be motivated from a variety of applications, such as the seemingly unrelated regressions (SUR) in econometrics and the conjoint analysis in marketing research.

Example 3 (Integrative multi-view regression) *Consider the general model* (1)*, which is called the integrative multi-view regression by Li et al. (2019). Assume $\mathbf{E}$ has i.i.d. Gaussian entries, and for simplicity that $q_{n,1} = q_{n,2} = \ldots = q_{n,p_n} = q_n$. Then by a similar argument used by Li et al. (2019) it follows that $\xi_E = O_p(\sqrt{n \log p_n}(\sqrt{d_n} + \sqrt{q_n}))$. Suppose the predictors $\mathbf{X}_j$, for $j = 1, 2, \ldots, p_n$, are distributed across computing nodes. TSRGA achieves*

$$\frac{1}{d_n}\sum_{j=1}^{p_n} \|\mathbf{B}_j^* - \hat{\mathbf{B}}_j\|_F^2 = O_p\left(\frac{s_n^4(d_n + q_n)\log p_n}{nd_n} + \frac{(d_n + q_n)\log p_n}{n\delta_n}\right) \tag{17}$$

with a communication complexity of

$$O_p\left((n + d_n)s_n^4 \log \frac{nd_n}{(d_n + q_n)\log p_n}\right).$$

Although Li et al. (2019) did not consider the feature-distributed data, they offer an ADMM-based algorithm, iRRR, for estimating (1). However, updating many parameters in each iteration causes significant computational bottleneck. In our Monte Carlo simulation, iRRR is unable to run efficiently with $p_n \geq 50$ even with centralized computing and a moderate sample size, whereas TSRGA can handle such data sizes easily.

In general, the statistical errors of TSRGA in the above examples ((13), (16), and (17)) are sub-optimal compared to the minimax rates unless $s_n = O(1)$, in which case the model is strongly sparse with a fixed number of relevant predictors. One reason is that Theorem 1 only guarantees sure-screening instead of predictor and rank selection consistency. In Examples 1 and 2, the statistical error could be improved if one applies hard-thresholding after the second-stage RGA, and then estimates the coefficients associated with the survived predictors again. This would not hurt the communication complexity in terms of the order of magnitude since this step takes even less number of iterations. Nevertheless, in our simulation studies, TSRGA performs on par with and in many cases even outperforms strong benchmarks in the finite-sample case.

Another reason for the sub-optimality comes from the dependence on δ_n in the error. In the second-stage, TSRGA relies on the sample SVD of the (scaled) marginal covariance $\mathbf{X}_j^\top \mathbf{Y}$ to estimate the singular subspaces of the unknown coefficient matrices. How well these sample singular vectors recover their noiseless counterparts depends on the strength of the marginal covariance, which is controlled by δ_n in Assumption (C5). This is needed because we try to avoid searching for the singular subspaces of the coefficient matrices, a challenging task for greedy algorithms. Unlike the scalar case, for the multivariate linear regression the dictionary for RGA contains all rank-one matrices and therefore the geometric structure is more intricate to exploit. For example, the argument used in Ing (2020) will not work with this dictionary.

Recently, Ding et al. (2020) and Ding et al. (2021) proposed new modifications of the Frank-Wolfe algorithm that directly search within the nuclear norm ball, under the assumptions of strict complementarity and quadratic growth. These algorithms rely on solving more complicated sub-problems. To illustrate one main difference between these modifications and TSRGA, note that for the usual reduced rank regression where $\min\{d_n, q_{n,1}\} > 1$ and $p_n = 1$, one of the leading examples in Ding et al. (2020) and Ding et al. (2021), our theoretical results for TSRGA still hold (though in this case the data are not feature-distributed because p_n is only one). In this case, (C5) and (C6) automatically hold with $\delta_n \leq d_n^{1/2}/(\mu s_n^{1/2})$. Consequently, Corollary 4 implies the error is of order $O_p(\frac{s_n^2 \xi_n^2}{n^2 d_n} \log \frac{n^2 d_n}{\xi_n^2})$ using $O_p(s_n^2 \log \frac{n^2 d_n}{\xi_n^2})$ iterations, regardless of whether strict complementarity holds. This advantage precisely comes from that TSRGA uses the singular vectors of $\mathbf{X}_1^\top \mathbf{Y}$ in its updates in the second stage instead of searching over the intricate space of nuclear norm ball in each iteration.

4. Simulation experiments

In this section, we apply TSRGA to synthetic data sets and compare its performance with some existing distributed as well as centralized methods. We first examine how well TSRGA and other algorithms estimate the unknown parameters. Then we apply TSRGA to a large-

scale feature-distributed data to measure its prowess in speed. In both experiments, TSRGA delivered superior performance.

4.1 Statistical performance of TSRGA

In this subsection, we compare the effectiveness of TSRGA in estimating the parameters. Specifically, it is applied to the well-known high-dimensional linear regression and the general multi-view regression (2).

Consider first the high-dimensional linear regression model:

$$y_t = \sum_{j=1}^{p_n} \beta_j^* x_{t,j} + \epsilon_t, \quad t = 1, \ldots, n,$$

which is sparse with only $a_n = \lfloor p_n^{1/3} \rfloor$ non-zero β_j^*'s, where $\lfloor x \rfloor$ denotes the integer part of x. We also generate $\{\epsilon_t\}$ as i.i.d. t-distributed random variables with five degrees of freedom.

To estimate this model, we employ the Hydra (Richtárik and Takáč, 2016) and Hydra^2 (Fercoq et al., 2014) algorithms to solve the Lasso problem, namely,

$$\min_{\{\beta_j\}_{j=1}^{p_n}} \left\{ \frac{1}{2n} \sum_{t=1}^{n} \left(y_t - \sum_{j=1}^{p_n} \beta_j x_{t,j} \right)^2 + \lambda \sum_{j=1}^{p_n} |\beta_j| \right\}. \tag{18}$$

The predictors are divided into 10 groups at random; each of the groups is owned by one node in the Hydra-type algorithm. The step size of the Hydra-type algorithms is set to the lowest value so that we observe convergence of the algorithms instead of divergence. As a benchmark, we also solve the Lasso problem with 5-fold cross validation using `glmnet` package in R. To further reduce the computational burden, we use the λ selected by the cross-validated Lasso in implementing Hydra-type algorithms.

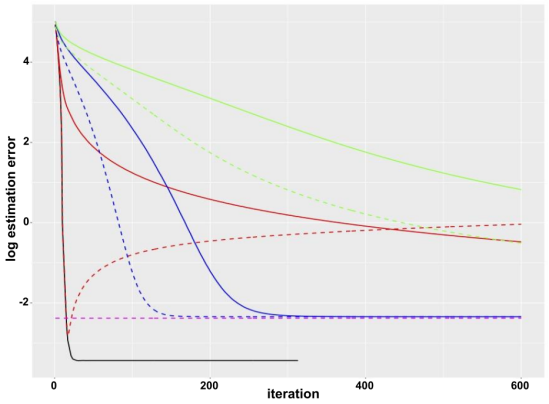
Choosing the hyperparameter for RGA-type methods is more straightforward, but there is one subtlety. It is well-known that the Lasso problem corresponds to the constrained minimization problem

$$\min_{\{\beta_j\}_{j=1}^{p_n}} \frac{1}{2n} \sum_{t=1}^{n} \left(y_t - \sum_{j=1}^{p_n} \beta_j x_{t,j} \right)^2 \text{ subject to } \sum_{j=1}^{p_n} |\beta_j| \le L_n.$$

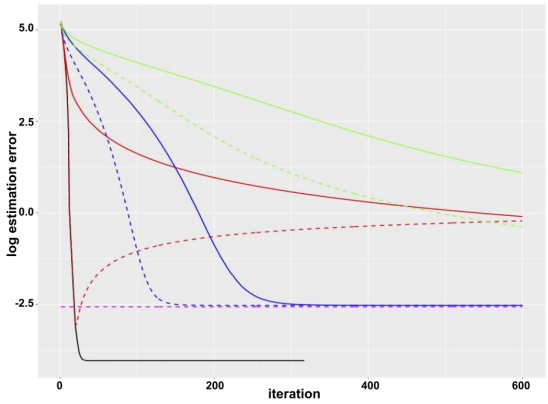
Moreover, setting L_n to $\sum_{j=1}^{p_n} |\beta_j^*|$, which is nonetheless unknown in practice, would yield the usual Lasso statistical guarantee (see, e.g., Theorem 10.6.1 of Vershynin, 2018). However, our theoretical results in Section 3.2 recommend setting L_n to a larger value than this conventionally recommended value. To illustrate the advantage of a larger L_n, we employ two versions of RGA: one with $L_n = 500$ and the other with $L_n = \sum_{j=1}^{p_n} |\beta_j^*|$. For TSRGA, we simply set $L_n = 500$ and $t_n = 1/(10 \log n)$, and the performance is not too sensitive to these choices.

For Specifications 1 and 2 below, we consider three cases with $(n, p_n) \in \{(800, 1200), (1200, 2000), (1500, 3000)\}$. In Specification 1, we simulate the predictors as independent, t-distributed data. Together with the t-distributed errors, this specification simulates the situation where heavy-tailed data are frequently observed.

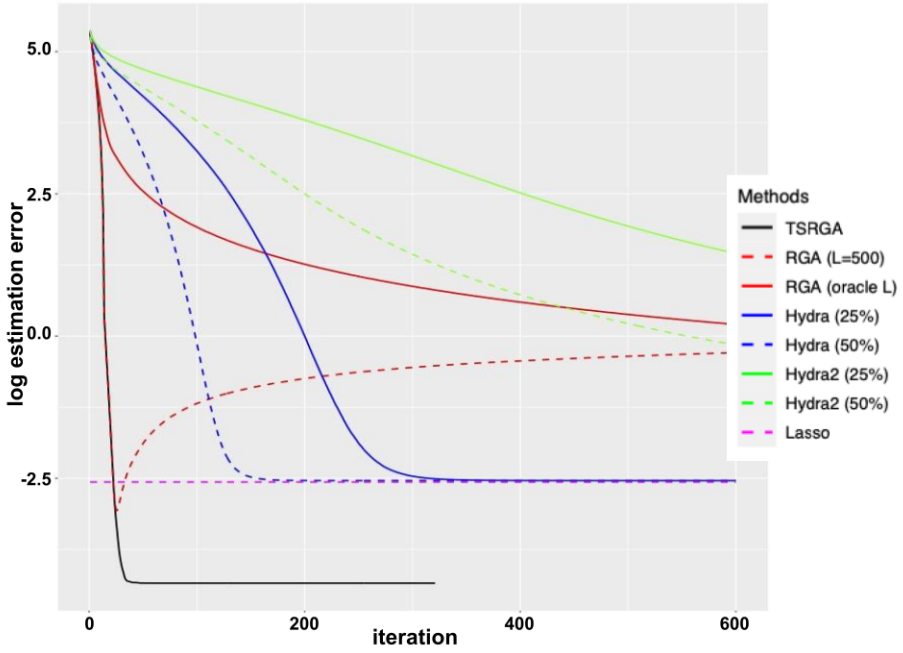
Specification 1 In the first experiment, we generate $x_{t,j}$ as i.i.d. $t(6)$ random variables, for all $t = 1, 2, \ldots, n$, and $j = 1, 2, \ldots, p_n$. Hence the predictors have heavy tails with only 6 finite moments. The nonzero coefficients are generated independently by $\beta_j^* = z_j u_j$, where z_j is uniform over $\{-1, +1\}$ and u_j is uniform over $[2.5, 5.5]$. The coefficients are drawn at the start of each of the 100 Monte Carlo simulations.



(a) $n = 800$, $p_n = 1200$



(b) $n = 1200$, $p_n = 2000$



(c) $n = 1500$, $p_n = 3000$

Figure 1: Logarithm of parameter estimation errors of various methods under Specification 1, where n is the sample size and p_n is the dimension of predictors. The results are averages of 100 simulations.

Figure 1 plots the logarithm of the parameter estimation error against the number of iterations. The parameter estimation error is defined as $\sum_{j=1}^{p_n}(\beta_j^* - \hat{\beta}_j)^2$, where $\{\hat{\beta}_j\}$ are the estimates made by the aforementioned methods. In the plot, the trajectories are averaged across 100 simulations. TSRGA (black) converges using the least number of iterations. Since the per-iteration communication costs of TSRGA and Hydra-type algorithms are similar ($O(n)$ bytes), this serves as a proxy for a smaller communication overhead of TSRGA. In addition, the parameter estimation error of TSRGA is also the smallest among the employed methods. RGA with $L_n = 500$ (dashed red) follows the same trajectories as TSRGA in the first few iterations, but without the two-step design, it suffers from over-fitting in later iterations and hence an increasing parameter estimation error. On the other hand, RGA with oracle $L_n = \sum_{j=1}^{p_n} |\beta_j^*|$ (solid red) converges much slower than TSRGA due to a sub-linear convergence rate. For Hydra (blue lines) and Hydra^2 (green lines) algorithms, we consider two implementations: updating 25% of the coordinates in each node (solid), and updating 50% of the coordinates in each node (dashed). Hydra converges to the centralized Lasso (dashed magenta) at a faster rate if 50% of the coordinates were updated in each iterations than the 25% counterparts. However, Hydra^2 converges much slower.

In the next specification, we generate the predictors so that they are correlated and the correlations are the same between any two predictors. This simulates the situation where one cannot simply divide groups of variables that have weak inter-group dependence into different computing nodes to alleviate the difficulties caused by feature-distributed data.

Specification 2 In this experiment, we generate the predictors by

$$x_{t,j} = \nu_t + w_{t,j}, \quad t = 1, \ldots, n; \quad j = 1, \ldots, p_n,$$

where $\{\nu_t\}$ and $\{w_{t,j}\}$ are independent $N(0,1)$ random variables. Consequently, $\mathrm{Cor}(x_{t,k}, x_{t,j}) = 0.5$, for $k \neq j$. The coefficients are set to $\beta_j^* = 2.5 + 1.2(j-1)$ for $j = 1, 2, \ldots, a_n = \lfloor p_n^{1/3} \rfloor$. The rest of the specification is the same as that of Specification 1.

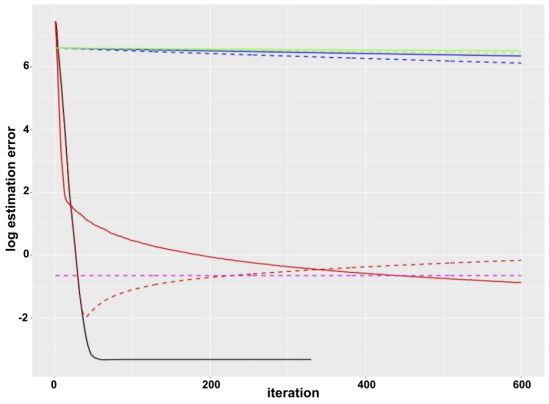
Figure 2 plots the parameter estimation errors under Specification 2. TSRGA remains the most effective method for estimating the unknown parameters, which converges within 100 iterations in all cases. It is worth noting that the Hydra-type algorithms display a substantially deteriorated rate of convergence compared to the previous specification, highlighting their sensitivity to the dependence between predictors, and potentially high computational expenses in certain scenarios.

It is also important to study the performance of these methods in terms of elapsed time and out-of-sample performance. To save space, we postpone the discussion to Appendix E, as most conclusions drawn above remain valid in examining the elapsed time and prediction performance.

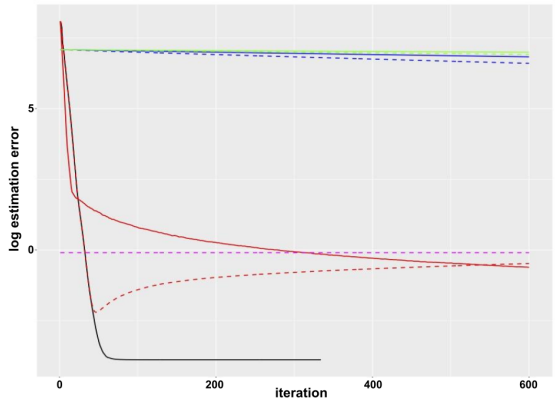
Next we consider the general model:

$$\mathbf{y}_t = \sum_{j=1}^{p_n} \mathbf{B}_j^{*\top} \mathbf{x}_{t,j} + \boldsymbol{\epsilon}_t, \quad t = 1, \ldots, n, \tag{19}$$

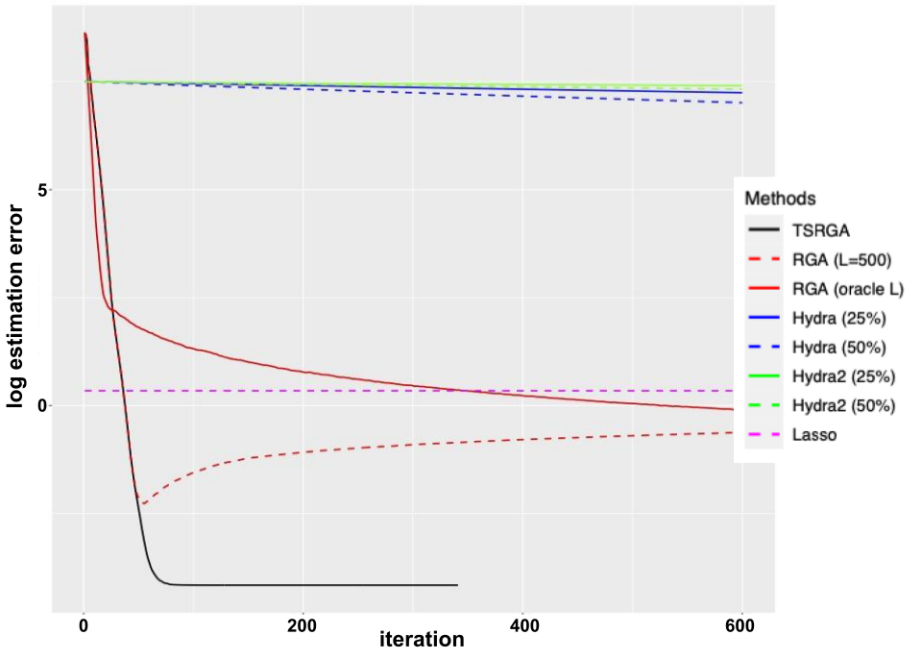
where $\mathbf{y}_t \in \mathbb{R}^{d_n}$ and $\mathbf{x}_{t,j} \in \mathbb{R}^{q_n}$, for $j = 1, 2, \ldots, p_n$. We generate $\boldsymbol{\epsilon}_t$ as i.i.d. random vectors with each entry having independent $t(5)$ distributions. In the following cases, the model is



(a) $n = 800$, $p_n = 1200$



(b) $n = 1200$, $p_n = 2000$



(c) $n = 1500$, $p_n = 3000$

Figure 2: Parameter estimation errors of various estimation methods under Specification 2, where n is the sample size and p_n is the number of predictors. The results are averages of 100 simulations.

sparse with a_n non-zero $\mathbf{B}_j^*$'s, each of which is only of rank r_n. In particular, we generate $\mathbf{B}_j^*$, for $j \leq a_n$, independently by

$$\mathbf{B}_j^* = \sum_{k=1}^{r_n} \sigma_{k,n} \mathbf{u}_{k,j} \mathbf{v}_{k,j}^\top, \tag{20}$$

where $\{\mathbf{u}_{k,j}\}_{k=1}^{r_n}$ and $\{\mathbf{v}_{k,j}\}_{k=1}^{r_n}$ are independently drawn (q_n- and d_n-dimensional) orthonormal vectors and $\{\sigma_{k,n}\}$ are i.i.d. uniform over [7,15].

We employ the iRRR method (Li et al., 2019) to estimate (19). To select its tuning parameter, we execute iRRR with a grid of tuning parameter values and opt for the one with the lowest mean square prediction error on an independently generated validation set of 500 observations. Although centralized computation is used to implement iRRR, it is too computationally demanding to implement the algorithm for the two cases with $n = 600$ and $n = 1200$. Therefore, we use the least squares estimator with only the relevant variables as another benchmark. For TSRGA, L_n is set to 10^5, and we hold one third of the training data as validation set to select the tuning parameter t_n for TSRGA over a grid of values[1].

Since iRRR is not a feature-distributed algorithm, we directly report their parameter estimation errors (averaged across 500 Monte Carlo simulations) defined as

$$\sqrt{\sum_{j=1}^{p_n} \|\mathbf{B}_j^* - \hat{\mathbf{B}}_j\|_F^2}, \tag{21}$$

where $\{\hat{\mathbf{B}}_j\}$ are the estimated coefficient matrices. Additionally, the out-of-sample prediction performance of these methods are evaluated on an independent test sample of size 500, measured by $(\|\mathbf{Y} - \hat{\mathbf{Y}}\|_F^2/(nd_n))^{1/2}$. We consider the cases $(n, d_n, q_n, p_n, a_n, r_n) \in \{(200, 10, 12, 20, 1, 2), (400, 15, 18, 50, 2, 2), (600, 20, 25, 400, 3, 2), (1200, 40, 45, 800, 3, 3)\}$.

Specification 3 In this specification, we consider (19) with the predictors generated as in Specification 1. Note that $\{\mathbf{B}_j^* : j \le a_n\}$ are drawn at the start of each of the 500 Monte Carlo simulations.

Table 1 reports the results of the methods averaged over 500 Monte Carlo simulations of data generated under Specification 3. TSRGA achieved the lowest estimation error in all constellations of problem sizes. On the other hand, iRRR yielded larger estimation error than the least squares method using exactly the relevant predictors when $n = 200$, but when n increases, iRRR outperforms the least squares method. However, the computational costs of iRRR became so high that completing 500 simulations would require more than days, even when parallelism with 15 cores is used. TSRGA circumvents such computational overhead and delivers superior estimates. The figures in the parentheses are p-values for testing equality between each method's errors and those of the least squares. They show that TSRGA indeed significantly outperforms the least squares in parameter estimation. Nevertheless, the difference in prediction capabilities is less discernible.

Specification 4 In this specification, we generalize (19) to group predictors as follows. Let $\{\boldsymbol{\nu}_t : t = 1, 2, \ldots\}$ and $\{\mathbf{w}_{t,j} : t = 1, 2, \ldots; j = 1, 2, \ldots, p_n\}$ be independent $N(\mathbf{0}, \mathbf{I}_{q_n})$ random vectors. The group predictors are then constructed as $\mathbf{x}_{t,j} = 2\boldsymbol{\nu}_t + \mathbf{w}_{t,j}$, $1 \le t \le n$, $1 \le j \le p_n$. Hence $\mathbb{E}(\mathbf{x}_{t,j}\mathbf{x}_{t,i}^\top) = 4\mathbf{I}_{q_n}$, for $1 \le i < j \le p_n$. Note that $\mathrm{Corr}(x_{t,i,l}, x_{t,j,l}) = 0.8$ for $i \ne j$, $1 \le l \le q_n$, where $\mathbf{x}_{t,i} = (x_{t,i,1}, \ldots, x_{t,i,q_n})^\top$. Hence, the l-th components in each of the group predictors are highly correlated.

1. t_n is selected among $\mathbf{t} = (0.01, 0.07, 1.10, 1.39, 1.61, 1.79, 1.95, 2.08, 2.20, 2.30)/\log n$.

	Parameter estimation			Prediction		
$(n, d_n, q_n, p_n, a_n, r_n)$	TSRGA	iRRR	Oracle LS	TSRGA	iRRR	Oracle LS
(200, 10, 12, 20, 1, 2)	0.666	0.929	0.851	1.318	1.339	1.331
	(0.000)	(0.000)		(0.000)	(0.000)	
(400, 15, 18, 50, 2, 2)	0.858	1.245	1.287	1.322	1.351	1.355
	(0.000)	(0.000)		(0.000)	(0.000)	
(600, 20, 25, 400, 3, 2)	1.223	-	1.787	1.361	-	1.381
	(0.000)			(0.158)		
(1200, 40, 45, 800, 3, 3)	1.388	-	2.378	1.345	-	1.371
	(0.000)			(0.069)		

Table 1: Parameter estimation and prediction errors of various methods under Specification 3. Figures in the parentheses are the p-values for testing if the errors are different from their Oracle LS counterparts. We do not report the results for iRRR with sample sizes of 600 and 1200 since the computation required for these cases is excessively time-consuming. In the table, n, d_n, q_n, p_n, a_n and r_n are the sample size, number of targeted variables, dimension of predictors, number of predictors, number of non-zero coefficient matrices, and rank of coefficient matrices, respectively. The results are based on 500 simulations.

	Parameter estimation			Prediction		
$(n, d_n, q_n, p_n, a_n, r_n)$	TSRGA	iRRR	Oracle LS	TSRGA	iRRR	Oracle LS
(200, 10, 12, 20, 1, 2)	0.401	0.616	0.460	1.324	1.337	1.330
	(0.000)	(0.000)		(0.000)	(0.000)	
(400, 15, 18, 50, 2, 2)	0.562	0.993	1.172	1.345	1.344	1.354
	(0.000)	(0.000)		(0.584)	(0.000)	
(600, 20, 25, 400, 3, 2)	0.812	-	1.817	1.362	-	1.379
	(0.000)			(0.478)		
(1200, 40, 45, 800, 3, 3)	0.751	-	2.419	1.310	-	1.371
	(0.000)			(0.000)		

Table 2: Parameter estimation and prediction errors under Specification 4. Figures in the parentheses are the p-values for testing if the errors are different from their Oracle LS counterparts. We do not report the results for iRRR with sample sizes of 600 and 1200 since the computation required for these sample sizes is excessively time-consuming. The same notations as those of Table 1 are used. The results are based on 500 simulations.

Table 2 reports the results for Specification 4. As in the previous specifications, TSRGA continues to surpass the benchmarks. When $n = 400$, iRRR gains an advantage over the least squares method, despite of a high computational cost. The results in Tables 1 and 2

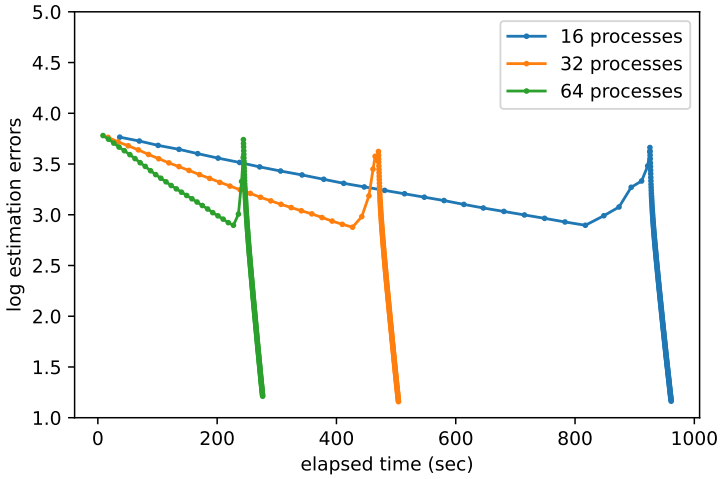


Figure 3: Logarithm of the average parameter estimation errors at each iteration of TSRGA, plotted against the average time elapsed at the end of each iteration. Various number of processes are employed for feature-distributed implementation. 10 simulations are used.

suggest that TSRGA is both a fast and a statistically effective tool for parameter estimation for model (19).

4.2 Large-scale performance of TSRGA

In this subsection, we apply TSRGA to large feature-distributed data. We have an MPI implementation of TSRGA through OpenMPI and the `Python` binding `mpi4py` (Dalcín et al., 2005; Dalcín and Fang, 2021). The algorithm runs on the high-performance computing cluster of the university, which comprises multiple computing nodes equipped with Intel Xeon Gold 6248R processors. We consider again Specification 4 in the previous subsection, with $(n, d_n, q_n, p_n, a_n, r_n) = (20000, 100, 100, 1024, 4, 4)$. In the following experiments we employ $M/4$ nodes, each of which runs 4 processes and each process owns p_n/M predictors, with M varying from 16 to 64. When combined, the data are approximately over 16 GB of size, exceeding the usual RAM capacity on most laptops.

There are two primary goals for the experiments. The first goal is to investigate the wall-clock time required by TSRGA to estimate (19). The second goal is to examine the effect of the number of nodes on the required wall-clock time. Each experiment is repeated 10 times, and we average the wall-clock time needed to complete the k-th iteration as well as the parameter estimation error (21) at the k-th iteration.

Figure 3 plots the (log) estimation errors against the wall-clock time of TSRGA iterations. When using 16 processes, TSRGA took about 16 minutes to estimate (19), and the time reduced to less than 5 minutes when 64 processes were employed. The acceleration primarily occurred in the first stage, because solving (3) becomes faster when each process

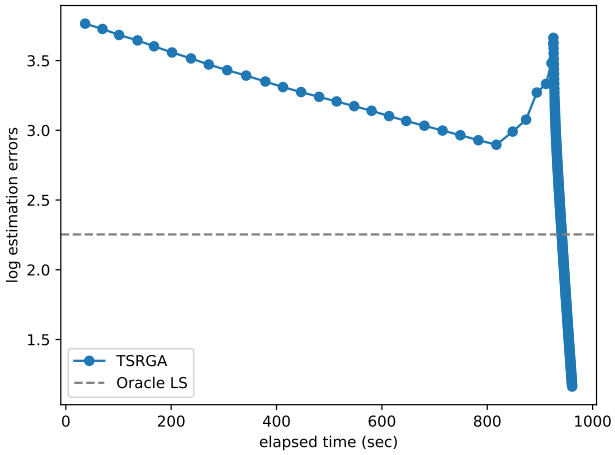


Figure 4: Logarithm of the estimation errors of TSRGA (running with 16 processes) and the oracle least squares. The oracle least squares method is performed by applying the second-stage RGA with exactly the relevant predictors and no rank constraints. 10 simulations are used.

handles only a small number of predictors. After screening, there is a spike in estimation error due to re-initialization of the estimators but subsequent second-stage RGA runs extremely fast in all cases and yields accurate estimates. Indeed, Figure 4 shows that the estimation error of TSRGA quickly drops below that of the oracle least squares in the second stage. We remark that with more diligent programming, one can apply the advanced protocols introduced in Section 6 of Richtárik and Takáč (2016) to TSRGA, using both multi-process and multi-thread techniques. It is anticipated that the required time will be further shortened.

5. Empirical application

This section showcases an application of TSRGA to financial data. In addition to the conventional financial data, we further collect the annual 10-K reports of firms under study to extract useful features for augmenting the predictors. Thus, in this application, both the response and predictors are multivariate, and the predictors may consist of large dense matrices, leading to potential computational challenges in practice.

5.1 Financial data and 10-K reports

We aim to predict four key financial outcomes for companies in the S&P 500 index: volatility, trading volume, market beta, and return. We obtain daily return series for each company from 2010 through 2019, calculate the sample variances of the daily returns in each month, and transform them by taking the logarithm to get the volatility series $\{V_{it}(m) : m = 1, 2, \ldots, 12\}$ for the i-th company in the m-th month of year $t \in \{2010, \ldots, 2019\}$. Next,

we regress each company's daily returns on the daily returns of the S&P 500 index for each month and use the slope estimates as market beta, $\{B_{it}(m) : m = 1, 2, \ldots, 12\}$. Finally, we also obtain data of the monthly returns series $\{R_{it}(m) : m = 1, 2, \ldots, 12\}$ and the logarithm of the trading volumes $\{M_{it}(m) : m = 1, 2, \ldots, 12\}$, for the ith company. All series are obtained from Yahoo! Finance via the `tidyquant` package in R.

After obtaining these series, some data cleaning is performed to facilitate subsequent analysis. First, the volume series exhibits a high degree of serial dependence, which could be due to unit-roots caused by the persistence in trading activities. Therefore, we apply a year-to-year difference, i.e., $\Delta M_{it}(m) = M_{i,t}(m) - M_{i,t-1}(m)$ for all i, $1 \leq m \leq 12$, and $t = 2011, \ldots, 2019$. Additionally, we remove companies that have outlying values in these series.

In addition to these financial time series, we also make use of the information from a pertinent collection of textual data: 10-K reports. Publicly traded companies in the U.S. are required to file these annual reports with the aim of increasing transparency and satisfying the regulation of exchanges. The reports are maintained by the Securities and Exchange Commission (SEC) in the Electronic Data Gathering, Analysis, and Retrieval system (EDGAR), and provide information about a company's risks, liabilities, and corporate agreements and operations. Due to their significance in communicating information to the public, the 10-K reports are an important corpus in finance, economics, and computational social sciences studies (Hanley and Hoberg, 2019; Kogan et al., 2009; Gandhi et al., 2019; Jegadeesh and Wu, 2013).

The corpus utilized in this application is sourced from the EDGAR-CORPUS, originally prepared by Loukas et al. (2021). Our analysis specifically focuses on Section 7, titled "Management's Discussion and Analysis." To process the reports, we preprocess each document using the default functionality in the `gensim` package in Python and discard the documents that consist of fewer than 50 tokens. As a result, we have data of both the financial time series and 10-K reports of 256 companies over the period from 2011 through 2019.

To extract features from the textual data, we employ a technique called Latent Semantic Indexing (LSI, see, e.g., Deerwester et al., 1990). We first construct the term-document matrix as follows. Suppose we have D documents in the training set, and there are V distinct tokens in these documents. The term-document matrix $\boldsymbol{\Theta}$ is a $V \times D$ matrix, whose entries are given by

$$\boldsymbol{\Theta}_{ij} = (\text{number of times the } i\text{-th token appears in document } j) \times \log \frac{D}{\sharp\{1 \leq k \leq D : \text{the } i\text{-th token appears in document } k\}},$$

for $1 \leq i \leq V$, $1 \leq j \leq D$. The entries are known as one form of the term-frequency inverse document frequency (TFIDF, see, e.g., Salton and Buckley, 1988). Then, to extract K features from the text data, LSI uses the singular value decomposition,

$$\boldsymbol{\Theta} = \mathbf{U}_{\boldsymbol{\Theta}} \boldsymbol{\Sigma}_{\boldsymbol{\Theta}} \mathbf{V}_{\boldsymbol{\Theta}}^{\top},$$

and the first K rows of $\boldsymbol{\Sigma}_{\boldsymbol{\Theta}} \mathbf{V}_{\boldsymbol{\Theta}}^{\top}$ are used as the features in the training set. For a new document in the test set, we compute its TFIDF representation $\boldsymbol{\theta} \in \mathbb{R}^V$, and then use $\mathbf{x} = \mathbf{U}_K^{\top} \boldsymbol{\theta}$ as its textual features, where $\mathbf{U}_K$ is the sub-matrix of the first K columns of $\mathbf{U}_{\boldsymbol{\Theta}}$.

5.2 Results

For each of the four financial response variables, we estimate the following model.

$$\mathbf{y}_{it} = \boldsymbol{\beta}_0 + \mathbf{A}_1^\top \mathbf{v}_{i,t-1} + \mathbf{A}_2^\top \mathbf{m}_{i,t-1} + \mathbf{A}_3^\top \mathbf{b}_{i,t-1} + \mathbf{A}_4^\top \mathbf{r}_{i,t-1} + \mathbf{A}_5^\top \mathbf{x}_{i,t-1} + \boldsymbol{\epsilon}_{it}, \tag{22}$$

where $\mathbf{y}_{it} = (y_{it}(1), \ldots, y_{it}(12))^\top$ is the response variable under study, $\mathbf{v}_{it} = (V_{it}(1), \ldots, V_{it}(12))^\top$, $\mathbf{m}_{it} = (\Delta M_{it}(1), \ldots, \Delta M_{it}(12))^\top$, $\mathbf{b}_{it} = (B_{it}(1), \ldots, B_{it}(12))^\top$, $\mathbf{r}_{it} = (R_{it}(1), \ldots, R_{it}(12))^\top$, $\mathbf{x}_{it} \in \mathbb{R}^K$ is the extracted text features, and $\{\boldsymbol{\beta}_0, \mathbf{A}_1, \ldots, \mathbf{A}_5\}$ are unknown parameters. When predicting each of the four financial outcomes, we replace $\mathbf{y}_{it}$ in (22) with the corresponding vector ($\mathbf{v}_{it}$, $\mathbf{m}_{it}$, $\mathbf{b}_{it}$, or $\mathbf{r}_{it}$), while keeping the same model structure. Since predicting next-year's financial outcomes in one month is related to predicting the same variable in other months, it is natural to expect low-rank coefficient matrices. (22) can also be viewed as a multi-step ahead prediction model, since we are predicting the next twelve months simultaneously.

When applying TSRGA to (22), we use a hold-out validation set from the training sample to select the just-in-time threshold t_n from the grid $(0.1, 0.2, \ldots, 1.0)/\log n$. In addition to TSRGA, we employ several benchmark prediction methods, including the vector autoregression (VAR), reduced rank regression (RR; see, e.g., Chen et al., 2013), the integrative reduced rank regression (iRRR, Li et al., 2019), and the Lasso. For VAR, we concatenate all response variables and estimate the model

$$\mathbf{z}_{it} = \mathbf{A}^\top \mathbf{z}_{i,t-1} + \mathbf{e}_{it},$$

where $\mathbf{z}_{it} = (\mathbf{v}_{it}^\top, \mathbf{m}_{it}^\top, \mathbf{b}_{it}^\top, \mathbf{r}_{it}^\top)^\top \in \mathbb{R}^{48}$. Alternatively, we can implement VAR in a group-wise fashion (gVAR henceforth). Specifically, we separately estimate the model

$$\mathbf{y}_{it} = \mathbf{A}^\top \mathbf{y}_{i,t-1} + \mathbf{e}_{it}, \tag{23}$$

for each response variable $\mathbf{y}_{it} \in \{\mathbf{v}_{it}, \mathbf{m}_{it}, \mathbf{b}_{it}, \mathbf{r}_{it}\}$. The reduced rank regression also estimates (23) with an intercept term and an additional rank constraint on the coefficient matrix $\mathbf{A}$ in (23). We use the generalized cross validation (GCV, Gene H. Golub and Wahba, 1979) to select the optimal rank. For Lasso, it is applied separately to each row of (22); namely, it estimates

$$\begin{aligned} y_{it}(m) = \beta_0 + \sum_{j=1}^{12} \alpha_{j,1} V_{i,t-1}(j) + \sum_{j=1}^{12} \alpha_{j,2} \Delta M_{i,t-1}(j) \\ + \sum_{j=1}^{12} \alpha_{j,3} B_{i,t-1}(j) + \sum_{j=1}^{12} \alpha_{j,4} R_{i,t-1}(j) + \epsilon_{it}, \end{aligned}$$

for $m = 1, 2, \ldots, 12$. Finally, we also apply the iRRR method of Li et al. (2019) to (22).

Table 3 presents the root mean squared prediction errors (RMSE) for different methods on the test set, for which we reserved the last year of data. The results show that gVAR consistently outperformed the usual VAR in all four financial variables, suggesting using simple least squares could be harmful in prediction when including many financial series as predictors. RR provides a slight improvement in predicting volatility, but performs

similarly as VAR and gVAR in predicting other targets. In the case of predicting volatility, the text data proved to be quite useful, and TSRGA, iRRR, and Lasso have all outperformed gVAR by more than 5% with different number of textual features K (except for Lasso with $K = 50$). TSRGA and iRRR, utilizing both the text information and low-rank coefficient estimates, yielded the smallest prediction errors. In some cases, they achieved 10% reduction in RMSE compared with gVAR and RR. In Appendix E, we also report the p-values for testing if the methods outperform gVAR for volatility and return (see Tables 5 and 6). The result suggests the advantages of Lasso, iRRR, and TSRGA in predicting volatility over gVAR are quite significant, whereas the differences in predicting the return are not as strong.

	Volatility	Volume	Beta	Return
VAR	0.782	0.323	0.583	0.077
gVAR	0.750	0.319	0.556	0.073
RR	0.732	0.325	0.555	0.071
$K = 50$				
Lasso	0.718	0.310	0.574	0.075
iRRR	**0.688**	0.318	0.568	0.072
TSRGA	**0.702**	0.345	0.572	0.072
$K = 100$				
Lasso	**0.700**	0.308	0.574	0.074
iRRR	**0.677**	0.316	0.566	0.072
TSRGA	**0.678**	0.330	0.571	0.072
$K = 150$				
Lasso	**0.693**	0.308	0.571	0.073
iRRR	**0.667**^{a}	0.314	0.566	0.072
TSRGA	**0.681**	0.332	0.573	0.072
$K = 200$				
Lasso	**0.684**	0.309	0.574	0.073
iRRR	**0.663**^{a}	0.314	0.567	0.072
TSRGA	**0.654**^{a,b}	0.345	0.574	0.072

Table 3: Root mean squared prediction errors on the test dataset. Entries in boldface are at least 5% below gVAR; ^{a} means 10% below gVAR, and ^{b} means 10% below RR.

In addition to the prediction performance, we make two more remarks on the empirical results. First, our finding that textual features are useful in predicting volatility is consistent with previous studies. For instance, Kogan et al. (2009) reported that one-hot text features are already effective in predicting volatility in a scalar linear regression, and Yeh et al. (2020) also observed gains of using neural word embedding to predict volatility. Our results suggest an alternative modeling choice: text data could explain each month's volatility via a low-rank channel. Second, low-rank models may not be suited for the trading volume series. The RR selected a full-rank model and TSRGA iterated more steps before the just-in-time stopping criterion was triggered.

The data set used in the application is relatively small, and can fit in most personal computer's memory. However, incorporating more sections of the 10-K reports or other financial corpus may pose computational challenges due to the increased number of dense text feature matrices. TSRGA can easily handle such cases when feature-distributed data are inevitable.

6. Horizontal partition for big feature-distributed data

In this section, we briefly discuss the usage of TSRGA when the sample size n, in addition to the dimension p_n, is also large so that storing $(\mathbf{Y}, \mathbf{X}_j)$ in one machine is infeasible. In this case, we also horizontally partition the (feature-distributed) data matrices and employ more computing nodes.

To fix ideas, for $h = 1, 2, \ldots, H$, let

$$\mathbf{Y}_{(h)} = (\mathbf{y}_{m_{h-1}+1}, \ldots, \mathbf{y}_{m_h})^\top, \text{ and } \quad \mathbf{X}_{j,(h)} = (\mathbf{x}_{m_{h-1}+1,j}, \ldots, \mathbf{x}_{m_h,j})^\top$$

be horizontal partitions of $\mathbf{Y}$ and $\mathbf{X}_j$, $j = 1, \ldots, p_n$, where $0 = m_0 < m_1 < \ldots < m_H = n$. In the distributed computing system, we label the nodes by (h, c), so that the (h, c)-th node owns data $\mathbf{Y}_{(h)}$ and $\{\mathbf{X}_{j,(h)} : j \in \mathcal{I}_c\}$, where $h \in [H]$, $c \in [M]$ and $\cup_{c\in[M]}\mathcal{I}_c = [p_n]$. For ease in illustration, we further assume $\{\mathcal{I}_c : c \in [M]\}$ forms a partition of $[p_n]$. Therefore, each computing node only owns a slice of the samples on a subset $\mathcal{I}_c$ of the predictors as well as the same slice of the response variables. Moreover, let $I(j) = \{(h, c) : j \in \mathcal{I}_c\}$ be the indices of the nodes that have some observations of predictor j.

We call the nodes that own the h-th slice of data "segment h". That is, $\{(k, c) : k = h\}$. Note that each segment is essentially the feature-distributed framework discussed in the previous sections. In what follows, quantities computed at nodes in segment h carry a subscript (h). For example, $\hat{\mathbf{\Sigma}}_{j,(h)} = n_h^{-1}\mathbf{X}_{j,(h)}^\top\mathbf{X}_{j,(h)}$, where $n_h = m_h - m_{h-1}$. For simplicity, we also assume $n_1 = \ldots = n_H$ in this section. Finally, we again assume there is at least one master node to coordinate all the computing nodes $\{(h, c) : h \in [H], c \in [M]\}$.

To estimate (2) with the horizontally partitioned feature-distributed data described above, we suggest the following procedure. First, we obtain a set of potentially relevant predictors $\hat{J}$ and their respective upper bounds on the coefficient ranks $\hat{r}_j$ by running the first-stage RGA with the just-in-time stopping criterion. This can be done by applying Algorithm 1 to one segment. Alternatively, one can apply it to multiple segments in parallel and set $\hat{J} = \cap_h \hat{J}_{(h)}$ and $\hat{r}_j = \min_h \hat{r}_{j,(h)}$. In either case, Theorem 1 ensures the sure-screening property as $n_1 \to \infty$ if (C1)-(C4) hold in each of the segments. By Lemma 2, this step costs $O_p(s_n^2(n_1 + d_n))$ bytes of communication per node in the segment(s) involved.

Next, for each $j \in \hat{J}$, each node $(h, c) \in I(j)$ computes $\mathbf{X}_{j,(h)}^\top\mathbf{X}_{j,(h)}$ and, if $q_{n,j} \wedge d_n > \hat{r} = \sum_j \hat{r}_j$, additionally computes $\mathbf{X}_{j,(h)}^\top\mathbf{Y}_{(h)}$. Then, send these matrices to the master node. The master node computes $\hat{\mathbf{\Sigma}}_j^{-1} = (\sum_{h=1}^H \mathbf{X}_{j,(h)}^\top\mathbf{X}_{j,(h)})^{-1}$ and the leading $\hat{r}$ singular vectors of $\sum_{h=1}^H \mathbf{X}_{j,(h)}^\top\mathbf{Y}_{(h)}$, which form the column vectors of $\mathbf{U}_j$ and $\mathbf{V}_j$. Then $(\hat{\mathbf{\Sigma}}_j^{-1}, \mathbf{U}_j, \mathbf{V}_j)$ (or just $\hat{\mathbf{\Sigma}}_j^{-1}$ if $q_{n,j} \wedge d_n \le \hat{r}$) are sent back to $I(j)$. This step costs $O_p(\sum_{j\in\hat{J}}\{q_{n,j}^2 + (q_{n,j}d_n + \hat{r}(q_{n,j} + d_n))\mathbf{1}\{q_{n,j} \wedge d_n > \hat{r}\}\}$ bytes of communication per node.

Now we can start the second-stage RGA iterations. Initializing $\hat{\mathbf{G}}_{(h)}^{(0)} = \mathbf{0}$ and $\hat{\mathbf{U}}_{(h)}^{(0)} = \mathbf{Y}_{(h)}$ for each computing nodes. At iteration k, for each $j \in \hat{J}$, nodes in $I(j)$ send

$\mathbf{U}_j^\top \hat{\mathbf{\Sigma}}_j^{-1} \mathbf{X}_{j,(h)}^\top \hat{\mathbf{U}}_{(h)}^{(k-1)} \mathbf{V}_j$ to the master. The master aggregates the matrices

$$\left\{ \mathbf{P}_j = \sum_{h=1}^{H} \mathbf{U}_j^\top \hat{\mathbf{\Sigma}}_j^{-1} \mathbf{X}_{j,(h)}^\top \hat{\mathbf{U}}_{(h)}^{(k-1)} \mathbf{V}_j : j \in \hat{J} \right\},$$

and decides $\hat{j}_k = \arg\max_{j \in \hat{J}} \sigma_1(\mathbf{P}_j)$ and $\hat{\mathbf{S}}_k = L_n \mathbf{u}\mathbf{v}^\top$, where $(\mathbf{u}, \mathbf{v})$ are the leading singular vectors of $\mathbf{P}_{\hat{j}_k}$. The master node sends $\hat{\mathbf{S}}_k$ to the nodes in $I(\hat{j}_k)$. Sending the matrix $\mathbf{U}_j^\top \hat{\mathbf{\Sigma}}_j^{-1} \mathbf{X}_{j,(h)}^\top \hat{\mathbf{U}}_{(h)}^{(k-1)} \mathbf{V}_j$ requires $O(\hat{r}^2)$ bytes of communication if $q_{n,j} \wedge d_n > \hat{r}$, and $O(q_{n,j} d_n)$ bytes otherwise. Each computing node also receives $O(\hat{r})$ or $O(q_{n,j} + d_n)$ bytes of data from the master, depending on whether $q_{n,\hat{j}_k} \wedge d_n$ is greater than $\hat{r}$.

To compute $\hat{\lambda}_k$, each node $(h, c) \in I(\hat{j}_k)$ computes and sends to the master

$$\mathbf{A}_h = \hat{\mathbf{U}}_{(h)}^{(k-1)\top} \mathbf{X}_{\hat{j}_k,(h)} \hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1} \mathbf{U}_{\hat{j}_k} \hat{\mathbf{S}}_k \mathbf{V}_{\hat{j}_k}^\top - \hat{\mathbf{U}}_{(h)}^{(k-1)\top} \hat{\mathbf{G}}_{(h)}^{(k-1)},$$

and

$$a_h = \| \mathbf{X}_{\hat{j}_k,(h)} \hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1} \mathbf{U}_{\hat{j}_k} \hat{\mathbf{S}}_k \mathbf{V}_{\hat{j}_k}^\top - \hat{\mathbf{G}}_{(h)}^{(k-1)} \|_F^2.$$

The master then is able to compute $\hat{\lambda}_k = \max\{\min\{\hat{\lambda}_{k,uc}, 1\}, 0\}$, where

$$\hat{\lambda}_{k,uc} = \frac{\mathrm{tr}(\sum_{h=1}^{H} \mathbf{A}_h)}{\sum_{h=1}^{H} a_h}.$$

Subsequently, $\hat{\lambda}_k$ is sent to all nodes. In this step, because $\hat{\mathbf{G}}_h^{(k-1)}$ is of rank at most $k-1$, sending $\mathbf{A}_h$ costs $O(d_n(k \wedge d_n))$ bytes of communication.

Finally, each node $(h, c) \in I(\hat{j}_k)$ updates

$$\begin{aligned}
\hat{\mathbf{G}}_{(h)}^{(k)} =& (1 - \hat{\lambda}_k) \hat{\mathbf{G}}_{(h)}^{(k-1)} + \hat{\lambda}_k \mathbf{X}_{\hat{j}_k,(h)} \hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1} \mathbf{U}_{\hat{j}_k} \hat{\mathbf{S}}_k \mathbf{V}_{\hat{j}_k}^\top, \\
\hat{\mathbf{U}}_{(h)}^{(k)} =& \mathbf{Y}_{(h)} - \hat{\mathbf{G}}_{(h)}^{(k)}, \\
\hat{\mathbf{B}}_{\hat{j}_k}^{(k)} =& (1 - \hat{\lambda}_k) \hat{\mathbf{B}}_{\hat{j}_k}^{(k-1)} + \hat{\lambda}_k \hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1} \mathbf{U}_{\hat{j}_k} \hat{\mathbf{S}}_k \mathbf{V}_{\hat{j}_k}^\top, \\
\hat{\mathbf{B}}_j^{(k)} =& (1 - \hat{\lambda}_k) \hat{\mathbf{B}}_j^{(k-1)}, \quad j \in \mathcal{I}_c - \{\hat{j}_k\},
\end{aligned}$$

and also sends (possibly via the master node) the matrix $\mathbf{X}_{\hat{j}_k,(h)} \hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1} \mathbf{U}_{\hat{j}_k} \hat{\mathbf{S}}_k \mathbf{V}_{\hat{j}_k}^\top$ (which is of rank one and costs $O(n_1 + d_n)$ bytes of communication) to the nodes $\{(h, c') : c' \neq c\}$. Then the node $(h, c') \notin I(\hat{j}_k)$ is able to update $\hat{\mathbf{G}}_{(h)}^{(k)}$, $\hat{\mathbf{U}}_{(h)}^{(k)}$, and $\hat{\mathbf{B}}_j^{(k)}$ as above.

It can be verified the above procedure implements the second-stage RGA. Moreover, the communication cost for node (h, c) at the k-th iteration is at most

$$O\left(\sum_{j \in \hat{J} \cap \mathcal{I}_c} \left(\hat{r}^2 \mathbf{1}\{q_{n,j} \wedge d_n > \hat{r}\} + q_{n,j} d_n \mathbf{1}\{q_{n,j} \wedge d_n \leq \hat{r}\} \right) + d_n k + n_1 \right).$$

As a result, the above procedure to implement TSRGA has the following guarantee.

Corollary 5 *Suppose* $\hat{J}$ *and* $\{\hat{r}_j : j \in \hat{J}\}$ *satisfy the sure-screening property* (12) *as* $n_1 \to \infty$*, and assume (C1)-(C6). If* $\max_{1\le j\le p_n} q_{n,j} = O(n_1^{\alpha})$*, then the above procedure achieves an error of order*

$$\frac{1}{d_n}\sum_{j=1}^{p_n} \|\mathbf{B}_j^* - \hat{\mathbf{B}}_j\|_F^2 = O_p\left(\frac{\mathfrak{s}_n \xi_n^2}{n^2 d_n}\log\frac{n^2 d_n}{\xi_n^2} + \frac{\xi_n^2}{n^2\delta_n^2}\mathbf{1}\{J_o \neq \emptyset\}\right)$$

with a communication complexity per computing node of order

$$O_p\left(n_1^{\max\{2\alpha,1\}} s_n^2 + (s_n^2 n_1^{\alpha} d_n + n_1)\log\frac{n^2 d_n}{\xi_n^2} + s_n^{10}\log\frac{n^2 d_n}{\xi_n^2} + d_n s_n^8\left(\log\frac{n^2 d_n}{\xi_n^2}\right)^2\right).$$

The proof of Corollary 5 is an accounting on the communication costs shown above, whose details are relegated to Appendix C. The communication complexity is still free of the ambient dimension p_n, but the dimension of the predictors $\max_{1\le j\le p_n} q_{n,j}$ comes into play, which was not a factor in the purely feature-distributed case. The additional communication between segments could inflate the communication complexity compared to the purely feature-distributed case. If $\alpha \le 0.5$ and $s_n = O(1)$, the communication complexity, up to poly-logarithmic factors, reduces to $O_p(n_1 + n_1^{\alpha} d_n + d_n)$, which is no larger than the purely feature-distributed case $O_p(n_1 + d_n)$ if $d_n = O(n_1^{1-\alpha})$. On the other hand, if $\alpha > 0.5$ and $s_n = O(1)$, the communication complexity becomes $O_p(n_1^{2\alpha} + n_1^{\alpha} d_n)$ (again ignoring poly-logarithmic terms), which is higher than the purely feature-distributed case. These costs are incurred in the greedy search as well as in the determination of $\hat{\lambda}_k$. Finally, we note that the above procedure is sequential, and certain improvements can be achieved with some carefully designed communication protocol. However, methods or algorithms for speeding up convergence or lowering communication of the proposed TSRGA with horizontal partition is left for future research.

7. Conclusion

This paper presented a two-stage relaxed greedy algorithm (TSRGA) for estimating high-dimensional multivariate linear regression models with feature-distributed data. Our main contribution is that the communication complexity of TSRGA is independent of the feature dimension, which is often very large in feature-distributed data. Instead, the complexity depends on the sparsity of the underlying model, making the proposed approach a highly scalable and efficient method for analyzing large data sets. We also briefly discussed applying TSRGA to huge data sets that require both vertical and horizontal partitions.

We would like to point out a possible future extension. In some applications, it is of paramount importance to protect the privacy of each node's data. Thus, modifying TSRGA so that privacy can be guaranteed for feature-distributed data is an important direction for future research.

Acknowledgments

We acknowledge the University of Chicago Research Computing Center for support of this work.

Appendix A. Second-stage RGA with feature-distributed data

The following algorithm presents the pseudo-code for the implementation of the second-stage RGA with feature-distributed data.

Algorithm 2: Feature-distributed second-stage RGA

Input: Number of required iterations K_n, $L_n > 0$, pre-selected $\hat{J}$.
Output: Each worker $1 \le c \le M$ has the coefficient matrices $\{\hat{\mathbf{B}}_j : j \in \mathcal{I}_c\}$ to use for prediction.
Initialization: $\hat{\mathbf{B}}_j = \mathbf{0}$, for all j, and $\hat{\mathbf{G}}^{(0)} = \mathbf{0}$

for $k = 1, 2, \ldots, K_n$ **do**
  Workers $c = 1, 2, \ldots, M$ **in parallel do**
    if $k > 1$ **then**
      Receive $(c^*, \hat{\lambda}_{k-1}, \sigma_{\hat{j}_{k-1}}, \mathbf{u}_{\hat{j}_{k-1}}, \mathbf{v}_{\hat{j}_{k-1}})$ from the master.
      $\hat{\mathbf{G}}^{(k-1)} = (1 - \hat{\lambda}_{k-1})\hat{\mathbf{G}}^{(k-2)} + \hat{\lambda}_{k-1}\sigma_{\hat{j}_{k-1}}\mathbf{u}_{\hat{j}_{k-1}}\mathbf{v}_{\hat{j}_{k-1}}^\top$.
      $\hat{\mathbf{B}}_j = (1 - \hat{\lambda}_{k-1})\hat{\mathbf{B}}_j$ for $j \in \mathcal{I}_c \cap \hat{J}$.
      if $c = c^*$ **then**
        $\hat{\mathbf{B}}_{\hat{j}_{k-1}^{(c)}} = \hat{\mathbf{B}}_{\hat{j}_{k-1}} + \hat{\lambda}_{k-1}\hat{\mathbf{\Sigma}}_{\hat{j}_{k-1}^{(c)}}^{-1}\mathbf{U}_{\hat{j}_{k-1}^{(c)}}\hat{\mathbf{S}}_{k-1}^{(c)}\mathbf{V}_{\hat{j}_{k-1}^{(c)}}^\top$
      end
    end
    $\hat{\mathbf{U}}^{(k-1)} = \mathbf{Y} - \hat{\mathbf{G}}^{(k-1)}$
    $(\hat{j}_k^{(c)}, \hat{\mathbf{S}}_k^{(c)}) \in \arg\max_{\substack{j \in \mathcal{I}_c \cap \hat{J} \\ \|\mathbf{S}_k\|_* \le L_n}} |\langle \hat{\mathbf{U}}^{(k-1)}, \mathbf{X}_j \hat{\mathbf{\Sigma}}_j^{-1}\mathbf{U}_j\mathbf{S}_k\mathbf{V}_j^\top \rangle|$
    $\rho_c = |\langle \hat{\mathbf{U}}^{(k-1)}, \mathbf{X}_{\hat{j}_k^{(c)}}\hat{\mathbf{\Sigma}}_{\hat{j}_k^{(c)}}^{-1}\mathbf{U}_{\hat{j}_k^{(c)}}\hat{\mathbf{S}}_k^{(c)}\mathbf{V}_{\hat{j}_k^{(c)}}^\top \rangle|$
    Find the leading singular value decomposition:

$$\mathbf{X}_{\hat{j}_k^{(c)}}\hat{\mathbf{\Sigma}}_{\hat{j}_k^{(c)}}^{-1}\mathbf{U}_{\hat{j}_k^{(c)}}\hat{\mathbf{S}}_{k^{(c)}}\mathbf{V}_{\hat{j}_k^{(c)}}^\top = \sigma_{\hat{j}_k^{(c)}}\mathbf{u}_{\hat{j}_k^{(c)}}\mathbf{v}_{\hat{j}_k^{(c)}}^\top$$

    Send $(\sigma_{\hat{j}_k^{(c)}}, \mathbf{u}_{\hat{j}_k^{(c)}}, \mathbf{v}_{\hat{j}_k^{(c)}}, \rho_c)$ to the master.
  end
  Master do
    Receives $\{(\sigma_{\hat{j}_k^{(c)}}, \mathbf{u}_{\hat{j}_k^{(c)}}, \mathbf{v}_{\hat{j}_k^{(c)}}, \rho_c) : c = 1, 2, \ldots, M\}$ from the workers.
    $c^* = \arg\max_{1 \le c \le M} \rho_c$
    $\sigma_{\hat{j}_k} = \sigma_{\hat{j}_k^{(c^*)}}, \mathbf{u}_{\hat{j}_k} = \mathbf{u}_{\hat{j}_k^{(c^*)}}, \mathbf{v}_{\hat{j}_k} = \mathbf{v}_{\hat{j}_k^{(c^*)}}$
    $\hat{\mathbf{G}}^{(k)} = (1 - \hat{\lambda}_k)\hat{\mathbf{G}}^{(k-1)} + \hat{\lambda}_k\sigma_{\hat{j}_k}\mathbf{u}_{\hat{j}_k}\mathbf{v}_{\hat{j}_k}^\top$, where $\hat{\lambda}_k$ is determined by

$$\hat{\lambda}_k \in \arg\min_{0 \le \lambda \le 1} \|\mathbf{Y} - (1 - \lambda)\hat{\mathbf{G}}^{(k-1)} - \lambda\sigma_{\hat{j}_k}\mathbf{u}_{\hat{j}_k}\mathbf{v}_{\hat{j}_k}^\top\|_F^2.$$

    Broadcast $(c^*, \hat{\lambda}_k, \sigma_{\hat{j}_k}, \mathbf{u}_{\hat{j}_k}, \mathbf{v}_{\hat{j}_k})$ to all workers.
  end
end

Appendix B. Proofs

This section presents the essential elements of the proofs of our main results. Further technical details are relegated to Appendix C.

The analysis of TSRGA relies on what we call the "noiseless updates," a theoretical device constructed as follows. Initialize $\mathbf{G}^{(0)} = \mathbf{0}$ and $\mathbf{U}^{(0)} = \tilde{\mathbf{Y}}$. For $1 \le k \le K_n$, suppose $(\hat{j}_k, \tilde{\mathbf{B}}_{\hat{j}_k})$ is chosen according to (3) by the first-stage RGA. The noiseless updates are defined as

$$\mathbf{G}^{(k)} = (1-\lambda_k)\hat{\mathbf{G}}^{(k-1)} + \lambda_k \mathbf{X}_{\hat{j}_k}\tilde{\mathbf{B}}_{\hat{j}_k}, \tag{24}$$

where

$$\lambda_k \in \arg\min_{0\le\lambda\le 1} \|\tilde{\mathbf{Y}} - (1-\lambda)\hat{\mathbf{G}}^{(k-1)} - \lambda \mathbf{X}_{\hat{j}_k}\tilde{\mathbf{B}}_{\hat{j}_k}\|_F^2. \tag{25}$$

Recall that $\tilde{\mathbf{Y}} = \sum_{j=1}^{p_n} \mathbf{X}_j \mathbf{B}_j^*$ is the noise-free part of the response. Thus $\mathbf{G}^{(k)}$ is unattainable in practice. Similarly, we can define the noiseless updates for the second-stage RGA, with $\tilde{\mathbf{B}}_{\hat{j}_k}$ replaced by $\hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1}\mathbf{U}_{\hat{j}_k}\hat{\mathbf{S}}_k\mathbf{V}_{\hat{j}_k}^\top$ in (24) and (25). By definition of the updates, for first- and second-stage RGA,

$$\begin{aligned}\|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k)}\|_F^2 \le& \|\tilde{\mathbf{Y}} - \mathbf{G}^{(k)}\|_F^2 + 2\langle \mathbf{E}, \hat{\mathbf{G}}^{(k)} - \mathbf{G}^{(k)}\rangle \\ \le& \|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}\|_F^2 + 2\langle \mathbf{E}, \hat{\mathbf{G}}^{(k)} - \mathbf{G}^{(k)}\rangle \end{aligned} \tag{26}$$

Recursively applying (26), we have for any $1 \le l \le k$,

$$\|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k)}\|_F^2 \le \|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-l)}\|_F^2 + 2\sum_{j=1}^{l}\langle \mathbf{E}, \hat{\mathbf{G}}^{(k-j+1)} - \mathbf{G}^{(k-j+1)}\rangle. \tag{27}$$

(27) bounds the empirical prediction error at step k by the empirical prediction error at step $k-l$ and a remainder term involving the noise and the noiseless updates up to step l. This will be handy in numerous places throughout the proofs.

Two other useful identities are

$$\max_{\substack{1\le j\le p_n \\ \|\mathbf{B}_j\|_*\le L_n}} \langle \mathbf{A}, \mathbf{X}_j\mathbf{B}_j\rangle = \sup_{\substack{\mathbf{B}_j\in\mathbb{R}^{q_n, j}\times d_n, j=1,2,\ldots,p_n \\ \sum_j \|\mathbf{B}_j\|_*\le L_n}} \left\langle \mathbf{A}, \sum_{j=1}^{p_n}\mathbf{X}_j\mathbf{B}_j \right\rangle \tag{28}$$

and

$$\max_{\substack{j\in\hat{J} \\ \|\mathbf{S}\|_*\le L_n}} \langle \mathbf{A}, \mathbf{X}_j\hat{\mathbf{\Sigma}}_j^{-1}\mathbf{U}_j\mathbf{S}\mathbf{V}_j^\top\rangle = \sup_{\sum_{j\in\hat{J}}\|\mathbf{S}_j\|_*\le L_n} \left\langle \mathbf{A}, \sum_{j\in\hat{J}}\mathbf{X}_j\hat{\mathbf{\Sigma}}_j^{-1}\mathbf{U}_j\mathbf{S}_j\mathbf{V}_j^\top \right\rangle, \tag{29}$$

where $\mathbf{A} \in \mathbb{R}^{n\times d_n}$ is arbitrary. These identities hold because the maximum of the inner product is attained at the extreme points in the ℓ_1 ball. The proofs are omitted for brevity.

We first prove an auxiliary lemma which guarantees sub-linear convergence of the empirical prediction error, whose proof makes use of the noiseless updates introduced above.

Lemma 6 *Assume (C1)-(C2) and that* $\sum_{j=1}^{p_n}\|\mathbf{B}_j^*\|_* \le d_n^{1/2}L$. *RGA has the following uniform rate of convergence.*

$$\max_{1\le k\le K_n}\frac{(nd_n)^{-1}\|\tilde{\mathbf{Y}}-\hat{\mathbf{G}}^{(k)}\|_F^2}{k^{-1}}=O_p(1). \tag{30}$$

Proof Let $1\le m\le K_n$ be arbitrary. Note that for any $1\le k\le K_n$,

$$\begin{aligned}
&\langle\tilde{\mathbf{Y}}-\hat{\mathbf{G}}^{(k-1)},\mathbf{X}_{\hat{j}_k}\tilde{\mathbf{B}}_{\hat{j}_k}-\hat{\mathbf{G}}^{(k-1)}\rangle\\
=&\langle\mathbf{Y}-\hat{\mathbf{G}}^{(k-1)},\mathbf{X}_{\hat{j}_k}\tilde{\mathbf{B}}_{\hat{j}_k}-\hat{\mathbf{G}}^{(k-1)}\rangle-\langle\mathbf{E},\mathbf{X}_{\hat{j}_k}\tilde{\mathbf{B}}_{\hat{j}_k}-\hat{\mathbf{G}}^{(k-1)}\rangle\\
\ge&\max_{\substack{1\le j\le p_n\\ \|\mathbf{B}_j\|_*\le L_n}}\{\langle\mathbf{Y}-\hat{\mathbf{G}}^{(k-1)},\mathbf{X}_j\mathbf{B}_j-\hat{\mathbf{G}}^{(k-1)}\rangle\}-2L_n\xi_E\\
\ge&\max_{\substack{1\le j\le p_n\\ \|\mathbf{B}_j\|_*\le L_n}}\{\langle\tilde{\mathbf{Y}}-\hat{\mathbf{G}}^{(k-1)},\mathbf{X}_j\mathbf{B}_j-\hat{\mathbf{G}}^{(k-1)}\rangle\}-4L_n\xi_E.
\end{aligned}\tag{31}$$

Put

$$\mathcal{E}_n(m)=\left\{\min_{1\le l\le m}\max_{\substack{1\le j\le p_n\\ \|\mathbf{B}_j\|_*\le L_n}}\langle\tilde{\mathbf{Y}}-\hat{\mathbf{G}}^{(l-1)},\mathbf{X}_j\mathbf{B}_j-\hat{\mathbf{G}}^{(l-1)}\rangle>\tilde{\tau}d_n^{1/2}\xi_E\right\}, \tag{32}$$

for some $\tilde{\tau}>4L_0$. It follows from (28) and (31) that on $\mathcal{E}_n(m)$, for all $1\le k\le m$,

$$\begin{aligned}
&\langle\tilde{\mathbf{Y}}-\hat{\mathbf{G}}^{(k-1)},\mathbf{X}_{\hat{j}_k}\tilde{\mathbf{B}}_{\hat{j}_k}-\hat{\mathbf{G}}^{(k-1)}\rangle\\
\ge&(1-\frac{4L_0}{\tilde{\tau}})\max_{\substack{1\le j\le p_n\\ \|\mathbf{B}_j\|_*\le L}}\{\langle\tilde{\mathbf{Y}}-\hat{\mathbf{G}}^{(k-1)},\mathbf{X}_j\mathbf{B}_j-\hat{\mathbf{G}}^{(k-1)}\rangle\}\\
\ge&(1-\frac{4L_0}{\tilde{\tau}})\|\tilde{\mathbf{Y}}-\hat{\mathbf{G}}^{(k-1)}\|_F^2\\
:=&\tau\|\tilde{\mathbf{Y}}-\hat{\mathbf{G}}^{(k-1)}\|_F^2\\
\ge&0,
\end{aligned}\tag{33}$$

where $\tau=1-4L_0/\tilde{\tau}$. This, together with Lemma 10(iii) in Appendix C, implies

$$\lambda_k=\frac{\langle\tilde{\mathbf{Y}}-\hat{\mathbf{G}}^{(k-1)},\mathbf{X}_{\hat{j}_k}\tilde{\mathbf{B}}_{\hat{j}_k}-\hat{\mathbf{G}}^{(k-1)}\rangle}{\|\mathbf{X}_{\hat{j}_k}\tilde{\mathbf{B}}_{\hat{j}_k}-\hat{\mathbf{G}}^{(k-1)}\|_F^2}$$

for $1\le k\le m$ on $\mathcal{E}_n(m)$ except for a vanishing event. This, combined with (26) and (33), yields

$$\begin{aligned}
\|\tilde{\mathbf{Y}}-\hat{\mathbf{G}}^{(k)}\|_F^2\le&\|\tilde{\mathbf{Y}}-\mathbf{G}^{(k)}\|_F^2+2\langle\mathbf{E},\hat{\mathbf{G}}^{(k)}-\mathbf{G}^{(k)}\rangle\\
=&\|\tilde{\mathbf{Y}}-\hat{\mathbf{G}}^{(k-1)}-\lambda_k(\mathbf{X}_{\hat{j}_k}\tilde{\mathbf{B}}_{\hat{j}_k}-\hat{\mathbf{G}}^{(k-1)})\|_F^2+2\langle\mathbf{E},\hat{\mathbf{G}}^{(k)}-\mathbf{G}^{(k)}\rangle\\
=&\|\tilde{\mathbf{Y}}-\hat{\mathbf{G}}^{(k-1)}\|_F^2-\frac{\langle\tilde{\mathbf{Y}}-\hat{\mathbf{G}}^{(k-1)},\mathbf{X}_{\hat{j}_k}\tilde{\mathbf{B}}_{\hat{j}_k}-\hat{\mathbf{G}}^{(k-1)}\rangle^2}{\|\mathbf{X}_{\hat{j}_k}\tilde{\mathbf{B}}_{\hat{j}_k}-\hat{\mathbf{G}}^{(k-1)}\|_F^2}+2\langle\mathbf{E},\hat{\mathbf{G}}^{(k)}-\mathbf{G}^{(k)}\rangle\\
\le&\|\tilde{\mathbf{Y}}-\hat{\mathbf{G}}^{(k-1)}\|_F^2\left\{1-\frac{\tau^2\|\tilde{\mathbf{Y}}-\hat{\mathbf{G}}^{(k-1)}\|_F^2}{\|\mathbf{X}_{\hat{j}_k}\tilde{\mathbf{B}}_{\hat{j}_k}-\hat{\mathbf{G}}^{(k-1)}\|_F^2}\right\}+2\langle\mathbf{E},\hat{\mathbf{G}}^{(k)}-\mathbf{G}^{(k)}\rangle
\end{aligned}\tag{34}$$

for all $1 \le k \le m$ on $\mathcal{E}_n(m)$ except for a vanishing event. By (C1), with probability tending to one, $\|\mathbf{X}_{\hat{j}_k}\tilde{\mathbf{B}}_{\hat{j}_k} - \hat{\mathbf{G}}^{(k-1)}\|_F^2 \le 4L_n^2 n\mu$ and $\|\tilde{\mathbf{Y}}\|_F^2 \le (1-\epsilon_L)^2 L_n^2 n\mu$. Now by Lemma 11 and Lemma 10(ii) in Appendix C, we have

$$\begin{aligned}\frac{1}{nd_n}\|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(m)}\|_F^2 \le& \frac{4L_0^2\mu}{1+m\tau^2} + 2\sum_{l=1}^{m}\frac{|\langle \mathbf{E}, \hat{\mathbf{G}}^{(l)} - \mathbf{G}^{(l)}\rangle|}{nd_n}\\ =& \frac{4L_0^2\mu}{1+m\tau^2} + 2\sum_{l=1}^{m}|\hat{\lambda}_l - \lambda_l|\frac{|\langle \mathbf{E}, \mathbf{X}_{\hat{j}_l}\tilde{\mathbf{B}}_{\hat{j}_l} - \hat{\mathbf{G}}^{(l-1)}\rangle|}{nd_n}\\ \le& \frac{4L_0^2\mu}{1+m\tau^2} + \frac{8}{1-\epsilon_L}\frac{m\xi_E^2}{n^2 d_n}, \end{aligned} \tag{35}$$

on $\mathcal{E}_n(m)$ except for a vanishing event. Note that by (C2), $m\xi_E^2/(n^2 d_n) \le m^{-1}(K_n\xi_E/(nd_n^{1/2}))^2 = O_p(m^{-1})$. Furthermore, it is shown in Appendix C that on $\mathcal{E}_n^c(m)$ except for a vanishing event,

$$\frac{1}{nd_n}\|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(m)}\|_F^2 \le \frac{\tilde{\tau}\xi_E}{n\sqrt{d_n}} + \frac{8m\xi_E^2}{(1-\epsilon_L)n^2 d_n}. \tag{36}$$

Combining (35) and (36) yields the desired result. ■

Now we are ready to prove the main results.

Proof [Proof of Theorem 1] Since $d_n^{1/2}L \ge \sum_{j=1}^{p_n}\|\mathbf{B}_j^*\|_* \ge \sharp(J_n)\min_{j\in J_n}\sigma_{r_j^*}(\mathbf{B}_j^*)$ and $s_n = o(K_n^2)$, it follows that $\sharp(J_n) = o(K_n)$, and by (C3), with probability tending to one, $\lambda_{\min}(\mathbf{X}(\hat{J}_k \cup J_n)^\top \mathbf{X}(\hat{J}_k \cup J_n)) \ge n\mu^{-1}$, for all $1 \le k \le K_n$, where $\hat{J}_k = \{\hat{j}_1, \hat{j}_2, \ldots, \hat{j}_k\}$. Let $\mathcal{G}_n = \{\text{there exists some } j \text{ such that } \mathrm{rank}(\mathbf{B}_j^*) > \mathrm{rank}(\hat{\mathbf{B}}_j^{(\hat{k})})\}$. Then on $\mathcal{G}_n$ except for a vanishing event, it follows from (28), (C3), Eckart-Young theorem and (C4) that

$$\begin{aligned}\min_{1\le m\le \hat{k}}\max_{\substack{1\le j\le p_n\\ \|\mathbf{B}_j\|_*\le L}}\langle \tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(m)}, \mathbf{X}_j\mathbf{B}_j - \hat{\mathbf{G}}^{(m)}\rangle \ge& \min_{1\le m\le\hat{k}}\|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(m)}\|_F^2\\ \ge& n\mu^{-1}\min_{1\le m\le\hat{k}}\|\mathbf{B}_j^* - \hat{\mathbf{B}}_j^{(m)}\|_F^2\\ \ge& n\mu^{-1}\min_{\mathrm{rank}(\mathbf{B})<r_j^*}\|\mathbf{B}_j^* - \mathbf{B}\|_F^2\\ \ge& \frac{nd_n}{\mu s_n}. \end{aligned} \tag{37}$$

By (37), (C2) and (C4), we have $\lim_{n\to\infty}\mathbb{P}\left(\mathcal{G}_n \cap \mathcal{E}_n^c(\hat{k})\right) \le \lim_{n\to\infty}\mathbb{P}(nd_n^{1/2} \le \tilde{\tau}\mu s_n\xi_E) = 0$, where $\mathcal{E}_n(\cdot)$ is defined in (32). Hence it suffices to show $\lim_{n\to\infty}\mathbb{P}(\mathcal{G}_n \cap \mathcal{E}_n(\hat{k})) = 0$. By (37) and the same argument as in (34), on $\mathcal{G}_n \cap \mathcal{E}_n(\hat{k})$ except for a vanishing event,

$$\begin{aligned}\|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k)}\|_F^2 \le& \|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}\|_F^2\left\{1 - \frac{\tau^2\|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}\|_F^2}{\|\mathbf{X}_{\hat{j}_k}\tilde{\mathbf{B}}_{\hat{j}_k} - \hat{\mathbf{G}}^{(k-1)}\|_F^2}\right\} + 2\langle \mathbf{E}, \hat{\mathbf{G}}^{(k)} - \mathbf{G}^{(k)}\rangle\\ \le& \|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}\|_F^2\left\{1 - \frac{\tau^2 s_n^{-1}}{4L_0^2\mu^2}\right\} + 2\langle \mathbf{E}, \hat{\mathbf{G}}^{(k)} - \mathbf{G}^{(k)}\rangle, \end{aligned}$$

and thus

$$nd_n\hat{\sigma}_k^2 \le \|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}\|_F^2 \left(1 - \frac{\tau^2 s_n^{-1}}{4L_0^2\mu^2}\right) + \|\mathbf{E}\|_F^2 + 2\langle \mathbf{E}, \tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k)}, \rangle$$

for $1 \le k \le \hat{k}$. It follows that

$$\begin{aligned}\frac{\hat{\sigma}_k^2}{\hat{\sigma}_{k-1}^2} \le & \frac{(nd_n)^{-1}\|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}\|_F^2 + (nd_n)^{-1}\|\mathbf{E}\|_F^2 + 4L_0\xi_E/(nd_n^{1/2})}{(nd_n)^{-1}\|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}\|_F^2 + (nd_n)^{-1}\|\mathbf{E}\|_F^2 - 4L_0\xi_E/(nd_n^{1/2})} \\ & - \frac{\tau^2 s_n^{-1}}{4L_0^2\mu^2} \frac{(nd_n)^{-1}\|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}\|_F^2}{(nd_n)^{-1}\|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}\|_F^2 + (nd_n)^{-1}\|\mathbf{E}\|_F^2 - 4L_0\xi_E/(nd_n^{1/2})} \\ := & A_k - B_k, \end{aligned} \tag{38}$$

for $1 \le k \le \hat{k}$ on $\mathcal{G}_n \cap \mathcal{E}_n(\hat{k})$ except for a vanishing event. We show in Appendix C that on $\mathcal{G}_n \cap \mathcal{E}_n(\hat{k})$ except for a vanishing event, for all $1 \le k \le \hat{k}$,

$$A_k \le 1 + \frac{12ML_0\xi_E}{nd_n^{1/2}}, \tag{39}$$

and

$$B_k \ge \frac{\tau^2}{4L_0^2\mu^2} s_n^{-1} \frac{1}{1 + \mu M s_n} \left(1 - \frac{4ML_0\xi_E}{nd_n^{1/2}}\right). \tag{40}$$

By (38)-(40), $\max_{1\le k\le\hat{k}} \hat{\sigma}_k^2/\hat{\sigma}_{k-1}^2 \le 1 - s_n^{-2}C_n$, where

$$C_n = \frac{\tau^2}{4L_0^2\mu^2} \frac{1}{\mu M + s_n^{-1}} \left(1 - \frac{4ML_0\xi_E}{nd_n^{1/2}}\right) - 12ML_0 \frac{s_n^2\xi_E}{nd_n^{1/2}}.$$

By (C2) and (C4), it can be shown that there exists some $v > 0$ such that $C_n \ge v$ with probability tending to one. Therefore, by the definition of $\hat{k}$,

$$\begin{aligned}\mathbb{P}(\mathcal{G}_n \cap \mathcal{E}_n(\hat{k})) \le & \mathbb{P}(\hat{k} < K_n, \mathcal{G}_n \cap \mathcal{E}_n(\hat{k})) + \mathbb{P}(\hat{k} = K_n, \mathcal{G}_n \cap \mathcal{E}_n(\hat{k})) \\ \le & \mathbb{P}(\max_{1\le k\le\hat{k}} \hat{\sigma}_k^2/\hat{\sigma}_{k-1}^2 \le 1 - vs_n^{-2}, \hat{k} < K_n) + \mathbb{P}(\hat{k} = K_n, \mathcal{G}_n \cap \mathcal{E}_n(\hat{k})) + o(1) \\ = & \mathbb{P}(\hat{k} = K_n, \mathcal{G}_n \cap \mathcal{E}_n(\hat{k})) + o(1), \end{aligned} \tag{41}$$

if $t_n = Cs_n^{-2}$ in (6) is chosen with $C < v$. In view of (41), it remains to show $\mathbb{P}(\hat{k} = K_n, \mathcal{G}_n \cap \mathcal{E}_n(\hat{k})) = o(1)$. Since $s_n = o(K_n)$ by (C4), it follows from (37) and Lemma 6 that

$$\begin{aligned}\mathbb{P}(\hat{k} = K_n, \mathcal{G}_n) \le & \mathbb{P}\left(\frac{1}{nd_n}\|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(K_n)}\|_F^2 \ge \frac{1}{\mu s_n}\right) + o(1) \\ = & \mathbb{P}\left(\frac{(nd_n)^{-1}\|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(K_n)}\|_F^2}{K_n^{-1}} \ge \frac{K_n}{\mu s_n}\right) + o(1) \\ = & o(1), \end{aligned}$$

which completes the proof. ■

Proof [Proof of Lemma 2] Letting $a_n = \lfloor Ds_n^2 \rfloor$ for some arbitrary $D > 0$, we have

$$\begin{aligned}\mathbb{P}(\hat{k} > a_n) \leq & \mathbb{P}\left(\frac{\hat{\sigma}^2_{a_n}}{\hat{\sigma}^2_{a_n-1}} < 1 - Cs_n^2\right) \\ = & \mathbb{P}\left(Cs_n^{-2} < \frac{\hat{\sigma}^2_{a_n-1} - \zeta_n^2 - (\hat{\sigma}^2_{a_n} - \zeta_n^2)}{\zeta_n^2 + \hat{\sigma}^2_{a_n-1} - \zeta_n^2}\right) \\ \leq & \mathbb{P}\left(Cs_n^{-2} < \frac{\hat{\sigma}^2_{a_n-1} - \zeta_n^2}{M^{-1} + \hat{\sigma}^2_{a_n-1} - \zeta_n^2} + \frac{4L_0\xi_E n^{-1} d_n^{-1/2}}{M^{-1} + \hat{\sigma}^2_{a_n-1} - \zeta_n^2}\right) + o(1). \end{aligned} \tag{42}$$

Put $A_n = \{\hat{\sigma}^2_{a_n-1} - \zeta_n^2 > 0\}$. Then (42) implies

$$\begin{aligned}\mathbb{P}(\hat{k} > a_n, A_n) \leq & \mathbb{P}\left(M^{-1} + \hat{\sigma}^2_{a_n-1} - \zeta_n^2 < \frac{\hat{\sigma}^2_{a_n-1} - \zeta_n^2}{Cs_n^{-2}} + \frac{4L_0 s_n^2 \xi_E}{Cnd_n^{1/2}}, A_n\right) + o(1) \\ \leq & \mathbb{P}\left(M^{-1} < Z_n \frac{s_n^2}{C(a_n - 1)} + \frac{4L_0}{C}\frac{s_n^2 \xi_E}{nd_n^{1/2}}\right) + o(1),\end{aligned}$$

where

$$Z_n := \max_{1 \leq k \leq K_n} \frac{|(nd_n)^{-1}\|\mathbf{Y} - \hat{\mathbf{G}}^{(k)}\|_F^2 - \zeta_n^2|}{k^{-1}}.$$

Since $|(nd_n)^{-1}\|\mathbf{Y} - \hat{\mathbf{G}}^{(k)}\|_F^2 - \zeta_n^2| \leq (nd_n)^{-1}\|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k)}\|_F^2 + 4L_0\xi_E n^{-1} d_n^{-1/2}$, where $\zeta_n^2 = (nd_n)^{-1}\|\mathbf{E}\|_F^2$, it follows from Lemma 6 that $Z_n = O_p(1)$. Thus $\limsup_{n\to\infty} \mathbb{P}(\hat{k} > a_n, A_n) \to 0$ as $D \to 0$. On A_n^c, it is not difficult to show that

$$\hat{\sigma}^2_{a_n} - \zeta_n^2 \leq \hat{\sigma}^2_{a_n-1} - \zeta_n^2 \leq 0$$

and

$$\max\left\{\frac{1}{nd_n}\|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(a_n-1)}\|_F^2, \frac{1}{nd_n}\|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(a_n)}\|_F^2\right\} \leq \frac{4L_0\xi_E}{nd_n^{1/2}}.$$

It follows that on A_n^c,

$$\begin{aligned}\frac{\hat{\sigma}^2_{a_n}}{\hat{\sigma}^2_{a_n-1}} = & 1 - \frac{\hat{\sigma}^2_{a_n-1} - \hat{\sigma}^2_{a_n}}{\hat{\sigma}^2_{a_n-1}} \\ \geq & 1 - \frac{\hat{\sigma}^2_{a_n-1} - \zeta_n^2 - (\hat{\sigma}^2_{a_n} - \zeta_n^2)}{\zeta_n^2 - 4L_0\xi_E n^{-1} d_n^{-1/2}} \\ \geq & 1 - \frac{1}{\zeta_n^2 - 4L_0\xi_E n^{-1} d_n^{-1/2}} \frac{16L_0\xi_E}{nd_n^{1/2}}.\end{aligned}$$

By (C4), we have

$$\mathbb{P}(\hat{k} > a_n, A_n^c) \leq \mathbb{P}\left(Cs_n^{-2} \leq \frac{1}{\zeta_n^2 - 4L_0\xi_E n^{-1} d_n^{-1/2}} \frac{16L_0\xi_E}{nd_n^{1/2}} \right) = o(1),$$

which completes the proof. ■

Before proving Theorem 3, we introduce the following uniform convergence rate for the second-stage RGA, which is also of independent interest.

Theorem 7 *Assume the same as Theorem 1, and additionally (C5) and (C6) hold. The second-stage RGA satisfies*

$$\max_{1\leq m\leq K_n} \frac{(nd_n)^{-1}\|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(m)}\|_F^2}{\left(1 - \frac{\tau^2}{64\mu^5\kappa_n}\right)^m + \frac{(m+\kappa_n)\xi_E^2}{n^2 d_n} + \frac{\xi_E^2}{\delta_n^2 n^2}\mathbf{1}\{J_o \neq \emptyset\}} = O_p(1), \tag{43}$$

where $\tau < 1$ is an absolute constant.

Proof By Theorem 1, we can assume $\text{rank}(\mathbf{B}_j^*) \leq \hat{r}_j$ holds for all j in the following analysis. Let $1 \leq m \leq K_n$ be arbitrary. Observe that for the second-stage RGA, each $\hat{\mathbf{G}}^{(k)}$, $k = 1, 2, \ldots$, lies in the set

$$\mathcal{C}_L = \left\{ \mathbf{H} = \sum_{j\in\hat{J}} \mathbf{X}_j \hat{\mathbf{\Sigma}}_j^{-1} \mathbf{U}_j \mathbf{D}_j \mathbf{V}_j^\top : \sum_{j\in\hat{J}} \|\mathbf{D}_j\|_* \leq L_n \right\}. \tag{44}$$

By (29) and a similar argument as (31)-(33), we have, for all $1 \leq k \leq m$,

$$\begin{aligned}
&\langle \tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}, \mathbf{X}_{\hat{j}_k} \hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1} \mathbf{U}_{\hat{j}_k} \hat{\mathbf{S}}_k \mathbf{V}_{\hat{j}_k}^T - \hat{\mathbf{G}}^{(k-1)} \rangle \\
\geq &\tau \max_{\substack{j\in\hat{J}_{\hat{k}} \\ \|\mathbf{S}\|_*\leq L_n}} \langle \tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}, \mathbf{X}_j \hat{\mathbf{\Sigma}}_j^{-1} \mathbf{U}_j \mathbf{S} \mathbf{V}_j^\top - \hat{\mathbf{G}}^{(k-1)} \rangle \\
= &\tau \sup_{\mathbf{H}\in\mathcal{C}_L} \langle \tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}, \mathbf{H} - \hat{\mathbf{G}}^{(k-1)} \rangle,
\end{aligned} \tag{45}$$

where $\tau = 1 - 4\mu L_0/\tilde{\tau}$ and $\tilde{\tau} > 4\mu L_0$ on the event

$$\mathcal{F}_n(m) = \left\{ \min_{1\leq k\leq m} \max_{\substack{j\in\hat{J}_{\hat{k}} \\ \|\mathbf{S}\|_*\leq L_n}} \langle \tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}, \mathbf{X}_j \hat{\mathbf{\Sigma}}_j^{-1} \mathbf{U}_j \mathbf{S} \mathbf{V}_j^\top - \hat{\mathbf{G}}^{(k-1)} \rangle > \tilde{\tau} d_n^{1/2} \xi_E \right\}.$$

Define

$$\mathcal{B} = \left\{ \mathbf{H} = \sum_{j\in\hat{J}_{\hat{k}}} \mathbf{X}_j \hat{\mathbf{\Sigma}}_j^{-1} \mathbf{U}_j \mathbf{D}_j \mathbf{V}_j^\top : \|\bar{\mathbf{Y}} - \mathbf{H}\|_F^2 \leq \frac{9nd_n L_0^2}{16\mu^3\kappa_n} \right\},$$

where

$$\bar{\mathbf{Y}} = \sum_{j \in \hat{J}_o} \mathbf{X}_j \hat{\mathbf{\Sigma}}_j^{-1} \mathbf{U}_j \mathbf{L}_j \mathbf{\Lambda}_j \mathbf{R}_j^\top \mathbf{V}_j^\top + \sum_{j \in \hat{J} - \hat{J}_o} \mathbf{X}_j \mathbf{B}_j^*, \tag{46}$$

in which $\hat{J}_o = \{j \in \hat{J} : \hat{r} < \min\{q_{n,j}, d_n\}\}$, $\mathbf{\Lambda}_j$ are defined in (C6), and $\mathbf{L}_j$, $\mathbf{R}_j$ are $\hat{r} \times \bar{r}_j$ matrices such that $\mathbf{L}_j^T \mathbf{L}_j = \mathbf{I}_{\bar{r}_j} = \mathbf{R}_j^T \mathbf{R}_j$ to be specified later (recall that $\hat{r} \geq \bar{r}_j = \operatorname{rank}(\mathbf{X}_j^\top \tilde{\mathbf{Y}})$ because of Theorem 1). We claim that

$$\lim_{n \to \infty} \mathbb{P}(\mathcal{B} \subseteq \mathcal{C}_L) = 1, \tag{47}$$

whose proof is relegated to Appendix C. Now put $\mathbf{H}^{(l)} = \hat{\mathbf{G}}^{(l)} + (1 + \alpha_l)(\bar{\mathbf{Y}} - \hat{\mathbf{G}}^{(l)})$ for $l = 1, 2, \ldots$, where

$$\alpha_l = \frac{3\sqrt{n d_n} L_0}{4\mu^{3/2} \sqrt{\kappa_n} \|\bar{\mathbf{Y}} - \hat{\mathbf{G}}^{(l)}\|_F} \geq 0.$$

Then (47) implies that $\mathbb{P}(\mathbf{H}^{(l)} \in \mathcal{C}_L, l = 1, 2, \ldots) \to 1$. Thus by (45),

$$\begin{aligned} &\langle \tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}, \mathbf{X}_{\hat{j}_k} \hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1} \mathbf{U}_{\hat{j}_k} \hat{\mathbf{S}}_k \mathbf{V}_{\hat{j}_k}^\top - \hat{\mathbf{G}}^{(k-1)} \rangle \\ &\geq \tau \langle \tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}, \mathbf{H}^{(k-1)} - \hat{\mathbf{G}}^{(k-1)} \rangle \end{aligned} \tag{48}$$

for all $1 \leq k \leq m$ on $\mathcal{F}_n(m)$ except for a vanishing event. Put $\mathcal{H}_n(m) = \{\|\tilde{\mathbf{Y}} - \bar{\mathbf{Y}}\|_F < 2^{-1} \min_{1 \leq l \leq m} \|\bar{\mathbf{Y}} - \hat{\mathbf{G}}^{(l-1)}\|_F\}$. On $\mathcal{F}_n(m) \cap \mathcal{H}_n(m)$ except for a vanishing event, (48) and Cauchy-Schwarz inequality yield

$$\begin{aligned} &\langle \tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}, \mathbf{X}_{\hat{j}_k} \hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1} \mathbf{U}_{\hat{j}_k} \hat{\mathbf{S}}_k \mathbf{V}_{\hat{j}_k}^\top - \hat{\mathbf{G}}^{(k-1)} \rangle \\ \geq &\tau \langle \tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}, \mathbf{H}^{(k-1)} - \hat{\mathbf{G}}^{(k-1)} \rangle \\ \geq &\tau (1 + \alpha_{k-1}) \left\{ \|\bar{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}\|_F^2 - \|\tilde{\mathbf{Y}} - \bar{\mathbf{Y}}\|_F \|\bar{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}\|_F \right\} \\ \geq &\frac{\tau (1 + \alpha_{k-1})}{2} \|\bar{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}\|_F^2 \geq 0 \end{aligned}$$

for all $1 \le k \le m$. Notice that $\|\bar{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}\|_F \ge (2/3)\|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}\|_F$ for all $1 \le k \le m$ on $\mathcal{H}_n(m)$. Hence, by Lemma 10(ii), (iii), and a similar argument used in (34),

$$\begin{aligned}
\|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k)}\|_F^2 \le & \|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}\|_F^2 - \frac{\langle \tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}, \mathbf{X}_{\hat{j}_k} \hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1} \mathbf{U}_{\hat{j}_k} \hat{\mathbf{S}}_k \mathbf{V}_{\hat{j}_k}^\top - \hat{\mathbf{G}}^{(k-1)} \rangle^2}{\|\mathbf{X}_{\hat{j}_k} \hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1} \mathbf{U}_{\hat{j}_k} \hat{\mathbf{S}}_k \mathbf{V}_{\hat{j}_k}^\top - \hat{\mathbf{G}}^{(k-1)}\|_F^2} \\
& + 2\langle \mathbf{E}, \hat{\mathbf{G}}^{(k)} - \mathbf{G}^{(k)} \rangle \\
\le & \|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}\|_F^2 - \frac{\tau^2 \langle \tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}, \mathbf{H}^{(k-1)} - \hat{\mathbf{G}}^{(k-1)} \rangle^2}{4n\mu L_n^2} + 2\langle \mathbf{E}, \hat{\mathbf{G}}^{(k)} - \mathbf{G}^{(k)} \rangle \\
\le & \|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}\|_F^2 - \frac{\tau^2 (1+\alpha_{k-1})^2}{16n\mu L_n^2} \|\bar{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}\|_F^4 + 2\langle \mathbf{E}, \hat{\mathbf{G}}^{(k)} - \mathbf{G}^{(k)} \rangle \\
\le & \|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}\|_F^2 - \frac{\tau^2 \|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}\|_F^2}{64\mu^4 \kappa_n} \\
& + 2(\hat{\lambda}_k - \lambda_k) \langle \mathbf{E}, \mathbf{X}_{\hat{j}_k} \hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1} \mathbf{U}_{\hat{j}_k} \hat{\mathbf{S}}_k \mathbf{V}_{\hat{j}_k}^\top - \hat{\mathbf{G}}^{(k-1)} \rangle \\
\le & \|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}\|_F^2 \left(1 - \frac{\tau^2}{64\mu^4 \kappa_n}\right) + \frac{8\mu}{1-\epsilon_L} \frac{\xi_E^2}{n}
\end{aligned}$$

for all $1 \le k \le m$ on $\mathcal{F}_n(m) \cap \mathcal{H}_n(m)$ except for a vanishing event. It follows that, on the same event,

$$\|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(m)}\|_F^2 \le \|\tilde{\mathbf{Y}}\|_F^2 \left(1 - \frac{\tau^2}{64\mu^4 \kappa_n}\right)^m + \frac{8\mu}{1-\epsilon_L} \frac{m\xi_E^2}{n}. \tag{49}$$

By (29), on $\mathcal{F}_n^c(m) \cap \mathcal{H}_n(m)$ there exists some $1 \le k \le m$ such that

$$\begin{aligned}
\tilde{\tau} d_n^{1/2} \xi_E \ge & \langle \tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}, \mathbf{H}^{(k-1)} - \hat{\mathbf{G}}^{(k-1)} \rangle \\
\ge & (1+\alpha_{k-1}) \langle \tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}, \bar{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)} \rangle \\
\ge & \frac{1}{2}(1+\alpha_{k-1}) \|\bar{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}\|_F^2 \\
\ge & \frac{3\sqrt{nd_n} L_0}{8\mu^{3/2}\sqrt{\kappa_n}} \|\bar{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}\|_F,
\end{aligned}$$

which implies

$$\begin{aligned}
\|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(m)}\|_F^2 \le & \|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}\|_F^2 + \frac{8\mu}{1-\epsilon_L} \frac{(m-k)\xi_E^2}{n} \\
\le & 2\|\tilde{\mathbf{Y}} - \bar{\mathbf{Y}}\|_F^2 + 2\|\bar{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}\|_F^2 + \frac{8\mu}{1-\epsilon_L} \frac{(m-k)\xi_E^2}{n} \\
\le & \frac{5}{2}\|\bar{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}\|_F^2 + \frac{8\mu}{1-\epsilon_L} \frac{(m-k)\xi_E^2}{n} \\
\le & \left(\frac{160\tilde{\tau}^2 \mu^3}{9L^2} \kappa_n + \frac{8\mu}{1-\epsilon_L}(m-k)\right) \frac{\xi_E^2}{n}.
\end{aligned} \tag{50}$$

Next, on $\mathcal{H}_n^c(m)$, there exists some $1 \le k \le m$ such that $\|\bar{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}\|_F^2 \le 4\|\tilde{\mathbf{Y}} - \bar{\mathbf{Y}}\|_F^2$. By (27) and the parallelogram law,

$$\begin{aligned}\|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(m)}\|_F^2 \le& \|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}\|_F^2 + 2\sum_{j=k}^{m}\langle \mathbf{E}, \hat{\mathbf{G}}^{(j)} - \mathbf{G}^{(j)}\rangle \\ \le& 10\|\tilde{\mathbf{Y}} - \bar{\mathbf{Y}}\|_F^2 + \frac{8\mu}{1-\epsilon_L}\frac{(m-k)\xi_E^2}{n} \end{aligned} \tag{51}$$

on $\mathcal{H}_n^c(m)$ except for a vanishing event. Finally, note that (49)-(51) are valid for any choice of $\mathbf{L}_j$ and $\mathbf{R}_j$ so long as $\mathbf{L}_j^\top \mathbf{L}_j = \mathbf{I}_{\bar{r}_j} = \mathbf{R}_j^\top \mathbf{R}_j$, $j \in \hat{J}$. In Appendix C, we show that $\mathbf{L}_j$, $\mathbf{R}_j$, $j \in \hat{J}_o$, can be chosen so that

$$\frac{1}{nd_n}\|\tilde{\mathbf{Y}} - \bar{\mathbf{Y}}\|_F^2 \le 8\mu L^2 \frac{\xi_E^2}{(n\delta_n - \xi_E)^2} = O_p\left(\frac{\xi_E^2}{n^2\delta_n^2}\right). \tag{52}$$

Hence, by (49)-(52), the desired result follows. ■

Now we are ready to prove our last main result.

Proof [Proof of Theorem 3] Note first that $\mathcal{C}_L$ (defined in (44)) is a convex compact set almost surely. Thus we can define $\mathbf{Y}^*$ to be the orthogonal projection of $\mathbf{Y}$ onto $\mathcal{C}_L$. Since $\hat{\mathbf{G}}^{(m)} \in \mathcal{C}_L$ and $\hat{\sigma}_m^2 \le \hat{\sigma}_{m_n}^2$ for $m \ge m_n$, it follows that for $m \ge m_n$,

$$\begin{aligned}\|\mathbf{Y}^* - \hat{\mathbf{G}}^{(m)}\|_F^2 =& \|\mathbf{Y} - \hat{\mathbf{G}}^{(m)}\|_F^2 - \|\mathbf{Y} - \mathbf{Y}^*\|_F^2 + 2\langle \mathbf{Y}^* - \mathbf{Y}, \mathbf{Y}^* - \hat{\mathbf{G}}^{(m)}\rangle \\ \le& \|\mathbf{Y} - \hat{\mathbf{G}}^{(m_n)}\|_F^2 - \|\mathbf{Y} - \mathbf{Y}^*\|_F^2 \\ =& \|\mathbf{Y}^* - \hat{\mathbf{G}}^{(m_n)}\|_F^2 - 2\langle \tilde{\mathbf{Y}} - \mathbf{Y}^*, \hat{\mathbf{G}}^{(m_n)} - \mathbf{Y}^*\rangle - 2\langle \mathbf{E}, \hat{\mathbf{G}}^{(m_n)} - \mathbf{Y}^*\rangle \\ \le& 2\|\mathbf{Y}^* - \hat{\mathbf{G}}^{(m_n)}\|_F^2 + \|\mathbf{Y}^* - \tilde{\mathbf{Y}}\|_F^2 - 2\langle \mathbf{E}, \hat{\mathbf{G}}^{(m_n)} - \mathbf{Y}^*\rangle. \end{aligned} \tag{53}$$

Note that if $\mathbf{H}, \mathbf{G}$ are in $\mathcal{C}_L$ with $\mathbf{H} = \sum_{j\in\hat{J}} \mathbf{X}_j \hat{\mathbf{\Sigma}}_j^{-1} \mathbf{U}_j \mathbf{S}_j^H \mathbf{V}_j^\top$ and $\mathbf{G} = \sum_{j\in\hat{J}} \mathbf{X}_j \hat{\mathbf{\Sigma}}_j^{-1} \mathbf{U}_j \mathbf{S}_j^G \mathbf{V}_j^\top$, then by Proposition 8 and (C3) we have

$$\|\mathbf{H} - \mathbf{G}\|_F^2 \ge \frac{n}{\mu^3 \kappa_n}\left\{\sum_{j\in\hat{J}} \|\mathbf{S}_j^H - \mathbf{S}_j^G\|_*\right\}^2.$$

Hence

$$|\langle \mathbf{E}, \mathbf{H} - \mathbf{G}\rangle| \le \mu\xi_E \sum_{j\in\hat{J}} \|\mathbf{S}_j^H - \mathbf{S}_j^G\|_* \le \xi_E \sqrt{\frac{\mu^5\kappa_n}{n}}\|\mathbf{H} - \mathbf{G}\|_F. \tag{54}$$

Combining (53) and (54) yields

$$\|\mathbf{Y}^* - \hat{\mathbf{G}}^{(m)}\|_F^2 \le 2\|\mathbf{Y}^* - \hat{\mathbf{G}}^{(m_n)}\|_F^2 + \|\mathbf{Y}^* - \tilde{\mathbf{Y}}\|_F^2 + 2\xi_E\sqrt{\frac{\mu^5\kappa_n}{n}}\|\mathbf{Y}^* - \hat{\mathbf{G}}^{(m)}\|_F.$$

Since $x^2 \le c + bx$ $(x, b, c \ge 0)$ implies $x \le (b + \sqrt{b^2 + 4c})/2$, we have

$$\|\mathbf{Y}^* - \hat{\mathbf{G}}^{(m)}\|_F^2 \le 2\|\mathbf{Y}^* - \tilde{\mathbf{Y}}\|_F^2 + 4\|\mathbf{Y}^* - \hat{\mathbf{G}}^{(m_n)}\|_F^2 + 4\mu^5 \frac{\kappa_n \xi_E^2}{n}. \tag{55}$$

By (55) and repeated applications of the parallelogram law, it is straightforward to show

$$\frac{1}{nd_n}\|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(m)}\|_F^2 \le \frac{C_1}{nd_n}\left\{\|\tilde{\mathbf{Y}} - \mathbf{Y}^*\|_F^2 + \|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(m_n)}\|_F^2 + \frac{\mu^5 \kappa_n \xi_E^2}{n}\right\}$$

for some absolute constant C_1. The right-hand side does not depend on m, so the inequality still holds if we take supremum over $m \ge m_n$ on the left-hand side. Moreover, by (C3) and Theorem 1, we have

$$\sup_{m \ge m_n} \frac{1}{d_n}\sum_{j=1}^{p_n} \|\mathbf{B}_j^* - \hat{\mathbf{B}}_j^{(m)}\|_F^2 = O_p\left(\frac{1}{nd_n}\left\{\|\tilde{\mathbf{Y}} - \mathbf{Y}^*\|_F^2 + \|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(m_n)}\|_F^2 + \frac{\mu^5 \kappa_n \xi_E^2}{n}\right\}\right) \tag{56}$$

By Theorem 7 and the choice of m_n, we have

$$\frac{1}{nd_n}\|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(m_n)}\|_F^2 = O_p\left(\frac{\kappa_n \xi_n^2}{n^2 d_n}\log\frac{n^2 d_n}{\xi_n^2} + \frac{\xi_n^2}{n^2 \delta_n^2}\right). \tag{57}$$

By (C6), it is not difficult to show $\bar{\mathbf{Y}}$, defined in (46), is in $\mathcal{C}_L$. It follows from the definition of $\mathbf{Y}^*$ that

$$\begin{aligned}
\|\tilde{\mathbf{Y}} - \mathbf{Y}^*\|_F^2 =& \|\mathbf{Y} - \mathbf{Y}^*\|_F^2 - \|\mathbf{E}\|_F^2 - 2\langle \mathbf{E}, \tilde{\mathbf{Y}} - \mathbf{Y}^*\rangle \\
\le& \|\mathbf{Y} - \bar{\mathbf{Y}}\|_F^2 - \|\mathbf{E}\|_F^2 - 2\langle \mathbf{E}, \tilde{\mathbf{Y}} - \mathbf{Y}^*\rangle \\
=& \|\tilde{\mathbf{Y}} - \bar{\mathbf{Y}}\|_F^2 + 2\langle \mathbf{E}, \mathbf{Y}^* - \bar{\mathbf{Y}}\rangle.
\end{aligned} \tag{58}$$

By (54) again,

$$|\langle \mathbf{E}, \mathbf{Y}^* - \bar{\mathbf{Y}}\rangle| \le \xi_E \left(\frac{\mu^5 \kappa_n}{n}\right)^{1/2} \|\bar{\mathbf{Y}} - \mathbf{Y}^*\|_F. \tag{59}$$

Now if $\|\bar{\mathbf{Y}} - \mathbf{Y}^*\|_F \ge 2\|\tilde{\mathbf{Y}} - \mathbf{Y}^*\|_F$, then $\|\bar{\mathbf{Y}} - \mathbf{Y}^*\|_F \le 2\|\bar{\mathbf{Y}} - \tilde{\mathbf{Y}}\|_F$. This, together with (58), (59), and (52), yields

$$\begin{aligned}
\|\tilde{\mathbf{Y}} - \mathbf{Y}^*\|_F^2 \le& \|\tilde{\mathbf{Y}} - \bar{\mathbf{Y}}\|_F^2 + 4\xi_E \left(\frac{\mu^5 \kappa_n}{n}\right)^{1/2} \|\bar{\mathbf{Y}} - \tilde{\mathbf{Y}}\|_F \\
\le& 2\|\tilde{\mathbf{Y}} - \bar{\mathbf{Y}}\|_F^2 + 4\mu^5 \frac{\kappa_n \xi_E^2}{n} \\
\le& 16\mu L_0^2 \frac{nd_n \xi_E^2}{(n\delta_n - \xi_E)^2} + 4\mu^5 \frac{\kappa_n \xi_E^2}{n}.
\end{aligned} \tag{60}$$

On the other hand, if $\|\bar{\mathbf{Y}} - \mathbf{Y}^*\|_F < 2\|\tilde{\mathbf{Y}} - \mathbf{Y}^*\|_F$, then (58) and (59) imply

$$\|\tilde{\mathbf{Y}} - \mathbf{Y}^*\|_F^2 \le \|\tilde{\mathbf{Y}} - \bar{\mathbf{Y}}\|_F^2 + 4\xi_E \left(\frac{\mu^5 \kappa_n}{n}\right)^{1/2} \|\tilde{\mathbf{Y}} - \mathbf{Y}^*\|_F.$$

By a similar argument used to obtain (55), this and (52) yield

$$\begin{aligned}\|\tilde{\mathbf{Y}} - \mathbf{Y}^*\|_F^2 \leq & 16\mu^5 \frac{\kappa_n \xi_E^2}{n} + 2\|\tilde{\mathbf{Y}} - \bar{\mathbf{Y}}\|_F^2 \\ \leq & 16\mu^5 \frac{\kappa_n \xi_E^2}{n} + 16\mu L_0^2 \frac{n d_n \xi_E^2}{(n\delta_n - \xi_E)^2}. \end{aligned} \tag{61}$$

In view of (56), (57), (60), (61) and (C5), the deisred result follows. ■

Appendix C. Further technical details

In this section, we present some additional auxiliary results along with the proofs of (36), (39), (40), (47), (52). Some existing results that are useful in our proofs are also stated here for completeness with the references to their proofs in the literature. These results are stated in the forms that are most convenient for our use, which may not be in full generality.

Proposition 8 (Ruhe, 1970) *Let $\mathbf{A}, \mathbf{B}$ be matrices with size $m \times n$ and $n \times p$ respectively. Then*

$$\sum_{j=1}^{n} \sigma_j^2(\mathbf{A})\sigma_j^2(\mathbf{B}) \geq \|\mathbf{AB}\|_F^2 \geq \sum_{j=1}^{n} \sigma_{n-j+1}^2(\mathbf{A})\sigma_j^2(\mathbf{B}).$$

Remark 9 One consequence of this inequality we frequently use is $\sigma_1^2(\mathbf{A})\|\mathbf{B}\|_F^2 \geq \|\mathbf{AB}\|_F^2 \geq \sigma_n^2(\mathbf{A})\|\mathbf{B}\|_F^2$. Note also that by transposition the roles of $\mathbf{A}$ and $\mathbf{B}$ can be interchanged on the left- and right-most expressions.

Lemma 10 *Assume (C1)-(C2) and that $\sum_{j=1}^{p_n} \|\mathbf{B}_j^*\|_* \leq L$. Suppose $L_n = d_n^{1/2} L_0$ is chosen so that $L_0 \geq L/(1-\epsilon_L)$ with $1 - \epsilon_L \leq 1/(4\mu^2)$. Then for first- and second-stage RGA, with probability tending to one,*

(i)

$$\inf_{k \geq 1} \frac{1}{n d_n} \|\mathbf{X}_{\hat{j}_k} \tilde{\mathbf{B}}_{\hat{j}_k} - \hat{\mathbf{G}}^{(k-1)}\|_F^2 \geq (1-\epsilon_L)\mu L_0^2 \tag{62}$$

$$\inf_{k \geq 1} \frac{1}{n d_n} \|\mathbf{X}_{\hat{j}_k} \hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1} \mathbf{U}_{\hat{j}_k} \hat{\mathbf{S}}_k \mathbf{V}_{\hat{j}_k}^\top - \hat{\mathbf{G}}^{(k-1)}\|_F^2 \geq (1-\epsilon_L)\mu L_0^2 \tag{63}$$

(ii)

$$\sup_{k \geq 1} |\lambda_k - \hat{\lambda}_k| \leq \frac{2}{(1-\epsilon_L)L_0} \frac{\xi_E}{n\sqrt{d_n}} \tag{64}$$

(iii)

$$\max_{1 \leq k \leq K_n} \lambda_k \leq 1. \tag{65}$$

Proof We shall prove the results for the second-stage RGA. The corresponding proofs for first-stage RGA follow similarly and thus are omitted. It is also sufficient to prove (i)-(iii) assuming the condition described in (C1) holds almost surely because the event that the condition holds has probability tending to one. It will greatly simplify the exposition (without repeating that the inequalities holds except on a vanishing event). Note that

$$\begin{aligned}
&\langle \mathbf{X}_{\hat{j}_k} \hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1} \mathbf{U}_{\hat{j}_k} \hat{\mathbf{S}}_k \mathbf{V}_{\hat{j}_k}^\top, \tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)} \rangle \\
=&\langle \mathbf{X}_{\hat{j}_k} \hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1} \mathbf{U}_{\hat{j}_k} \hat{\mathbf{S}}_k \mathbf{V}_{\hat{j}_k}^\top, \mathbf{Y} - \hat{\mathbf{G}}^{(k-1)} \rangle - \langle \mathbf{X}_{\hat{j}_k} \hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1} \mathbf{U}_{\hat{j}_k} \hat{\mathbf{S}}_k \mathbf{V}_{\hat{j}_k}^\top, \mathbf{E} \rangle \\
\geq& - |\langle \mathbf{X}_{\hat{j}_k} \hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1} \mathbf{U}_{\hat{j}_k} \hat{\mathbf{S}}_k \mathbf{V}_{\hat{j}_k}^\top, \mathbf{E} \rangle| \\
\geq& - \|\hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1} \mathbf{X}_{\hat{j}_k}^\top \mathbf{E}\|_{op} \|\mathbf{U}_{\hat{j}_k} \hat{\mathbf{S}}_k \mathbf{V}_{\hat{j}_k}^\top\|_* \\
\geq& - \mu L_n \xi_E,
\end{aligned}$$

where the first inequality follows because $\langle \mathbf{X}_{\hat{j}_k} \hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1} \mathbf{U}_{\hat{j}_k} \hat{\mathbf{S}}_k \mathbf{V}_{\hat{j}_k}^T, \mathbf{Y} - \hat{\mathbf{G}}^{(k-1)} \rangle \geq 0$ with probability one and the second inequality follows because the dual norm of the nuclear norm is the operator norm. By Proposition 8, we have

$$\begin{aligned}
&\|\mathbf{X}_{\hat{j}_k} \hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1} \mathbf{U}_{\hat{j}_k} \hat{\mathbf{S}}_k \mathbf{V}_{\hat{j}_k}^\top - \hat{\mathbf{G}}^{(k-1)}\|_F^2 \\
\geq& \|\mathbf{X}_{\hat{j}_k} \hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1} \mathbf{U}_{\hat{j}_k} \hat{\mathbf{S}}_k \mathbf{V}_{\hat{j}_k}^\top\|_F^2 - 2\langle \mathbf{X}_{\hat{j}_k} \hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1} \mathbf{U}_{\hat{j}_k} \hat{\mathbf{S}}_k \mathbf{V}_{\hat{j}_k}^\top, \hat{\mathbf{G}}^{(k-1)} \rangle \\
\geq& n\mu^{-1} \|\mathbf{U}_{\hat{j}_k} \hat{\mathbf{S}}_k \mathbf{V}_{\hat{j}_k}^\top\|_F^2 \\
&+ 2\langle \mathbf{X}_{\hat{j}_k} \hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1} \mathbf{U}_{\hat{j}_k} \hat{\mathbf{S}}_k \mathbf{V}_{\hat{j}_k}^\top, \tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)} \rangle - 2\langle \mathbf{X}_{\hat{j}_k} \hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1} \mathbf{U}_{\hat{j}_k} \hat{\mathbf{S}}_k \mathbf{V}_{\hat{j}_k}^\top, \tilde{\mathbf{Y}} \rangle \\
\geq& n\mu^{-1} L_n^2 - 2\mu L_n \xi_E - 2\langle \mathbf{X}_{\hat{j}_k} \hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1} \mathbf{U}_{\hat{j}_k} \hat{\mathbf{S}}_k \mathbf{V}_{\hat{j}_k}^\top, \tilde{\mathbf{Y}} \rangle,
\end{aligned}$$

where the last inequality follows from the fact that $\hat{\mathbf{S}}_k$ is rank-one with singular value L_n. Thus, by writing $\hat{\mathbf{S}}_k = L_n \mathbf{a}\mathbf{b}^T$ for some unit vectors $\mathbf{a}, \mathbf{b}$, we have $\|\mathbf{U}_{\hat{j}_k} \hat{\mathbf{S}}_k \mathbf{V}_{\hat{j}_k}^T\|_F^2 = L_n^2 \|\mathbf{U}_{\hat{j}_k} \mathbf{a}\mathbf{b}^T \mathbf{V}_{\hat{j}_k}^T\|_F^2 = L_n^2$. Next, observe that

$$\begin{aligned}
|\langle \mathbf{X}_{\hat{j}_k} \hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1} \mathbf{U}_{\hat{j}_k} \hat{\mathbf{S}}_k \mathbf{V}_{\hat{j}_k}^\top, \tilde{\mathbf{Y}} \rangle| =& \left| \sum_{j=1}^{p_n} \langle \mathbf{X}_{\hat{j}_k} \hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1} \mathbf{U}_{\hat{j}_k} \hat{\mathbf{S}}_k \mathbf{V}_{\hat{j}_k}^\top, \mathbf{X}_j \mathbf{B}_j^* \rangle \right| \\
\leq& \sum_{j=1}^{p_n} \|\mathbf{B}_j^*\|_* \|\mathbf{X}_j^\top \mathbf{X}_{\hat{j}_k} \hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1} \mathbf{U}_{\hat{j}_k} \hat{\mathbf{S}}_k \mathbf{V}_{\hat{j}_k}^\top\|_{op} \\
\leq& (1 - \epsilon_L) L_n^2 n\mu.
\end{aligned}$$

Therefore,

$$\begin{aligned}
(nd_n)^{-1} \|\mathbf{X}_{\hat{j}_k} \hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1} \mathbf{U}_{\hat{j}_k} \hat{\mathbf{S}}_k \mathbf{V}_{\hat{j}_k}^\top - \hat{\mathbf{G}}^{(k-1)}\|_F^2 \geq& \mu^{-1} L_0^2 - 2(1 - \epsilon_L) L_0^2 \mu - 2\mu L_0 \frac{\xi_E}{n\sqrt{d_n}} \\
\geq& 2(1 - \epsilon_L) L_0^2 \mu - 2\mu L_0 \frac{\xi_E}{n\sqrt{d_n}}.
\end{aligned}$$

Since $\xi_E = o_p(n\sqrt{d_n})$ by (C2), (63) follows.

For (64), note first that if the solutions to the line search problems (9) and (25) (with $\tilde{\mathbf{B}}_{\hat{j}_k}$ replaced by $\hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1}\mathbf{U}_{\hat{j}_k}\hat{\mathbf{S}}_k\mathbf{V}_{\hat{j}_k}^\top$) for second-stage RGA are not constrained to be in $[0,1]$, then they are given by

$$\begin{aligned}\hat{\lambda}_{k,uc} =& \frac{\langle \mathbf{Y} - \hat{\mathbf{G}}^{(k-1)}, \mathbf{X}_{\hat{j}_k}\hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1}\mathbf{U}_{\hat{j}_k}\hat{\mathbf{S}}_k\mathbf{V}_{\hat{j}_k}^\top - \hat{\mathbf{G}}^{(k-1)}\rangle}{\|\mathbf{X}_{\hat{j}_k}\hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1}\mathbf{U}_{\hat{j}_k}\hat{\mathbf{S}}_k\mathbf{V}_{\hat{j}_k}^\top - \hat{\mathbf{G}}^{(k-1)}\|_F^2},\\ \lambda_{k,uc} =& \frac{\langle \tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}, \mathbf{X}_{\hat{j}_k}\hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1}\mathbf{U}_{\hat{j}_k}\hat{\mathbf{S}}_k\mathbf{V}_{\hat{j}_k}^\top - \hat{\mathbf{G}}^{(k-1)}\rangle}{\|\mathbf{X}_{\hat{j}_k}\hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1}\mathbf{U}_{\hat{j}_k}\hat{\mathbf{S}}_k\mathbf{V}_{\hat{j}_k}^\top - \hat{\mathbf{G}}^{(k-1)}\|_F^2}.\end{aligned}$$

Since $\hat{\mathbf{G}}^{(l)}$ can always be expressed as $\hat{\mathbf{G}}^{(l)} = \sum_{j\in\hat{J}}\mathbf{X}_j\hat{\mathbf{\Sigma}}_j^{-1}\mathbf{U}_j\mathbf{A}_j\mathbf{V}_j^\top$ with $\sum_{j\in\hat{J}}\|\mathbf{A}_j\|_* \le L_n$, it follows that

$$\begin{aligned}|\hat{\lambda}_k - \lambda_k| \le |\hat{\lambda}_{k,uc} - \lambda_{k,uc}| =& \frac{|\langle \mathbf{E}, \mathbf{X}_{\hat{j}_k}\hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1}\mathbf{U}_{\hat{j}_k}\hat{\mathbf{S}}_k\mathbf{V}_{\hat{j}_k}^\top - \hat{\mathbf{G}}^{(k-1)}\rangle|}{\|\mathbf{X}_{\hat{j}_k}\hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1}\mathbf{U}_{\hat{j}_k}\hat{\mathbf{S}}_k\mathbf{V}_{\hat{j}_k}^\top - \hat{\mathbf{G}}^{(k-1)}\|_F^2}\\ \le& \frac{2L_n\mu\xi_E}{\|\mathbf{X}_{\hat{j}_k}\hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1}\mathbf{U}_{\hat{j}_k}\hat{\mathbf{S}}_k\mathbf{V}_{\hat{j}_k}^\top - \hat{\mathbf{G}}^{(k-1)}\|_F^2}\\ \le& \frac{2\xi_E}{nd_n^{1/2}(1-\epsilon_L)L_0},\end{aligned}$$

with probability tending to one, where the last inequality follows from (63).

For (65), it suffices to prove that $\lim_{n\to\infty}\mathbb{P}(E_n) = 1$, where $E_n = \{\max_{1\le k\le K_n}\lambda_{k,uc} \le 1\}$. On E_n^c, there exists some k such that, by Cauchy-Schwarz inequality and (27),

$$\begin{aligned}\|\mathbf{X}_{\hat{j}_k}\hat{\mathbf{\Sigma}}_{\hat{j}_k}^{-1}\mathbf{U}_{\hat{j}_k}\hat{\mathbf{S}}_k\mathbf{V}_{\hat{j}_k}^T - \hat{\mathbf{G}}^{(k-1)}\|_F^2 \le& \|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(k-1)}\|_F^2\\ \le& \|\tilde{\mathbf{Y}}\|_F^2 + 2\sum_{j=1}^{k-1}\langle \mathbf{E}, \hat{\mathbf{G}}^{(k-j)} - \mathbf{G}^{(k-j)}\rangle\\ =& \|\tilde{\mathbf{Y}}\|_F^2 + 2\sum_{l=1}^{k-1}(\hat{\lambda}_l - \lambda_l)\langle \mathbf{E}, \mathbf{X}_{\hat{j}_l}\hat{\mathbf{\Sigma}}_{\hat{j}_l}^{-1}\mathbf{U}_{\hat{j}_l}\hat{\mathbf{S}}_l\mathbf{V}_{\hat{j}_l}^\top - \hat{\mathbf{G}}^{(l-1)}\rangle\\ \le& \|\tilde{\mathbf{Y}}\|_F^2 + 4K_nL_n\mu\xi_E\max_{1\le l\le k}|\hat{\lambda}_l - \lambda_l|. \qquad (66)\end{aligned}$$

It is easy to see that

$$\|\tilde{\mathbf{Y}}\|_F = \left\|\sum_{j=1}^{p_n}\mathbf{X}_j\mathbf{B}_j^*\right\|_F \le (1-\epsilon_L)L_n\sqrt{n\mu}. \qquad (67)$$

Thus, by (63), (64) and (66)-(67), we have

$$\mathbb{P}(E_n^c) \le \mathbb{P}\left((1-\epsilon_L)L_0^2\mu\{1-(1-\epsilon_L)\} \le \frac{8\mu}{1-\epsilon_L}\frac{K_n\xi_E^2}{n^2d_n}\right) + o(1) = o(1),$$

where the last equality follows from (C2). ■

Lemma 11 *Let $\{a_m\}$ be a nonnegative sequence of reals. If*

$$a_0 \le A, \text{ and } a_m \le a_{m-1}\left(1 - \frac{\xi^2 a_{m-1}}{A}\right) + b_m,$$

for $m = 1, 2, \ldots$, where $b_m \ge 0$ with $b_0 = 0$, then for each m,

$$a_m \le \frac{A}{1 + m\xi^2} + \sum_{k=0}^{m} b_k. \tag{68}$$

Proof We prove by induction. When $m = 0$, (68) holds by assumption. Suppose now that (68) holds for some $m \ge 1$. Then

$$\begin{aligned}
a_{m+1} \le & a_m\left(1 - \frac{\xi^2 a_m}{A}\right) + b_{m+1} \\
\le & \frac{1}{a_m^{-1} + \xi^2/A} + b_{m+1} \\
\le & \frac{1}{\left(\frac{A}{1+m\xi^2} + \sum_{k=0}^{m} b_k\right)^{-1} + \xi^2/A} + b_{m+1} \\
= & \frac{\frac{A}{1+m\xi^2} + \sum_{k=0}^{m} b_k}{1 + \frac{\xi^2}{A}\left(\frac{A}{1+m\xi^2} + \sum_{k=0}^{m} b_k\right)} + b_{m+1} \\
\le & \frac{A}{1 + (m+1)\xi^2} + \sum_{k=0}^{m+1} b_k,
\end{aligned}$$

where the second inequality follows from $1 - x \le 1/(1 + x)$ for $x \ge 0$. ■

Remark 12 Lemma 11 is a slight modification of Lemma 3.1 of Temlyakov (2000).

Proof [Proof of (36)] On $\mathcal{E}_n^c(m)$, there exists some $l \le m$ such that

$$\tilde{\tau} d_n^{1/2} \xi_E \ge \max_{\substack{1 \le j \le p_n \\ \|\mathbf{B}_j\|_* \le L_n}} \langle \tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(l-1)}, \mathbf{X}_j \mathbf{B}_j - \hat{\mathbf{G}}^{(l-1)} \rangle \ge \|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(l-1)}\|_F^2.$$

By (27) and Lemma 10(ii), it follows that, on $\mathcal{E}_n^c(m)$ except for a vanishing event,

$$\begin{aligned}
\|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(m)}\|_F^2 \le & \|\tilde{\mathbf{Y}} - \hat{\mathbf{G}}^{(l-1)}\|_F^2 + 2\sum_{k=l}^{m} \langle \mathbf{E}, \hat{\mathbf{G}}^{(k)} - \mathbf{G}^{(k)} \rangle \\
\le & \tilde{\tau} d_n^{1/2} \xi_E + 2\sum_{k=l}^{m} (\hat{\lambda}_k - \lambda_k) \langle \mathbf{E}, \mathbf{X}_{\hat{j}_k} \tilde{\mathbf{B}}_{\hat{j}_k} - \hat{\mathbf{G}}^{(k-1)} \rangle \\
\le & \tilde{\tau} d_n^{1/2} \xi_E + \frac{8m\xi_E^2}{n(1 - \epsilon_L)},
\end{aligned}$$

which is the desired result. ■

Proof [Proof of (39) and (40)] Note first that for any $D > 0$, $(D+x)/(D-x) \le 1+3x/D$ for all $0 \le x \le (1-\sqrt{2/3})D$. It is not difficult to see that

$$\begin{aligned}
&\mathbb{P}\left\{\frac{4L_0\xi_E}{nd_n^{1/2}} \le (1-\sqrt{\frac{2}{3}})\left((nd_n)^{-1}\|\tilde{\mathbf{Y}}-\hat{\mathbf{G}}^{(k)}\|_F^2+(nd_n)^{-1}\|\mathbf{E}\|_F^2\right), 1\le k\le \hat{k}, \mathcal{G}_n\right\}\\
\ge&\mathbb{P}\left\{\frac{4L_0\xi_E}{nd_n^{1/2}} \le (1-\sqrt{\frac{2}{3}})M^{-1}\right\}-o(1)\\
\to&1.
\end{aligned}$$

Thus, on $\mathcal{G}_n$ except for a vanishing event,

$$\begin{aligned}
A_k \le& 1+\frac{12L_0\xi_E/(nd_n^{1/2})}{(nd_n)^{-1}\|\tilde{\mathbf{Y}}-\hat{\mathbf{G}}^{(k)}\|_F^2+(nd_n)^{-1}\|\mathbf{E}\|_F^2}\\
\le& 1+12ML_0\frac{\xi_E}{nd_n^{1/2}},
\end{aligned}$$

for all $1 \le k \le \hat{k}$. This proves (39). We now turn to (40). Since for any positive A and B, $A/(B+x) \ge A(1-x/B)/B$ for all $x \ge 0$, it follows from (37) that on $\mathcal{G}_n$ except for a vanishing event,

$$\begin{aligned}
B_k \ge& \frac{\tau^2 s_n^{-1}}{4L_0^2\mu^2}\frac{(nd_n)^{-1}\|\tilde{\mathbf{Y}}-\hat{\mathbf{G}}^{(k-1)}\|_F^2}{(nd_n)^{-1}\|\tilde{\mathbf{Y}}-\hat{\mathbf{G}}^{(k-1)}\|_F^2+(nd_n)^{-1}\|\mathbf{E}\|_F^2}\\
&\times\left(1-\frac{4L_0\xi_E/(nd_n^{1/2})}{(nd_n)^{-1}\|\tilde{\mathbf{Y}}-\hat{\mathbf{G}}^{(k-1)}\|_F^2+(nd_n)^{-1}\|\mathbf{E}\|_F^2}\right)\\
\ge& \frac{\tau^2 s_n^{-1}}{4L_0^2\mu^2}\frac{1}{1+\mu M s_n}\left(1-\frac{4ML_0\xi_E}{nd_n^{1/2}}\right)
\end{aligned}$$

for $1 \le k \le \hat{k}$, which proves (40). ■

Proof [Proof of (47)] Let

$$\mathbf{H} = \sum_{j\in\hat{J}} \mathbf{X}_j\hat{\mathbf{\Sigma}}_j^{-1}\mathbf{U}_j\mathbf{D}_j\mathbf{V}_j^\top \in \mathcal{B}.$$

Note that Proposition 8 and (C3) imply

$$\begin{aligned}\|\bar{\mathbf{Y}}-\mathbf{H}\|_F^2 \geq & n\mu^{-1}\left\{\sum_{j\in\hat{J}_o}\|\hat{\mathbf{\Sigma}}_j^{-1}\mathbf{U}_j(\mathbf{L}_j\mathbf{\Lambda}_j\mathbf{R}_j^\top-\mathbf{D}_j)\mathbf{V}_j^\top\|_F^2+\sum_{j\in\hat{J}-\hat{J}_o}\|\hat{\mathbf{\Sigma}}_j^{-1}\mathbf{U}_j\mathbf{D}_j\mathbf{V}_j^\top-\mathbf{B}_j^*\|_F^2\right\}\\ \geq & n\mu^{-3}\left\{\sum_{j\in\hat{J}_o}\|\mathbf{L}_j\mathbf{\Lambda}_j\mathbf{R}_j^\top-\mathbf{D}_j\|_F^2+\sum_{j\in\hat{J}-\hat{J}_o}\|\mathbf{U}_j^\top\hat{\mathbf{\Sigma}}_j\mathbf{B}_j^*\mathbf{V}_j-\mathbf{D}_j\|_F^2\right\}\\ \geq & \frac{n}{\mu^3\kappa_n}\left\{\sum_{j\in\hat{J}_o}\|\mathbf{L}_j\mathbf{\Lambda}_j\mathbf{R}_j^\top-\mathbf{D}_j\|_*+\sum_{j\in\hat{J}-\hat{J}_o}\|\mathbf{U}_j^\top\hat{\mathbf{\Sigma}}_j\mathbf{B}_j^*\mathbf{V}_j-\mathbf{D}_j\|_*\right\}^2.\end{aligned}$$

Since $\mathbf{H}\in\mathcal{B}$, we have

$$\left\{\sum_{j\in\hat{J}_o}\|\mathbf{L}_j\mathbf{\Lambda}_j\mathbf{R}_j^\top-\mathbf{D}_j\|_*+\sum_{j\in\hat{J}-\hat{J}_o}\|\mathbf{U}_j^\top\hat{\mathbf{\Sigma}}_j\mathbf{B}_j^*\mathbf{V}_j-\mathbf{D}_j\|_*\right\}^2\leq\frac{9d_nL_0^2}{16}=\frac{9L_n^2}{16}.$$

By the triangle inequality, we have $\sum_{j\in\hat{J}}\|\mathbf{D}_j\|_*\leq 3L_n/4+\sum_{j\in\hat{J}_o}\|\mathbf{\Lambda}_j\|_*+\sum_{j\in\hat{J}-\hat{J}_o}\|\hat{\mathbf{\Sigma}}_j\mathbf{B}_j^*\|_*$. Because of (C6), and $\hat{J}_o\subset J_o$ (with probability tending to one), $\sum_{j\in\hat{J}_o}\|\mathbf{\Lambda}_j\|_*+\sum_{j\in\hat{J}-\hat{J}_o}\|\hat{\mathbf{\Sigma}}_j\mathbf{B}_j^*\|_*\leq\sum_{j\in\hat{J}}\|\hat{\mathbf{\Sigma}}_j\mathbf{B}_j^*\|_*\leq\mu(1-\epsilon_L)L_n\leq 4^{-1}\mu^{-1}L_n\leq L_n/4$. Hence $\sum_{j\in\hat{J}_{\hat{k}}}\|\mathbf{D}_j\|_*\leq L_n$, which proves $\mathbf{H}\in\mathcal{C}_L$. ■

Proposition 13 *Let $\mathbf{A}^*$ be an $m\times n$ matrix and $\mathbf{A}=\mathbf{A}^*+\mathbf{E}$ be its perturbed version. Let $\mathbf{U}_*\mathbf{\Sigma}_*\mathbf{V}_*^\top$ and $\mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$ be their truncated SVD of rank r_*, respectively. If $\sigma_{r_*}(\mathbf{A}^*):=\sigma_{r_*}>\sigma_{r_*+1}(\mathbf{A}^*)=0$, and if $\|\mathbf{E}\|_{op}<\sigma_{r_*}$, then*

$$\max\{\mathrm{dist}(\mathbf{U}_*,\mathbf{U}),\mathrm{dist}(\mathbf{V}_*,\mathbf{V})\}\leq\frac{\sqrt{2}\max\{\|\mathbf{E}^\top\mathbf{U}_*\|_{op},\|\mathbf{E}\mathbf{V}_*\|_{op}\}}{\sigma_{r_*}-\|\mathbf{E}\|_{op}},$$

where $\mathrm{dist}(\mathbf{Q},\mathbf{Q}_*)=\min_{\mathbf{R}}\|\mathbf{Q}\mathbf{R}-\mathbf{Q}_*\|_{op}$ for any two orthogonal matrices $\mathbf{Q}$, $\mathbf{Q}^*$ with r columns, where the minimum is taken over all $r\times r$ orthonormal matrices.

Remark 14 Proposition 13 is a consequence of the perturbation bounds for singular values (Wedin, 1972). A proof can be found in Chen et al. (2021).

Proof [Proof of (52)] Note first that

$$\begin{aligned}\bar{\mathbf{Y}}-\tilde{\mathbf{Y}}=&\sum_{j\in\hat{J}_o}\mathbf{X}_j\hat{\mathbf{\Sigma}}_j^{-1}(\mathbf{U}_j\mathbf{L}_j-\tilde{\mathbf{U}}_j)\mathbf{\Lambda}_j\tilde{\mathbf{V}}_j^\top\\ &+\sum_{j\in\hat{J}_o}\mathbf{X}_j\hat{\mathbf{\Sigma}}_j^{-1}\mathbf{U}_j\mathbf{L}_j\mathbf{\Lambda}_j(\mathbf{V}_j\mathbf{R}_j-\tilde{\mathbf{V}}_j)^\top.\end{aligned}$$

By triangle inequality,

$$\|\bar{\mathbf{Y}} - \tilde{\mathbf{Y}}\|_F \leq \sqrt{n\mu} \left(\sum_{j \in \hat{J}_o} \|\mathbf{\Lambda}_j\|_F \right) \left\{ \max_{j \in \hat{J}_o} \|\mathbf{U}_j \mathbf{L}_j - \tilde{\mathbf{U}}_j\|_{op} + \max_{j \in \hat{J}_o} \|\mathbf{V}_j \mathbf{R}_j - \tilde{\mathbf{V}}_j\|_{op} \right\}. \quad (69)$$

Let $\mathbf{U}_{j,\bar{r}_j}$ and $\mathbf{V}_{j,\bar{r}_j}$ be sub-matrices of $\mathbf{U}_j$ and $\mathbf{V}_j$ consisting of column vectors that correspond to the leading $\bar{r}_j$ singular vectors. Write $\mathbf{U}_j = (\mathbf{U}_{j,\bar{r}_j}, \mathbf{U}_{j,-\bar{r}_j})$ and $\mathbf{V}_j = (\mathbf{V}_{j,\bar{r}_j}, \mathbf{V}_{j,-\bar{r}_j})$. Since $\mathbf{X}_j^\top \tilde{\mathbf{Y}} = \mathbf{X}_j^\top \mathbf{Y} - \mathbf{X}_j^\top \mathbf{E}$, it follows from Proposition 13 and (C5) that there exist $\bar{r}_j \times \bar{r}_j$ orthonormal matrices $\tilde{\mathbf{L}}_j$ and $\tilde{\mathbf{R}}_j$ such that with probability tending to one,

$$\begin{aligned} \max \left\{ \|\mathbf{U}_{j,\bar{r}_j} \tilde{\mathbf{L}}_j - \tilde{\mathbf{U}}_j\|_{op}, \|\mathbf{V}_{j,\bar{r}_j} \tilde{\mathbf{R}}_j - \tilde{\mathbf{V}}_j\|_{op} \right\} \leq & \frac{\sqrt{2} \max\{\|\mathbf{E}^\top \mathbf{X}_j \tilde{\mathbf{U}}_j\|_{op}, \|\mathbf{X}_j^\top \mathbf{E} \tilde{\mathbf{V}}_j\|_{op}\}}{n\delta_n - \|\mathbf{X}_j^\top \mathbf{E}\|_{op}} \\ \leq & \frac{\sqrt{2}\xi_E}{n\delta_n - \xi_E}. \end{aligned}$$

Set $\mathbf{L}_j^\top = (\tilde{\mathbf{L}}_j^\top, \mathbf{0}_{\bar{r}_j \times (\hat{r} - \bar{r}_j)})$ and $\mathbf{R}_j^\top = (\tilde{\mathbf{R}}_j^\top, \mathbf{0}_{\bar{r}_j \times (\hat{r} - \bar{r}_j)})$ for $j \in \hat{J}_o$ in (69). Then by (C4) and (C6), it follows that

$$\|\bar{\mathbf{Y}} - \tilde{\mathbf{Y}}\|_F^2 \leq n\mu \left(\sum_{j \in \hat{J}_o} \|\mathbf{\Lambda}_j\|_F \right)^2 \left(\frac{2\sqrt{2}\xi_E}{n\delta_n - \xi_E} \right)^2 \leq 8\mu L^2 n d_n \frac{\xi_E^2}{(n\delta_n - \xi_E)^2}.$$

■

Proof [Proof of Corollary 5] By Lemma 2, $\sharp(\hat{J}) + \hat{r} = O_p(s_n^2)$. Thus running the first-stage RGA with the just-in-time stopping criterion costs

$$O_p(s_n^2(n_1 + d_n)) \quad (70)$$

bytes of communication per computing node. In addition, preparing $\{\hat{\mathbf{\Sigma}}_j^{-1} : j \in \hat{J}\}$ and $(\mathbf{U}_j, \mathbf{V}_j)$ for $j \in \hat{J}$ with $q_{n,j} \wedge d_n > \hat{r}$ costs

$$\begin{aligned} & O_p \left(\sum_{j \in \hat{J}} \{q_{n,j}^2 + (q_{n,j} d_n + \hat{r}(q_{n,j} + d_n)) \mathbf{1}\{q_{n,j} \wedge d_n > \hat{r}\}\} \right) \\ = & O_p(n_1^{2\alpha} s_n^2 + n_1^\alpha d_n s_n^2 + s_n^4 (n_1^\alpha + d_n)). \end{aligned} \quad (71)$$

Since the communication costs per node at the k-th iteration of the second-stage RGA is at most

$$\begin{aligned} & O_p \left(\sum_{j \in \hat{J}} \left(\hat{r}^2 \mathbf{1}\{q_{n,j} \wedge d_n > \hat{r}\} + q_{n,j} d_n \mathbf{1}\{q_{n,j} \wedge d_n \leq \hat{r}\} \right) + d_n k + n_1 \right) \\ = & O_p \left(s_n^6 + n_1^\alpha d_n s_n^2 + d_n k + n_1 \right), \end{aligned}$$

running $m_n = O_p(s_n^4 \log(n^2 d_n/\xi_n^2))$ iterations (see Theorem 3 for the definition of m_n) costs

$$O_p\left((s_n^6 + s_n^2 n_1^\alpha d_n + n_1) s_n^4 \log \frac{n^2 d_n}{\xi_n^2} + d_n s_n^8 \left(\log \frac{n^2 d_n}{\xi_n^2}\right)^2\right). \tag{72}$$

Combining (70)-(72) yields the desired result. ■

Appendix D. TSRGA for high-dimensional generalized linear models

In this section, we apply the idea of TSRGA to and propose a modified algorithm for estimating the generalized linear model (GLM). Focusing on the case of a scalar response y_t, the GLM postulates that the probability density function f of y_t (or the probability mass function if y_t is discrete) belongs to the exponential family. In particular,

$$f(y;\theta) = \exp[y\theta - r(\theta) + h(y)],$$

and

$$\mathbb{E}(y_t | x_{t,1}, \ldots, x_{t,p_n}) = r'\left(\sum_{j=1}^{p_n} \beta_j^* x_{t,j}\right)$$

where θ is called the natural parameter; r, h are known functions, and r' is the derivative of r, which is also known as the inverse of the link function (see, e.g., Dunn and Smyth, 2018; Han et al., 2023). To maximize the log-likelihood function, scaled as $y\theta - r(\theta)$, one can minimize the following loss function

$$\mathcal{L}_n(\mathbf{X}\boldsymbol{\beta}) = \frac{1}{n}\sum_{t=1}^{n}\left[-y_t\left(\sum_{j=1}^{p_n}\beta_j x_{t,j}\right) + r\left(\sum_{j=1}^{p_n}\beta_j x_{t,j}\right)\right],$$

where $\mathcal{L}_n(\boldsymbol{\tau}) = n^{-1}\sum_{t=1}^{n}(y_t\tau_t - r(\tau_t))$ for $\boldsymbol{\tau} = (\tau_1, \ldots, \tau_n)^\top$.

Interpreting $y_t - r'(\sum_{j=1}^{p_n}\beta_j x_{t,j})$ as the residual, we can implement RGA as follows. First initialize $\hat{\mathbf{G}}^{(0)} = 0$. Then for $k = 1, 2, \ldots, K_n$, find

$$\hat{j}_k \in \arg\max_{1\le j\le p_n}\left|\frac{1}{n}\sum_{t=1}^{n}\left(y_t - r'(\hat{G}_t^{(k-1)})\right)x_{t,j}\right| \tag{73}$$

and update

$$\hat{\mathbf{G}}^{(k)} = (1-\hat{\lambda}_k)\hat{\mathbf{G}}^{(k-1)} + \hat{\lambda}_k L s_k \mathbf{z}_{\hat{j}_k}, \tag{74}$$

where $\hat{\mathbf{G}}^{(k)} = (\hat{G}_1^{(k)}, \ldots, \hat{G}_n^{(k)})^\top$, $L > 0$ is given, $\mathbf{z}_j = (x_{1,j}, \ldots, x_{n,j})^\top$,

$$s_k = \operatorname{sgn}\left(\frac{1}{n}\sum_{t=1}^{n}\left(y_t - r'(\hat{G}_t^{(k-1)})\right)x_{t,\hat{j}_k}\right),$$

and $\hat{\lambda}_k$ is determined by

$$\hat{\lambda}_k = \arg\min_{\lambda\in[0,1]} \mathcal{L}_n((1-\lambda)\hat{\mathbf{G}}^{(k-1)} + \lambda L s_k \mathbf{z}_{\hat{j}_k}).$$

It is not difficult to see that (73) can be easily solved for feature-distributed data and constructing $\hat{\mathbf{G}}^{(k)}$ in each node requires a communication cost of $O(n)$ bytes. The second-stage RGA can be implemented similarly with the set of predictors considered in (73) restricted to $\hat{J}$, the set of predictors chosen by the first-stage when the just-in-time criterion is met. Finally, since $\mathcal{L}_n$ could take negative values, we modify the just-in-time criterion (6) as

$$\hat{k} = \min\left\{1 \le k \le K_n : \left|\frac{\mathcal{L}_n(\hat{\mathbf{G}}^{(k)})}{\mathcal{L}_n(\hat{\mathbf{G}}^{(k-1)})} - 1\right| < t_n\right\}. \tag{75}$$

In the same spirit as (6), (75) terminates the first-stage RGA as soon as the improvement in the loss function is below certain threshold, which would save some communication costs and speed up the algorithm.

Next, we examine the performance of this version of TSRGA ((73)-(75)) using simulations. In the following experiments, the predictors $x_{t,j}$ are generated as in Specification 2. We consider the following two specifications.

Specification 5 (Logit model) The response y_t takes only values in $\{0, 1\}$ and is generated via

$$\mathbb{P}(y_t = y, \theta_t) = \theta_t^y (1-\theta_t)^{1-y}, \quad \theta_t = \frac{1}{1+\exp(-\sum_{j=1}^{p_n} \beta_j^* x_{t,j})}$$

where $(\beta_1^*, \beta_2^*, \beta_3^*, \beta_4^*, \beta_5^*) = (-2.4, 1.8, -1.9, 2.8, -2.2)$, $\beta_j^* = 0$ for $j > 5$. For this model, we have $r(\theta) = \log(1+\exp(\theta))$.

Specification 6 (Poisson model) The response y_t takes values in $\{0, 1, 2, \ldots\}$ and is generated via

$$\mathbb{P}(y_t = y, \theta_t) = \frac{\theta_t^y e^{-\theta_t}}{y!}, \quad \theta_t = \exp(\sum_{j=1}^{p_n} \beta_j^* x_{t,j})$$

and $(\beta_1^*, \beta_2^*, \beta_3^*, \beta_4^*, \beta_5^*) = (0.15, -0.25, 0.35, -0.45, 0.55)$, $\beta_j^* = 0$ for $j > 5$. For this model, we have $r(\theta) = \exp(\theta)$.

As a benchmark, we compare with the ℓ_1-regularized GLM which solves

$$\min_{\boldsymbol{\beta}} \mathcal{L}_n(\mathbf{X}\boldsymbol{\beta}) + \lambda\|\boldsymbol{\beta}\|_1 \tag{76}$$

with λ selected by 5-fold cross validation. Table 4 reports the parameter estimation error $\|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}^*\|_2$, the number of irrelevant variables selected (false positives, FP) and the number of relevant variables not selected (false negatives, FN). For the logit model, we additionally report the out-of-sample prediction accuracy on a test set of size 500. For the Poisson model,

Logit	$n = 800, p = 1200$		$n = 1200, p = 2000$	
	TSRGA	ℓ_1-GLM	TSRGA	ℓ_1-GLM
$\|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}^*\|_2$	0.698	2.185	0.689	2.036
FP	0.018	82.808	0	105.070
FN	0	0	0	0
Accuracy	0.901	0.888	0.901	0.892
Poisson				
$\|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}^*\|_2$	0.135	0.190	0.060	0.144
FP	6.638	25.470	1.830	25.146
FN	0.114	0.008	0.020	0
RMSE	1.324	1.363	1.283	1.329

Table 4: Simulation results for estimating high-dimensional GLMs. ℓ_1-GLM is defined in (76). The results are based on 500 simulations.

the out-of-sample prediction error is measured by RMSE. All these figures are averages over 500 independent simulations.

The results show that for both the logit and Poisson models, TSRGA yields parsimonious and accurate coefficient estimates, with comparable out-of-sample prediction accuracy to the ℓ_1-GLM defined by (76). In particular, the low FP and FN values of TSRGA may be due to its variable selection properties. Though we expect the general conclusions about TSRGA in this paper, such as the sure-screening property, to hold under the GLM framework, the rigorous mathematical treatment is left for future work.

Appendix E. Complementary empirical results

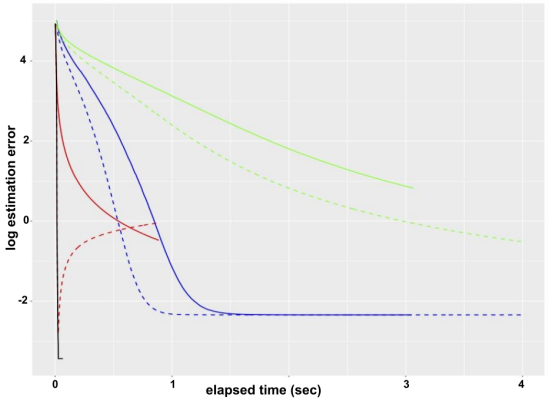
In this section, we present some additional simulation results regarding Specifications 1 and 2, and supplementary results associated with the real data example.

Figures 5 and 6 plot the parameter estimation error, as in Figures 1 and 2, against the elapsed time. Clearly, TSRGA converges within the least amount of time. In particular, its second-stage only takes a very short amount of time, thanks to the dimension reduction after the just-in-time stopping criterion. Other methods behave similarly as those in Figures 1 and 2, as their implementation cost scales directly with the number of iterations.

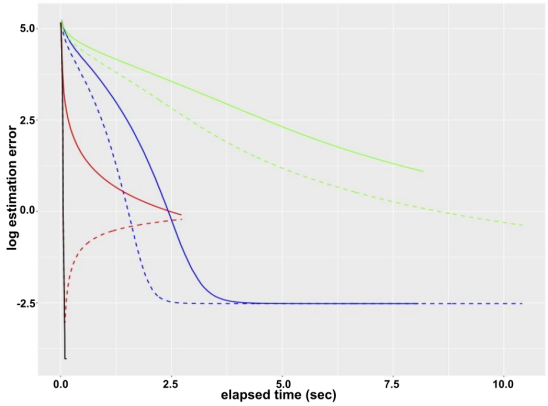
Figures 7 and 8 plot the out-of-sample prediction error (measures by the root mean square prediction error on an independent test sample) of the methods under Specifications 1 and 2. For Specification 1, the final prediction accuracy of TSRGA, cross-validated Lasso, and Hydra are similar. However, for Specification 2, TSRGA clearly is the most desirable prediction tool among the methods under consideration.

Tables 5 and 6 report the p-values of testing, for each month, whether Lasso, iRRR, and TSRGA produced smaller errors compared to the group-wise VAR when predicting volatility and return respectively. A smaller p-value indicates the method outperforms gVAR with higher confidence. See Section 5 for the description of the data and the experiment. For volatility, the results show that the methods achieve statistically significant improvements

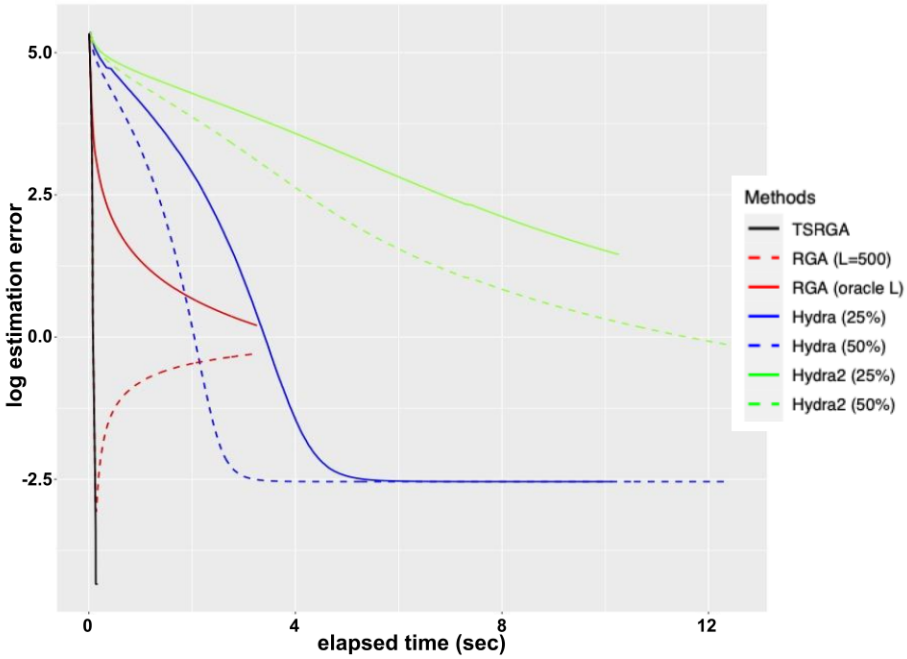
over gVAR in most months, consistent with the much smaller RMSEs reported in Table 3. For return, the results are more mixed. For example, TSRGA performed well only on January, February, July, September, and December, which implies the differences in RMSEs reported in Table 3 might not be significant.



(a) $n = 800$, $p_n = 1200$

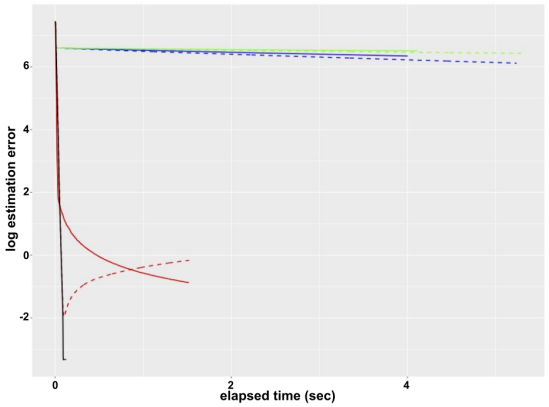


(b) $n = 1200$, $p_n = 2000$

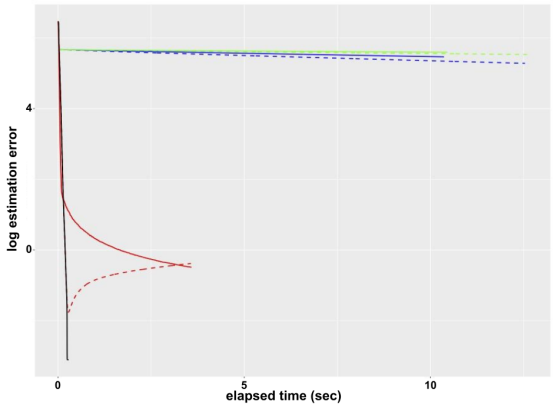


(c) $n = 1500$, $p_n = 3000$

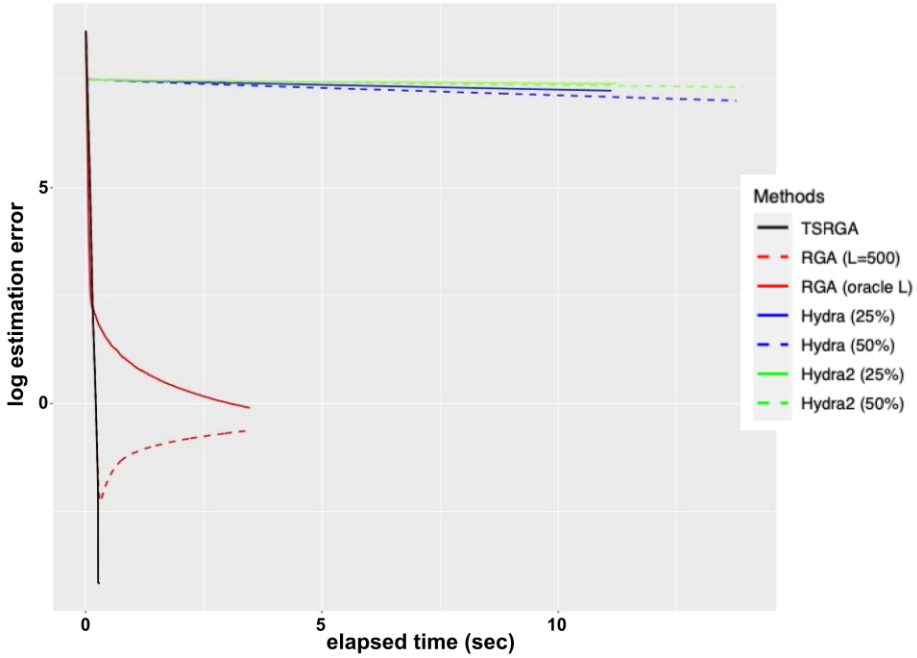
Figure 5: Logarithm of parameter estimation errors of various methods against the elapsed time under Specification 1, where n is the sample size and p_n is the dimension of predictors. The results are based on 100 simulations.



(a) $n = 800$, $p_n = 1200$

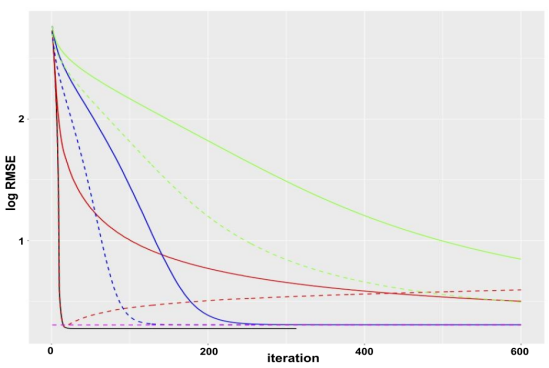


(b) $n = 1200$, $p_n = 2000$

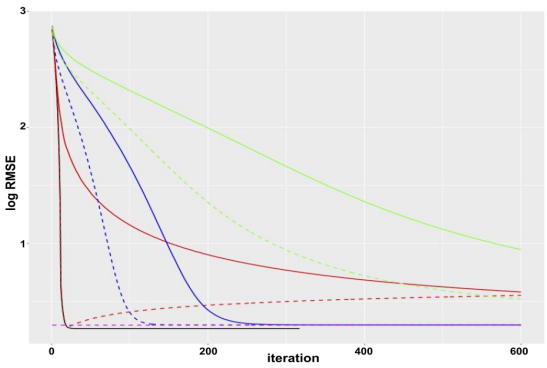


(c) $n = 1500$, $p_n = 3000$

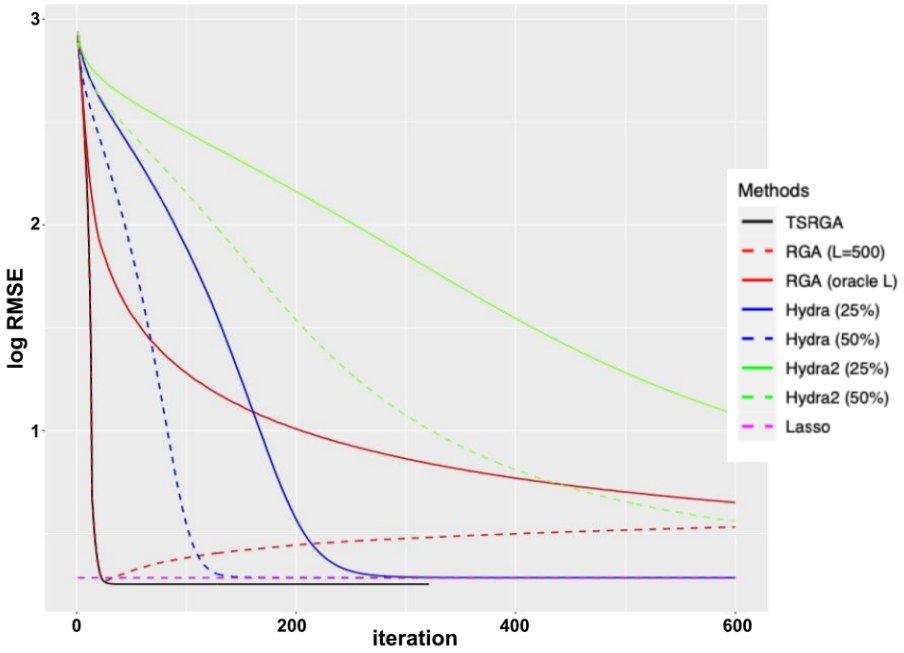
Figure 6: Logarithm of parameter estimation errors of various methods against the elapsed time under Specification 2, where n is the sample size and p_n is the dimension of predictors. The results are based on 100 simulations.



(a) $n = 800$, $p_n = 1200$

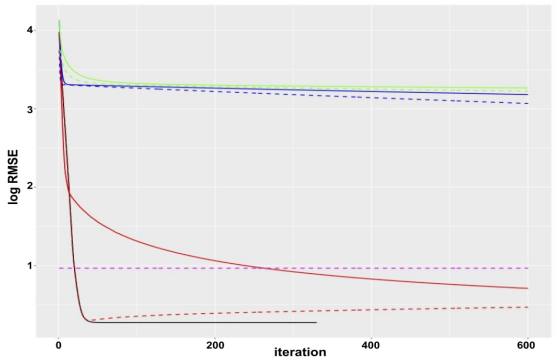


(b) $n = 1200$, $p_n = 2000$

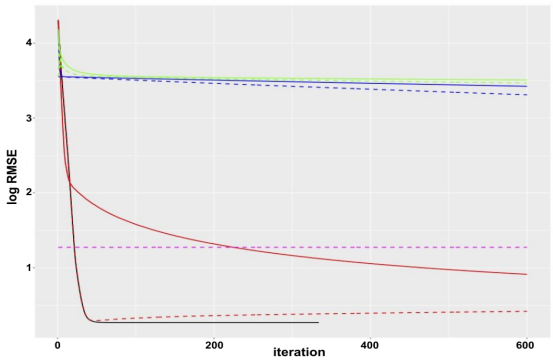


(c) $n = 1500$, $p_n = 3000$

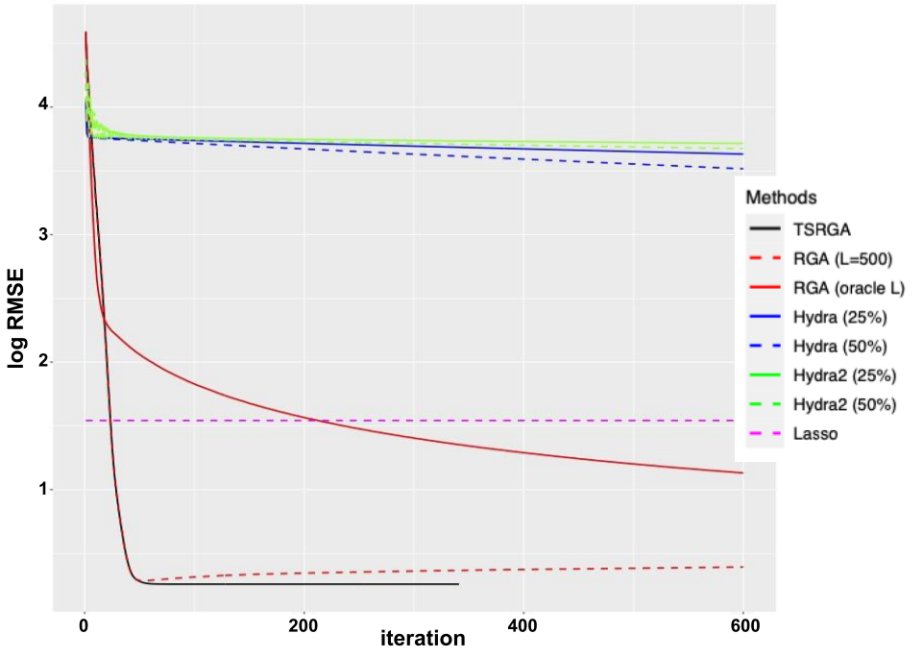
Figure 7: Logarithm of out-of-sample prediction errors of various methods under Specification 1, where n is the sample size and p_n is the dimension of predictors. The results are based on 100 simulations.



(a) $n = 800$, $p_n = 1200$



(b) $n = 1200$, $p_n = 2000$



(c) $n = 1500$, $p_n = 3000$

Figure 8: Logarithm of out-of-sample prediction errors of various methods under Specification 2, where n is the sample size and p_n is the dimension of predictors. The results are based on 100 simulations.

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
$K = 50$												
Lasso	0.50	0.97	0.00	0.72	0.46	1.00	0.00	0.22	0.00	0.02	0.09	0.00
iRRR	0.00	0.55	0.00	0.01	0.97	1.00	0.00	1.00	0.00	0.00	0.00	0.00
TSRGA	0.00	0.59	0.00	0.01	0.98	0.95	0.00	1.00	0.00	0.00	0.02	0.00
$K = 100$												
Lasso	0.26	0.72	0.00	0.61	0.28	0.99	0.00	0.33	0.00	0.01	0.01	0.00
iRRR	0.00	0.01	0.00	0.00	0.97	0.99	0.00	1.00	0.00	0.00	0.00	0.00
TSRGA	0.00	0.00	0.00	0.02	0.85	0.62	0.00	0.92	0.00	0.00	0.00	0.00
$K = 150$												
Lasso	0.07	0.42	0.00	0.47	0.45	0.99	0.00	0.61	0.00	0.00	0.01	0.00
iRRR	0.00	0.00	0.00	0.00	1.00	0.90	0.00	1.00	0.00	0.00	0.00	0.00
TSRGA	0.00	0.00	0.00	0.07	1.00	1.00	0.00	0.84	0.00	0.04	0.00	0.00
$K = 200$												
Lasso	0.14	0.22	0.00	0.45	0.24	0.99	0.00	0.52	0.00	0.00	0.00	0.00
iRRR	0.00	0.00	0.00	0.00	0.99	0.98	0.00	1.00	0.00	0.00	0.00	0.00
TSRGA	0.00	0.00	0.00	0.02	1.00	0.97	0.01	1.00	0.00	0.00	0.00	0.00

Table 5: p-values for testing against the alternative hypotheses of whether the methods yield smaller errors in predicting volatility compared to the gVAR.

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
$K = 50$												
Lasso	0.87	0.00	0.80	1.00	0.08	0.05	0.00	0.99	1.00	1.00	0.16	0.33
iRRR	1.00	0.00	0.34	1.00	0.00	0.55	0.00	1.00	0.05	0.91	0.01	0.00
TSRGA	0.00	0.00	0.09	0.92	1.00	1.00	0.00	0.84	0.00	0.98	0.72	0.00
$K = 100$												
Lasso	0.93	0.00	0.70	1.00	0.63	0.01	0.00	0.98	1.00	1.00	0.00	0.14
iRRR	1.00	0.00	0.39	1.00	0.00	0.56	0.00	1.00	0.04	0.91	0.01	0.00
TSRGA	0.00	0.00	0.09	0.92	1.00	1.00	0.00	0.84	0.00	0.98	0.72	0.00
$K = 150$												
Lasso	0.03	0.00	0.85	1.00	0.96	0.02	0.00	0.99	0.99	1.00	0.00	0.02
iRRR	1.00	0.00	0.52	1.00	0.00	0.71	0.00	1.00	0.07	0.91	0.00	0.00
TSRGA	0.00	0.00	0.09	0.92	1.00	1.00	0.00	0.84	0.00	0.98	0.72	0.00
$K = 200$												
Lasso	0.02	0.00	0.66	1.00	0.98	0.03	0.00	0.98	0.99	1.00	0.00	0.01
iRRR	1.00	0.00	0.58	1.00	0.00	0.76	0.00	1.00	0.13	0.91	0.00	0.00
TSRGA	0.00	0.00	0.09	0.92	1.00	1.00	0.00	0.84	0.00	0.98	0.72	0.00

Table 6: p-values for testing against the alternative hypotheses of whether the methods yield smaller errors in predicting return compared to the gVAR.

References

Aurélien Bellet, Yingyu Liang, Alireza Bagheri Garakani, Maria-Florina Balcan, and Fei Sha. A distributed Frank-Wolfe algorithm for communication-efficient sparse learning. In *SIAM International Conference on Data Mining*, pages 478–486, 2015.

Alexander Bertrand and Marc Moonen. Distributed adaptive node-specific signal estimation in fully connected sensor networks—part i: Sequential node updating. *IEEE Transactions on Signal Processing*, 58(10):5277–5291, 2010.

Alexander Bertrand and Marc Moonen. Distributed adaptive estimation of covariance matrix eigenvectors in wireless sensor networks with application to distributed pca. *Signal Processing*, 104:120–135, 2014.

Alexander Bertrand and Marc Moonen. Distributed canonical correlation analysis in wireless sensor networks with application to distributed blind source separation. *IEEE Transactions on Signal Processing*, 63(18):4800–4813, 2015.

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.

Leland Bybee, Bryan T Kelly, Asaf Manela, and Dacheng Xiu. Business news and business cycles. Working Paper 29344, National Bureau of Economic Research, October 2021.

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

Yuxin Chen, Yuejie Chi, Jianqing Fan, and Cong Ma. Spectral methods for data science: A statistical perspective. *Foundations and Trends®in Machine Learning*, 14(5):566–806, 2021.

Lisandro Dalcín and Yao-Lung L. Fang. mpi4py: Status update after 12 years of development. *Computing in Science and Engineering*, 23(4):47–54, 2021.

Lisandro Dalcín, Rodrigo Paz, and Mario Storti. Mpi for python. *Journal of Parallel and Distributed Computing*, 65(9):1108–1115, 2005.

Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407, 1990.

Lijun Ding, Yingjie Fei, Qiantong Xu, and Chengrun Yang. Spectral Frank-Wolfe algorithm: Strict complementarity and linear convergence. In *International Conference on Machine Learning*, volume 119, pages 2535–2544, 2020.

Lijun Ding, Jicong Fan, and Madeleine Udell. kFW: A Frank-Wolfe style algorithm with stronger subproblem oracles. *arXiv preprint arXiv:2006.16142*, 2021.

Peter K. Dunn and Gordon K. Smyth. *Generalized Linear Models With Examples in R*. Springer Texts in Statistics. Springer New York, 2018.

Jianqing Fan and Jinchi Lv. Sure independence screening for ultrahigh dimensional feature space. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(5): 849–911, 2008.

Yingying Fan, Gareth M. James, and Peter Radchenko. Functional additive regression. *The Annals of Statistics*, 43(5):2296–2325, 2015.

Olivier Fercoq, Zheng Qu, Peter Richtárik, and Martin Takáč. Fast distributed coordinate descent for non-strongly convex losses. In *IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6, 2014.

Marguerite Frank and Philip Wolfe. An algorithm for quadratic programming. *Naval Research Logistics Quarterly*, 3:95–110, 1956.

Priyank Gandhi, Tim Loughran, and Bill McDonald. Using annual report sentiment as a proxy for financial distress in u.s. banks. *Journal of Behavioral Finance*, 20(4):424–436, 2019.

Zhaoxing Gao and Ruey S. Tsay. Divide-and-conquer: A distributed hierarchical factor approach to modeling large-scale time series data. *Journal of the American Statistical Association*, 118(544):2698–2711, 2023.

Dan Garber. Revisiting Frank-Wolfe for polytopes: Strict complementarity and sparsity. In *Advances in Neural Information Processing Systems*, volume 33, pages 18883–18893, 2020.

Michael Heath Gene H. Golub and Grace Wahba. Generalized cross-validation as a method for choosing a good ridge parameter. *Technometrics*, 21(2):215–223, 1979.

Yuefeng Han, Ruey S. Tsay, and Wei Biao Wu. High dimensional generalized linear models for temporal dependent data. *Bernoulli*, 29(1):105–131, 2023.

Kathleen Weiss Hanley and Gerard Hoberg. Dynamic interpretation of emerging risks in the financial sector. *The Review of Financial Studies*, 32(12):4543–4603, 2019.

Christina Heinze, Brian McWilliams, and Nicolai Meinshausen. Dual-loco: Distributing statistical estimation using random projections. In *International Conference on Artificial Intelligence and Statistics*, volume 51, pages 875–883, 2016.

Yaochen Hu, Di Niu, Jianming Yang, and Shengping Zhou. Fdml: A collaborative machine learning framework for distributed features. In *ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2232–2240, 2019.

Ching-Kang Ing. Model selection for high-dimensional linear regression with dependent observations. *The Annals of Statistics*, 48(4):1959–1980, 2020.

Ching-Kang Ing and Tze Leung Lai. A stepwise regression method and consistent model selection for high-dimensional sparse linear models. *Statistica Sinica*, 21(4):1473–1513, 2011.

Martin Jaggi. Revisiting Frank-Wolfe: Projection-free sparse convex optimization. *International Conference on Machine Learning*, 28(1):427–435, 2013.

Martin Jaggi and Simon Lacoste-Julien. On the global linear convergence of Frank-Wolfe optimization variants. *Advances in Neural Information Processing Systems*, 28, 2015.

Narasimhan Jegadeesh and Di Wu. Word power: A new approach for content analysis. *Journal of Financial Economics*, 110(3):712–729, 2013.

Shimon Kogan, Dimitry Levin, Bryan R Routledge, Jacob S Sagi, and Noah A Smith. Predicting risk from financial reports with regression. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 272–280, 2009.

Qi Lei, Jiacheng Zhuo, Constantine Caramanis, Inderjit S Dhillon, and Alexandros G Dimakis. Primal-dual block generalized frank-wolfe. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Gen Li, Xiaokang Liu, and Kun Chen. Integrative multi-view regression: Bridging group-sparse and low-rank models. *Biometrics*, 75(2):593–602, 2019.

Lefteris Loukas, Manos Fergadiotis, Ion Androutsopoulos, and Prodromos Malakasiotis. Edgar-corpus: Billions of tokens make the world go round. In *Proceedings of the Third Workshop on Economics and Natural Language Processing*, pages 13–18, Punta Cana, Dominican Republic, 2021.

Karim Lounici, Massimiliano Pontil, Sara Van De Geer, and Alexandre B Tsybakov. Oracle inequalities and optimal inference under group sparsity. *The Annals of Statistics*, 39(4): 2164–2204, 2011.

Mert Pilanci and Martin J Wainwright. Iterative hessian sketch: Fast and accurate solution approximation for constrained least-squares. *Journal of Machine Learning Research*, 17 (1):1842–1879, 2016.

Gregory C. Reinsel, Raja P. Velu, and Kun Chen. *Multivariate Reduced-Rank Regression*. Springer New York, NY, 2022.

Peter Richtárik and Martin Takáč. Distributed coordinate descent method for learning with big data. *Journal of Machine Learning Research*, 17(75):1–25, 2016.

Axel Ruhe. Perturbation bounds for means of eigenvalues and invariant subspaces. *BIT Numerical Mathematics*, 10:343–354, 1970.

Gerard Salton and Christopher Buckley. Term-weighting approaches in automatic text retrieval. *Information Processing and Management*, 24(5):513–523, 1988.

Vladimir Temlyakov. Weak greedy algorithms. *Advances in Computational Mathematics*, 12(2):213–227, 2000.

Vladimir Temlyakov. Greedy approximation in convex optimization. *Constructive Approximation*, 41(2):269–296, 2015.

Roman Vershynin. *High-Dimensional Probability: An Introduction with Applications in Data Science*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2018.

Jialei Wang, Jason D. Lee, Mehrdad Mahdavi, Mladen Kolar, and Nathan Srebro. Sketching meets random projection in the dual: A provable recovery algorithm for big and high-dimensional data. *Electronic Journal of Statistics*, 11(2):4896–4944, 2017.

Xiangyu Wang, David Dunson, and Chenlei Leng. Decorrelated feature space partitioning for distributed sparse regression. In *International Conference on Neural Information Processing Systems*, pages 802–810, 2016.

Per-Åke Wedin. Perturbation bounds in connection with singular value decomposition. *BIT Numerical Mathematics*, 12(1):99–111, 1972.

Jiyan Yang, Michael W. Mahoney, Michael A. Saunders, and Yuekai Sun. Feature-distributed sparse regression: A screen-and-clean approach. In *International Conference on Neural Information Processing Systems*, pages 2711–2719, 2016.

Hsiang-Yuan Yeh, Yu-Ching Yeh, and Da-Bai Shen. Word vector models approach to text regression of financial risk prediction. *Symmetry*, 12(1), 2020.

Wenjie Zheng, Aurélien Bellet, and Patrick Gallinari. A distributed Frank-Wolfe framework for learning low-rank matrices with the trace norm. *Machine Learning*, 107(8):1457–1475, 2018.

Jiacheng Zhuo, Qi Lei, Alex Dimakis, and Constantine Caramanis. Communication-efficient asynchronous stochastic Frank-Wolfe over nuclear-norm balls. In *International Conference on Artificial Intelligence and Statistics*, pages 1464–1474, 2020.