

Efficient Multi-Round Distributed PCA Without Eigengap Assumptions: Enlarged Eigenspace Estimation with Communication Reduction

Chris Junchi Li[◊]

Department of Electrical Engineering and Computer Sciences[◊]
University of California, Berkeley

October 4, 2024

Abstract

We present a novel distributed algorithm for principal component analysis (PCA) that efficiently handles large-scale data in a distributed environment. By removing traditional assumptions on eigengap constraints and number of machines, our approach leverages shift-and-invert preconditioning to achieve fast convergence rates. The proposed multi-round method computes the top- L eigenspace without eigengap requirements, extending previous divide-and-conquer strategies. This allows the estimator to handle complex distributed setups, such as sensor networks, with reduced communication costs while maintaining optimal statistical convergence. Our theoretical analysis demonstrates that the method remains robust in scenarios where the number of machines is large, and no eigengap condition is assumed. Numerical experiments confirm that our algorithm outperforms existing approaches and provides near-oracle performance even in challenging distributed environments.

Keywords: Distributed PCA, Shift-and-Invert Preconditioning, Eigenspace Estimation, Large-Scale Data, Communication Efficiency

1 Introduction

The rapid expansion of modern datasets, fueled by technological advancements, has introduced significant challenges for traditional statistical methods, particularly in distributed systems handling large-scale data. Principal component analysis (PCA), a widely used method in machine learning, faces computational and memory constraints when data is scattered across multiple machines. In such environments, transferring and processing all the data centrally is often impractical due to high communication costs and the limitations of central systems to handle massive datasets.

Existing approaches, like divide-and-conquer (DC), address these challenges by computing local eigenvectors on individual machines and merging them centrally. However, these methods often rely on restrictive assumptions, such as the presence of an eigengap or limitations on the number of machines, which can hinder optimal convergence rates in large distributed systems.

In this paper, we propose a new distributed algorithm for principal eigenspace estimation that overcomes these limitations. Our method combines shift-and-invert preconditioning with distributed optimization, eliminating the need for an eigengap assumption and enabling the algorithm to scale efficiently without constraints on the number of machines, even in large distributed networks.

Backgrounds Distributed statistical approaches have drawn a lot of attentions these days and methods are developed for various statistics problems, such as sparse regression (see, e.g., [LLST17]),

likelihood-based inference (see, e.g., [BFL⁺18, JLY19]), kernel ridge regression [ZDW15], semi-parametric partial linear models [ZCL16], quantile regression (see, e.g., [VCC19, CLZ⁺19, CLMY20]), linear support vector machine [WYCL19]), Newton-type estimator [CLZ21], and M -estimators with cubic rate [SLS18, BDS⁺19]. All these works are seeking for distributed statistical methods that are able to handle massive computation tasks efficiently for large-scale data and achieve the same convergence rate as those classical methods as well.

In a typical distributed environment, each machine has access to a different subset of samples of the whole data set. The communication and computation follow from a hierarchical master-slave-type architecture, where a central machine acts as a fusion node. Computation tasks for local machines and the central machine are different. After local machines finish their computation, the local results will be transferred to the master machine, where they will be merged together and the fused result will be transferred back to all local machines for the next step.

In this paper, we study the problem of principal component analysis (PCA) in a distributed environment. PCA [Pea01, Hot33] is one of the most important and fundamental tools in statistical machine learning. For random vectors $\mathbf{a}_1, \dots, \mathbf{a}_n$ in \mathbb{R}^d with mean zero and covariance matrix Σ , its empirical covariance matrix is $\widehat{\Sigma} = \frac{1}{n} \sum_{i=1}^n \mathbf{a}_i \mathbf{a}_i^\top$. The L -PCA ($L \leq d$) finds a L -dimension subspace projection that preserves the most variation in the data set, which is equivalent to the following optimization problem:

$$\max_{\mathbf{U} \in \mathbb{R}^{d \times L}: \mathbf{U}^\top \mathbf{U} = \mathbf{I}_L} \|\widehat{\Sigma} \mathbf{U}\|_F \quad (1)$$

where $\|\cdot\|_F$ denotes the matrix Frobenius norm and \mathbf{I}_L is the $L \times L$ identity matrix. In other words, $\mathbf{U} \in \mathbb{R}^{d \times L}$ is the top- L -dim eigenspace of $\widehat{\Sigma}$. PCA has been widely used in many aspects of statistical machine learning, e.g., principal component regression [Jef67, Jol82], single index model [Li92], representation learning [BCV13].

Under distributed regime, [FWWZ19] proposed a novel one-shot type of algorithm which is often called divide-and-conquer (DC) method. In [FWWZ19], DC method first computes local covariance matrices $\widehat{\Sigma}_i$ on each machine $k = 1, \dots, K$. Eigenspaces $\widehat{\mathbf{U}}_k, k = 1, \dots, K$ are then computed locally using the traditional PCA algorithm and transmitted to the central machine. Central machine combines local eigenspaces $\widehat{\mathbf{U}}_k$ into an aggregated covariance estimator, $\widetilde{\Sigma} = \frac{1}{K} \sum_{k=1}^K \widehat{\mathbf{U}}_k \widehat{\mathbf{U}}_k^\top$. The final estimator is obtained as the top- L -dim eigenspace of $\widetilde{\Sigma}$. DC method is easy to implement and requires only $\mathcal{O}(dL)$ communications for each local machine, where d denotes the data dimension, n the total sample size, and m the sample size on each local machine. Let us denote the condition number of the population covariance matrix Σ by ρ , i.e., $\rho = \lambda_1 / (\lambda_L - \lambda_{L+1})$, and the effective rank of Σ by $r = \text{Tr}(\Sigma) / \lambda_1$. For asymmetric innovation distributions, [FWWZ19] showed that when the number of machines is not very large (no greater than $\mathcal{O}(m / (\rho^2 r))$), DC method enjoys a optimal statistical convergence rate of order $\mathcal{O}(\rho \sqrt{Lr/n})$. However, when the number of machines becomes larger, DC method only achieves a slow convergence rate of $\mathcal{O}(\rho \sqrt{Lr/n} + \rho^2 \sqrt{Lr/m})$. This feature may not be desirable in distributed settings. For example, in a sensor network with a vast number of sensors, the number of machines may exceed the constraint set for the optimal rate. The precise definition of asymmetric innovation above is given in Section 4.2 of [FWWZ19]. Roughly speaking, a random variable $\mathbf{a} \in \mathbb{R}^d$ is distributed under asymmetric innovation if flipping the sign of one component of \mathbf{a} changes its distribution.

One question naturally arises from the analysis of DC method, *can we possibly relax the restriction on the number of machines?* Motivated by this question, our paper presents a multi-round distributed algorithm for top- L -dim eigenspace estimation.

The main results of our method is two-fold. First, as compared to DC method in [FWWZ19], we completely remove the assumption on the number of machines. Our method leverages shift-and-invert preconditioning (a.k.a., Rayleigh quotient iteration) from numerical analysis [VLG12] together with quadratic programming and achieves a fast convergence rate. Moreover, most previous convergence analysis of eigenspace estimation relies on the assumption of an explicit eigengap between the L -th and the $(L + 1)$ -th population eigenvalues λ_L and λ_{L+1} , i.e., $\lambda_L - \lambda_{L+1} > 0$, or other specific eigen-structures of Σ . The second contribution of our paper is that we propose an enlarged eigenspace estimator that does not require any eigengap assumption.

In particular, let \mathbf{U}_L denote the top- L -dim eigenspace of the population covariance matrix Σ , and $\widehat{\mathbf{U}}_L$ the top- L -dim eigenspace of the empirical covariance $\widehat{\Sigma}$. Estimation consistency of $\widehat{\mathbf{U}}_L$ is guaranteed by the (variant of) Davis-Kahan Theorem [DK70, YWS14]: there exists an orthogonal matrix $\mathbf{Q} \in \mathbb{R}^{L \times L}$, such that

$$\left\| \mathbf{U}_L - \widehat{\mathbf{U}}_L \mathbf{Q} \right\|_2 \leq \frac{\sqrt{2} \left\| \widehat{\Sigma} - \Sigma \right\|_2}{\min(|\widehat{\lambda}_{L-1} - \lambda_L|, |\widehat{\lambda}_{L+1} - \lambda_L|)} \quad (2)$$

where $\|\cdot\|_2$ denotes the matrix spectrum norm. Since the empirical eigenvalue $\widehat{\lambda}_l$ is expected to be concentrated around its population counterpart λ_l for all $l \in [d]$, the consistency of $\widehat{\mathbf{U}}_L$ relies on an eigengap condition requiring $\min(\lambda_{L-1} - \lambda_L, \lambda_L - \lambda_{L+1})$ to be strictly away from zero. Unfortunately, without such an eigenvalue gap condition, the top- L -dim subspace \mathbf{U}_L is not statistically identifiable and estimation error from $\widehat{\mathbf{U}}_L$ can be arbitrarily large (cf. a counter-example provided in [YWS14]). Fortunately, in many statistical applications of PCA such as the principal component regression (see Example 1 below), it suffices to retrieve the variation captured by the top eigenspace rather than exactly recover the top eigenspace in order to achieve a small in-sample prediction risk. To address the challenge of no explicit eigengap, we choose a different perspective. In particular, we consider an *enlarged estimator* $\mathbf{V}_{>(1-\delta)\widehat{\lambda}_L}$ (see Equation (3)), where δ is a pre-specified constant to quantify the amount of enlargement.

$$\underbrace{\mathbf{u}_1, \dots, \mathbf{u}_L, \mathbf{u}_{L+1}, \dots, \mathbf{u}_S, \mathbf{u}_{S+1}, \dots, \mathbf{u}_d}_{\mathbf{U}_L} \quad (3)$$

$$\underbrace{\mathbf{v}_1, \dots, \mathbf{v}_L, \mathbf{v}_{L+1}, \dots, \mathbf{v}_S, \mathbf{v}_{S+1}, \dots, \mathbf{v}_d}_{\mathbf{V}_{>(1-\delta)\widehat{\lambda}_L}} \quad \underbrace{\mathbf{v}_{\leq(1-\delta)\widehat{\lambda}_L}}$$

Roughly speaking, we prove that our distributed estimator $\mathbf{V}_{>(1-\delta)\widehat{\lambda}_L}$ satisfies inequality (2) with the following property: the angle between the target \mathbf{U}_L and the complement of our estimator $\mathbf{V}_{\leq(1-\delta)\widehat{\lambda}_L}$ is sufficiently small (please see Theorem 3 for more details). Such a property shows that the enlarged estimator $\mathbf{V}_{>(1-\delta)\lambda_L}$ almost cover the \mathbf{U}_L even without an eigengap condition.

Our method is motivated by the shift-and-invert preconditioning. The idea of solving PCA via shift-and-invert preconditioning has long history in numerical analysis [VLG12]. It is an iterative method that sequentially solves linear system to obtain increasingly accurate eigenvector estimates. Its connection with convex optimization has been studied in the past decade. In a single-machine setting, [GHJ⁺16, AZL16] formulate each round of shift-and-invert preconditioning as a quadratic optimization problem and it can be solved with first-order deterministic (accelerated) gradient method like Nesterov accelerated method. [GH15, Sha16, Xu18] also relate the same convex optimization problem with variance-reduction stochastic technique (SVRG, see, e.g., [JZ13]). Furthermore, in distributed settings, [GSS17] perform a multi-round algorithm but they

only consider the estimation task of the first eigenvector. This paper proposes a general distributed algorithm that estimates the top- L -dim eigenspace without a restriction on the eigengap.

The proposed algorithm can facilitate many fundamental applications based on PCA in distributed environment. In particular, we illustrate two important applications, namely principal component regression (see Appendix B.1) and single index model (see Appendix B.2).

Example 1: principal component regression

Introduced by [Jef67, Jol82], principal component regression (PCR) is a regression analysis technique based on PCA. Typically, PCR assumes a linear model $\mathbf{y} = \mathbf{A}\beta^* + \epsilon$ with the further assumption that coefficient β^* lies in the low-rank eigenspace of data covariance matrix. Therefore, PCA can be performed to obtain the principal components $\widehat{\mathbf{U}}_L$ of the observed covariance matrix $\widehat{\Sigma} = \frac{1}{n}\mathbf{A}^\top\mathbf{A}$ and the data matrix \mathbf{A} is then projected on $\widehat{\mathbf{U}}_L$. The estimator $\widehat{\beta}$ of β^* is then obtained by regress \mathbf{y} on this projected data matrix $\mathbf{A}\widehat{\mathbf{U}}_L$. Many previous work has analyze the statistical property of PCR, see [FF93, BHPT06]. Under a distributed environment, our distributed PCA algorithm can replace the traditional PCA algorithm in the above procedure and lead to a distributed algorithm for PCR. As we will show in Appendix B.1, this distributed estimator achieves a similar error as in the single-machine setting.

Example 2: single index model

Single index model [Li92] considers a semi-parametric regression model $y = f(\langle\beta^*, \mathbf{a}\rangle) + \epsilon$. Under some mild condition on the link function $f(\cdot)$, we would like to make estimation on the coefficient β^* using observed data $\{\mathbf{a}_i, y_i\}_{i=1}^n$ without knowing $f(\cdot)$. Some previous methods include semi-parametric maximum likelihood estimator [Hor09] and gradient-based estimator [HJS01]. Moreover, many works propose to use Stein's identity [Ste81, JSA14] to estimate β^* (see, e.g., [Li92, YBL17] and references therein). Specifically, under Gaussian innovation where \mathbf{a} is standard multi-variate normal random vector, the estimator $\widehat{\beta}$ can be calculated from the top eigenvector of $\frac{1}{n}\sum_{i=1}^n y_i \cdot (\mathbf{a}_i \mathbf{a}_i^\top - \mathbf{I}_d)$. This method can be naturally extended to a distributed manner with a distributed eigen-decomposition of $\frac{1}{n}\sum_{i=1}^n y_i \cdot (\mathbf{a}_i \mathbf{a}_i^\top - \mathbf{I}_d)$.

1.1 Notations

We first introduce the notations related to our work. We write vectors in \mathbb{R}^d in boldface lower-case letters (e.g., \mathbf{a}), matrices in boldface upper-case letters (e.g., \mathbf{A}), and scalars are written in lightface letters (e.g., t). Let $\|\cdot\|$ denote vector norm (e.g., $\|\cdot\|_2$ is standard Euclidean norm for vectors). Matrix norm is written as $\|\cdot\|$. For a matrix $\mathbf{A} \in \mathbb{R}^{n \times d}$, $\|\mathbf{A}\|_2$ and $\|\mathbf{A}\|_F$ represent the spectral norm and Frobenius norm respectively. Furthermore, $\mathbf{0}$ represents zero vector with corresponding dimension and identity matrix with dimension $d \times d$ is shortened as \mathbf{I}_d . We use $\mathbf{e}_1, \dots, \mathbf{e}_d$ to denote the standard unit vectors in \mathbb{R}^d , i.e., $\mathbf{e}_i = [0, \dots, 0, 1, 0, \dots, 0]$ where only the i -th element of \mathbf{e}_i is 1.

We use \mathcal{O}_p to describe a high probability bound with constant term omitted. We also use $\tilde{\mathcal{O}}_p$ to further omit the logarithm factors.

We adopt the standard definition of sub-Gaussian random vectors (see, e.g., [Ver12, RH15]) that a random vector $\mathbf{a} \in \mathbb{R}^d$ is said to be a d -dimensional sub-Gaussian with variance proxy σ if

$\mathbb{E}[\mathbf{a}] = \mathbf{0}$ and for any unit vector \mathbf{u} ,

$$\mathbb{E}[\exp(s\mathbf{a}^\top \mathbf{u})] \leq \exp\left(\frac{\sigma^2 s^2}{2}\right), \forall s \in \mathbb{R}$$

1.2 Contributions

Our contributions can be summarized as follows:

- We propose a novel multi-round algorithm for distributed PCA that leverages shift-and-invert preconditioning to handle large-scale datasets without eigengap assumptions.
- The algorithm achieves optimal statistical convergence rates and significantly reduces communication costs compared to existing DC methods.
- We provide extensive theoretical analysis that establishes fast convergence rates and robustness in complex distributed environments.
- Numerical experiments demonstrate the superior performance of our method, achieving near-oracle results across various distributed settings.

1.3 Organization of the Paper

The remainder of this paper is structured as follows. In Section 2, we introduce the problem setup for distributed PCA and present our algorithms. Section 3 develops the theoretical convergence analysis of our proposed estimator. Section 4 provides extensive numerical experiments demonstrating the algorithm’s performance compared to existing methods. Additional technical proofs and supplementary experimental results are provided in the supplementary material. Finally, in the Appendix, we discuss two application scenarios—principal component regression and the single index model—where convergence analysis is explored for both single-machine and distributed settings.

2 Problem Setups

In the following section, we collect the setups for our distributed PCA and present the algorithms.

Assume that there are n *i.i.d.* zero mean vectors \mathbf{a}_i sampling from some distribution \mathcal{D} in \mathbb{R}^d . Let $\mathbf{A} = [\mathbf{a}_1, \dots, \mathbf{a}_n]^\top \in \mathbb{R}^{n \times d}$ be the data matrix. Let Σ be the population covariance matrix $\Sigma = \mathbb{E}_{\mathbf{a} \sim \mathcal{D}}[\mathbf{a}\mathbf{a}^\top]$ with the eigenvalues $\lambda_1(\Sigma) \geq \lambda_2(\Sigma) \geq \dots \lambda_d(\Sigma) \geq 0$ and the associated eigenvectors are $\mathbf{U} = [\mathbf{u}_1, \dots, \mathbf{u}_d] \in \mathbb{R}^{d \times d}$.

In the distributed principal component analysis, for a given number L , $1 \leq L \leq d$, we are interested in estimating the eigenspace spanned by $\mathbf{U}_L := \{\mathbf{u}_1, \dots, \mathbf{u}_L\}$ in a distributed environment. We assume n samples are split uniformly at random on K machines, where each machine contains m samples, i.e., $n = mK$. We note that since our algorithm aggregates gradient information across machines, it can handle the unbalanced data case without any modification. We choose to present the balanced data case only for the ease of presentation (see Remark for more details). The data matrix on each machine k is denoted by $\mathbf{A}_k \in \mathbb{R}^{m \times d}$ for $k \in [K]$.

Let us first discuss a special case (illustrated in Algorithm 1), where we estimate the top eigenvector, i.e., $L = 1$. The basic idea of our Algorithm 1 is as follows.

Algorithm 1 Distributed Top Eigenvector (Distri-Eigen)

Input: Data matrix \mathbf{A}_k on each machine $k = 1, \dots, K$. The initial top eigenvalue estimator $\bar{\lambda}_1$ and eigenvector estimator $\mathbf{w}^{(0)}$. The number of outer iterations T and the number of inner iterations T' .

- 1: Distribute $\bar{\lambda}_1$ to each local machine and each local machine computes $\mathbf{H}_k = \bar{\lambda}_1 \mathbf{I} - \mathbf{A}_k^\top \mathbf{A}_k / m$.
 - 2: **for** $t = 0, 1, \dots, (T - 1)$ **do**
 - 3: Distribute $\mathbf{w}^{(t)}$ to each local machine and each local machine sets $\mathbf{w}_0^{(t+1)} = \mathbf{w}^{(t)}$
 - 4: **for** $j = 0, 1, \dots, (T' - 1)$ **do**
 - 5: **for** each local machine $k = 1, \dots, K$ **do**
 - 6: Compute the local gradient information $\mathbf{g}_k = \mathbf{H}_k \mathbf{w}_j^{(t+1)} - \mathbf{w}^{(t)}$
 - 7: Transmit the local gradient information \mathbf{g}_k to the central machine.
 - 8: **end for**
 - 9: Calculate the global gradient information $\mathbf{g} = \frac{1}{K} \sum_{k=1}^K \mathbf{g}_k$.
 - 10: Perform the approximate Newton's step: $\mathbf{w}_{j+1}^{(t+1)} = \mathbf{w}_j^{(t+1)} - \mathbf{H}_1^{-1} \mathbf{g}$.
 - 11: **end for**
 - 12: The central machine updates $\mathbf{w}^{(t+1)} = \frac{\mathbf{w}_{T'}^{(t+1)}}{\|\mathbf{w}_{T'}^{(t+1)}\|_2}$.
 - 13: **end for**
 - 14: **Output:** $\mathbf{w}^{(T)}$.
-

Let $\mathbf{w}^{(0)}$ be the initial estimator of the top eigenvector and $\bar{\lambda}_1$ a crude estimator of an upper bound of the top eigenvalue. Here we propose to compute $\mathbf{w}^{(0)}$ and $\bar{\lambda}_1$ only using the data from the first machine, and thus there does not incur any communication cost. For example, $\bar{\lambda}_1$ can be computed with $\bar{\lambda}_1 = \lambda_1(\mathbf{A}_1^\top \mathbf{A}_1 / m) + 3\eta/2$, where $\lambda_1(\mathbf{A}_1^\top \mathbf{A}_1 / m)$ is the top eigenvalue for the empirical covariance matrix on the first machine and η is a special constant defined later in Equation (11). The $\mathbf{w}^{(0)}$ can be simply computed via eigenvalue decomposition of $\mathbf{A}_1^\top \mathbf{A}_1 / m$. We note the that Algorithm 1 is almost tuning free. The only parameter in constructing $\bar{\lambda}_1$ is η . According to our theory, we could set $\eta = c_0 \sqrt{d/m}$ for some sufficiently large c_0 and the result is not sensitive to c_0 . There are other tuning-free ways to obtain a crude top-eigenvalue estimator $\bar{\lambda}_1$ only using the sample on the first machine (e.g., the adaptive Algorithm 1 in [GHJ⁺16] without tuning parameters).

Given $\mathbf{w}^{(0)}$ and $\bar{\lambda}_1$, we perform the *shift-and-invert preconditioning* iteration in a distributed manner. In particular, for each iteration $t = 0, 1, \dots$,

$$\tilde{\mathbf{w}}^{(t+1)} = \left(\bar{\lambda}_1 \mathbf{I} - \frac{1}{n} \mathbf{A}^\top \mathbf{A} \right)^{-1} \mathbf{w}^{(t)}, \quad \mathbf{w}^{(t+1)} = \frac{\tilde{\mathbf{w}}^{(t+1)}}{\|\tilde{\mathbf{w}}^{(t+1)}\|_2} \quad (4)$$

Therefore, the non-convex eigenvector estimation problem (1) is reduced to solving a sequence of linear system. The key challenge is how to implement $(\bar{\lambda}_1 \mathbf{I} - \mathbf{A}^\top \mathbf{A} / n)^{-1}$ in a distributed setup.

To address this challenge, we formulate (4) into a quadratic optimization problem. In particular, the update $\tilde{\mathbf{w}}^{(t+1)} = (\bar{\lambda}_1 \mathbf{I} - \mathbf{A}^\top \mathbf{A} / n)^{-1} \mathbf{w}^{(t)}$ is equivalent to the following problem,

$$\begin{aligned} \tilde{\mathbf{w}}^{(t+1)} &= \arg \min_{\mathbf{w}} \left[Q(\mathbf{w}) := \frac{1}{2} \mathbf{w}^\top \mathbf{H} \mathbf{w} - \mathbf{w}^\top \mathbf{w}^{(t)} \right], \\ \mathbf{H} &\triangleq \bar{\lambda}_1 \mathbf{I} - \frac{1}{n} \mathbf{A}^\top \mathbf{A} \end{aligned} \quad (5)$$

Algorithm 2 Distributed Top- L -dim principal subspace

Input: The data matrix \mathbf{A}_k on each machine $k = 1, \dots, K$. The number of top-eigenvectors L .

- 1: Initialize $\mathbf{V}_0 = []$, $\mathbf{A}_{k,0} = \mathbf{A}_k$
 - 2: **for** $l = 1, \dots, L$ **do**
 - 3: Compute the initial l -th eigenvalue estimator $\bar{\lambda}_l$ and eigenvector estimator $\mathbf{w}_l^{(0)}$.
 - 4: Call Algorithm 1 with $\{\mathbf{A}_{k,l-1}\}_{k=1}^K$ on each local machine to obtain \mathbf{w}_l on the central machine.
 - 5: Project \mathbf{w}_l to \mathbf{V}_{l-1}^\perp by computing $\mathbf{v}_l = \frac{(\mathbf{I} - \mathbf{V}_{l-1} \mathbf{V}_{l-1}^\top) \mathbf{w}_l}{\|(\mathbf{I} - \mathbf{V}_{l-1} \mathbf{V}_{l-1}^\top) \mathbf{w}_l\|_2}$ as the estimated l -th eigenvector
 - 6: Update $\mathbf{V}_l = [\mathbf{V}_{l-1}, \mathbf{v}_l]$
 - 7: Transmit \mathbf{v}_l to each local machine.
 - 8: **for** each local machine $k = 1, \dots, K$ **do**
 - 9: Update the data matrix $\mathbf{A}_{k,l} = \mathbf{A}_{k,l-1} (\mathbf{I} - \mathbf{v}_l \mathbf{v}_l^\top)$
 - 10: **end for**
 - 11: **end for**
 - 12: **Output:** \mathbf{V}_L .
-

To solve this quadratic programming, the standard Newton's approach computes a sequence for $j = 0, \dots$, with a starting point $\mathbf{w}_0^{(t+1)} = \mathbf{w}^{(t)}$:

$$\mathbf{w}_{j+1}^{(t+1)} = \mathbf{w}_j^{(t+1)} - \left(\nabla^2 Q(\mathbf{w}_j^{(t+1)}) \right)^{-1} \left[\nabla Q(\mathbf{w}_j^{(t+1)}) \right] \quad (6)$$

where the Hessian matrix $\nabla^2 Q(\mathbf{w}_j^{(t+1)})$ is indeed \mathbf{H} . If we define, for each machine $k \in [K]$,

$$\begin{aligned} \mathbf{H}_k &= \bar{\lambda}_1 \mathbf{I} - \frac{1}{m} \mathbf{A}_k^\top \mathbf{A}_k, \\ Q_k(\mathbf{w}) &= \frac{1}{2} \mathbf{w}^\top \mathbf{H}_k \mathbf{w} - \mathbf{w}^\top \mathbf{w}^{(t)} \end{aligned} \quad (7)$$

It is easy to see that $\mathbf{H} = \sum_{k=1}^K \mathbf{H}_k / K$ and $Q(\mathbf{w}) = \sum_{k=1}^K Q_k(\mathbf{w}) / K$. Therefore, in the Newton's update (6), computing the full Hessian matrix $\nabla^2 Q(\mathbf{w}_j^{(t+1)})$ requires each machine to communicate a $d \times d$ local Hessian matrix \mathbf{H}_k to the central machine. This procedure incurs a lot of communication cost. Moreover, taking the inverse of the whole sample Hessian matrix \mathbf{H} almost solves the original linear system (4). To address this challenge, we adopt the idea from [SSZ14, JLY19, FGW19]. In particular, we approximate the Newton's iterates by only using the Hessian information on the *first machine*, which significantly reduces the communication cost. This approximated Newton's update can be written as,

$$\begin{aligned} \mathbf{w}_{j+1}^{(t+1)} &= \mathbf{w}_j^{(t+1)} - \left(\nabla^2 Q_1(\mathbf{w}_j^{(t+1)}) \right)^{-1} \left[\nabla Q(\mathbf{w}_j^{(t+1)}) \right] \\ &= \mathbf{w}_j^{(t+1)} - \mathbf{H}_1^{-1} \left[\frac{1}{K} \sum_{k=1}^K (\mathbf{H}_k \mathbf{w}_j^{(t+1)} - \mathbf{w}^{(t)}) \right] \end{aligned} \quad (8)$$

where \mathbf{H}_1 is the Hessian matrix of the first machine. This procedure can be computed easily in a distributed manner, i.e., each machine computes local gradient $\mathbf{g}_k = \mathbf{H}_k \mathbf{w}_j^{(t+1)} - \mathbf{w}^{(t)}$, and these gradient vectors are communicated to the central machine for a final update $\mathbf{g} = \sum_{k=1}^K \mathbf{g}_k / K$. Therefore, in each inner iteration, the communication cost for each local nodes is only $\mathcal{O}(d)$. See Algorithm 1 for a complete description.

Remark. In this remark, we explain why we choose the Newton approach in the inner loop (8), instead of the quasi-Newton method (the Broyden–Fletcher–Goldfarb–Shanno (BFGS) method) or other gradient methods with line search (e.g., Barzilai–Borwein gradient method in [WY13]). Due to the special structure of the PCA problem in our quadratic programming (5), the Hessian matrix is fixed and will not change over iterations. In other words, as shown in Algorithm 1, we compute each local Hessian \mathbf{H}_k for $k = 1, \dots, m$ (see Equation (7)), and the inverse of the local Hessian on the first machine \mathbf{H}_1 only once. Therefore, the distributed Newton method is computationally more efficient for the PCA problem. In comparison, BFGS is often used when the inverse of Hessian matrix \mathbf{H} is hard to compute and changes over iterations, which is not the scenario of our PCA problem. Moreover, as we will show later in Section 3, the Newton method has already achieved a linear convergence rate in the inner loop (see Lemma 2), BFGS cannot be faster than that. In fact, although BFGS will eventually achieve a linear convergence rate, it can be quite slow at the very beginning with a crude Hessian inverse estimation.

Remark. In this remark, we compare the computational and communication costs between our method and the DC approach. Notice that the communication cost of our Algorithm 1 from each local machine is $\mathcal{O}(TT'd)$, where TT' is the total number of iterations. By our theoretical results in Section 3 (see Corollary 1), for a targeting error rate ε , we only require T and T' and to be an logarithmic order of $1/\varepsilon$ (i.e., $TT' = \mathcal{O}(\log^2(1/\varepsilon))$). Therefore, the total number of iterations is quite small. While it is more than $\mathcal{O}(d)$ communication cost of the DC approach, it is still considered as a communication efficient protocol. In distributed learning literature (e.g., [JLY19]), a communication-efficient algorithm usually refers to an algorithm that only transmits an $\mathcal{O}(d)$ vector (instead of $\mathcal{O}(d^2)$ Hessian matrices) at each iteration.

When the full data of n samples can be stored in the memory, the oracle PCA method incurs a computation cost (i.e., runtime) of $\mathcal{O}(nd^2 + d^3)$, where nd^2 is for the computation of the sample covariance matrix and d^3 is for performing the eigen-decomposition. In the distributed setting with m samples on each local machine, the DC approach incurs the computation cost of $\mathcal{O}(md^2 + d^3)$ since it is a one-shot algorithm. In comparison, our method incurs the $\mathcal{O}(md^2 + d^3 + TT'd^2)$ computational cost, in order to achieve the optimal convergence rate. We note that our method incurs one-time computational of the Hessian inverse with $\mathcal{O}(d^3)$ and each iteration only involves the efficient computation of the gradient (i.e., $\mathcal{O}(d^2)$). Therefore, the extra computational overhead over the DC $\mathcal{O}(TT'd^2)$ is a smaller order term in d as compared to $\mathcal{O}(d^3)$. Moreover, the number of iterations TT' is relatively small and thus the extra computation as compared to the DC is rather limited. In practice, one can easily combine two approaches. For example, one can initialize the estimator using the DC method, and further improve its accuracy using our method.

For the top- L -dim eigenspace estimation, we extend a framework from [AZL16] to our distributed settings. In our Algorithm 2, we first compute the leading eigenvector \mathbf{v}_1 of $\mathbf{A}^\top \mathbf{A}/n$ in a distributed manner with Algorithm 1. The \mathbf{v}_1 is then transferred back to local machines and used to right-project data matrix, i.e., $\mathbf{A}_k(\mathbf{I}_d - \mathbf{v}_1 \mathbf{v}_1^\top)$ for $k \in [K]$. The next eigenvector \mathbf{v}_2 is obtained with these projected data matrices and Algorithm 1. In other words, we estimate the top eigenvector of $(\mathbf{I}_d - \mathbf{v}_1 \mathbf{v}_1^\top) \hat{\Sigma} (\mathbf{I}_d - \mathbf{v}_1 \mathbf{v}_1^\top)$ in distributed settings. This procedure is repeated L times until we obtain all the L top eigenvectors $\mathbf{V}_L = [\mathbf{v}_1, \dots, \mathbf{v}_L]$. This deflation technique is quite straight-forward and performs well in our later convergence analysis.

Remark. Our paper, and also the earlier works [AZL16, FWWZ19] all assume data vectors are centered, i.e., zero-mean data vectors $\mathbb{E}[\mathbf{a}] = 0$. When the data is non-centered, we could adopt a two stage estimator, where the first stage centralizes the data in a distributed fashion and second stage applies our distributed PCA algorithm. In particular, each local machine k first computes the

mean of local samples, i.e., $\bar{\mathbf{a}}_k = \frac{1}{m_k} \sum_{i \in \mathcal{D}_k} \mathbf{a}_i$, where \mathcal{D}_k denotes the sample indices on the k -local machine and $m_k = |\mathcal{D}_k|$. Then each local machine transmits (\mathbf{a}_k, m_k) to the center. The center computes their average $\bar{\mathbf{a}} = \frac{\sum_{k=1}^K m_k \bar{\mathbf{a}}_k}{\sum_{k=1}^K m_k}$, which will be transmitted back to each local machine to center the data (i.e., each sample \mathbf{a}_i will be $\mathbf{a}_i - \bar{\mathbf{a}}$). Given the centralized data, we can directly apply our distributed PCA algorithm. This centralization step only incurs one extra round of communication and each local machine only transmits an $O(d)$ vector to the center (which is the same amount of communication as in our algorithm that transmits the gradient).

3 Theoretical Properties

This section exhibits the theoretical results for our setups in Section 2. The technical proofs will be relegated to the supplementary material (see Appendix A).

3.1 Distributed top eigenvector estimation

We first investigate the theoretical properties of the top eigenvector estimation in Algorithm 1. Let $\hat{\Sigma}_k = \mathbf{A}_k^\top \mathbf{A}_k / m \in \mathbb{R}^{d \times d}$ denote the local sample covariance matrix on machine $k \in [K]$, and $\hat{\Sigma} = K^{-1} \sum_{k=1}^K \hat{\Sigma}_k$ the global sample covariance matrix using all data. Let $\hat{\lambda}_1 \geq \hat{\lambda}_2 \geq \dots \geq \hat{\lambda}_d \geq 0$ and $\hat{\mathbf{u}}_1, \hat{\mathbf{u}}_2, \dots, \hat{\mathbf{u}}_d$ denote the sorted eigenvalues and associated eigenvectors of $\hat{\Sigma}$. We are interested in quantifying the quality of some estimator $\mathbf{w}^{(t)}$. More specifically, we will reserve the letter δ to denote the relative eigenvalue gap threshold, and will measure the closeness between $\mathbf{w}^{(t)}$ and the top eigenvector $\hat{\mathbf{u}}_1$ via proving

$$\sum_{l: \hat{\lambda}_l \leq (1-\delta) \hat{\lambda}_1} |\langle \hat{\mathbf{u}}_l, \mathbf{w}^{(t)} \rangle|^2 \leq \frac{\varepsilon^2}{\delta^2} \quad (9)$$

for error $\varepsilon > 0$ and any constant $\delta \in (0, 1)$. In particular, the result (9) above is always stronger (modulo constants) than the usual bound

$$\hat{\theta}^{(t)} := \arccos |\langle \hat{\mathbf{u}}_1, \mathbf{w}^{(t)} \rangle| \leq C \varepsilon \frac{\hat{\lambda}_1}{\hat{\lambda}_1 - \hat{\lambda}_2} \quad (10)$$

that involves the relative gap between the first two eigenvalues of $\hat{\Sigma}$. Here C is a constant. To see this, we can simply choose $\delta = (\hat{\lambda}_1 - \hat{\lambda}_2) / \hat{\lambda}_1$ in Equation (9). Then $\sin^2 \hat{\theta}^{(t)} = 1 - |\langle \hat{\mathbf{u}}_1, \mathbf{w}^{(t)} \rangle|^2 = \sum_{l: \hat{\lambda}_l \leq (1-\delta) \hat{\lambda}_1} |\langle \hat{\mathbf{u}}_l, \mathbf{w}^{(t)} \rangle|^2 \leq \varepsilon^2 / \delta^2$, implying $\hat{\theta}^{(t)} \leq \arcsin(\varepsilon / \delta) \leq C \varepsilon \hat{\lambda}_1 / (\hat{\lambda}_1 - \hat{\lambda}_2)$ for some universal constant $C > 0$. Moreover, it has to be assumed that $\hat{\lambda}_1 > \hat{\lambda}_2$ in the usual bound (10), which may not be held in some applications.

From our enlarged eigenspace viewpoint, the result in Equation (9) indicates that the top eigenvector estimator $\mathbf{w}^{(t)}$ is almost covered by the span of $\{\hat{\mathbf{u}}_l : \hat{\lambda}_l > (1-\delta) \hat{\lambda}_1\}$.

As we will show in the theoretical analysis later, the success of our algorithm relies on the initial values of both eigenvalue and eigenvector. We first clarify our choice of initial eigenvalue estimates.

For the top eigenvector estimation in Algorithm 1, since we have the following high probability bound (see Equation (31) in Lemma 4 in the appendix for the justification), $\|\hat{\Sigma} - \hat{\Sigma}_1\|_2 \leq \eta/2$ for some constant $\eta > 0$. If we choose $\bar{\lambda}_1^{(0)} = \lambda_1(\hat{\Sigma}_1) + 3\eta/2$, then it is guaranteed that, $2\eta \geq \bar{\lambda}_1^{(0)} - \hat{\lambda}_1 \geq \eta$. Lemma 4 (Equation (32)) also provides a concentration bound on our initial value

of eigenvectors, $\sum_{l: \hat{\lambda}_l \leq (1-\delta)\hat{\lambda}_1} |\langle \hat{\mathbf{u}}_l, \mathbf{w}^{(0)} \rangle|^2 \leq 3/4$ with high probability. Here $\{\hat{\mathbf{u}}_l : \hat{\lambda}_l \leq (1-\delta)\hat{\lambda}_1\}$ are all the eigenvectors for the full sample covariance matrix $\hat{\Sigma}$ whose associated eigenvalues have a relative gap δ from the largest eigenvalue $\hat{\lambda}_1$ and $\mathbf{w}^{(0)}$ is the top eigenvector for the sample covariance matrix on the first machine.

Given our initial estimators $\bar{\lambda}_1^{(0)}$ and $\mathbf{w}^{(0)}$, we have the following convergence guarantee for our Algorithm 1. With the above guarantees of initial estimator $\bar{\lambda}_1$ and $\mathbf{w}^{(0)}$, our first lemma characterizes the convergence rate of the *outer loop* in Algorithm 1.

Lemma 1. *Suppose the initial estimator $\bar{\lambda}_1$ satisfies*

$$\eta \leq \bar{\lambda}_1 - \hat{\lambda}_1 \leq 2\eta \quad \text{for some } \eta > 0 \quad (11)$$

For any $\mathbf{w} \in \mathbb{R}^d$, and $\mathbf{v} \in \mathbb{R}^d$ that satisfies

$$\|\mathbf{w}\|_2 = 1, \quad \sum_{l: \hat{\lambda}_l \leq (1-\delta)\hat{\lambda}_1} |\langle \hat{\mathbf{u}}_l, \mathbf{w} \rangle|^2 \leq \frac{3}{4} \quad (12)$$

and

$$\|\mathbf{v} - \mathbf{H}^{-1}\mathbf{w}\|_2 \leq \varepsilon \leq (8\eta)^{-1} \quad (13)$$

and for each index $l = 1, \dots$ such that $\hat{\lambda}_l \leq (1-\delta)\hat{\lambda}_1$, we have

$$\frac{|\langle \hat{\mathbf{u}}_l, \mathbf{v} \rangle|}{\|\mathbf{v}\|_2} \leq \frac{8\eta}{\delta\hat{\lambda}_1} \frac{|\langle \hat{\mathbf{u}}_l, \mathbf{w} \rangle|}{\|\mathbf{w}\|_2} + 8\eta\varepsilon \quad (14)$$

Moreover, we have

$$\sum_{l: \hat{\lambda}_l \leq (1-\delta)\hat{\lambda}_1} \frac{|\langle \hat{\mathbf{u}}_l, \mathbf{v} \rangle|^2}{\|\mathbf{v}\|_2^2} \leq \frac{128\eta^2}{\delta^2\hat{\lambda}_1^2} \sum_{l: \hat{\lambda}_l \leq (1-\delta)\hat{\lambda}_1} \frac{|\langle \hat{\mathbf{u}}_l, \mathbf{w} \rangle|^2}{\|\mathbf{w}\|_2^2} + 128\eta^2\varepsilon^2 \quad (15)$$

For the outer loop in our Algorithm 1, \mathbf{w} and $\mathbf{v}/\|\mathbf{v}\|_2$ in Lemma 1 can be explained as the t -th round and $(t+1)$ -th round estimators $\mathbf{w}^{(t)}$ and $\mathbf{w}^{(t+1)}$, respectively. This lemma implies that up to a numerical tolerance ε for inverting \mathbf{H} (Condition (13)), each application of the outer loop reduces the magnitude of the projection of $\mathbf{w}^{(t)}$ onto $\hat{\mathbf{u}}_l$ by a factor of $\mathcal{O}((\delta\hat{\lambda}_1)^{-1}\eta) \ll 1$ given $\eta \ll 1$ (if we have a good initial estimator of $\hat{\lambda}_1$ and $\delta\hat{\lambda}_1 = \Omega(1)$). Notice that if $\mathbf{w}^{(t)}$ satisfies condition (12), our Equation (15) claims that $\mathbf{w}^{(t+1)} = \mathbf{v}/\|\mathbf{v}\|_2$ satisfies Condition (12) as well. This condition is justified if $\mathbf{w}^{(0)}$ satisfies Condition (12), which is a conclusion from Lemma 4 in the appendix.

Our second lemma characterizes the convergence rate of distributively solving the linear system $\mathbf{H}\mathbf{w} = \mathbf{w}^{(t)}$ in the *inner loop* of Algorithm 1. Recall that in Equation (4), $\tilde{\mathbf{w}}^{(t+1)} = \mathbf{H}^{-1}\mathbf{w}^{(t)}$ denote the exact solution of this linear system.

Lemma 2. *Suppose the initial estimator $\bar{\lambda}_1$ satisfies*

$$\bar{\lambda}_1 - \hat{\lambda}_1 \geq \eta \geq \frac{1}{2} \left\| \hat{\Sigma} - \hat{\Sigma}_1 \right\|_2$$

Then for each $j = 0, 1, \dots, (T' - 1)$, we have

$$\|\mathbf{w}_{j+1}^{t+1} - \tilde{\mathbf{w}}^{(t+1)}\|_2 \leq \frac{2 \left\| \hat{\Sigma} - \hat{\Sigma}_1 \right\|_2}{\eta} \|\mathbf{w}_j^{t+1} - \tilde{\mathbf{w}}^{(t+1)}\|_2 \quad (16)$$

Here $\|\widehat{\Sigma} - \widehat{\Sigma}_1\|_2$ on the RHS of (16) is due to the approximation using the Hessian matrix \mathbf{H}_1 on the first machine in place of original Hessian matrix \mathbf{H} . As we will show later, by standard matrix concentration inequalities, we have $\|\widehat{\Sigma} - \widehat{\Sigma}_1\|_2 = \mathcal{O}\left(\sqrt{d/m}\right)$ with high probability. As a consequence, the inner loop of Algorithm 1 has a contraction rate of order $\mathcal{O}(\eta^{-1}\sqrt{d/m})$, which is inversely proportional to the gap $\bar{\lambda}_1 - \widehat{\lambda}_1$ (due to the condition number of the Hessian \mathbf{H}).

Combining these two lemmas, we come to our first main theoretical result for the convergence rate of Algorithm 1.

Theorem 1. *Let $\kappa := \|\widehat{\Sigma} - \widehat{\Sigma}_1\|_2 = \mathcal{O}_P(\sqrt{d/m})$. Assume*

$$2\eta \geq \bar{\lambda}_1 - \widehat{\lambda}_1 \geq \eta \geq \frac{1}{2}\kappa$$

and the initial eigenvector estimator $\mathbf{w}^{(0)}$ satisfies

$$\sum_{l: \widehat{\lambda}_l \leq (1-\delta)\widehat{\lambda}_1} |\langle \widehat{\mathbf{u}}_l, \mathbf{w}^{(0)} \rangle|^2 \leq \frac{3}{4}$$

Then for each T and T' as the outer and inner iterations in Algorithm 1, respectively, and the relative eigenvalue gap $\delta \in (0, 1)$, we have

$$\sum_{l: \widehat{\lambda}_l \leq (1-\delta)\widehat{\lambda}_1} |\langle \widehat{\mathbf{u}}_l, \mathbf{w}^{(t)} \rangle|^2 \leq \left(\frac{128\eta^2}{\delta^2\widehat{\lambda}_1^2}\right)^T + \frac{512\eta}{1 - 128\eta^2/(\delta\widehat{\lambda}_1)^2} \left(\frac{4\kappa^2}{\eta^2}\right)^{T'} \quad (17)$$

We can further simplify Equation (17) by choosing proper η and T' .

Corollary 1. *In particular, if $\eta \leq \delta\widehat{\lambda}_1/16$, and we choose $T' = T$, and $\eta = (\kappa\delta\widehat{\lambda}_1)^{1/2}/3 = \mathcal{O}_P(\sqrt[4]{d/m})$, then the final output $\mathbf{w}^{(T)}$ satisfies*

$$\sum_{l: \widehat{\lambda}_l \leq (1-\delta)\widehat{\lambda}_1} |\langle \widehat{\mathbf{u}}_l, \mathbf{w}^{(T)} \rangle|^2 \leq 257 \left(\frac{6\kappa}{\delta\widehat{\lambda}_1}\right)^{2T} \quad (18)$$

As indicated in (18), when $6\kappa/\delta\widehat{\lambda}_1 \ll 1$, our Algorithm 1 enjoys a linear convergence rate. Moreover, to ensure this convergence, when the absolute eigengap $\delta\widehat{\lambda}_1$ is small, $\kappa = \mathcal{O}_P(\sqrt{d/m})$ needs to be smaller, i.e., $\kappa = o(\delta\widehat{\lambda}_1)$. Recall that $\kappa := \|\widehat{\Sigma} - \widehat{\Sigma}_1\|_2$, which is defined in Theorem 1. This indicates that more samples are needed on each local machine.

Remark. Under the setting without an explicit eigengap, our goal is not to construct a good estimator of the top eigenvector. Instead, we aim to construct an estimator that captures a similar amount of variability in the sample data as the top eigenvector. Recall that by Theorem 1, we construct an estimator \mathbf{w} such that $\sum_{l: \widehat{\lambda}_l \leq (1-\delta)\widehat{\lambda}_1} |\langle \widehat{\mathbf{u}}_l, \mathbf{w} \rangle|^2 \leq \varepsilon$ for some error term $\varepsilon > 0$. We can see that,

$$\mathbf{w}^\top \widehat{\Sigma} \mathbf{w} > (1 - \delta)(1 - \varepsilon)\widehat{\lambda}_1 \quad (19)$$

This fact can be easily derived as follows (see also the proof of Theorem 3.1 in [AZL16]),

$$\mathbf{w}^\top \widehat{\Sigma} \mathbf{w} = \sum_{i=1}^d \widehat{\lambda}_i (\mathbf{w}^\top \widehat{\mathbf{u}}_i)^2 \geq \sum_{l: \widehat{\lambda}_l > (1-\delta)\widehat{\lambda}_1} \widehat{\lambda}_l (\mathbf{w}^\top \widehat{\mathbf{u}}_l)^2 \geq (1 - \delta)\widehat{\lambda}_1 \sum_{l: \widehat{\lambda}_l > (1-\delta)\widehat{\lambda}_1} (\mathbf{w}^\top \widehat{\mathbf{u}}_l)^2$$

$$\geq (1 - \delta)(1 - \varepsilon)\widehat{\lambda}_1$$

According to (19), our estimator \mathbf{w} captures almost the same amount of variability of the sampled data (up to a $(1 - \delta)(1 - \varepsilon)$ multiplicative factor). This type of results are also known as the gap-free bound in some optimization literature (see e.g., [AZL16]).

When the eigengap is extremely small, identifying the top eigenvector is an information-theoretically difficult problem. As an extreme case, when the gap is zero, it is impossible to distinguish between the top and the second eigenvectors. In contrast, our setting is favorable in practice since the main goal of PCA/dimension reduction is to capture the variability of the data.

Moreover, we note that the parameter δ is a pre-specified parameter that measures the proportion of the variability explain by the estimator \mathbf{w} . For example, when setting $\delta = \varepsilon$, the estimator \mathbf{w} will capture at least $(1 - 2\varepsilon)$ of the variability captured by the top eigenvector according to (19). We can also choose $\delta = c_0/\widehat{\lambda}_1$ for some constant c_0 so that $\delta\widehat{\lambda}_1 = \Omega(1)$.

3.2 Distributed top- L -dim principal subspace estimation

With the theoretical results for the top eigenvector estimation in place, we further present convergence analysis on the top- L -dim eigenspace estimation in Algorithm 2.

Let $\widehat{\mathbf{U}}_{\leq(1-\delta)\widehat{\lambda}_L} = [\widehat{\mathbf{u}}_{S+1}, \dots, \widehat{\mathbf{u}}_d]$ denote the column orthogonal matrix composed of all eigenvectors of $\widehat{\Sigma}$ whose associated eigenvalues have a relative gap δ from the L -th largest eigenvalue $\widehat{\lambda}_L$, that is, $S := \arg \max\{l : \widehat{\lambda}_l > (1 - \delta)\widehat{\lambda}_L\}$. We also denote $\widehat{\mathbf{U}}_{>(1-\delta)\widehat{\lambda}_L} = [\widehat{\mathbf{u}}_1, \dots, \widehat{\mathbf{u}}_S]$ to be the enlarged eigenspace corresponding to the eigenvalues larger than $(1 - \delta)\widehat{\lambda}_L$.

We also use the notation $\widehat{\Sigma}^{(l)} = (\mathbf{I} - \mathbf{V}_{l-1}\mathbf{V}_{l-1}^\top)\widehat{\Sigma}(\mathbf{I} - \mathbf{V}_{l-1}\mathbf{V}_{l-1}^\top)$ for $l = 0, 1, \dots, L - 1$. Here $\mathbf{V}_l = [\mathbf{v}_1, \dots, \mathbf{v}_l]$ consists of all the top- l eigenvector estimations and $\mathbf{V}_0 = \mathbf{0}$. Notice that $\widehat{\Sigma}^{(l)}$ is just the matrix $\mathbf{A}^{(l)}\mathbf{A}^{(l)}/n$ where $\mathbf{A}^{(l)} := [\mathbf{A}_{1,l}^\top, \dots, \mathbf{A}_{K,l}^\top]^\top$ and $\mathbf{A}_{k,l}^T$ is the projected data matrix on machine k ($k \in [K]$) for the l -th eigenvector estimation.

We first provide our choices of initial eigenvalue estimates. For Algorithm 2 for the top- L -dim principal, let $\widehat{\Sigma}_k^{(l)} = \mathbf{A}_{k,l}^\top\mathbf{A}_{k,l}/m$ and $\widehat{\Sigma}^{(l)} = K^{-1}\sum_{k=1}^K \widehat{\Sigma}_k^{(l)}$ denote the local and global projected sample covariance matrices at the outer iteration l . For the same constant η defined above in (11), we choose $\bar{\lambda}_l = \lambda_1(\widehat{\Sigma}_1^{(l)}) + 3\eta/2$ for $l \in [L]$. This follows from

$$\left\| \widehat{\Sigma}^{(l)} - \widehat{\Sigma}_1^{(l)} \right\|_2 = \left\| (\mathbf{I} - \mathbf{V}_l\mathbf{V}_l^\top)(\widehat{\Sigma} - \widehat{\Sigma}_1)(\mathbf{I} - \mathbf{V}_l\mathbf{V}_l^\top) \right\|_2 \leq \left\| \widehat{\Sigma} - \widehat{\Sigma}_1 \right\|_2 \leq \frac{\eta}{2}$$

which implies that,

$$2\eta \geq \bar{\lambda}_l - \lambda_1(\widehat{\Sigma}^{(l)}) \geq \eta$$

Our main result is summarized as follows.

Theorem 2. Let $\kappa = \left\| \widehat{\Sigma} - \widehat{\Sigma}_1 \right\|_2 = \mathcal{O}_P(\sqrt{d/m})$. Assume

$$2\eta \geq \bar{\lambda}_l - \lambda_1(\widehat{\Sigma}^{(l)}) \geq \eta \geq \frac{1}{2}\kappa$$

for each $l \in [L]$, where $\lambda_1(\widehat{\Sigma}^{(l)})$ denotes the largest eigen value of $\widehat{\Sigma}^{(l)}$. Then we have

$$\left\| \widehat{\mathbf{U}}_{\leq(1-\delta)\widehat{\lambda}_L}^\top \mathbf{V}_L \right\|_2^2 \leq \frac{64\widehat{\lambda}_1 L^2}{\widehat{\lambda}_L \delta} \sqrt{\left(\frac{128\eta^2}{\delta^2 \widehat{\lambda}_L^2}\right)^T + \frac{512\eta}{1 - 128\eta^2/(\delta\widehat{\lambda}_L)^2} \left(\frac{4\kappa^2}{\eta^2}\right)^{T'}} \quad (20)$$

By choosing specific settings of some parameters, the result in Theorem 2 can be simplified as shown in the following corollary.

Corollary 2. *Similarly, if $\eta \leq \delta\hat{\lambda}_1/16$, and we choose $T' = T$, and $\eta = (\kappa\delta\hat{\lambda}_1)^{1/2}/3 = \mathcal{O}_P(\sqrt[4]{d/m})$, then our estimator \mathbf{V}_L satisfies*

$$\left\| \widehat{\mathbf{U}}_{\leq(1-\delta)\hat{\lambda}_L}^\top \mathbf{V}_L \right\|_2^2 = \mathcal{O}\left(\frac{\hat{\lambda}_1 L^2}{\hat{\lambda}_L \delta} \left(\frac{6\kappa}{\delta\hat{\lambda}_L}\right)^T\right)$$

Here we could also interpret our results in Theorem 2 from an “angle” point of view corresponding to the classical $\sin \Theta$ result. Since there is no eigengap assumption, it is impossible to directly estimate \mathbf{U}_L . Therefore, we choose a parameter δ , and consider *an enlarged eigenspace* $\mathbf{U}_{>(1-\delta)\hat{\lambda}_L}$. Our theoretical results (see Theorem 2 and Corollary 2) imply that the “angle” between our estimator \mathbf{V}_L and $\widehat{\mathbf{U}}_{\leq(1-\delta)\hat{\lambda}_L}$ is sufficiently small. This result extends the classical $\sin \Theta$ result.

Similar to the top-eigenvector case in Remark , our estimator \mathbf{V}_L can also capture a similar amount of variability in the sampled data to $\widehat{\mathbf{U}}_L := \{\widehat{\mathbf{u}}_1, \dots, \widehat{\mathbf{u}}_L\}$. We further describe this property in the following Corollary 3.

Corollary 3. *Assume our estimator \mathbf{V}_L from Algorithm 2 satisfies $\left\| \widehat{\mathbf{U}}_{\leq(1-\delta)\hat{\lambda}_L}^\top \mathbf{V}_L \right\|_2 \leq \frac{\delta}{16\hat{\lambda}_1/\hat{\lambda}_{L+1}}$, then we have,*

$$\hat{\lambda}_{L+1} \leq \left\| (\mathbf{I}_d - \mathbf{V}_L \mathbf{V}_L^\top) \widehat{\Sigma} (\mathbf{I}_d - \mathbf{V}_L \mathbf{V}_L^\top) \right\|_2 \leq \frac{\hat{\lambda}_{L+1}}{1-\delta}, \quad (21)$$

$$(1-\delta)\hat{\lambda}_l \leq \mathbf{v}_l^\top \widehat{\Sigma} \mathbf{v}_l \leq \frac{1}{1-\delta} \hat{\lambda}_l, \quad \forall l \in [L] \quad (22)$$

Now we further extend the result in Corollary 2 to quantify the “angle” between our estimator \mathbf{V}_L and the population eigenspace $\mathbf{U}_{\leq(1-2\delta)\lambda_L}$.

Corollary 4. *Assume our estimator \mathbf{V}_L from Algorithm 2 satisfies $\left\| \widehat{\mathbf{U}}_{\leq(1-\delta)\hat{\lambda}_L}^\top \mathbf{V}_L \right\|_2 \leq \varepsilon$ for some error term $\varepsilon > 0$, then we have,*

$$\left\| \mathbf{U}_{\leq(1-2\delta)\lambda_L}^\top \mathbf{V}_L \right\|_2 \leq \frac{\left\| \Sigma - \widehat{\Sigma} \right\|_2}{(1-\delta)(\hat{\lambda}_L - \lambda_L) + \delta\lambda_L} + \varepsilon \quad (23)$$

where $\mathbf{U}_{\leq(1-2\delta)\lambda_L}$ is the eigenvectors of the population covariance matrix Σ corresponding to eigenvalues less than or equal to $(1-2\delta)\lambda_L$.

We further provide a different “angle” result on quantifying the complement of an enlarged space of V_L . Recall our definition $S := \arg \max\{l : \hat{\lambda}_l > (1-\delta_L)\hat{\lambda}_L\}$. We can classify $\widehat{\mathbf{u}}_1, \dots, \widehat{\mathbf{u}}_d$ and correspondingly our estimators $\mathbf{v}_1, \dots, \mathbf{v}_d$ from Algorithm 2 into three regimes:

$$\begin{aligned} & \underbrace{\widehat{\mathbf{u}}_1, \dots, \widehat{\mathbf{u}}_L}_{\widehat{\mathbf{U}}_L}, \underbrace{\widehat{\mathbf{u}}_{L+1}, \dots, \widehat{\mathbf{u}}_S}_{\widehat{\mathbf{U}}_{\leq(1-\delta)\hat{\lambda}_L}}, \underbrace{\widehat{\mathbf{u}}_{S+1}, \dots, \widehat{\mathbf{u}}_d}_{\widehat{\mathbf{U}}_{\geq(1-\delta)\hat{\lambda}_L}} \\ & \underbrace{\mathbf{v}_1, \dots, \mathbf{v}_L}_{\mathbf{V}_L}, \underbrace{\mathbf{v}_{L+1}, \dots, \mathbf{v}_S}_{\mathbf{V}_{\leq(1-\delta)\hat{\lambda}_L}}, \underbrace{\mathbf{v}_{S+1}, \dots, \mathbf{v}_d}_{\mathbf{V}_{\geq(1-\delta)\hat{\lambda}_L}} \end{aligned} \quad (24)$$

Corollary 2 shows that the “angle” between our estimator \mathbf{V}_L and $\widehat{\mathbf{U}}_{\leq(1-\delta)\widehat{\lambda}_L}$ is sufficiently small. Similarly, we can show the counterpart of this result, which indicates that the “angle” between $\widehat{\mathbf{U}}_L$ and $\mathbf{V}_{\leq(1-\delta)\widehat{\lambda}_L}$ is also very small. This result will be useful in our principal component regression example. To introduce our result, we denote $\widehat{\lambda}_S$ to be the S -th largest eigenvalue of $\widehat{\Sigma}$.

Theorem 3. *By running Algorithm 2 for obtaining the distributed top- S -dim principal subspace estimator \mathbf{V}_S , if there exists $\delta_S < \delta$ such that $\left\| \widehat{\mathbf{U}}_{\leq(1-\delta_S)\widehat{\lambda}_S}^\top \mathbf{V}_S \right\|_2 \leq \frac{\delta_S \widehat{\lambda}_1}{16\widehat{\lambda}_1/\widehat{\lambda}_{S+1}}$, then we have for the empirical eigenspace $\widehat{\mathbf{U}}_L$*

$$\left\| \widehat{\mathbf{U}}_L^\top \mathbf{V}_{\leq(1-\delta)\widehat{\lambda}_L} \right\|_2 \leq S \frac{\delta_S \widehat{\lambda}_1}{\widehat{\lambda}_L(1 - \delta_S) - \widehat{\lambda}_S} \quad (25)$$

Furthermore, for the population eigenspace \mathbf{U}_L , we can derive that

$$\left\| \mathbf{U}_L^\top \mathbf{V}_{\leq(1-2\delta)\widehat{\lambda}_L} \right\|_2 \leq S \frac{\delta_S \widehat{\lambda}_1}{\widehat{\lambda}_L(1 - \delta_S) - \widehat{\lambda}_S} + \frac{\left\| \Sigma - \widehat{\Sigma} \right\|_2}{(1 - \delta)(\widehat{\lambda}_L - \lambda_L) + \delta \lambda_L} \quad (26)$$

The reason why we impose the upper bound on $\left\| \widehat{\mathbf{U}}_{\leq(1-\delta_S)\widehat{\lambda}_S}^\top \mathbf{V}_S \right\|_2$ is mainly to obtain the result in Equation (21) for \mathbf{V}_S . We also note that this upper bound can be easily satisfied as long as we run Algorithm 2 for sufficiently large number of iterations.

Let us recall the classical Davis-Kahan result for PCA in Equation (2). As we explained in the introduction, without an eigengap condition, the estimation error can be arbitrarily large. However, our enlarged eigenspace estimator \mathbf{V}_S in (24) (i.e., $\mathbf{V}_{>(1-\delta)\widehat{\lambda}_L}$) will almost contain the top- L -dim eigenspace of the population covariance matrix. In particular, by Equation (26) and Lemma 3 in the appendix, we have shown that there exists a matrix \mathbf{Q} satisfying $\|\mathbf{Q}\|_2 \leq 1$ such that the error bound $\left\| \mathbf{U}_L - \mathbf{V}_{>(1-2\delta)\widehat{\lambda}_L} \mathbf{Q} \right\|_2$ is sufficiently small.

Our enlarged eigenspace results find important applications to many statistical problems. In particular, in Appendix B in the supplement, we illustrate how the theoretical results can be applied to the principal component regression (Example 1) and the single index model (Example 2). We also provide simulation studies of these two applications in Appendix D in the supplement.

Remark. It is also worthwhile to note that we assume the data are evenly split only for the ease of discussions. In fact, the local sample size m in our theoretical results is the sample size on the first machine (or any other machine that used to compute the estimation of Hessian \mathbf{H}) in Algorithm 1 and Algorithm 2. As long as the sample size m on the first machine is specified, our method does not depend on the partition of the entire dataset.

4 Numerical Study

In this section, we provide simulation experiments to illustrate the empirical performance of our distributed PCA algorithm.

Our data follows a normal distribution, $\mathbb{E}[\mathbf{a}] = 0$ and the population covariance matrix $\mathbb{E}[\mathbf{a}\mathbf{a}^\top] = \Sigma$ is generated as follows:

$$\Sigma = \mathbf{U} \Lambda \mathbf{U}^T$$

where \mathbf{U} is an orthogonal matrix generated randomly and Λ is a diagonal matrix. Since our experiments mainly estimate the top-3 eigenvectors, Λ has the following form,

$$\Lambda = \text{diag}(1 + 3\delta, 1 + 2\delta, 1 + \delta, 1, \dots, 1) \quad (27)$$

For example, when the relative eigengap δ is 1, $\Lambda = \text{diag}(4, 3, 2, 1, \dots, 1)$.

For orthogonal matrix $\mathbf{U} = [u_{ij}] \in \mathbb{R}^{d \times d}$, we first generate all elements $u_{ij}, i, j = 1, \dots, d$ such that they are *i.i.d.* standard normal variables. We then use Gram-Schmidt process to orthonormalize the matrix and obtain the \mathbf{U} .

We will compare our estimator with the following two estimators:

(1) Oracle estimator: the PCA estimator is computed in the single-machine setting with pooled data, i.e., we gather all the sampled data and compute the top eigenspace of $\widehat{\Sigma} = \frac{1}{n} \mathbf{A} \mathbf{A}^\top$, where $\mathbf{A} \in \mathbb{R}^{n \times d}$ is the data matrix.

(2) DC estimator (Algorithm 1 in [FWWZ19]): it first computes the top- L -dim eigenspace estimation $\widehat{\mathbf{U}}_L^{(k)}, k = 1, \dots, K$ on each machine, and merges every local result together with $\widetilde{\Sigma} = \frac{1}{K} \sum_{k=1}^K \widehat{\mathbf{U}}_L^{(k)} \widehat{\mathbf{U}}_L^{(k)\top}$. The final estimator is given by the eigenvalue decomposition of $\widetilde{\Sigma}$.

Note that all the reported estimation errors are computed based on the average of 100 Monte-Carlo simulations. Since the standard deviations of Monte-Carlo estimators for all the methods are similar and sufficiently small, we omit standard deviation terms in the following Figures and only report the average errors for better visualization. As shown in the following subsections, our distributed algorithm gets to a very close performance with the oracle one when the number of outer iterations T is large enough and outperforms its divide-and-conquer counterpart.

For distributed PCA, we adopt the following error measurements from the bound (17) and bound (20) with population eigenvectors replacing the oracle estimator. To be more specific, for the top eigenvector case, with the estimator $\widehat{\mathbf{u}}_1$, population eigenvectors $\mathbf{u}_1, \dots, \mathbf{u}_d$, population eigenvalues $\lambda_1, \dots, \lambda_d$ and relative eigenvalue gap $\delta \in (0, 1)$, the error measurement is defined as

$$\text{error}(\widehat{\mathbf{u}}_1) = \sum_{l: \lambda_l \leq (1-\delta)\lambda_1} |\langle \mathbf{u}_l, \widehat{\mathbf{u}}_1 \rangle|^2 \quad (28)$$

As for the top- L -dim eigenspace estimation, let $\widetilde{\mathbf{U}} = [\mathbf{u}_{l_\delta}, \dots, \mathbf{u}_d]$ be the column orthogonal matrix composed of all eigenvectors of population covariance Σ whose associated eigenvalues have a relative gap δ from the L -th largest eigenvalue λ_L . That is, $l_\delta := \arg \min\{l : \widehat{\lambda}_l \leq (1 - \delta) \widehat{\lambda}_L\}$. Recall that $\widehat{\mathbf{U}}_L$ is the estimator of the top- L eigenvectors. Then the corresponding error should be

$$\text{error}(\widehat{\mathbf{U}}_L) = \left\| \widetilde{\mathbf{U}}^\top \widehat{\mathbf{U}}_L \right\|_2^2 \quad (29)$$

4.1 Varying the number of outer iterations

In this section, we present tests on how the performance of our distributed PCA changes with the number of outer iterations T in Algorithm 1. Consider data dimension d to be 50, sample size on each machine to be 500, and the number of machines to be 200, i.e., $\mathbf{a} \in \mathbb{R}^{50}$, $m = 500$ and $K = 200$.

We will report the *logarithmic error*. As shown in Theorem 1, the logarithmic error follows an approximately linear decrease with respect to the number of outer iterations. A linear relationship between the number of outer iterations and logarithmic error verifies our theoretical findings.

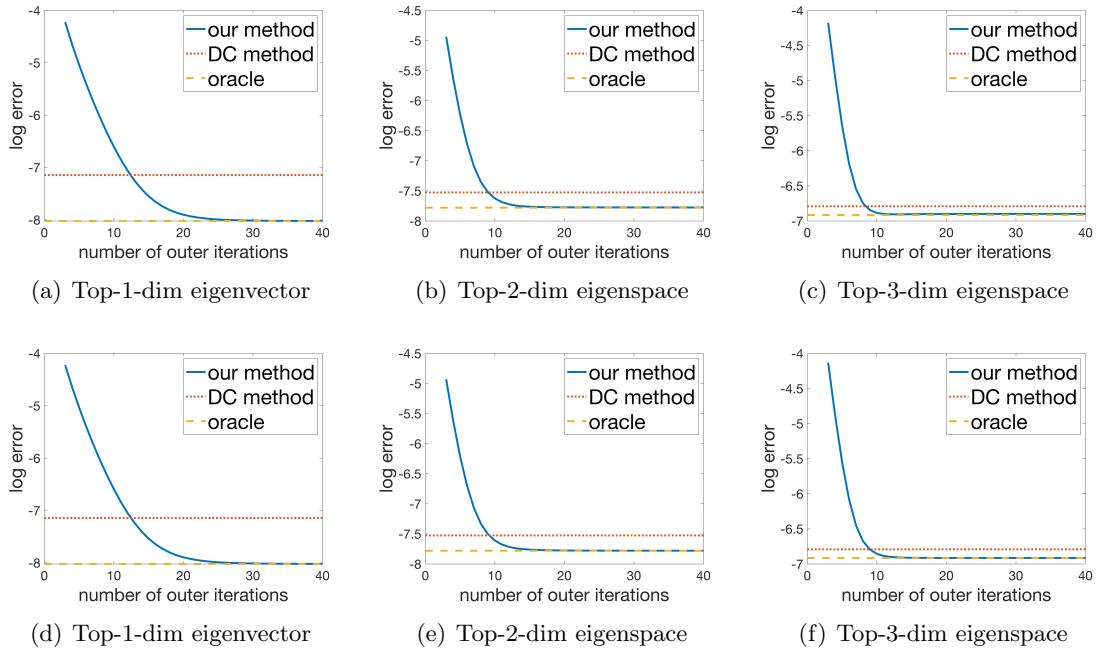


Figure 1. Comparison between algorithms when the number of outer iterations varies. The x -axis is the number of outer iterations and the y -axis is the *logarithmic error*. The blue line is our error, the red line is the DC method performance and the yellow one is logarithmic error for the oracle estimator. Subfigures (a) to (c) represent the experiments with 5 inner loops. Subfigures (d) to (f) represent the experiments with 10 inner loops. Eigengap δ is fixed to be 1.0.

We now check the performance of these three approaches (oracle one, our method and DC method) under the setting of a small eigengap. Specifically, we let eigengap δ to be 1.0 and 2.0. Our data is drawn independently, and $\mathbf{a}_i \sim \mathcal{N}(\mathbf{0}, \Sigma)$ for $i = 1, \dots, mK$. We vary the number of outer iterations T to evaluate the performance.

As we fix the total sample size $n = 10^5$, the errors of oracle estimator and DC estimator should be constants (illustrated by two horizontal dash lines in the graphs since they are not iterative algorithms). As shown below in Figure 1 and Figure 2, our method converges to the oracle estimator in around 20 iterations and outperforms the DC method. Moreover, as expected, we observe a approximately linear relation between logarithmic error and the number of outer iterations. We also observe that, empirically, setting the number of inner iterations $T' = 5$ in Algorithm 1 is good enough for most cases.

4.2 Varying the eigengap

In the convergence analysis of both our distributed algorithm and DC method, eigengap plays a central role in the error bound. When the eigengap between λ_L and λ_{L+1} becomes smaller, the estimation task turns to be harder and more rounds are needed for the same error. Theorem 4 in [FWWZ19] also shows a similar conclusion. In this part, we continue our experiment in Section 4.1, and examine the relationship between estimation error and eigengap.

We fix the number of inner iterations to be 10, and the number of outer iterations to be 40, which, from Section 4.1, is large enough for top-3-dim eigenspace. We still consider data dimension d to be 50, sample size on each machine to be 500, and the number of machines to be 200, i.e.,

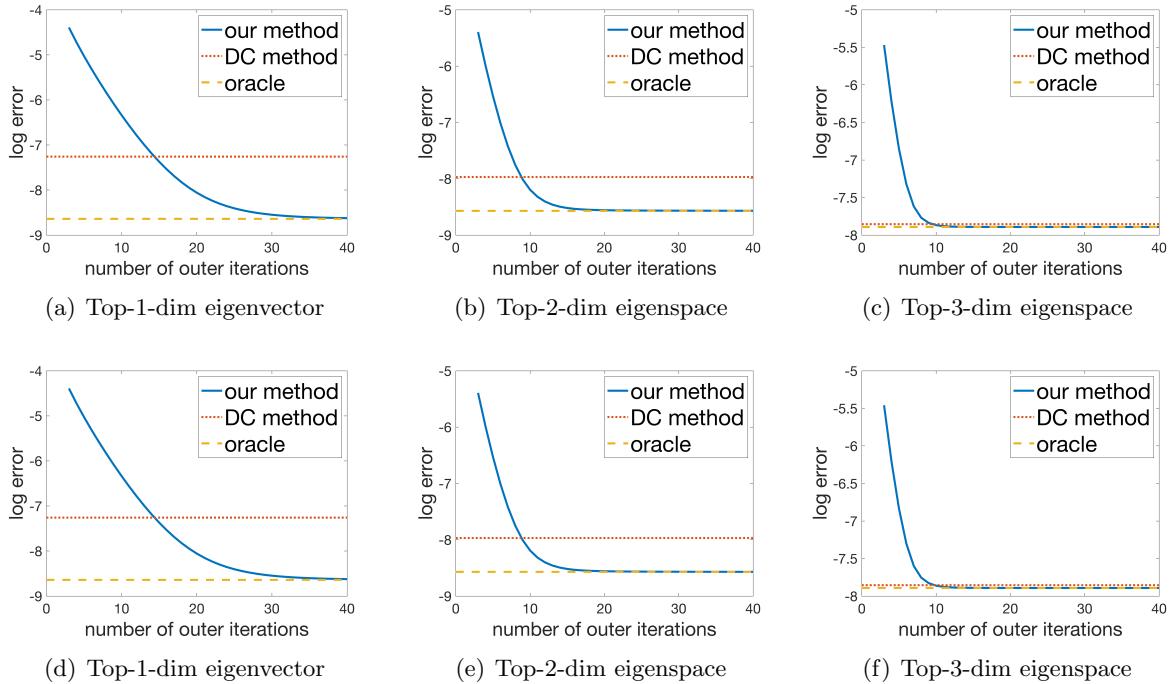


Figure 2. Comparison between algorithms when the number of outer iterations varies, under the same setting as in Figure 1. Subfigures (a) to (c) represent the experiments with 5 inner loops. Subfigures (d) to (f) represent the experiments with 10 inner loops. Eigengap δ is fixed to be 2.0.

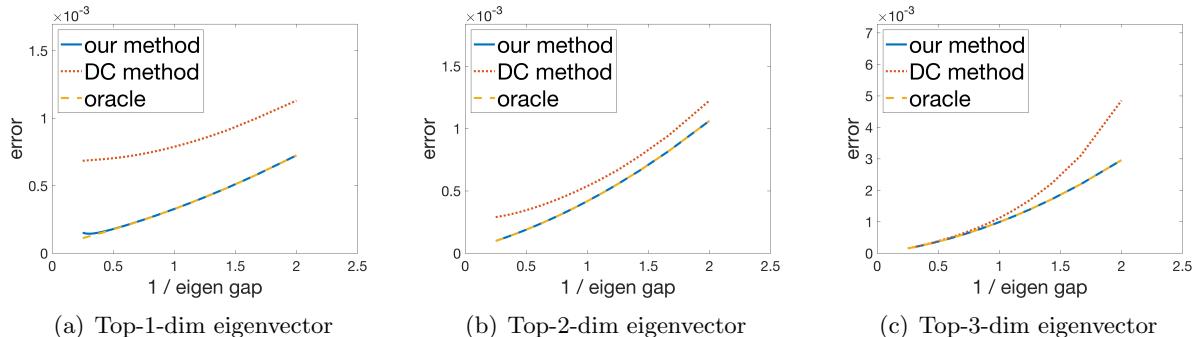


Figure 3. Comparison between algorithms when the eigengap varies. The x -axis is the reciprocal of eigengap and the y -axis is the logarithmic error.

$\mathbf{a} \in \mathbb{R}^{50}$, $m = 500$ and $K = 200$. Under this setting, we vary δ in (27) and the results is shown in Figure 3. In Figure 3, the logarithmic error increases with respect to $1/\delta$, which agrees with our theoretical findings. Furthermore, our estimator has the same performance as the oracle one.

4.3 Varying the number of machines for asymmetric innovation distributions

In this section, we compare our method to the DC method by varying the number of local machines. As mentioned in Theorem 4 in [FWWZ19], DC method has a slower convergence rate (of order $\mathcal{O}(\rho\sqrt{Lr/n}) + \mathcal{O}(\rho^2\sqrt{Lr}/m)$ instead of the optimal rate $\mathcal{O}(\rho\sqrt{Lr/n})$) when the number of machines is greater than $\mathcal{O}(m/(\rho^2r))$ in the asymmetric innovation distributions (defined in Section 1.1)

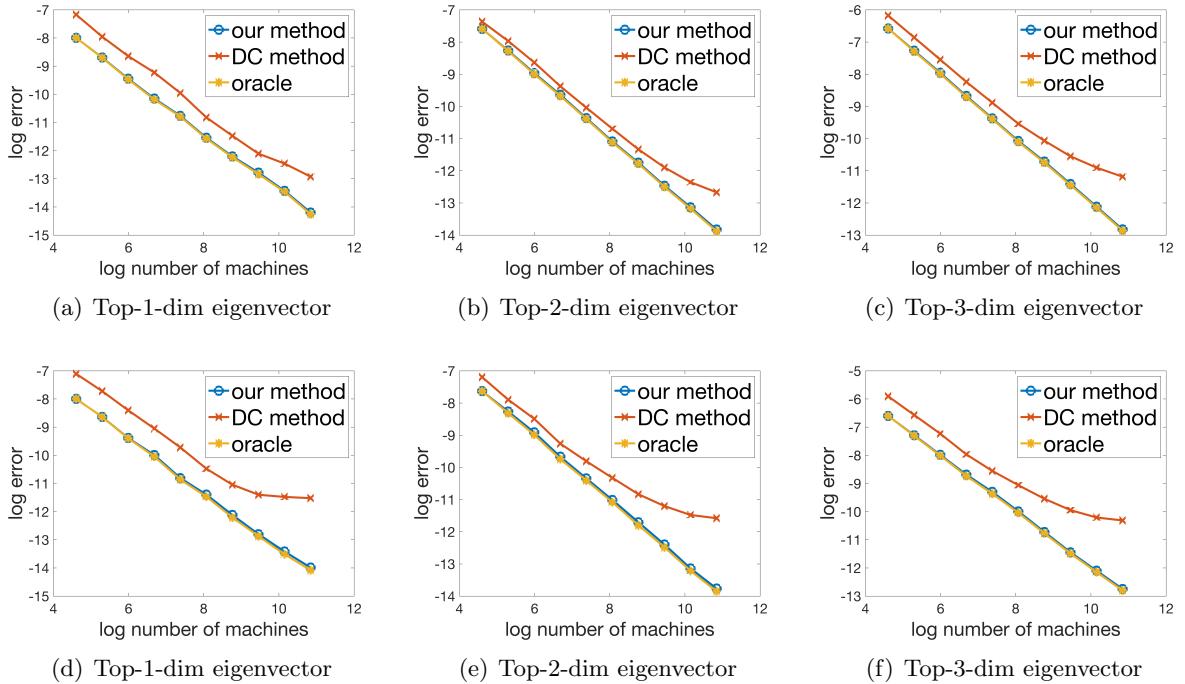


Figure 4. Comparison between algorithms when the number of machines varies. The x -axis is the log the number of machines and the y -axis is the logarithmic error. Subfigures (a) to (c) represent the experiments of top-1-dim to top-3-dim eigenvector estimation with skewness 4.0 and (d) to (f), 6.0.

setting. Here ρ is the condition number of the population covariance matrix, i.e., $\rho = \lambda_1/(\lambda_L - \lambda_{L+1})$, and $r = \text{Tr}(\Sigma)/\lambda_1$ is the effective rank of Σ .

We set data dimension d to be 50, local sample size to be 500, i.e., $\mathbf{a} \in \mathbb{R}^{50}$, $m = 500$. We choose eigengap δ to be 0.5, thus $\Lambda = \text{diag}(2.5, 2, 1.5, 1, \dots, 1)$. Here, without sticking on our Gaussian setting, we consider to use skew-distributed random variables. In particular, we generate $\mathbf{a} = [a_1, \dots, a_d]^\top \in \mathbb{R}^d$ from beta distribution family such that for each $a_i, i = 1, \dots, d$, we set its mean to be zero, variance to be Λ_{ii} and skewness to be 4 or 6, respectively.

We set the iteration parameters as in Section 4.2 and the number of machines is varied from 100 to 51,200. Our results are shown in Figure 4. As can be seen from Figure 4, our method achieves the same statistical convergence rate as the oracle one. When the number of machines is small, the estimation error of the DC method also decreases at the same rate as the number of machines increases. However, the estimation error the of DC method becomes flat (or decreases at a much slower rate) when the number of machines is larger than a certain threshold. In that regime, our approach is still comparable to its oracle counterpart.

We also conduct simulation studies on principal component regression and Gaussian single index model cases and compare our approach with the oracle and the DC ones. Due to the space limitation, we defer these results to Appendix D in the supplementary material.

5 Conclusion

In this paper, we address the problem of distributed estimation for principal eigenspaces and propose a novel multi-round algorithm for top- L -dimensional eigenspace estimation in distributed environments. Our method leverages shift-and-invert preconditioning and convex optimization to achieve fast convergence rates while removing constraints on the number of machines, a limitation in existing divide-and-conquer approaches. Unlike traditional methods that rely on an eigengap assumption, our analysis takes an alternative approach by showing that our estimator can *cover* the targeted eigenspace, eliminating the need for explicit eigengap conditions.

Our distributed PCA algorithm is communication-efficient and applies to various PCA-based problems, such as principal component regression and single index models. Simulation studies confirm the superior performance of our estimator compared to existing methods.

Future work could explore extensions to high-dimensional PCA and sparse PCA, investigating computational strategies and theoretical tools for these more complex distributed settings.

References

- [AZL16] Zeyuan Allen-Zhu and Yuanzhi Li. LazySVD: Even faster SVD decomposition yet without agonizing pain. In *Proceedings of the Advances in Neural Information Processing Systems (NIPS)*, 2016.
- [BCV13] Y. Bengio, A. Courville, and P. Vincent. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828, 2013.
- [BDS⁺19] Moulinath Banerjee, Cecile Durot, Bodhisattva Sen, et al. Divide and conquer in nonstandard problems and the super-efficiency phenomenon. *The Annals of Statistics*, 47(2):720–757, 2019.
- [BFL⁺18] Heather Battey, Jianqing Fan, Han Liu, Junwei Lu, and Ziwei Zhu. Distributed testing and estimation under sparse high dimensional models. *The Annals of Statistics*, 46(3):1352, 2018.
- [BHPT06] Eric Bair, Trevor Hastie, Debashis Paul, and Robert Tibshirani. Prediction by supervised principal components. *Journal of the American Statistical Association*, 101(473):119–137, 2006.
- [CLMY20] Xi Chen, Weidong Liu, Xiaojun Mao, and Zhuoyi Yang. Distributed high-dimensional regression under a quantile loss function. *Journal of Machine Learning Research*, 21(182):1–43, 2020.
- [CLZ⁺19] Xi Chen, Weidong Liu, Yichen Zhang, et al. Quantile regression under memory constraint. *Annals of Statistics*, 47(6):3244–3273, 2019.
- [CLZ21] Xi Chen, Weidong Liu, and Yichen Zhang. First-order newton-type estimator for distributed estimation and inference. *Journal of the American Statistical Association*, page to appear, 2021.
- [DK70] Chandler Davis and William Morton Kahan. The rotation of eigenvectors by a perturbation. iii. *SIAM Journal on Numerical Analysis*, 7(1):1–46, 1970.
- [FF93] LLDiko E Frank and Jerome H Friedman. A statistical view of some chemometrics regression tools. *Technometrics*, 35(2):109–135, 1993.
- [FGW19] Jianqing Fan, Yongyi Guo, and Kaizheng Wang. Communication-efficient accurate statistical estimation. *arXiv preprint arXiv:1906.04870*, 2019.
- [FWWZ19] Jianqing Fan, Dong Wang, Kaizheng Wang, and Ziwei Zhu. Distributed estimation of principal eigenspaces. *The Annals of Statistics*, 47(6):3009–3031, 2019.
- [GH15] Dan Garber and Elad Hazan. Fast and simple PCA via convex optimization. *arXiv preprint arXiv:1509.05647*, 2015.

- [GHJ⁺16] Dan Garber, Elad Hazan, Chi Jin, Sham M. Kakade, Cameron Musco, Praneeth Netrapalli, and Aaron Sidford. Faster eigenvector computation via shift-and-invert preconditioning. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2016.
- [GSS17] Dan Garber, Ohad Shamir, and Nathan Srebro. Communication-efficient algorithms for distributed stochastic principal component analysis. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2017.
- [HJS01] Marian Hristache, Anatoli Juditsky, and Vladimir Spokoiny. Direct estimation of the index coefficient in a single-index model. *The Annals of Statistics*, pages 595–623, 2001.
- [Hor09] Joel L Horowitz. *Semiparametric and nonparametric methods in econometrics*, volume 12. Springer, 2009.
- [Hot33] Harold Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24(6):417–441, 1933.
- [Jeff67] JNR Jeffers. Two case studies in the application of principal component analysis. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 16(3):225–236, 1967.
- [JLY19] Michael I Jordan, Jason D Lee, and Yun Yang. Communication-efficient distributed statistical inference. *Journal of the American Statistical Association*, 114(526):668–681, 2019.
- [Jol82] Ian T Jolliffe. A note on the use of principal components in regression. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 31(3):300–303, 1982.
- [JSA14] Majid Janzamin, Hanie Sedghi, and Anima Anandkumar. Score function features for discriminative learning: Matrix and tensor framework. *arXiv preprint arXiv:1412.2863*, 2014.
- [JZ13] Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in Neural Information Processing Systems (NIPS)*, 2013.
- [Li92] Ker-Chau Li. On principal hessian directions for data visualization and dimension reduction: Another application of Stein’s lemma. *Journal of the American Statistical Association*, 87(420):1025–1039, 1992.
- [LLST17] Jason D Lee, Qiang Liu, Yuekai Sun, and Jonathan E Taylor. Communication-efficient sparse regression. *Journal of Machine Learning Research*, 18(5):1–30, 2017.
- [LT13] Michel Ledoux and Michel Talagrand. *Probability in Banach Spaces: isoperimetry and processes*. Springer Science & Business Media, 2013.
- [MW15] Tengyu Ma and Avi Wigderson. Sum-of-squares lower bounds for sparse PCA. In *Advances in Neural Information Processing Systems (NIPS)*, 2015.
- [Pea01] Karl Pearson. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572, 1901.
- [RH15] Phillippe Rigollet and Jan-Christian Hütter. High dimensional statistics. *Lecture notes for course 18S997*, 2015.
- [Sha16] Ohad Shamir. Fast stochastic algorithms for SVD and PCA: Convergence properties and convexity. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2016.
- [SLS18] Chengchun Shi, Wenbin Lu, and Rui Song. A massive data framework for M-estimators with cubic-rate. *Journal of American Statistical Association*, 113(524):1698–1709, 2018.
- [SSZ14] Ohad Shamir, Nathan Srebro, and Tong Zhang. Communication efficient distributed optimization using an approximate newton-type method. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2014.
- [Ste81] Charles M Stein. Estimation of the mean of a multivariate normal distribution. *The Annals of Statistics*, 9(6):1135–1151, 1981.

- [T⁺15] Joel A Tropp et al. An introduction to matrix concentration inequalities. *Foundations and Trends® in Machine Learning*, 8(1-2):1–230, 2015.
- [VCC19] Stanislav Volgushev, Shih-Kang Chao, and Guang Cheng. Distributed inference for quantile regression processes. *The Annals of Statistics*, 47(3):1634–1662, 2019.
- [VCLR13] Vincent Q Vu, Juhee Cho, Jing Lei, and Karl Rohe. Fantope projection and selection: A near-optimal convex relaxation of sparse PCA. In *Advances in Neural Information Processing Systems (NIPS)*, 2013.
- [Ver12] Roman Vershynin. Introduction to the non-asymptotic analysis of random matrices. *Compressed Sensing*, pages 210–268, 2012.
- [VLG12] CF Van Loan and GH Golub. *Matrix Computations*. Johns Hopkins University Press, 3rd edition, 2012.
- [WY13] Zaiwen Wen and Wotao Yin. A feasible method for optimization with orthogonality constraints. *Mathematical Programming*, 142:397–434, 2013.
- [WYCL19] Xiaozhou Wang, Zhuoyi Yang, Xi Chen, and Weidong Liu. Distributed inference for linear support vector machine. *Journal of Machine Learning Research*, 20(113):1–41, 2019.
- [Xu18] Zhiqiang Xu. Gradient descent meets shift-and-invert preconditioning for eigenvector computation. In *Advances in Neural Information Processing Systems (NIPS)*, 2018.
- [YBL17] Zhuoran Yang, Krishnakumar Balasubramanian, and Han Liu. On Stein’s identity and near-optimal estimation in high-dimensional index models. *arXiv preprint arXiv:1709.08795*, 2017.
- [YWS14] Yi Yu, Tengyao Wang, and Richard J Samworth. A useful variant of the davis–kahan theorem for statisticians. *Biometrika*, 102(2):315–323, 2014.
- [ZCL16] Tianqi Zhao, Guang Cheng, and Han Liu. A partially linear framework for massive heterogeneous data. *The Annals of Statistics*, 44(4):1400–1437, 2016.
- [ZDW15] Yuchen Zhang, John Duchi, and Martin Wainwright. Divide and conquer kernel ridge regression: A distributed algorithm with minimax optimal rates. *Journal of Machine Learning Research*, 16:3299–3340, 2015.

A Proofs of Distributed PCA

A.1 Technical lemmas

We start with a useful result.

Lemma 3. *Let A, B be two matrices with orthonormal columns such that for some $\eta > 0$,*

$$\left\| A^\top B^\perp \right\|_2 \leq \eta$$

where B^\perp denotes a matrix whose columns consist an orthonormal basis of $\text{span}(B)^\perp$. Then there exists a matrix Q , $\|Q\|_2 \leq 1$, such that

$$\|A - BQ\|_2 \leq \eta$$

Proof. Note that $\|A^\top B^\perp\|_2 \leq \eta$ implies $A^\top BB^\top A^\top = I - A^\top B^\perp \succeq (1 - \eta)I$. Therefore, the claimed result is a consequence by applying Proposition B.1 in [AZL16]. \square

Next we provide a standard lemma to justify the claims that $\|\widehat{\Sigma} - \widehat{\Sigma}_1\|_2 = \mathcal{O}_p(\sqrt{d/m})$ and initial estimator conditions in Theorem 1.

Lemma 4. *If our samples $\mathbf{a}_i, i = 1, \dots, n$ are sub-Gaussian(σ) vectors, then with high probability, we have,*

$$\left\| \widehat{\Sigma} - \widehat{\Sigma}_1 \right\|_2 = \mathcal{O}\left(\sqrt{\frac{d}{m}}\right) \quad (30)$$

The top eigenvalue on the first machine $\lambda_1(\mathbf{A}^\top \mathbf{A}/m)$ satisfies,

$$|\widehat{\lambda}_1 - \lambda_1(\mathbf{A}^\top \mathbf{A}/m)| \leq \left\| \widehat{\Sigma} - \widehat{\Sigma}_1 \right\|_2 \quad (31)$$

Furthermore, let $\mathbf{w}^{(0)}$ be the top eigenvector of $\widehat{\Sigma}_1$ on the first machine. We have the following gap-free concentration bound for $\mathbf{w}^{(0)}$ and $\widehat{\mathbf{u}}_1, \dots, \widehat{\mathbf{u}}_d$,

$$\sum_{l: \widehat{\lambda}_l \leq (1-\delta)\widehat{\lambda}_1} \left| \langle \widehat{\mathbf{u}}_l, \mathbf{w}^{(0)} \rangle \right|^2 \leq \frac{\left\| \widehat{\Sigma} - \widehat{\Sigma}_1 \right\|_2^2}{\delta \widehat{\lambda}_1} \quad (32)$$

Proof. By Corollary 5.50 in [Ver12], with probability at least $1 - 2e^{-d\sigma^2/C}$,

$$\left\| \widehat{\Sigma} - \Sigma \right\|_2 \leq \left\| \widehat{\Sigma} - \Sigma \right\|_{\text{F}} \leq \sqrt{\frac{d}{n}}\sigma$$

and

$$\left\| \widehat{\Sigma}_1 - \Sigma \right\|_2 \leq \left\| \widehat{\Sigma}_1 - \Sigma \right\|_{\text{F}} \leq \sqrt{\frac{d}{m}}\sigma$$

where C is a constant which only depends on the sub-Gaussian norm of the random vector \mathbf{a} . Therefore, our first inequality (30) is a direct result of above matrix concentrations as well as triangle inequality for matrix spectral norm.

Denote $\hat{\mathbf{u}}_1$ and $\mathbf{w}^{(0)}$ to be the top eigenvector for $\widehat{\Sigma}$ and $\widehat{\Sigma}_1$, without loss of generality, we can assume $\widehat{\lambda}_1 > \lambda_1(\mathbf{A}_1^\top \mathbf{A}_1/m)$, then we have

$$\left| \widehat{\lambda}_1 - \lambda_1(\mathbf{A}_1^\top \mathbf{A}_1/m) \right| = \widehat{\mathbf{u}}_1^\top \widehat{\Sigma} \widehat{\mathbf{u}}_1 - \mathbf{w}^{(0)\top} \widehat{\Sigma}_1 \mathbf{w}^{(0)} \leq \widehat{\mathbf{u}}_1^\top \widehat{\Sigma} \widehat{\mathbf{u}}_1 - \widehat{\mathbf{u}}_1^\top \widehat{\Sigma}_1 \widehat{\mathbf{u}}_1 \leq \left\| \widehat{\Sigma} - \widehat{\Sigma}_1 \right\|_2$$

With Davis-Kahan Theorem [YWS14], it is easy to see,

$$\sum_{l: \widehat{\lambda}_l \leq (1-\delta) \widehat{\lambda}_1} \left| \langle \widehat{\mathbf{u}}_l, \mathbf{w}^{(0)} \rangle \right|^2 \leq \frac{\left\| \widehat{\Sigma} - \widehat{\Sigma}_1 \right\|_2}{\delta \widehat{\lambda}_1}$$

□

A.2 Proofs of distributed top eigenvector estimation

Proof of Lemma 1

Proof. Write $\mathbf{v} = \mathbf{H}^{-1}\mathbf{w} + \mathbf{e}$, where $\|\mathbf{e}\|_2 \leq \varepsilon$. Since $\mathbf{H}^{-1} = (\bar{\lambda}_1 \mathbf{I} - \widehat{\Sigma})^{-1} = \sum_{l=1}^d (\bar{\lambda}_1 - \widehat{\lambda}_l)^{-1} \widehat{\mathbf{u}}_l \widehat{\mathbf{u}}_l^\top$, we have, for each $l = 1, \dots, d$,

$$\langle \widehat{\mathbf{u}}_l, \mathbf{v} \rangle = (\bar{\lambda}_1 - \widehat{\lambda}_l)^{-1} \langle \widehat{\mathbf{u}}_l, \mathbf{w} \rangle + \langle \widehat{\mathbf{u}}_l, \mathbf{e} \rangle \quad (33)$$

This can yield a lower bound on $\|\mathbf{v}\|_2$,

$$\begin{aligned} \|\mathbf{v}\|_2^2 &= \sum_{l=1}^d |\langle \widehat{\mathbf{u}}_l, \mathbf{v} \rangle|^2 \geq \sum_{l: \widehat{\lambda}_l > (1-\delta) \widehat{\lambda}_1} |\langle \widehat{\mathbf{u}}_l, \mathbf{v} \rangle|^2 \\ &\geq \sum_{l: \widehat{\lambda}_l > (1-\delta) \widehat{\lambda}_1} \left[\frac{1}{2} (\bar{\lambda}_1 - \widehat{\lambda}_l)^{-2} |\langle \widehat{\mathbf{u}}_l, \mathbf{w} \rangle|^2 - |\langle \widehat{\mathbf{u}}_l, \mathbf{e} \rangle|^2 \right] \\ &\geq (32\eta^2)^{-1} - \|\mathbf{e}\|_2^2 \geq (32\eta^2)^{-1} - \varepsilon^2 \geq (64\eta^2)^{-1} \end{aligned} \quad (34)$$

where we used the upper bound of $\bar{\lambda}_1 - \widehat{\lambda}_l$, and the conditions (12) and (13) in Lemma 1. On the other hand, for each l such that $\widehat{\lambda}_l \leq (1-\delta) \widehat{\lambda}_1$, we have $\bar{\lambda}_1 - \widehat{\lambda}_l \geq \widehat{\lambda}_1 - \widehat{\lambda}_l \geq \delta \widehat{\lambda}_1$. Consequently, equation (33) implies

$$|\langle \widehat{\mathbf{u}}_l, \mathbf{v} \rangle| \leq (\delta \widehat{\lambda}_1)^{-1} |\langle \widehat{\mathbf{u}}_l, \mathbf{w} \rangle| + |\langle \widehat{\mathbf{u}}_l, \mathbf{e} \rangle| \leq (\delta \widehat{\lambda}_1)^{-1} |\langle \widehat{\mathbf{u}}_l, \mathbf{w} \rangle| + \varepsilon$$

A combination of the last two displays yields the first claimed bound. Similarly, the second claim bound follows by combining inequality (34) with

$$\begin{aligned} \sum_{l: \widehat{\lambda}_l \leq (1-\delta) \widehat{\lambda}_1} |\langle \widehat{\mathbf{u}}_l, \mathbf{v} \rangle|^2 &\leq 2(\delta \widehat{\lambda}_1)^{-2} \sum_{l: \widehat{\lambda}_l \leq (1-\delta) \widehat{\lambda}_1} |\langle \widehat{\mathbf{u}}_l, \mathbf{w} \rangle|^2 + 2 \sum_{l: \widehat{\lambda}_l \leq (1-\delta) \widehat{\lambda}_1} |\langle \widehat{\mathbf{u}}_l, \mathbf{e} \rangle|^2 \\ &\leq 2(\delta \widehat{\lambda}_1)^{-2} \sum_{l: \widehat{\lambda}_l \leq (1-\delta) \widehat{\lambda}_1} |\langle \widehat{\mathbf{u}}_l, \mathbf{w} \rangle|^2 + 2\varepsilon^2 \end{aligned}$$

□

Proof of Lemma 2

Proof. It is easy to verify the following two identities:

$$\begin{aligned}\tilde{\mathbf{w}}^{(t+1)} &= \mathbf{w}_j^{(t+1)} - \mathbf{H}^{-1}(\mathbf{H}\mathbf{w}_j^{(t+1)} - \mathbf{w}^{(t)}), \\ \mathbf{w}_{j+1}^{(t+1)} &= \mathbf{w}_j^{(t+1)} - \mathbf{H}_1^{-1}(\mathbf{H}\mathbf{w}_j^{(t+1)} - \mathbf{w}^{(t)})\end{aligned}$$

By taking the difference we obtain

$$\begin{aligned}\|\mathbf{w}_{j+1}^{(t+1)} - \tilde{\mathbf{w}}^{(t+1)}\|_2 &= \|(\mathbf{H}^{-1} - \mathbf{H}_1^{-1})(\mathbf{H}\mathbf{w}_j^{(t+1)} - \mathbf{w}^{(t)})\|_2 \\ &= \|(\mathbf{I} - \mathbf{H}_1^{-1}\mathbf{H})(\mathbf{w}_j^{(t+1)} - \mathbf{H}^{-1}\mathbf{w}^{(t)})\|_2 \\ &\leq \|\mathbf{I} - \mathbf{H}_1^{-1}\mathbf{H}\|_2 \|\mathbf{w}_j^{(t+1)} - \tilde{\mathbf{w}}^{(t+1)}\|_2\end{aligned}$$

We bound the first factor on the r.h.s. as

$$\|\mathbf{I} - \mathbf{H}_1^{-1}\mathbf{H}\|_2 \leq \|\mathbf{H}_1^{-1}\|_2 \|\mathbf{H}_1 - \mathbf{H}\|_2 \leq \frac{2}{\eta} \|\widehat{\Sigma} - \widehat{\Sigma}_1\|_2$$

where the last step follows from the fact $\mathbf{H}_1 - \mathbf{H} = -(\widehat{\Sigma} - \widehat{\Sigma}_1)$, and the inequality

$$\begin{aligned}\|\mathbf{H}_1^{-1}\|_2^{-1} &= \lambda_{\min}(\mathbf{H}_1) = \lambda_{\min}(\bar{\lambda}_1 \mathbf{I} - \widehat{\Sigma} + (\widehat{\Sigma} - \widehat{\Sigma}_1)) \geq \lambda_{\min}(\bar{\lambda}_1 \mathbf{I} - \widehat{\Sigma}) - \|\widehat{\Sigma} - \widehat{\Sigma}_1\|_2 \\ &\geq \eta - \|\widehat{\Sigma} - \widehat{\Sigma}_1\|_2 \geq \eta/2\end{aligned}$$

where $\lambda_{\min}(\mathbf{A})$ denotes the smallest singular value of symmetric matrix \mathbf{A} . □

Proof of Theorem 1

Proof. Applying Lemma 2, we have (recall $\mathbf{w}_0^{t+1} = \mathbf{w}_{T'}^t$)

$$\|\mathbf{w}_{T'}^{t+1} - \mathbf{H}^{-1}\mathbf{w}_{T'}^t\|_2 \leq \left(\frac{2\kappa}{\eta}\right)^{T'} \|\mathbf{w}_{T'}^t - \mathbf{H}^{-1}\mathbf{w}_{T'}^t\|_2 \leq \left(\frac{2\kappa}{\eta}\right)^{T'} \frac{2}{\eta} := \varepsilon_{T'}$$

where we used the fact that $\|\mathbf{I} - \mathbf{H}^{-1}\|_2 \leq 1 + \|\mathbf{H}^{-1}\|_2 = 1 + (\bar{\lambda}_1 - \widehat{\lambda}_1)^{-1} \leq 2/\eta$, and $\|\mathbf{w}_{T'}^t\|_2 = 1$. Now we can recursively apply inequality (15) with $\varepsilon \leftarrow \varepsilon_{T'}$ to obtain

$$\begin{aligned}\sum_{l: \widehat{\lambda}_l \leq (1-\delta)\widehat{\lambda}_1} |\langle \widehat{\mathbf{u}}_l, \mathbf{w}^{(t)} \rangle|^2 &\leq \frac{128\eta^2}{\delta^2\widehat{\lambda}_1^2} \sum_{l: \widehat{\lambda}_l \leq (1-\delta)\widehat{\lambda}_1} |\langle \widehat{\mathbf{u}}_l, \mathbf{w}^{(t-1)} \rangle|^2 + 128\eta^2\varepsilon_{T'}^2 \\ &\leq \left(\frac{128\eta^2}{\delta^2\widehat{\lambda}_1^2}\right)^2 \sum_{l: \widehat{\lambda}_l \leq (1-\delta)\widehat{\lambda}_1} |\langle \widehat{\mathbf{u}}_l, \mathbf{w}^{(t-2)} \rangle|^2 + \left(1 + \frac{128\eta^2}{\delta^2\widehat{\lambda}_1^2}\right) 128\eta^2\varepsilon_{T'}^2 \\ &\leq \dots \leq \left(\frac{128\eta^2}{\delta^2\widehat{\lambda}_1^2}\right)^T \sum_{l: \widehat{\lambda}_l \leq (1-\delta)\widehat{\lambda}_1} |\langle \widehat{\mathbf{u}}_l, \mathbf{w}^{(0)} \rangle|^2 + 128\eta^2\varepsilon_{T'}^2 \sum_{t=0}^T \left(\frac{128\eta^2}{\delta^2\widehat{\lambda}_1^2}\right)^t \\ &\leq \left(\frac{128\eta^2}{\delta^2\widehat{\lambda}_1^2}\right)^T + \frac{128\eta^2\varepsilon_{T'}^2}{1 - 128\eta^2/(\delta\widehat{\lambda}_1)^2}\end{aligned}$$

□

A.3 Proofs of distributed top- L -dim principal subspace estimation

Proof of Theorem 2

Proof. Our proof adapts the proof of Theorem 4.1(a) in [AZL16] to our settings.

Let $\widehat{\boldsymbol{\mu}} = \left\| \widehat{\boldsymbol{\Sigma}}^{(L-1)} \right\|_2$. Due to the Courant minimax principle, we have $\mu \geq \widehat{\lambda}_L$.

Note that column vectors in \mathbf{V}_l are already eigenvectors of $\widehat{\boldsymbol{\Sigma}}^{(l)}$ with eigenvalues zero. Let \mathbf{W}_l be column orthogonal matrix whose columns are eigenvectors in \mathbf{V}_l^\perp of $\widehat{\boldsymbol{\Sigma}}^{(l)}$ with eigenvalues in the range $[0, (1 - \delta + \tau_l) \widehat{\boldsymbol{\mu}}]$, where $\tau_l = \frac{l}{2L} \delta$, for $l = 0, 1, \dots, L$.

We will show that for each $l = 0, 1, \dots, L$, there exists a matrix \mathbf{Q}_l such that

$$\left\| \widehat{\mathbf{U}}_{\leq(1-\delta)\widehat{\lambda}_L} - \mathbf{W}_l \mathbf{Q}_l \right\|_2 \leq \varepsilon_l \in [0, 1) \quad \text{and} \quad \|\mathbf{Q}_l\|_2 \leq 1 \quad (35)$$

for some sequence $\{\varepsilon_l\}_{l=1}^L$ of small numbers. This would imply our claimed bound. In fact, the first inequality in the preceding display implies $\left\| \mathbf{I} - \widehat{\mathbf{U}}_{\leq(1-\delta)\widehat{\lambda}_L}^\top \mathbf{W}_L \mathbf{Q}_L \right\|_2 \leq \varepsilon_l$. Therefore, the smallest singular value of $\widehat{\mathbf{U}}_{\leq(1-\delta)\widehat{\lambda}_L}^\top \mathbf{W}_L \mathbf{Q}_L$ is at least $1 - \varepsilon_L > 0$. This lower bound combined with $\|\mathbf{Q}_L\|_2 \leq 1$ implies the smallest singular value of $\widehat{\mathbf{U}}_{\leq(1-\delta)\widehat{\lambda}_L}^\top \mathbf{W}_L$ to be at least $1 - \varepsilon_L$, or

$$\mathbf{I} - \widehat{\mathbf{U}}_{\leq(1-\delta)\widehat{\lambda}_L}^\top \mathbf{W}_L \mathbf{W}_L^\top \widehat{\mathbf{U}}_{\leq(1-\delta)\widehat{\lambda}_L} \preceq 1 - (1 - \varepsilon_L)^2 \mathbf{I}$$

Now since \mathbf{V}_L is (column) orthogonal to \mathbf{W}_L , we obtain

$$\widehat{\mathbf{U}}_{\leq(1-\delta)\widehat{\lambda}_L}^\top \mathbf{V}_L \mathbf{V}_L^\top \widehat{\mathbf{U}}_{\leq(1-\delta)\widehat{\lambda}_L} \preceq \widehat{\mathbf{U}}_{\leq(1-\delta)\widehat{\lambda}_L}^\top (\mathbf{I} - \mathbf{W}_L \mathbf{W}_L^\top) \widehat{\mathbf{U}}_{\leq(1-\delta)\widehat{\lambda}_L} \preceq 2\varepsilon_L \mathbf{I}$$

which implies $\left\| \widehat{\mathbf{U}}_{\leq(1-\delta)\widehat{\lambda}_L}^\top \mathbf{V}_L \right\|_2 \leq \sqrt{2\varepsilon_L}$.

When $l = 0$, we simply choose $\mathbf{W}_0 = \widehat{\mathbf{U}}_{\leq(1-\delta)\widehat{\lambda}_L}$, $\varepsilon_0 = 0$ and $\mathbf{Q}_0 = \mathbf{I}$. Suppose for every $l \in \{0, \dots, L-1\}$, there exists a matrix \mathbf{Q}_l with $\|\mathbf{Q}_l\|_2 \leq 1$ satisfying $\left\| \widehat{\mathbf{U}}_{\leq(1-\delta)\widehat{\lambda}_L} - \mathbf{W}_l \mathbf{Q}_l \right\|_2 \leq \varepsilon_l$ for some $\varepsilon_l \in [0, 1)$. Now we construct \mathbf{Q}_{l+1} as follows.

Since $\kappa = \left\| \widehat{\boldsymbol{\Sigma}} - \widehat{\boldsymbol{\Sigma}}_1 \right\|_2 \geq \left\| (\mathbf{I} - \mathbf{V}_l \mathbf{V}_l^\top) (\widehat{\boldsymbol{\Sigma}} - \widehat{\boldsymbol{\Sigma}}_1) (\mathbf{I} - \mathbf{V}_l \mathbf{V}_l^\top) \right\|_2$, we can apply Theorem 1 to \mathbf{w}_{l+1} with $\delta \leftarrow \delta/2$ to obtain (note that columns of \mathbf{W}_l and \mathbf{V}_l corresponds to eigenvectors of $\widehat{\boldsymbol{\Sigma}}^{(l)}$ with eigenvectors less than or equal to $(1 - \delta + \tau_l) \left\| \widehat{\boldsymbol{\Sigma}}^{(L-1)} \right\|_2 \leq (1 - \delta/2) \left\| \widehat{\boldsymbol{\Sigma}}^{(l)} \right\|_2$)

$$\|\mathbf{W}_l^\top \mathbf{w}_{l+1}\|_2^2 \leq \varepsilon_{T,T'}^{(l)} \quad \text{and} \quad \|\mathbf{V}_l^\top \mathbf{w}_{l+1}\|_2^2 \leq \varepsilon_{T,T'}^{(l)}$$

where $\varepsilon_{T,T'}^{(l)} := \left(\frac{128\eta^2}{\delta^2 \lambda_1(\widehat{\boldsymbol{\Sigma}}^{(l)})^2} \right)^T + \frac{512\eta}{1 - 128\eta^2 / (\delta \lambda_1(\widehat{\boldsymbol{\Sigma}}^{(l)}))^2} \left(\frac{4\kappa^2}{\eta^2} \right)^T \leq 1/2$. Since \mathbf{v}_{l+1} is the projection of \mathbf{w}_{l+1} into \mathbf{V}_l^\perp , we have

$$\|\mathbf{W}_l^\top \mathbf{v}_{l+1}\|_2^2 \leq \frac{\|\mathbf{W}_l^\top \mathbf{w}_{l+1}\|_2^2}{\|(\mathbf{I} - \mathbf{V}_l \mathbf{V}_l^\top) \mathbf{w}_{l+1}\|_2^2} \leq \frac{\varepsilon_{T,T'}^{(l)}}{1 - \varepsilon_{T,T'}^{(l)}} \leq \frac{3}{2} \varepsilon_{T,T'}^{(l)}$$

We will make use of the following lemma from [AZL16] (Lemma B.4).

Lemma 5 (Eigen-space perturbation lemma). *Let $\mathbf{M} \in \mathbb{R}^{d \times d}$ be a positive semidefinite matrix with eigenvalues $\lambda_1 \geq \dots \geq \lambda_r \geq \lambda_{r+1} = \dots = \lambda_d = 0$, and corresponding eigenvectors $\mathbf{u}_1, \dots, \mathbf{u}_d$. Define $\mathbf{U} = [\mathbf{u}_{j+1}, \dots, \mathbf{u}_r] \in \mathbb{R}^{d \times (r-j)}$ to be the matrix composing of all top r -eigenvectors with eigenvalues less than or equal to μ . Let $\mathbf{v} \in \mathbb{R}^d$ be a unit vector such that $\|\mathbf{U}^\top \mathbf{v}\|_2 \leq \varepsilon \leq 1/2$, $\mathbf{v}^\top \mathbf{u}_{r+1} = \dots = \mathbf{v}^\top \mathbf{u}_d = 0$. Define*

$$\widetilde{\mathbf{M}} = (\mathbf{I} - \mathbf{v}\mathbf{v}^\top) \mathbf{M} (\mathbf{I} - \mathbf{v}\mathbf{v}^\top)$$

Denote $[\widetilde{\mathbf{V}}, \widetilde{\mathbf{U}}, \mathbf{v}, \mathbf{u}_{r+1}, \dots, \mathbf{u}_d] \in \mathbb{R}^{d \times d}$ as the orthogonal matrix composed of eigenvectors of $\widetilde{\mathbf{M}}$, where $\widetilde{\mathbf{U}}$ consists of all eigenvectors (other than $\mathbf{v}, \mathbf{u}_{r+1}, \dots, \mathbf{u}_d$) with eigenvalues less than or equal to $\mu + \tau$. Then there exists a matrix \mathbf{Q} such that $\|\mathbf{Q}\|_2 \leq 1$ and

$$\left\| \mathbf{U} - \widetilde{\mathbf{U}} \mathbf{Q} \right\|_2 \leq \sqrt{\frac{169\lambda_1^2\varepsilon^2}{\tau^2} + \varepsilon^2}$$

By applying Lemma 5 with $\mathbf{M} = \widehat{\Sigma}^{(l)}$,
 $\widetilde{\mathbf{M}} = \widehat{\Sigma}^{(l+1)} = (\mathbf{I} - \mathbf{v}_{l+1}\mathbf{v}_{l+1}^\top)\widehat{\Sigma}^{(l)}(\mathbf{I} - \mathbf{v}_{l+1}\mathbf{v}_{l+1}^\top)$, $r = d - l$, $\mathbf{V} = \mathbf{W}_l$, $\widetilde{\mathbf{V}} = \mathbf{W}_{l+1}$, $\mathbf{v} = \mathbf{v}_{l+1}$,
 $\mu = (1 - \delta + \tau_l)\widehat{\mu}$, $\tau = (\tau_{l+1} - \tau_l)\widehat{\mu}$, we obtain a matrix $\widetilde{\mathbf{Q}}_l$ such that $\|\widetilde{\mathbf{Q}}_l\|_2 \leq 1$ and

$$\left\| \mathbf{W}_l - \mathbf{W}_{l+1} \widetilde{\mathbf{Q}}_l \right\|_2 \leq \sqrt{\frac{507\widehat{\lambda}_1^2\varepsilon_{T,T'}^{(l)}}{2(\tau_{s+1} - \tau_s)^2\widehat{\mu}^2} + \frac{3}{2}\varepsilon_{T,T'}^{(l)}} \leq \frac{32\widehat{\lambda}_1 L \sqrt{\varepsilon_{T,T'}^{(l)}}}{\widehat{\lambda}_L \delta}$$

This inequality combined with inequality (35) together implies

$$\begin{aligned} \left\| \mathbf{W}_{l+1} \widetilde{\mathbf{Q}}_l \mathbf{Q}_l - \widehat{\mathbf{U}}_{\leq(1-\delta)\widehat{\lambda}_L} \right\|_2 &\leq \left\| \mathbf{W}_{l+1} \widetilde{\mathbf{Q}}_l \mathbf{Q}_l - \mathbf{W}_l \mathbf{Q}_l \right\|_2 + \left\| \mathbf{W}_l \mathbf{Q}_l - \widehat{\mathbf{U}}_{\leq(1-\delta)\widehat{\lambda}_L} \right\|_2 \\ &\leq \varepsilon_l + \frac{32\widehat{\lambda}_1 L \sqrt{\varepsilon_{T,T'}^{(l)}}}{\widehat{\lambda}_L \delta} \end{aligned}$$

By defining $\mathbf{Q}_{l+1} = \widetilde{\mathbf{Q}}_l \mathbf{Q}_l$, we obtain

$$\begin{aligned} \left\| \widehat{\mathbf{U}}_{\leq(1-\delta)\widehat{\lambda}_L} - \mathbf{W}_{l+1} \mathbf{Q}_{l+1} \right\|_2 &\leq \varepsilon_{l+1} := \varepsilon_l + \frac{32\widehat{\lambda}_1 L \sqrt{\varepsilon_{T,T'}^{(l)}}}{\widehat{\lambda}_L \delta} = \dots \\ &= \sum_{k=0}^{l+1} \frac{32\widehat{\lambda}_1 L \sqrt{\varepsilon_{T,T'}^{(k)}}}{\widehat{\lambda}_L \delta} \\ &\leq (l+1) \frac{32\widehat{\lambda}_1 L \sqrt{\varepsilon_{T,T'}^{(l)}}}{\widehat{\lambda}_L \delta} \\ &\leq (l+1) \frac{32\widehat{\lambda}_1 L}{\widehat{\lambda}_L \delta} \sqrt{\left(\frac{128\eta^2}{\delta^2 \widehat{\lambda}_k^2}\right)^T + \frac{512\eta}{1 - 128\eta^2/(\delta\widehat{\lambda}_k)^2} \left(\frac{4\kappa^2}{\eta^2}\right)^T} \end{aligned}$$

for $l = 0, 1, \dots, L$.

□

Proof of Corollary 3

Please refer to Theorem 4.1 in [AZL16].

Proof of Corollary 4

Proof. Notice that $\widehat{\mathbf{U}}_{>(1-\delta)\widehat{\lambda}_L} \widehat{\mathbf{U}}_{>(1-\delta)\widehat{\lambda}_L}^\top + \widehat{\mathbf{U}}_{\leq(1-\delta)\widehat{\lambda}_L} \widehat{\mathbf{U}}_{\leq(1-\delta)\widehat{\lambda}_L}^\top = \mathbf{I}_d$, we have

$$\begin{aligned} \left\| \mathbf{V}_L^\top \mathbf{U}_{\leq(1-2\delta)\lambda_L} \right\|_2 &= \left\| \mathbf{V}_L^\top \left(\widehat{\mathbf{U}}_{>(1-\delta)\widehat{\lambda}_L} \widehat{\mathbf{U}}_{>(1-\delta)\widehat{\lambda}_L}^\top + \widehat{\mathbf{U}}_{\leq(1-\delta)\widehat{\lambda}_L} \widehat{\mathbf{U}}_{\leq(1-\delta)\widehat{\lambda}_L}^\top \right) \mathbf{U}_{\leq(1-2\delta)\lambda_L} \right\|_2 \\ &\leq \left\| \mathbf{V}_L^\top \widehat{\mathbf{U}}_{>(1-\delta)\widehat{\lambda}_L} \widehat{\mathbf{U}}_{>(1-\delta)\widehat{\lambda}_L}^\top \mathbf{U}_{\leq(1-2\delta)\lambda_L} \right\|_2 + \left\| \mathbf{V}_L^\top \widehat{\mathbf{U}}_{\leq(1-\delta)\widehat{\lambda}_L} \widehat{\mathbf{U}}_{\leq(1-\delta)\widehat{\lambda}_L}^\top \mathbf{U}_{\leq(1-2\delta)\lambda_L} \right\|_2 \\ &\leq \left\| \widehat{\mathbf{U}}_{>(1-\delta)\widehat{\lambda}_L}^\top \mathbf{U}_{\leq(1-2\delta)\lambda_L} \right\|_2 + \varepsilon \\ &\leq \frac{\left\| \boldsymbol{\Sigma} - \widehat{\boldsymbol{\Sigma}} \right\|_2}{(1-\delta)(\widehat{\lambda}_L - \lambda_L) + \delta\lambda_L} + \varepsilon \end{aligned}$$

where the last inequality follows from the Gap-free Wedin Theorem [AZL16, Lemma B.3]. \square

Proof of Theorem 3

Proof. Denote $\widetilde{\boldsymbol{\Sigma}}^{(S)} = \widehat{\boldsymbol{\Sigma}}^{(S)} + \sum_{j=1}^S \widehat{\lambda}_j \mathbf{v}_j \mathbf{v}_j^\top$ where $\widehat{\boldsymbol{\Sigma}}^{(S)} = (I - \mathbf{V}_S \mathbf{V}_S^\top) \widehat{\boldsymbol{\Sigma}} (I - \mathbf{V}_S \mathbf{V}_S^\top)$. Then by Corollary 3, $\mathbf{V}_{(1-\delta)\widehat{\lambda}_L}$ is the eigenspace of $\widetilde{\boldsymbol{\Sigma}}^{(S)}$ with eigenvalues less than or equal to $(1-\delta_S)\widehat{\lambda}_S$, and $\widehat{\mathbf{U}}_L$ is the eigenspace of $\widehat{\boldsymbol{\Sigma}}$ with eigenvalues greater than or equal to $\widehat{\lambda}_L$, by Gap-free Wedin Theorem [AZL16, Lemma B.3], we have

$$\left\| \widehat{\mathbf{U}}_L^\top \mathbf{V}_{(1-\delta)\widehat{\lambda}_L} \right\|_2 \leq \frac{\left\| \widetilde{\boldsymbol{\Sigma}}^{(S)} - \widehat{\boldsymbol{\Sigma}} \right\|_2}{\widehat{\lambda}_L - \widehat{\lambda}_S / (1-\delta_S)}$$

When $\widehat{\lambda}_L - \widehat{\lambda}_S / (1-\delta_S)$ (given by $\delta > \delta_S$), we only need to bound $\left\| \widetilde{\boldsymbol{\Sigma}}^{(S)} - \widehat{\boldsymbol{\Sigma}} \right\|_2$. Towards this goal, we first bound the following quantities, $\mathbf{v}_j^\top (\widetilde{\boldsymbol{\Sigma}}^{(S)} - \widehat{\boldsymbol{\Sigma}}) \mathbf{v}_k$, $\mathbf{v}_j^\top (\widetilde{\boldsymbol{\Sigma}}^{(S)} - \widehat{\boldsymbol{\Sigma}}) \mathbf{v}$ and $\mathbf{v}^\top (\widetilde{\boldsymbol{\Sigma}}^{(S)} - \widehat{\boldsymbol{\Sigma}}) \mathbf{v}$, for all $j, k \in [S]$ and for $\mathbf{v} \in \mathbf{V}_S^\perp$:

- (1) Term $\mathbf{v}^\top (\widetilde{\boldsymbol{\Sigma}}^{(S)} - \widehat{\boldsymbol{\Sigma}}) \mathbf{v}$ for $\mathbf{v} \in \mathbf{V}_S^\perp$: a simple calculation yields $\mathbf{v}^\top (\widetilde{\boldsymbol{\Sigma}}^{(S)} - \widehat{\boldsymbol{\Sigma}}) \mathbf{v} = \mathbf{v}^\top (\widehat{\boldsymbol{\Sigma}} - \widehat{\boldsymbol{\Sigma}}) \mathbf{v} = 0$.
- (2) Term $\mathbf{v}_j^\top (\widetilde{\boldsymbol{\Sigma}}^{(S)} - \widehat{\boldsymbol{\Sigma}}) \mathbf{v}_k$ for $j = k \in [S]$: we have

$$\left| \mathbf{v}_j^\top (\widetilde{\boldsymbol{\Sigma}}^{(S)} - \widehat{\boldsymbol{\Sigma}}) \mathbf{v}_j \right| = \left| \widehat{\lambda}_j - \mathbf{v}_j^\top \widehat{\boldsymbol{\Sigma}} \mathbf{v}_j \right| \leq \frac{\delta_S}{1-\delta_S} \widehat{\lambda}_j$$

where the last step is due to Theorem 4.1(c) in [AZL16].

- (3) Term $\mathbf{v}_j^\top (\widetilde{\boldsymbol{\Sigma}}^{(S)} - \widehat{\boldsymbol{\Sigma}}) \mathbf{v}_k$ for $1 \leq j < k \leq [S]$: according to our construction, we have $\mathbf{v}_u^\top \mathbf{v}_v = 0$ for each pair $u, v \in [S]$ and $u \neq v$. Thus we have $\mathbf{v}_j^\top \widetilde{\boldsymbol{\Sigma}}^{(S)} \mathbf{v}_k = \sum_{s=1}^S \widehat{\lambda}_s \mathbf{v}_j^\top \mathbf{v}_s \mathbf{v}_s^\top \mathbf{v}_k = 0$ and the following inequality

$$\left| \mathbf{v}_j^\top (\widetilde{\boldsymbol{\Sigma}}^{(S)} - \widehat{\boldsymbol{\Sigma}}) \mathbf{v}_k \right| = \left| \mathbf{v}_j^\top \widehat{\boldsymbol{\Sigma}} \mathbf{v}_k \right| = \left| \mathbf{v}_j^\top \widehat{\boldsymbol{\Sigma}} \mathbf{v}_k - \widehat{\lambda}_k \mathbf{v}_j^\top \mathbf{v}_k \right| \leq \left\| (\widehat{\boldsymbol{\Sigma}} - \widehat{\lambda}_k \mathbf{I}_d) \mathbf{v}_k \right\|_2$$

Recall that $\widehat{\Sigma}^{(k)} = (\mathbf{I} - V_{k-1}V_{k-1}^\top)\widehat{\Sigma}(\mathbf{I} - V_{k-1}V_{k-1}^\top)$ and $\mathbf{v}_k \perp V_{k-1}$. Therefore, $\mathbf{v}_k^\top \widehat{\Sigma} \mathbf{v}_k = \mathbf{v}_k^\top \widehat{\Sigma}^{(k)} \mathbf{v}_k$, and

$$\begin{aligned} \left\| (\widehat{\Sigma} - \widehat{\lambda}_k \mathbf{I}_d) \mathbf{v}_k \right\|_2^2 &= \mathbf{v}_k^\top (\widehat{\Sigma} - \widehat{\lambda}_k \mathbf{I}_d)^2 \mathbf{v}_k = \mathbf{v}_k^\top \widehat{\Sigma}^2 \mathbf{v}_k - 2\widehat{\lambda}_k \mathbf{v}_k^\top \widehat{\Sigma} \mathbf{v}_k + \widehat{\lambda}_k \|\mathbf{v}_k\|_2^2 \\ &= \mathbf{v}_k^\top \widehat{\Sigma}^{(k)2} \mathbf{v}_k - 2\widehat{\lambda}_k \mathbf{v}_k^\top \widehat{\Sigma}^{(k)} \mathbf{v}_k + \widehat{\lambda}_k \|\mathbf{v}_k\|_2^2 = \mathbf{v}_k^\top (\widehat{\Sigma}^{(k)} - \widehat{\lambda}_k \mathbf{I}_d)^2 \mathbf{v}_k \end{aligned} \quad (36)$$

According to Theorem 4.1(b) in [AZL16], we have

$$\widehat{\lambda}_k \leq \left\| \widehat{\Sigma}^{(k)} \right\|_2 \leq \frac{\widehat{\lambda}_k}{1 - \delta}$$

. Let $\widehat{\lambda}_1^{(k)} \geq \widehat{\lambda}_2^{(k)} \geq \dots \geq \widehat{\lambda}_d^{(k)} \geq 0$ denote the sorted eigenvalues of $\widehat{\Sigma}^{(k)}$, and $\widehat{\mathbf{u}}_j^{(k)}$, $j \in [d]$ the associated eigenvectors. From the preceding display, we have $\widehat{\lambda}_k \leq \widehat{\lambda}_1^{(k)} \leq (1 - \delta)^{-1} \widehat{\lambda}_k$. Simple calculations yield

$$\begin{aligned} \mathbf{v}_k^\top (\widehat{\Sigma}^{(k)} - \widehat{\lambda}_k \mathbf{I}_d)^2 \mathbf{v}_k &= \sum_{j=1}^d (\widehat{\lambda}_1^{(k)} - \widehat{\lambda}_j^{(k)})^2 (\mathbf{v}_k^\top \widehat{\mathbf{u}}_j^{(k)})^2 \\ &\leq \sum_{j: \widehat{\lambda}_j^{(k)} \leq (1 - \delta_S) \widehat{\lambda}_1^{(k)}} (\widehat{\lambda}_1^{(k)})^2 (\mathbf{v}_k^\top \widehat{\mathbf{u}}_j^{(k)})^2 + \sum_{j: \widehat{\lambda}_j^{(k)} > (1 - \delta_S) \widehat{\lambda}_1^{(k)}} \delta_S^2 (\widehat{\lambda}_1^{(k)})^2 (\mathbf{v}_k^\top \widehat{\mathbf{u}}_j^{(k)})^2 \\ &\leq \varepsilon (\widehat{\lambda}_1^{(k)})^2 + \delta_S^2 (\widehat{\lambda}_1^{(k)})^2 \leq 2\delta_S^2 (\widehat{\lambda}_1^{(k)})^2 \leq 2\delta_S^2 (1 - \delta)^{-2} \widehat{\lambda}_k^2 \end{aligned}$$

when ε is small enough, $\varepsilon \leq \delta_S^2$. Notice that for $2 \leq k \leq S$, the above result remains the same once we notice that $\mathbf{v}_k^\top \widehat{\Sigma}^{(k-1)} \mathbf{v}_k = \mathbf{v}_k^\top \widehat{\Sigma} \mathbf{v}_k$.

(4) Term $\mathbf{v}_j^\top (\widetilde{\Sigma}^{(S)} - \widehat{\Sigma}) \mathbf{v} = 0$ for $\mathbf{v} \in \mathbf{V}_S^\perp$ and $j \in [S]$.

Combine the result above, for any $\mathbf{x} \in \mathbb{R}^d$, $\|\mathbf{x}\|_2 = 1$, denote $\mathbf{x} = \sum_{j=1}^d \alpha_j \mathbf{v}_j$, $\sum_{j=1}^d \alpha_j^2 = 1$. We can show that,

$$\begin{aligned} \left| \mathbf{x}^\top (\widetilde{\Sigma}^{(S)} - \widehat{\Sigma}) \mathbf{x} \right| &= \left| \sum_{j,k=1}^d \alpha_j \alpha_k \mathbf{v}_j^\top (\widetilde{\Sigma}^{(S)} - \widehat{\Sigma}) \mathbf{v}_k \right| \\ &\leq \sum_{j,k=1}^S \frac{\delta_S}{1 - \delta_S} |\alpha_j \alpha_k| \widehat{\lambda}_1 \\ &\leq \frac{\delta_S}{1 - \delta_S} S \widehat{\lambda}_1 \end{aligned}$$

Therefore, we have,

$$\left\| \widehat{\mathbf{U}}_L^\top \mathbf{V}_{\leq (1-\delta) \widehat{\lambda}_L} \right\|_2 \leq S \frac{\delta_S \widehat{\lambda}_1}{\widehat{\lambda}_L (1 - \delta_S) - \widehat{\lambda}_S} \quad (37)$$

Using the same argument as in the proof of Theorem 2 (c), we have the desired result. \square

B Applications for Distributed PCA

Our distributed PCA can be applied to a wide range of important applications. In this section, we discuss two applications to principal component regression and single index model. Model assumptions and theoretical results are provided in this section and further numerical experiments will be presented in Appendix D in the supplementary material.

B.1 Distributed PCA for principal component regression

Principal component regression [Jef67, Jol82] is built on the following multivariate linear model,

$$\mathbf{y} = \mathbf{A}\boldsymbol{\beta}^* + \boldsymbol{\epsilon} \quad (38)$$

In (38), $\mathbf{A} = [\mathbf{a}_1, \dots, \mathbf{a}_n]^\top$ is the $n \times d$ observed covariate matrix with *i.i.d.* rows, where each \mathbf{a}_i is a zero-mean random vector with the covariance matrix $\boldsymbol{\Sigma}$, $\boldsymbol{\beta}^*$ is the $d \times 1$ coefficient, and \mathbf{y} is the $n \times 1$ response vector. The noise $\boldsymbol{\epsilon} = [\epsilon_1, \dots, \epsilon_n]^\top$ is the error term with $\mathbb{E}(\epsilon_i) = 0$ and $\boldsymbol{\epsilon}$ is independent from \mathbf{a} . Since our main purpose here is to illustrate PCR in our distributed algorithm, we assume that data dimension d is a constant. Of course, it would be interesting to extend to high-dimensional case, and we leave it for future investigation. Moreover, for the ease of technical derivation and presentation, we assume that $\boldsymbol{\beta}^*$ is normalized with $\|\boldsymbol{\beta}^*\|_2 = 1$.

Our goal is to estimate the coefficient $\boldsymbol{\beta}^*$ from \mathbf{A}, \mathbf{y} . Denote $\mathbf{U}_L = [\mathbf{u}_1, \dots, \mathbf{u}_L]$, $1 \leq L \leq d$ the subspace spanned by the top- L eigenvectors of $\boldsymbol{\Sigma}$. In principal component regression (denoted as PCR below), $\boldsymbol{\beta}^*$ is assumed to lie in the same subspace, i.e.,

$$\boldsymbol{\beta}^* = \mathbf{U}_L \boldsymbol{\gamma}^*$$

for some vector $\boldsymbol{\gamma}^* \in \mathbb{R}^L$. Our goal is to estimate $\boldsymbol{\beta}^*$.

Let $\widehat{\boldsymbol{\Sigma}} = \frac{1}{n} \mathbf{A}^\top \mathbf{A}$ be the sample covariance matrix with eigenvalues $\widehat{\lambda}_1 \geq \dots \geq \widehat{\lambda}_d > 0$. In the traditional setting with an explicit eigengap, we assume $\lambda_L > \lambda_{L+1}$ and estimate \mathbf{U}_L by the top- L eigenspace of the empirical covariance matrix $\widehat{\boldsymbol{\Sigma}}$, i.e., $\widehat{\mathbf{U}}_L = [\widehat{\mathbf{u}}_1, \dots, \widehat{\mathbf{u}}_L]$. Then the covariate data matrix \mathbf{A} is projected on this estimated subspace, $\widetilde{\mathbf{A}} = \mathbf{A}\widehat{\mathbf{U}}_L$ and an estimator $\widehat{\boldsymbol{\gamma}}$ of $\boldsymbol{\gamma}^*$ is obtained by ordinary least squares regression of the response vector \mathbf{y} on the projected data matrix $\widetilde{\mathbf{A}}$,

$$\widehat{\boldsymbol{\gamma}} = (\widetilde{\mathbf{A}}^\top \widetilde{\mathbf{A}})^{-1} \widetilde{\mathbf{A}}^\top \mathbf{y}$$

Therefore, the standard PCR estimator is $\widehat{\boldsymbol{\beta}} = \widehat{\mathbf{U}}_L \widehat{\boldsymbol{\gamma}}$.

In a gap-free setting, we cannot directly estimate \mathbf{U}_L due to the lack of an eigengap assumption. Instead, we consider an enlarged eigenspace estimator $\mathbf{V}_S = [\mathbf{v}_1, \dots, \mathbf{v}_S]$ given by our Algorithm 2 where $S > L$ is defined as before, i.e., $S := \arg \max\{l : \lambda_l > (1 - \delta) \widehat{\lambda}_L\}$, for some pre-determined parameter $\delta > 0$. In a distributed environment, our data is split uniformly on K local machines. The data on each machine is denoted by $\mathbf{A}_k \in \mathbb{R}^{m \times d}$, $\mathbf{y}_k \in \mathbb{R}^{m \times 1}$ for $k = 1, \dots, K$. Now we can obtain the corresponding projected data matrix $\widetilde{\mathbf{A}}_k = \mathbf{A}_k \mathbf{V}_S$ on each machine. The $\widetilde{\mathbf{A}}_k^\top \widetilde{\mathbf{A}}_k$, $\widetilde{\mathbf{A}}_k^\top \mathbf{y}_k$ are then computed locally and collected by the central machine. Then the central machine computes an OLS estimator based on $\{\mathbf{y}_k\}$ and $\{\widetilde{\mathbf{A}}_k\}$,

$$\widetilde{\boldsymbol{\gamma}} = \left(\sum_{k=1}^K \widetilde{\mathbf{A}}_k^\top \widetilde{\mathbf{A}}_k \right)^{-1} \left(\sum_{k=1}^K \widetilde{\mathbf{A}}_k^\top \mathbf{y}_k \right) = (\widetilde{\mathbf{A}}^\top \widetilde{\mathbf{A}})^{-1} \widetilde{\mathbf{A}}^\top \mathbf{y} \quad (39)$$

Finally, our distributed estimator is obtained from $\tilde{\boldsymbol{\beta}} = \mathbf{V}_S \tilde{\boldsymbol{\gamma}}$.

Proposition 1 below first describes the upper bound of estimation error for the usual PCR result where data matrix is on one machine and explicit eigengap is assumed. The technical proof in this subsection will be deferred to Appendix C in the supplementary material.

Proposition 1. *Assume the noise term $\{\epsilon_i\}_{i=1}^n$ are sub-Gaussian(σ^2) random variables that are independent from each other and from covariate \mathbf{A} . We further assume $\lambda_L > \lambda_{L+1}$, then the single-machine estimator $\hat{\boldsymbol{\beta}}$ satisfies*

$$\frac{1}{n} \left\| \mathbf{A}\hat{\boldsymbol{\beta}} - \mathbf{A}\boldsymbol{\beta}^* \right\|_2^2 \leq \hat{\lambda}_1 \hat{\lambda}_L^{-2} \frac{d}{n} + \hat{\lambda}_1 \frac{\left\| \hat{\boldsymbol{\Sigma}} - \boldsymbol{\Sigma} \right\|_2^2}{(\lambda_L - \hat{\lambda}_{L+1})^2} \quad (40)$$

with high probability. Here the omitted constant in \lesssim depend on σ^2 and $\|\boldsymbol{\gamma}^*\|_2$.

We would like to make some remarks on Proposition 1. By Theorem 5.39 in [Ver12], with high probability $\hat{\lambda}_L \geq \lambda_L + o(1)$. Therefore, $\hat{\lambda}_L^{-1}$ is bounded with high probability. Moreover, since $\left\| \boldsymbol{\Sigma} - \hat{\boldsymbol{\Sigma}} \right\|_2 = \mathcal{O}_p(\sqrt{d/n})$, this result indicates that the oracle estimator of PCR enjoys a statistical convergence rate of order $\mathcal{O}_p(\sqrt{d/n})$.

Now we are ready to provide our result on the distributed PCR with no eigengap assumption. The proof of Theorem 4 will be provided in Appendix C in the supplementary material.

Theorem 4. *Assume the noise term $\{\epsilon_i\}_{i=1}^n$ are sub-Gaussian(σ^2) random variables that are independent from each other and from covariate \mathbf{A} . If there exists $\delta_S < \delta/2$ such that $\left\| \hat{\mathbf{U}}_{\leq(1-\delta_S)\hat{\lambda}_S}^\top \mathbf{V}_S \right\|_2 \leq \frac{\delta_S}{16\hat{\lambda}_1/\hat{\lambda}_{S+1}}$, then with high probability, the prediction error of our distributed PCR estimator $\tilde{\boldsymbol{\beta}} = \mathbf{V}_S \tilde{\boldsymbol{\gamma}}$ satisfies,*

$$\begin{aligned} \frac{1}{n} \left\| \mathbf{A}\tilde{\boldsymbol{\beta}} - \mathbf{A}\boldsymbol{\beta}^* \right\|_2^2 &\lesssim \hat{\lambda}_1 \left[(1 - \delta_S) \hat{\lambda}_S \right]^{-2} \frac{d}{n} + \hat{\lambda}_1 \frac{\left\| \boldsymbol{\Sigma} - \hat{\boldsymbol{\Sigma}} \right\|_2^2}{(\lambda_L - (1 - \delta) \hat{\lambda}_L)^2} \\ &+ \frac{S^2 \delta_S^2}{(1 - \delta_S)^2} \frac{\hat{\lambda}_1^3}{(1 - \delta/2) \hat{\lambda}_L - \hat{\lambda}_S / (1 - \delta_S)} \end{aligned} \quad (41)$$

Here the omitted constant in \lesssim depend on σ^2 and $\|\boldsymbol{\gamma}^*\|_2$.

Notice that when $\lambda_L > \lambda_{L+1}$ as in the explicit eigengap case, we can simply set $\delta = \hat{\lambda}_L / (\hat{\lambda}_L - \hat{\lambda}_{L+1})$, $S = L$, and $\delta_S = 0$. Then our error bound for distributed estimator $\tilde{\boldsymbol{\beta}}$ in Equation (41) will be the same as the classical bound in (40) (up to a constant factor).

B.2 Distributed PCA for single index model

In a standard single index model (denoted as SIM below), we assume,

$$y = f(\langle \boldsymbol{\beta}^*, \mathbf{a} \rangle) + \epsilon$$

where $y \in \mathbb{R}$ is the response, \mathbf{a} is the d -dimensional covariate vector, $\boldsymbol{\beta}^* \in \mathbb{R}^d$ is the parametric component and ϵ is a zero-mean noise that is independent of \mathbf{a} . Here, the so-called link function $f : \mathbb{R} \mapsto \mathbb{R}$ is the nonparametric component. We also focus on the low-dimensional setting where

d does not grow with the sample size n . For the model identifiability, we assume that $\|\beta^*\|_2 = 1$ since $\|\beta^*\|_2$ can be absorbed into f . Following [Li92, JSA14] and references therein, we can use the second order Stein's identity to estimate β^* .

Proposition 2. *Assume that the density p of \mathbf{a} is twice differentiable. In addition, we define the second-order score function $T : \mathbb{R}^d \mapsto \mathbb{R}^{d \times d}$ as*

$$T(\mathbf{a}) = \nabla^2 p(\mathbf{a})/p(\mathbf{a})$$

Then, for any twice differentiable function $g : \mathbb{R}^d \mapsto \mathbb{R}$ such that $\mathbb{E}[\nabla^2 g(\mathbf{a})]$ exists, we have

$$\mathbb{E}[g(\mathbf{a}) \cdot T(\mathbf{a})] = \mathbb{E}[\nabla^2 g(\mathbf{a})]$$

Now we consider the SIM with Gaussian distribution as a special case, where $\mathbf{a} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d)$. The second order score function now becomes

$$T(\mathbf{a}) = \mathbf{a}\mathbf{a}^\top - \mathbf{I}_d$$

By Proposition 2 we have

$$\mathbb{E}[y(\mathbf{a}\mathbf{a}^\top - \mathbf{I}_d)] = C_0 \cdot \beta^* \beta^{*\top} \quad (42)$$

where $C_0 = 2\mathbb{E}[f''(\langle \mathbf{a}, \beta^* \rangle)]$.

Therefore, one way to estimator β^* is to obtain the leading eigenvector of $\mathbb{E}[y \cdot (\mathbf{a}\mathbf{a}^\top - \mathbf{I}_d)]$ from samples. Given n i.i.d. sample $\{\mathbf{a}_i, y_i\}_{i=1}^n$, we can calculate the estimator $\hat{\beta}$ by extracting the leading eigenvector of $\frac{1}{n} \sum_{i=1}^n y_i \cdot (\mathbf{a}_i \mathbf{a}_i^\top - \mathbf{I}_d)$, the empirical estimation of $\mathbb{E}[y(\mathbf{a}\mathbf{a}^\top - \mathbf{I}_d)]$. This can also be extended to our distributed setting, where we estimate $\hat{\beta}$ by $\tilde{\beta}$ from the distributed PCA Algorithm 1.

Let $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_d$ denote the eigenvalues of population matrix $\mathbb{E}[y(\mathbf{a}\mathbf{a}^\top - \mathbf{I}_d)]$ and $\hat{\lambda}_1 \geq \hat{\lambda}_2 \geq \dots \geq \hat{\lambda}_d$ the eigenvalues for the empirical matrix $\frac{1}{n} \sum_{i=1}^n y_i \cdot (\mathbf{a}_i \mathbf{a}_i^\top - \mathbf{I}_d)$ calculated with pooled data. Before presenting our theoretical results, we make some standard assumptions.

Assumption 1. *Under the Gaussian SIM model given above, we further assume that*

- (1) *We assume f and \mathbf{a} are such that $\mathbb{E}[f''(\langle \mathbf{a}, \beta^* \rangle)] > 0$, and moreover, $f(\langle \beta^*, \mathbf{a} \rangle)$ is bounded.*
- (2) *We assume the noise term $\{\epsilon\}_{i=1}^n$ to be independent, zero-mean sub-Gaussian(σ) random variables.*

Item (1) is commonly assumed in many references, for example, Definition 2.6 in [YBL17]. We further make a boundedness assumption on $f(\cdot)$. This assumption make use of the fact that \mathbf{a} is spherical Gaussian, thus with high probability, $\langle \beta^*, \mathbf{a} \rangle$ is bounded and $f(\cdot)$ only need to be finite in this domain.

Still, we consider the single-machine case first. The following proposition quantifies the statistical rate of convergence of the non-distributed estimator $\hat{\beta}$. We defer the proof of this theorem to Appendix C in the supplement.

Theorem 5. *Under Gaussian SIM model and Assumption 1, our estimator $\hat{\beta}$ satisfies*

$$\min_{t \in \{-1, +1\}} \|t\hat{\beta} - \beta^*\|_2 = \tilde{\mathcal{O}}_p\left(\frac{d}{\sqrt{n}}\right) \quad (43)$$

with high probability.

We can extend the Gaussian SIM model to our distributed framework where data are stored on different machines. The transformed “covariance matrix” on each machine k has the form $\frac{1}{m} \sum_{j=1}^m y_j^{(k)} (\mathbf{a}_j^{(k)} \mathbf{a}_j^{(k)\top} - \mathbf{I}_d)$ for $k = 1, \dots, K$. Then it is straightforward to apply Algorithm 1 to obtain a distributed estimation $\tilde{\beta}$ of β^* . Combining the results in Theorem 1, Corollary 1, and Theorem 5, we can obtain the non-asymptotic upper bound for our distributed estimator $\tilde{\beta}$.

Proposition 3. *In a distributed environment, if we estimate β^* with $\tilde{\beta}$ using Algorithm 1. We set the numbers of inner iterations and outer iterations to be T simultaneously. Under Assumption 1 and if $\eta \leq \delta \hat{\lambda}_1 / 16$ in Corollary 1, our distributed estimator $\tilde{\beta}$ satisfies,*

$$\min_{t \in \{-1, +1\}} \|t\tilde{\beta} - \beta^*\|_2 \leq \tilde{\mathcal{O}}_p \left(\frac{d}{\sqrt{n}} \right) + \mathcal{O}_p \left(\left[\frac{d \log d}{m} \right]^{T/2} \right) \quad (44)$$

Proposition 3 indicates that when the number of iterations T is sufficiently large, the first error term in (44) will dominate the second one and therefore our estimator $\tilde{\beta}$ will have the convergence rate of $\tilde{\mathcal{O}}_p(d/\sqrt{n})$.

C Proofs of Theoretical Results for Applications

C.1 Proofs of theoretical results in Section B.1

Proof of Proposition 1

Proof. First, notice that

$$\frac{1}{n} \left\| \mathbf{A}\hat{\beta} - \mathbf{A}\beta^* \right\|_2^2 \leq \frac{1}{n} \|\mathbf{A}\|_2^2 \left\| \hat{\beta} - \beta^* \right\|_2^2 = \hat{\lambda}_1 \left\| \hat{\beta} - \beta^* \right\|_2^2$$

Now consider $\|\hat{\beta} - \beta^*\|_2$, we have

$$\begin{aligned} \left\| \hat{\beta} - \beta^* \right\|_2 &= \left\| \hat{\mathbf{U}}_L \mathbf{O} \mathbf{O}^\top \hat{\gamma} - \mathbf{U}_L \gamma^* \right\|_2 \\ &= \left\| \hat{\mathbf{U}}_L \mathbf{O} \mathbf{O}^\top \hat{\gamma} - \hat{\mathbf{U}}_L \mathbf{O} \gamma^* + \hat{\mathbf{U}}_L \mathbf{O} \gamma^* - \mathbf{U}_L \gamma^* \right\|_2 \\ &\leq \|\mathbf{O}^\top \hat{\gamma} - \gamma^*\|_2 + \|\gamma^*\|_2 \left\| \hat{\mathbf{U}}_L \mathbf{O} - \mathbf{U}_L \right\|_2 \end{aligned} \quad (45)$$

for any orthogonal matrix $\mathbf{O} \in \mathbb{R}^{L \times L}$, $\mathbf{O}^\top \mathbf{O} = \mathbf{I}_L$.

Now consider the first part on the RHS.

$$\begin{aligned} \left\| \mathbf{O}^\top \hat{\gamma} - \gamma^* \right\|_2 &= \left\| \mathbf{O}^\top \left(\tilde{\mathbf{A}}^\top \tilde{\mathbf{A}} \right)^{-1} \tilde{\mathbf{A}}^\top \left(\mathbf{A} \left(\mathbf{U}_L - \hat{\mathbf{U}}_L \mathbf{O} \right) \gamma^* + \epsilon \right) \right\|_2 \\ &\leq \left\| \left(\tilde{\mathbf{A}}^\top \tilde{\mathbf{A}} \right)^{-1} \tilde{\mathbf{A}}^\top \epsilon \right\|_2 + \left\| \left(\tilde{\mathbf{A}}^\top \tilde{\mathbf{A}} \right)^{-1} \tilde{\mathbf{A}}^\top \mathbf{A} \right\|_2 \left\| \left(\mathbf{U}_L - \hat{\mathbf{U}}_L \mathbf{O} \right) \gamma^* \right\|_2 \\ &\leq \left\| \left(\tilde{\mathbf{A}}^\top \tilde{\mathbf{A}} \right)^{-1} \tilde{\mathbf{A}}^\top \epsilon \right\|_2 + \left\| \left(\mathbf{U}_L - \hat{\mathbf{U}}_L \mathbf{O} \right) \gamma^* \right\|_2 \end{aligned} \quad (46)$$

Therefore, plug inequality (46) into inequality (45) we have

$$\left\| \hat{\beta} - \beta^* \right\|_2 \leq \|(\tilde{\mathbf{A}}^\top \tilde{\mathbf{A}})^{-1} \tilde{\mathbf{A}}^\top \epsilon\|_2 + 2 \left\| \hat{\mathbf{U}}_L \mathbf{O} - \mathbf{U}_L \right\|_2 \|\gamma^*\|_2$$

Take the $L \times L$ orthogonal matrix $\mathbf{O} = \widehat{\mathbf{U}}_L^\top \mathbf{U}_L$, we have

$$\begin{aligned} \left\| \widehat{\mathbf{U}}_L \mathbf{O} - \mathbf{U}_L \right\|_2 &= \left\| \widehat{\mathbf{U}}_L \widehat{\mathbf{U}}_L^\top \mathbf{U}_L - \mathbf{U}_L \right\|_2 = \sqrt{\left\| \mathbf{U}_L^\top (I_d - \widehat{\mathbf{U}}_L \widehat{\mathbf{U}}_L^\top) \mathbf{U}_L \right\|_2} \\ &= \left\| \mathbf{U}_L^\top \widehat{\mathbf{U}}_L^\perp \right\|_2 \leq \frac{\left\| \widehat{\boldsymbol{\Sigma}} - \boldsymbol{\Sigma} \right\|_2}{\lambda_L - \widehat{\lambda}_{L+1}} \end{aligned} \quad (47)$$

where the third equality uses the fact that $\widehat{\mathbf{U}}_L \widehat{\mathbf{U}}_L^\top + \widehat{\mathbf{U}}_L^\perp \widehat{\mathbf{U}}_L^{\perp\top} = I_d$ and the last inequality follows from Lemma B.3 in [AZL16]. Also,

$$\left\| (\widetilde{\mathbf{A}}^\top \widetilde{\mathbf{A}})^{-1} \widetilde{\mathbf{A}}^\top \boldsymbol{\epsilon} \right\|_2 \leq \left\| \left(\frac{\widetilde{\mathbf{A}}^\top \widetilde{\mathbf{A}}}{n} \right)^{-1} \right\|_2 \left\| \frac{\widetilde{\mathbf{A}}^\top \boldsymbol{\epsilon}}{n} \right\|_2 = \widehat{\lambda}_L^{-1} \left\| \frac{\mathbf{A}^\top \boldsymbol{\epsilon}}{n} \right\|_2$$

Denote $\xi = \frac{\widetilde{\mathbf{A}}^\top \boldsymbol{\epsilon}}{n}$. Notice that ξ is a d -dimension sub-Gaussian vector with variance proxy $\frac{\sigma^2}{n}$. Therefore, with probability at least $1 - e^{-C^2}$

$$\|\xi\|_2 = \sqrt{\sum_{i=1}^d \xi_i^2} \leq \sqrt{2} C \sigma \sqrt{\frac{d}{n}}$$

Thus we have,

$$\left\| (\widetilde{\mathbf{A}}^\top \widetilde{\mathbf{A}})^{-1} \widetilde{\mathbf{A}}^\top \boldsymbol{\epsilon} \right\|_2 \leq \sqrt{2} C \sigma \widehat{\lambda}_L^{-1} \sqrt{\frac{d}{n}} \quad (48)$$

Combine inequality (47) with inequality (48) we obtain the desired result,

$$\frac{1}{n} \left\| \mathbf{A} \widehat{\boldsymbol{\beta}} - \mathbf{A} \boldsymbol{\beta}^* \right\|_2^2 \leq 4C^2 \sigma^2 \widehat{\lambda}_1 \widehat{\lambda}_L^{-2} \frac{d}{n} + 2\widehat{\lambda}_1 \frac{\left\| \widehat{\boldsymbol{\Sigma}} - \boldsymbol{\Sigma} \right\|_2^2}{(\lambda_L - \widehat{\lambda}_{L+1})^2} \|\boldsymbol{\gamma}^*\|_2^2$$

with probability at least $1 - e^{-C^2}$.

□

Proof of Theorem 4

Proof. Notice that

$$\begin{aligned} \frac{1}{n} \left\| \mathbf{A} \widehat{\boldsymbol{\beta}} - \mathbf{A} \boldsymbol{\beta}^* \right\|_2^2 &= \frac{1}{n} \left\| \mathbf{A} \mathbf{V}_S \left(\mathbf{V}_S^\top \mathbf{A}^\top \mathbf{A} \mathbf{V}_S \right)^{-1} \mathbf{V}_S^\top \mathbf{A}^\top (\mathbf{A} \mathbf{U}_L \boldsymbol{\gamma}^* + \boldsymbol{\epsilon}) - \mathbf{A} \mathbf{U}_L \boldsymbol{\gamma}^* \right\|_2^2 \\ &\leq \frac{2}{n} \left\| \mathbf{A} \mathbf{V}_S \left(\mathbf{V}_S^\top \mathbf{A}^\top \mathbf{A} \mathbf{V}_S \right)^{-1} \mathbf{V}_S^\top \mathbf{A}^\top \boldsymbol{\epsilon} \right\|_2^2 \\ &\quad + \frac{2}{n} \left\| \left(\mathbf{A} \mathbf{V}_S \left(\mathbf{V}_S^\top \mathbf{A}^\top \mathbf{A} \mathbf{V}_S \right)^{-1} \mathbf{V}_S^\top \mathbf{A}^\top - I_d \right) \mathbf{A} \mathbf{U}_L \boldsymbol{\gamma}^* \right\|_2^2 \end{aligned} \quad (49)$$

Similar to (48) in the previous proof, the first term on the RHS satisfies,

$$\frac{2}{n} \left\| \mathbf{A} \mathbf{V}_S \left(\mathbf{V}_S^\top \mathbf{A}^\top \mathbf{A} \mathbf{V}_S \right)^{-1} \mathbf{V}_S^\top \mathbf{A}^\top \boldsymbol{\epsilon} \right\|_2^2 \leq 4C^2 \sigma^2 \widehat{\lambda}_1 \left[(1 - \delta_S) \widehat{\lambda}_S \right]^{-2} \frac{d}{n} \quad (50)$$

with probability at least $1 - e^{-C^2}$. Here the $(1 - \delta_S)\widehat{\lambda}_S$ term is given by inequality (22) in Corollary 3. For the second term on the RHS, notice that we have the following identity

$$\left(\mathbf{A} \mathbf{V}_S \left(\mathbf{V}_S^\top \mathbf{A}^\top \mathbf{A} \mathbf{V}_S \right)^{-1} \mathbf{V}_S^\top \mathbf{A}^\top - \mathbf{I}_d \right) \mathbf{A} \mathbf{V}_S \mathbf{w} = 0$$

for any $\mathbf{w} \in \mathbb{R}^S$. We set $\mathbf{w} = \mathbf{V}_S^\top \mathbf{U}_L \boldsymbol{\gamma}^*$ and can obtain that

$$\begin{aligned} & \left\| \left(\mathbf{A} \mathbf{V}_S \left(\mathbf{V}_S^\top \mathbf{A}^\top \mathbf{A} \mathbf{V}_S \right)^{-1} \mathbf{V}_S^\top \mathbf{A}^\top - \mathbf{I}_d \right) \mathbf{A} \mathbf{U}_L \boldsymbol{\gamma}^* \right\|_2 \\ &= \left\| \left(\mathbf{A} \mathbf{V}_S \left(\mathbf{V}_S^\top \mathbf{A}^\top \mathbf{A} \mathbf{V}_S \right)^{-1} \mathbf{V}_S^\top \mathbf{A}^\top - \mathbf{I}_d \right) \left(\mathbf{A} \mathbf{U}_L \boldsymbol{\gamma}^* - \mathbf{A} \mathbf{V}_S \mathbf{V}_S^\top \mathbf{U}_L \boldsymbol{\gamma}^* \right) \right\|_2 \\ &\leq \left\| \mathbf{A} \mathbf{V}_S \left(\mathbf{V}_S^\top \mathbf{A}^\top \mathbf{A} \mathbf{V}_S \right)^{-1} \mathbf{V}_S^\top \mathbf{A}^\top - \mathbf{I}_d \right\|_2 \| \mathbf{A} \mathbf{U}_L \boldsymbol{\gamma}^* - \mathbf{A} \mathbf{V}_S \mathbf{V}_S^\top \mathbf{U}_L \boldsymbol{\gamma}^* \|_2 \end{aligned} \quad (51)$$

Denote the singular value decomposition of $\mathbf{A} \mathbf{V}_S$ as $\mathbf{P} \mathbf{W} \mathbf{Q}^\top$, where $\mathbf{P} \in \mathbb{R}^{d \times d}$, $\mathbf{Q} \in \mathbb{R}^{S \times S}$ are two orthogonal matrix and $\mathbf{W} \in \mathbb{R}^{d \times S}$ is a diagonal matrix. Therefore,

$$\left\| \mathbf{A} \mathbf{V}_S \left(\mathbf{V}_S^\top \mathbf{A}^\top \mathbf{A} \mathbf{V}_S \right)^{-1} \mathbf{V}_S^\top \mathbf{A}^\top - \mathbf{I}_d \right\|_2 = \left\| \sum_{i=S+1}^d \mathbf{p}_i \mathbf{p}_i^\top \right\|_2 = 1 \quad (52)$$

Now plug equality (52) into inequality (51), it can be shown that

$$\begin{aligned} & \frac{2}{n} \left\| \left(\mathbf{A} \mathbf{V}_S \left(\mathbf{V}_S^\top \mathbf{A}^\top \mathbf{A} \mathbf{V}_S \right)^{-1} \mathbf{V}_S^\top \mathbf{A}^\top - \mathbf{I}_d \right) \mathbf{A} \mathbf{U}_L \boldsymbol{\gamma}^* \right\|_2^2 \\ &\leq \frac{1}{n} \|\mathbf{A}\|_2^2 \left\| \mathbf{U}_L - \mathbf{V}_S \mathbf{V}_S^\top \mathbf{U}_L \right\|_2^2 \|\boldsymbol{\gamma}^*\|_2^2 \\ &\leq 2\widehat{\lambda}_1 \left\| \mathbf{U}_L^\top \left(\mathbf{I}_d - \mathbf{V}_S \mathbf{V}_S^\top \right) \mathbf{U}_L \right\|_2 \|\boldsymbol{\gamma}^*\|_2^2 \\ &\leq 2\widehat{\lambda}_1 \left\| \mathbf{U}_L^\top \mathbf{V}_S^\perp \mathbf{V}_S^{\perp\top} \mathbf{U}_L \right\|_2 \|\boldsymbol{\gamma}^*\|_2^2 \\ &\leq 2\widehat{\lambda}_1 \left\| \mathbf{U}_L^\top \mathbf{V}_S^\perp \right\|_2^2 \|\boldsymbol{\gamma}^*\|_2^2 \\ &\leq 2\widehat{\lambda}_1 \|\boldsymbol{\gamma}^*\|_2^2 \left(\frac{\left\| \Sigma - \widehat{\Sigma} \right\|_2^2}{\left(\lambda_L - (1 - \delta/2)\widehat{\lambda}_L \right)^2} + 4S^2 \frac{\delta_S^2 \widehat{\lambda}_1^2}{(1 - \delta/2)\widehat{\lambda}_L - \widehat{\lambda}_S/(1 - \delta_S)} \right) \\ &\leq 2\widehat{\lambda}_1 \|\boldsymbol{\gamma}^*\|_2^2 \left(\frac{\left\| \Sigma - \widehat{\Sigma} \right\|_2^2}{\left(\lambda_L - (1 - \delta)\widehat{\lambda}_L \right)^2} + \frac{S^2 \delta_S^2}{(1 - \delta_S)^2} \frac{\widehat{\lambda}_1^2}{(1 - \delta/2)\widehat{\lambda}_L - \widehat{\lambda}_S/(1 - \delta_S)} \right) \end{aligned} \quad (53)$$

Here the last inequality follows from Theorem 3 and the proof in Corollary 4. Our proof is completed once we combine inequality (49), (50) and (53). \square

C.2 Proofs of theoretical sesults in Section B.2

Proof of Theorem 5

We will need a definition on Orlicz norm from [LT13], in order to deal with random variables whose tail is heavier than sub-Exponential variables.

Definition 1. For $1 \leq \alpha < \infty$, let $\psi_\alpha = \exp(x^\alpha) - 1$. For $0 < \alpha < 1$, let $\psi_\alpha(x) = \exp(x^\alpha) - 1$ for large enough $x \geq x_\alpha$ and ψ_α is linear in $[0, x_\alpha]$ in order to remain global convexity. The Orlicz norm ψ_α of a random variable X is defined as

$$\|X\|_{\psi_\alpha} \triangleq \inf \{c \in (0, \infty) | \mathbb{E}[\psi_\alpha(|X|/c)] \leq 1\}$$

Proof. Denote $\Sigma = \mathbb{E}[y(\mathbf{a}\mathbf{a}^\top - \mathbf{I}_d)]$, $\widehat{\Sigma} = \frac{1}{n} \sum_{i=1}^n y_i (\mathbf{a}_i \mathbf{a}_i^\top - \mathbf{I}_p)$. We first obtain a high probability bound on $\|\widehat{\Sigma} - \Sigma\|_2$.

Without loss of generality, we can assume $M = 1$, otherwise we can always multiply M on the bound we obtain. Note that

$$\begin{aligned} & \|\widehat{\Sigma} - \Sigma\|_2 \\ &= \left\| \frac{1}{n} \sum_{i=1}^n y_i (\mathbf{a}_i \mathbf{a}_i^\top - \mathbf{I}_p) - \mathbb{E}[f(\langle \beta^*, \mathbf{a} \rangle)(\mathbf{a}\mathbf{a}^\top - \mathbf{I}_d)] \right\|_2 \\ &\leq \left\| \frac{1}{n} \sum_{i=1}^n f(\langle \beta^*, \mathbf{a}_i \rangle)(\mathbf{a}_i \mathbf{a}_i^\top - \mathbf{I}_p) - \mathbb{E}[f(\langle \beta^*, \mathbf{a} \rangle)(\mathbf{a}\mathbf{a}^\top - \mathbf{I}_d)] \right\|_2 + \left\| \frac{1}{n} \sum_{i=1}^n \epsilon_i (\mathbf{a}_i \mathbf{a}_i^\top - \mathbf{I}_d) \right\|_2 \end{aligned} \quad (54)$$

Under Lemma 5.4 in [Ver12], We can evaluate the operator norm on the RHS of inequality (54) on a $\frac{1}{4}$ -net \mathcal{E} of the unit sphere \mathcal{S}^{d-1} :

$$\begin{aligned} & \left\| \frac{1}{n} \sum_{i=1}^n f(\langle \beta^*, \mathbf{a}_i \rangle)(\mathbf{a}_i \mathbf{a}_i^\top - \mathbf{I}_p) - \mathbb{E}[f(\langle \beta^*, \mathbf{a} \rangle)(\mathbf{a}\mathbf{a}^\top - \mathbf{I}_d)] \right\|_2 \\ &\leq 2 \max_{\mathbf{v} \in \mathcal{E}} \left| \frac{1}{n} \sum_{i=1}^n f(\langle \beta^*, \mathbf{a}_i \rangle)(z(\mathbf{v})_i^2 - 1) - \mathbb{E}[f(\langle \beta^*, \mathbf{a} \rangle)(z(\mathbf{v})^2 - 1)] \right| \end{aligned}$$

where $z(\mathbf{v}) = \mathbf{v}^\top \mathbf{a} \sim \mathcal{N}(0, 1)$. Notice that $z(\mathbf{v})^2 - 1$ is a sub-Exponential with parameter $(2, 4)$, and $\mathbb{E}\{f(\langle \beta^*, \mathbf{a} \rangle)(z(\mathbf{v})^2 - 1)\} \geq -2$. Therefore, denote $\theta = \max\{1, 2M\}$, $\forall |s| \leq \frac{1}{8\theta}$,

$$\begin{aligned} & \mathbb{E}[\exp\{s(f(\langle \beta^*, \mathbf{a} \rangle)(z(\mathbf{v})^2 - 1) - \mathbb{E}f(\langle \beta^*, \mathbf{a} \rangle)(z(\mathbf{v})^2 - 1))\}] \\ &\leq 1 + \sum_{k=2}^{\infty} \frac{|s|^k}{k!} \mathbb{E}|f(\langle \beta^*, \mathbf{a} \rangle)(z(\mathbf{v})^2 - 1) - \mathbb{E}(f(\langle \beta^*, \mathbf{a} \rangle)(z(\mathbf{v})^2 - 1))|^k \\ &\leq 1 + \sum_{k=2}^{\infty} \frac{|s|^k}{k!} 2^k \mathbb{E}|f(\langle \beta^*, \mathbf{a} \rangle)(z(\mathbf{v})^2 - 1)|^k \\ &\leq 1 + \sum_{k=2}^{\infty} \frac{|s|^k}{k!} (2M)^k \mathbb{E}|(z(\mathbf{v})^2 - 1)|^k \\ &\leq 1 + \sum_{k=2}^{\infty} (|s|8M)^k \\ &\leq \exp\{s^2(16M)^2/2\} \end{aligned}$$

Thus, $\tilde{X} \triangleq f(\langle \beta^*, \mathbf{a} \rangle)(z(\mathbf{v})^2 - 1)$ is a sub-Exponential with parameter $(16M, 8\theta)$. Therefore, by Proposition 5.16 in [Ver12], we can obtain a Bernstein-type inequality:

$$\mathbb{P}\left[\left|\frac{1}{n} \sum_{i=1}^n \tilde{X}_i - \mathbb{E}\tilde{X}\right| > 8\theta\epsilon\right] \leq 2 \exp\left[-\frac{n}{2}(\epsilon^2 \wedge \epsilon)\right]$$

for any $\epsilon > 0$. Now let $\epsilon = \max(\gamma, \gamma^2)$, where $\gamma = C\sqrt{\frac{d}{n}} + \frac{\delta_1}{\sqrt{n}}$, for some constant C and $\delta_1 > 0$. Now we have,

$$\mathbb{P} \left[\left| \frac{1}{n} \sum_{i=1}^n \tilde{X}_i - \mathbb{E} \tilde{X} \right| > 8\theta\gamma \right] \leq 2 \exp \left[-\frac{n}{2} \gamma^2 \right] \leq 2 \exp \left[-\frac{1}{2} (C^2 d + \delta_1^2) \right]$$

Notice that by Lemma 5.2 in [Ver12], we can choose the net \mathcal{E} so that it has cardinality $|\mathcal{E}| \leq 9^d$. Therefore, we take the union bound over all vectors $\mathbf{v} \in \mathcal{E}$, we obtain

$$\mathbb{P} \left[\max_{\mathbf{v} \in \mathcal{E}} \left| \frac{1}{n} \sum_{i=1}^n \tilde{X}_i - \mathbb{E} \tilde{X} \right| > 8\theta\gamma \right] \leq 2 \times 9^d \exp \left[-\frac{1}{2} (C^2 d + \delta_1^2) \right] = 2 \exp \left(-\frac{\delta_1^2}{2} \right) \quad (55)$$

where we can choose C sufficiently large, e.g. $C = 2\sqrt{\log 3}$.

For the second part on the RHS of the inequality (54), we have

$$\left\| \frac{1}{n} \sum_{i=1}^n \epsilon_i (\mathbf{a}_i \mathbf{a}_i^\top - \mathbf{I}_d) \right\|_2 \leq \left\| \frac{1}{n} \sum_{i=1}^n \epsilon_i \mathbf{a}_i \mathbf{a}_i^\top \right\|_2 + \left| \frac{1}{n} \sum_{i=1}^n \epsilon_i \right|$$

Given that ϵ s are independent sub-Gaussian(σ) random variables with mean 0, we have

$$\left| \frac{1}{n} \sum_{i=1}^n \epsilon_i \right| \leq \sigma \delta_2 \frac{1}{\sqrt{n}} \quad (56)$$

with probability at least $1 - 2 \exp(-\delta_2^2/2)$.

Now, in order to control $\left\| \frac{1}{n} \sum_{i=1}^n \epsilon_i \mathbf{a}_i \mathbf{a}_i^\top \right\|_2$, we consider $\left\| \sum_{i=1}^n \|\mathbf{a}_i\|_2^2 \mathbf{a}_i \mathbf{a}_i^\top \right\|_2$ first. Under a $\frac{1}{4}$ -net \mathcal{E}_2 of the unit sphere \mathcal{S}^{d-1} ,

$$\begin{aligned} \left\| \sum_{i=1}^n \|\mathbf{a}_i\|_2^2 \mathbf{a}_i \mathbf{a}_i^\top - \mathbb{E} \sum_{i=1}^n \|\mathbf{a}_i\|_2^2 \mathbf{a}_i \mathbf{a}_i^\top \right\|_2 &\leq 2 \max_{\mathbf{v} \in \mathcal{E}_2} \left| \sum_{i=1}^n \|\mathbf{a}_i\|_2^2 (\mathbf{v}^\top \mathbf{a}_i)^2 - \mathbb{E} \sum_{i=1}^n \|\mathbf{a}_i\|_2^2 (\mathbf{v}^\top \mathbf{a}_i)^2 \right| \\ &= 2 \max_{\mathbf{v} \in \mathcal{E}_2} \left| \sum_{i=1}^n \|\mathbf{a}_i\|_2^2 (\mathbf{v}^\top \mathbf{a}_i)^2 - \mathbb{E} \sum_{i=1}^n \|\mathbf{a}_i\|_2^2 (\mathbf{v}^\top \mathbf{a}_i)^2 \right| \\ &= 2 \max_{\mathbf{v} \in \mathcal{E}_2} \left| \sum_{i=1}^n \sum_{j=1}^d \mathbf{a}_{ij}^2 z(\mathbf{v})_i^2 - \mathbb{E} \sum_{i=1}^n \sum_{j=1}^d \mathbf{a}_{ij}^2 z(\mathbf{v})_i^2 \right| \end{aligned}$$

where $\mathbf{a}_{ij} \sim \mathcal{N}(0, 1)$ is the j -th term of \mathbf{a}_i and $z(\mathbf{v})_i = \mathbf{v}^\top \mathbf{a}_i \sim \mathcal{N}(0, 1)$. Notice that $\mathbf{a}_{ij}^2, z(\mathbf{v})_i^2 \sim \chi^2(1)$, and

$$\mathbb{P}(\mathbf{a}_{ij}^2 z(\mathbf{v})_i^2 \geq t) \leq \mathbb{P}(\mathbf{a}_{ij}^2 \geq \sqrt{t}) + \mathbb{P}(z(\mathbf{v})_i^2 \geq \sqrt{t}) \leq 2 \exp(-\sqrt{t}/2)$$

for all $i \in \{1, \dots, n\}, j \in \{1, \dots, d\}$. Therefore, denote

$$\|\mathbf{a}_{ij}^2 z(\mathbf{v})_i^2\|_{\psi_{1/2}} \leq K_\psi$$

where K_ψ is a finite constant. By Theorem 8.4 in [MW15], there exists a constant K_α such that

$$\left\| \sum_{i=1}^n \sum_{j=1}^d \mathbf{a}_{ij}^2 z(\mathbf{v})_i^2 - \mathbb{E} \sum_{i=1}^n \sum_{j=1}^d \mathbf{a}_{ij}^2 z(\mathbf{v})_i^2 \right\|_{\psi_{1/2}} \leq K_\alpha K_\psi \sqrt{nd} \log(nd)$$

Denote $Z = \sum_{i=1}^n \sum_{j=1}^d \mathbf{a}_{ij}^2 z(\mathbf{v})_i^2 - \mathbb{E} \sum_{i=1}^n \sum_{j=1}^d \mathbf{a}_{ij}^2 z(\mathbf{v})_i^2$ and $K_Z = \|\sum_{i=1}^n \sum_{j=1}^d \mathbf{a}_{ij}^2 z(\mathbf{v})_i^2 - \mathbb{E} \sum_{i=1}^n \sum_{j=1}^d \mathbf{a}_{ij}^2 z(\mathbf{v})_i^2\|_{\psi_{1/2}}$. Using Markov inequality, we have

$$\mathbb{P}(|Z| > t) \leq \frac{\mathbb{E}[\psi_{1/2}(Z/K_Z) + 1]}{\psi_{1/2}(t/K_Z) + 1} \leq 2 \exp \left\{ - \left(\frac{t}{K_Z} \right)^{1/2} \right\}$$

By a union bound, we have

$$\mathbb{P} \left[\left\| \sum_{i=1}^n \|\mathbf{a}_i\|_2^2 \mathbf{a}_i \mathbf{a}_i^\top - \mathbb{E} \sum_{i=1}^n \|\mathbf{a}_i\|_2^2 \mathbf{a}_i \mathbf{a}_i^\top \right\|_2 \geq t \right] \leq 2 \times 9^d \exp \left\{ - \left(\frac{t}{K_Z} \right)^{1/2} \right\}$$

Therefore, with probability at least $1 - 2 \exp(-\delta_3)$,

$$\left\| \sum_{i=1}^n \|\mathbf{a}_i\|_2^2 \mathbf{a}_i \mathbf{a}_i^\top - \mathbb{E} \sum_{i=1}^n \|\mathbf{a}_i\|_2^2 \mathbf{a}_i \mathbf{a}_i^\top \right\|_2 \leq 5K_\alpha K_\psi \sqrt{nd} \log(nd)d^2 + \delta_3^2 K_\alpha K_\psi \sqrt{nd} \log(nd)$$

Note that

$$\left\| \mathbb{E} \sum_{i=1}^n \|\mathbf{a}_i\|_2^2 \mathbf{a}_i \mathbf{a}_i^\top \right\|_2 \leq n \left\| \mathbb{E} \|\mathbf{a}_i\|_2^2 \mathbf{a}_i \mathbf{a}_i^\top \right\|_2 \leq 3nd$$

Thus, with probability at least $1 - 2 \exp(-\delta_3)$,

$$\begin{aligned} \left\| \sum_{i=1}^n \|\mathbf{a}_i\|_2^2 \mathbf{a}_i \mathbf{a}_i^\top \right\|_2 &\leq 5K_\alpha K_\psi \sqrt{nd} \log(nd)d^2 + \delta_3^2 K_\alpha K_\psi \sqrt{nd} \log(nd) + 3nd \\ &\leq 10K_\alpha K_\psi \sqrt{nd} \log nd^2 + 2\delta_3^2 K_\alpha K_\psi \sqrt{nd} \log n + 3nd \end{aligned}$$

By Theorem 4.1.1 in [T⁺15],

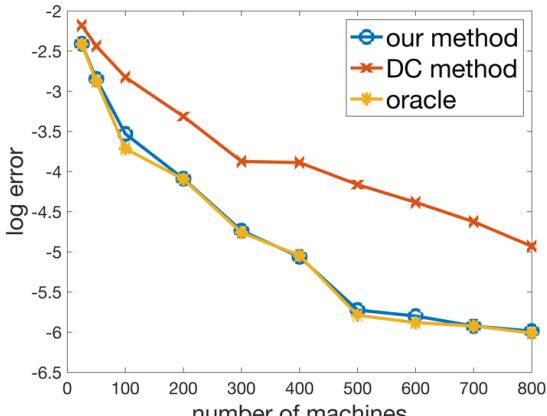
$$\mathbb{P} \left(\left\| \frac{1}{n} \sum_{i=1}^n \epsilon_i \mathbf{a}_i \mathbf{a}_i^\top \right\|_2 \geq t \right) \leq 2d \exp \left\{ - \frac{t^2}{2 \left\| \frac{1}{n^2} \sum_{i=1}^n \|\mathbf{a}_i\|_2^2 \mathbf{a}_i \mathbf{a}_i^\top \right\|_2} \right\}$$

which yields that with a probability over $1 - 2 \exp(-\delta_3) - 2 \exp(-\delta_4)$,

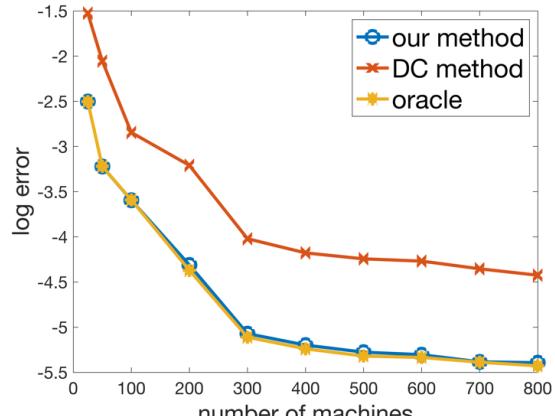
$$\begin{aligned} \left\| \frac{1}{n} \sum_{i=1}^n \epsilon_i \mathbf{a}_i \mathbf{a}_i^\top \right\|_2 &\leq \sqrt{2 \log d \left\| \frac{1}{n^2} \sum_{i=1}^n \|\mathbf{a}_i\|_2^2 \mathbf{a}_i \mathbf{a}_i^\top \right\|_2} + \sqrt{2\delta_4 \left\| \frac{1}{n^2} \sum_{i=1}^n \|\mathbf{a}_i\|_2^2 \mathbf{a}_i \mathbf{a}_i^\top \right\|_2} \\ &\leq \left(\frac{\sqrt{\delta_4 d} + \sqrt{d \log d}}{\sqrt{n}} \right) \sqrt{\frac{(20 + 4\delta_3^2)K_\alpha K_\psi d \sqrt{d \log n}}{\sqrt{n}} + 6} \\ &\leq \sqrt{(20 + 4\delta_3^2)K_\alpha K_\psi C + 6} \left(\frac{\sqrt{\delta_4 \log n} d + d \sqrt{\log n \log d}}{\sqrt{n}} \right) \end{aligned} \tag{57}$$

where we use Assumption 1. Combine inequality (54), (55), (56) as well as inequality (57), we have with probability at least $1 - 2 \exp(-\delta_1^2/2) - 2 \exp(-\delta_2^2/2) - 2 \exp(-\delta_3) - 2 \exp(-\delta_4)$,

$$\left\| \widehat{\Sigma} - \Sigma \right\|_2 \leq 17\theta \sqrt{\frac{d}{n}} + \sqrt{(20 + 4\delta_3^2)K_\alpha K_\psi C \delta_4 + 6\delta_4} \frac{d \sqrt{\log n}}{\sqrt{n}} + \frac{8\theta\delta_1 + \sigma\delta_2}{\sqrt{n}}$$



(a) Distributed PCR ($\sigma^2 = 0.2$)



(b) Distributed PCR ($\sigma^2 = 0.5$)

Figure D.5. Comparison between algorithms in PCR when the number of machines varies. The x -axis is the the number of machines and the y -axis is the log- l_2 error. On the left, the noise term has variance 0.2 and on the right figure, 0.5.

$$+ \sqrt{(20 + 4\delta_3^2)K_\alpha K_\psi C + 6} \frac{d\sqrt{\log d \log n}}{\sqrt{n}} \quad (58)$$

By Corollary 3.1 in [VCLR13],

$$\min_{t \in \{-1, +1\}} \|t\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}^*\|_2 = \min_{t \in \{-1, +1\}} \sqrt{2} \left\| \sin \Theta(t\hat{\boldsymbol{\beta}}, \boldsymbol{\beta}^*) \right\|_{\text{F}} \leq 2 \left\| \hat{\boldsymbol{\Sigma}} - \boldsymbol{\Sigma} \right\|_2 \quad (59)$$

Combine bound (59) with (58), we have the desired result.

D Additional Experiments

In this section, we present the experimental results on distributed PCR and distributed Gaussian SIM.

D.1 Numerical results of distributed PCR

We provide numerical results of distributed PCR in this section. Recall the problem setting in Section B.1. We assume the real coefficient β^* lies in the top-3-dim eigenspace of Σ , i.e., $\beta^* = \mathbf{U}_3\gamma^*$, where $\mathbf{U}_3 = [\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3]^\top$ consists of the top-3 eigenvectors of Σ . As in previous experiments, data dimension d is set to 50, and sample size on each machine is set to 500, i.e., $\mathbf{a} \in \mathbb{R}^{50}$, $m = 500$. We vary the number of machines. The response vector $\mathbf{y} \in \mathbb{R}^{mK}$ is generated by $\mathbf{y} = \mathbf{A}\beta^* + \epsilon$, where noise term $\epsilon \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I}_{mK})$, and σ^2 is a constant, which is set to 0.2 and 0.5 in the following experiments. Here covariate data matrix $\mathbf{A} = [\mathbf{a}_1, \dots, \mathbf{a}_{mK}]^\top \in \mathbb{R}^{mK \times d}$ is drawn *i.i.d.* from $\mathcal{N}(\mathbf{0}, \Sigma)$ with $\Sigma = \mathbf{U}\Lambda\mathbf{U}^\top$, $\Lambda = \text{diag}(2.5, 2, 1.5, 1, \dots, 1)$. γ is sampled only once from $\mathcal{N}(\mathbf{0}, \mathbf{I}_3)$ and is fixed in our 100 Monte-Carlo simulations. Moreover, the underlying true regression coefficient is $\beta^* = \mathbf{U}_3 \frac{\gamma}{\|\gamma\|_2}$.

We estimate \mathbf{U}_3 using 3 different estimation methods and compare their performances. The measurement we use here is the l_2 distance between estimator $\hat{\boldsymbol{\beta}}$ and real coefficient $\boldsymbol{\beta}^*$, i.e.,

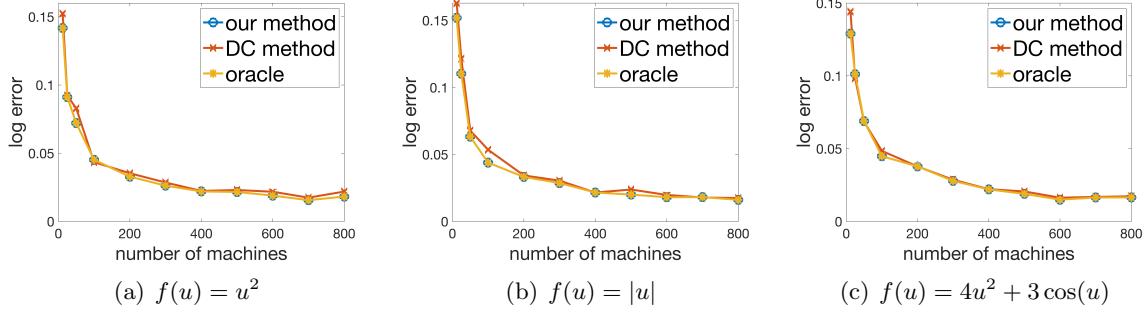


Figure D.6. Comparison between algorithms for SIM. The x -axis is the the number of machines and the y -axis is the $\log-l_2$ error.

error($\hat{\beta}$) = $\|\hat{\beta} - \beta^*\|_2$. The numbers of outer iterations and inner iterations in our algorithm are fixed as 40 and 10, respectively. The results are shown in Figure D.5. In accordance with previous experiments, our method almost keeps the same error rate as the oracle one.

D.2 Numerical results of distributed SIM

In our last part of the experiments, we conduct simulations on Gaussian single index model. Consider data dimension d to be 50, sample size on each machine to be 500, i.e., $\mathbf{a} \in \mathbb{R}^{50}$, $m = 500$. Our covariate data matrix $\mathbf{A} \in \mathbb{R}^{mK \times d}$ is drawn independently, where each row $\mathbf{a}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d)$ follows a standard normal distribution. For the data generating process of $\mathbf{y} = (y_1, \dots, y_{mK})^\top \in \mathbb{R}^{mK}$, we have $y_i = f(\langle \beta^*, \mathbf{a}_i \rangle) + \epsilon_i$, $\forall i \in \{1, \dots, mK\}$, where $f(\cdot)$ is our specific choice of link function, β^* is a normalized vector only drawn once during Monte-Carlo process from $\mathcal{N}(\mathbf{0}, \mathbf{I}_d)$, i.e., $\beta^* = \beta / \|\beta\|_2$, $\beta \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d)$ and $\{\epsilon_i\}$ are *i.i.d.* normal $\mathcal{N}(\mathbf{0}, \sigma^2)$ with the constant variance σ^2 fixed to be 0.2. During our estimation process, we estimate top eigenvector $\hat{\beta}$ of $\frac{1}{mK} \sum_{i=1}^{mK} y_i \cdot (\mathbf{a}_i \mathbf{a}_i^\top - \mathbf{I}_d)$.

In the following experiment, we consider three different link functions: $f(u) = u^2$, $f(u) = |u|$ and $f(u) = 4u^2 + 3\cos(u)$. The l_2 distance $\|\hat{\beta} - \beta^*\|_2$ is used here to measure the performance. In Figure D.6, for all choices of link function, our estimators have the same errors as the oracle results. For this experiment, the DC method also works well, which is mainly because the problem of estimating the top eigenvector is relatively simple and \mathbf{a} follows a symmetric normal distribution.