Journal of Machine Learning Research 25 (2024) 1-82 Submitted 7/22; Revised 8/23; Published 2/24

Personalized PCA: Decoupling Shared and Unique Features

Naichen Shi NAICHENS@UMICH.EDU
Raed Al Kontar ALKONTAR@UMICH.EDU
Department of Industrial & Operations Engineering
University of Michigan
Ann Arbor, MI 48109-2117, USA

Editor: Martin Jaggi

Abstract

In this paper, we tackle a significant challenge in PCA: heterogeneity. When data are collected from different sources with heterogeneous trends while still sharing some congruency, it is critical to extract shared knowledge while retaining the unique features of each source. To this end, we propose personalized PCA (`PerPCA`), which uses mutually orthogonal global and local principal components to encode both unique and shared features. We show that, under mild conditions, both unique and shared features can be identified and recovered by a constrained optimization problem, even if the covariance matrices are immensely different. Also, we design a fully federated algorithm inspired by distributed Stiefel gradient descent to solve the problem. The algorithm introduces a new group of operations called generalized retractions to handle orthogonality constraints, and only requires global PCs to be shared across sources. We prove the linear convergence of the algorithm under suitable assumptions. Comprehensive numerical experiments highlight `PerPCA`'s superior performance in feature extraction and prediction from heterogeneous datasets. As a systematic approach to decouple shared and unique features from heterogeneous datasets, `PerPCA` finds applications in several tasks, including video segmentation, topic extraction, and feature clustering.

Keywords: Principal component analysis, personalization, heterogeneity.

1. Introduction

Principal component analysis (PCA) (F.R.S., 1901; Hotelling, 1933) unravels data features by finding a few principal components (PCs) from high dimensional data that explain the largest portion of the variance. Due to its effective feature learning and dimension reduction capability, PCA has seen immense success across various domains, including image processing (Deledalle et al., 2011; Jégou and Chum, 2012), time series modeling (Yang and Shahabi, 2004; Aguilera et al., 1999), bio-information (Reich et al., 2008; Novembre and Stephens, 2008), condition monitoring (Pozo et al., 2018; Li et al., 2018b), and many more.

However, since all data are equally weighted in standard PCA, an underlying assumption is that these data come from homogeneous distributions. This assumption, however, is often challenged in various scenarios, including the Internet of Things (IoT), where data do not come from a single source but a large number of distinct edge devices (or clients). The edge devices, from smartphones to connected vehicles, usually operate in different environments and conditions (Kontar et al., 2017, 2018). The data collected by edge devices are also

subject to changes in external conditions (Kontar et al., 2021) or user preferences (Kulkarni et al., 2020). Thus, it is common for the datasets to contain significant heterogeneity and even conflicting trends while sharing some congruity.

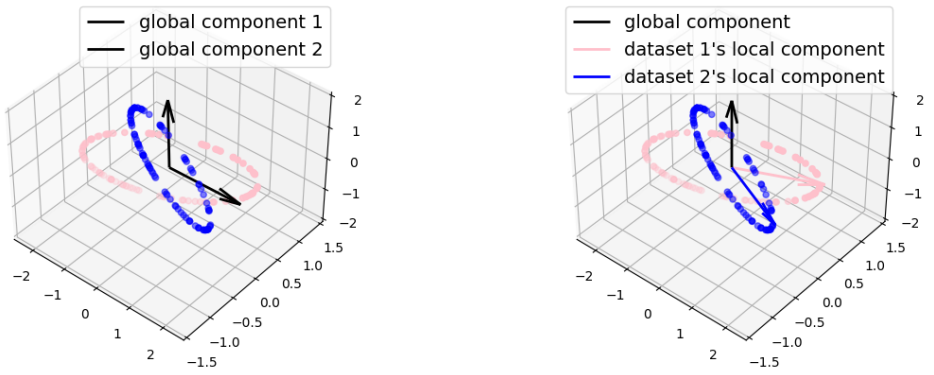
Standard PCA often does not work well when data homogeneity is not guaranteed (Oba et al., 2007; Hong et al., 2021). Few works have endeavored to extend the PCA philosophy to incorporate data heterogeneity. For example, Heterogeneous PCA (Oba et al., 2007) considers the case where data from different sources have different noise levels. They propose a reweighting technique to alleviate noise heteroscedasticity. Such an approach is shown to be useful in identifying PCs from heteroscedastic noises. However, simply treating the discrepancy among datasets as different levels of noise might be inadequate to understand the intrinsic features within the data and insufficient to encode both unique and shared features across devices and clients. As such, personalized solutions are needed.

To transmute the heterogeneity from a bane into a blessing, in this work, we propose personalized PCA (`PerPCA`) that fits personalized features on each client in addition to common features shared by all clients. In our model, data are driven by several mutually orthogonal global (shared) and local (personalized) PCs. The global PCs model the common patterns among different datasets, while the local PCs model the idiosyncratic features of one specific dataset. Global and local PCs work together to fit the observations. Figure 1 is an illustration of using homogeneous PCA and personalized PCA to fit two heterogeneous datasets. As shown in the figure, simply pooling together all data across datasets using homogeneous PCA will fail to encode the unique features within each dataset, and the horizontal PC is a misleading one that is not representative of any source. In contrast, personalized PCA aims at decoupling unique and shared features so that heterogeneity across data sources is accounted for.

There are several benefits to personalization. Firstly, employing several local PCs to fit individual data patterns enables us to describe immensely heterogeneous trends in datasets accurately. Also, global PCs shared by all data can be estimated more precisely without being affected by disagreeing drifts from different sources. What's more, disentangling local features from global ones provides a systematic and interpretable approach to analyzing the heterogeneity structure of datasets and leveraging this knowledge for better analytics. These include: (i) *Improving classification and clustering*: instead of using raw data, operating on unique features may yield better performance as differences become more explicit when removing shared features, (ii) *Transforming personalized, predictive analytics*: Through selectively transferring common knowledge from one data source to another, we can reduce the negative transfer of knowledge and enhance personalized predictive and prescriptive models, (iii) *Anomaly Detection*: Through monitoring changes in the unique features, we envision that anomalies can be better and faster detected.

To enable personalized PCA, we propose an optimization framework to provably recover both global and local PCs from noisy observations. The objective is to minimize the empirical reconstruction error under orthogonality constraints. The formulation stands on solid theoretical ground: We prove that, under an identifiability condition, the optimal solution can recover the true global and local PCs.

Not only can the PCs be solved, but they can also be solved efficiently. We design an algorithm based on Stiefel manifold gradient descent that can be proved to converge linearly into the global optimum under mild conditions. The algorithm relies on a new operation



Homogeneous PCA Personalized PCA

Figure 1: Comparison between homogeneous PCA (standard PCA) and personalized PCA (`PerPCA`). There are two datasets, one colored blue and the other pink. Dots represent the observations. Observations from one dataset are on a 2-dimensional plane. The black arrows represent the global PCs learned, and the colored arrows represent learned local PCs. Homogeneous PCA is a standard PCA on the pooled dataset. We will revisit the example in Section 7.

called generalized retraction to handle the orthogonality constraints. It is worth noting that our algorithm is designed in a *federated manner*, as the need to share raw data or place all data in a central location is circumvented, and only the updates of global PCs need to be shared across clients. Compared with centralized PCA, where all datasets are uploaded to a central server where PCA is learned on the aggregated dataset, our algorithm reaps the benefits of distributed and federated analytics. Those include communication, cost, storage, and privacy benefits (Kontar et al., 2021). We will show the advantages of `PerPCA` over existing distributed PCA methods in Section 2.1.

Furthermore, `PerPCA` proposes a novel provable paradigm of decoupling shared and unique features. Its applications go beyond simple data dimension reduction. We show that `PerPCA` has remarkable performance in video segmentation and topic extraction tasks. Hence `PerPCA` opens up new possibilities for broader applications.

Moving forward, we will use client, edge device, data source, and local dataset interchangeably to represent the entities of interest. Here, entities are broadly defined, encompassing various levels of granularity. For instance, we can extract shared and unique features across dispersed datasets, output classes within a dataset, or even among observations (such as images) within a single dataset.

1.1 Main contributions

We summarize our contributions in the following:

- **Modeling**: We propose a personalized PCA model that learns both global and local features from *distributed* datasets. These features can be recovered from observations by solving a nonconvex optimization problem designed to minimize reconstruction error.

- **Consistency**: We find that there exists a simple sufficient condition based on the "misalignment" of local PCs to ensure the identifiability of the global and local PCs: the maximum eigenvalue of the average of projections into local subspaces should be smaller than 1. We show that, under the identifiability condition, both global and local PCs can be estimated from noisy observations with an error that is upper bounded by $O(\frac{1}{n})$, where n is the number of observations on each client. As the error decreases to 0 when n approaches infinity, the error bound essentially implies the consistency of `PerPCA`. The analysis extends conventional matrix perturbation bounds (Bhatia, 1997; Vu et al., 2013) into personalized settings where the change in one client's covariance matrix can affect the PC estimates on all clients. We also use a minimax statistical lower bound to show that the statistical error upper bound is almost tight in terms of the eigengap and misalignment parameter.

- **Algorithm**: We design an algorithm based on Stiefel manifold gradient descent (St-GD hereon) to obtain global and local PC estimates. The major difficulty for the algorithm is handling the orthogonality constraints. To tackle it, we introduce a correction step that relies on a group of operations called *generalized retractions*. A generalized retraction extends retraction in literature (Edelman et al., 1998) as it is defined on the entire $\mathbb{R}^{d\times r}$ rather than the tangent bundle of the Stiefel manifold $St(d, r)$. In our algorithm, clients only need to share iterates of global PCs, thus preserving privacy and minimizing communication costs.

- **Convergence**: The proposed algorithm has a local linear convergence rate. To our best knowledge, this is the first theoretical guarantee for an algorithm that simultaneously learns global and local PCs. Interestingly, the convergence is faster when local PCs are more heterogeneous, a result that lies in sharp contrast to conventional predictive federated or transfer learning (Zhuang et al., 2020) theory as it highlights that heterogeneity can be a blessing in disguise. On the technical side, we introduce a novel Lyapunov function to study distributed St-GD with generalized retractions.

- **Numerical results**: Empirical evidence on both synthetic and real-life datasets confirms `PerPCA`'s ability to decouple shared and unique features. Also, `PerPCA` has exciting applications in video segmentation and topic extraction. For instance, on video segmentation tasks, `PerPCA` has significant advantages over the popular `Robust PCA` (Candes et al., 2011) when heterogeneity patterns are not sparse.

1.2 Organization

The paper is organized as follows: We review related work and introduce notations in Section 2. In Section 3, we propose the formulation of `PerPCA` and link it with constrained optimization. Section 4 includes the theoretical analysis on identifiability and consistency. A federated algorithm to solve `PerPCA` is developed in Section 5, and its convergence guarantee

is established in Section 6. Numerical experimentation results are demonstrated in Section 7. Finally, Section 8 concludes the paper with a brief discussion. Readers mainly interested in the implementation and applications of `PerPCA` can focus on Sections 3, 5, and 7. An implementation of the proposed method is in the linked Github repository.

2. Preliminaries

In this section, we will review related work in the literature and introduce needed notations.

2.1 Related work

Structural PCA Structural PCA attempts to build structural models for data and noise. Research on structural PCA abounds. A seminal algorithm along this line is `Robust PCA` (Candes et al., 2011). The authors point out that traditional PCA is sensitive to noise in the observations and tackle this issue by decomposing an observation matrix $\boldsymbol{Y}$ into a low-rank part $\boldsymbol{L}$ and a sparse noise part $\boldsymbol{S}$: $\boldsymbol{Y} = \boldsymbol{L} + \boldsymbol{S}$. The low-rank matrix $\boldsymbol{L}$ corresponds to the signal, and $\boldsymbol{S}$ represents the noise. It turns out that the two parts can be exactly identified under regularity conditions with carefully designed algorithms. `Robust PCA` has become a useful technique in image denoising and video processing (Bouwmans et al., 2018), collaborative filtering (Xu et al., 2012), and many more. Sparse PCA (Zou et al., 2006) adds sparse constraints on the PCs, encouraging each PC to depend on a minimal number of variables. While these methods are powerful in handling large noise or high dimensional data, they mainly analyze homogeneous data.

Several algorithms have also been invented to leverage variance heterogeneity in different samples. Heterogeneous component analysis (`HCA`) (Oba et al., 2007) assumes data come from different sources with different levels of noise. To better learn the PCs with heteroscedastic variance, `HCA` reweights the empirical loss of each observation according to the inverse of its variance so that noisier samples contribute less to the total loss. Hong et al. (2021) calculates the optimal weights in the asymptotic case by considering the signal-to-noise ratio. Though these methods have superior performance compared to uniform weighting PCA, heterogeneity among different sources is only modeled by the noise magnitude. A few heuristic methods also attempt to use low-rank features to characterize heterogeneity, including joint and individual variance explained (`JIVE`) (Lock et al., 2013), common and individual feature extraction (`CIFE`) (Zhou et al., 2015). However, it is difficult to distribute these methods for federated learning and provide theoretical guarantees for their outputs.

Distributed PCA There has been a recent push to calculate PCs on distributed devices. Oftentimes, the clients/edge devices use their local data to estimate PCs and communicate with a central server to update their estimates. One round of information exchange between clients and the central server is referred to as a communication round. Based on the number of communication rounds between edge devices and the server, research can be roughly divided into two categories (1) those that require only one round of communication and (2) those that require multiple rounds of communication.

For one-round PCA algorithms, clients estimate PCs from local datasets and send summary statistics to the server. The server then analyzes the aggregated statistics to calculate the PCs of the entire dataset. There are several ways for the server to calculate PCs. Qu et al. (2002) proposes a method to reconstruct the aggregated covariance matrix

by averaging the clients' covariance matrices approximated by a few top PCs. Global PCs can be obtained by learning the top eigenvalues of the averaged covariance matrix. `distPCA` (Fan et al., 2019) provides an alternative approach, where the server stacks locally calculated PCs into a large matrix and runs another PCA on the stacked matrix. Liang et al. (2014) uses a similar method, where clients calculate a singular value decomposition (SVD) of the local observation matrix, and then send the singular values and singular vectors to the server. The server stacks the scaled singular vectors and runs SVD on the stacked matrix. Federated PCA (Grammenos et al., 2020) considers streaming data applications where edge devices have limited memory budgets. In their work, locally estimated subspaces are hierarchically merged to form the global subspace. Feldman et al. (2013) also focuses on streaming data and reduces large datasets into smaller ones. In spite of the reductions in communication or memory cost, these algorithms are often not guaranteed to recover true PCs exactly. Also, they are built upon homogeneity assumptions and neglect statistical heterogeneity among the distributed datasets.

To obtain more refined estimates of PCs from distributed datasets, a series of works propose to use multiple rounds of communication (Chen et al., 2020; Garber et al., 2017; Huang and Pan, 2020; Alimisis et al., 2021). Among them, Chen et al. (2020) and Garber et al. (2017) design PC updates by shift-and-invert iterations. The shift-and-inverse method (Garber and Hazan, 2015) reformulates inverse power iteration as an unconstrained convex optimization problem and uses gradient-based iterative algorithms to solve it. With a similar rationale, Chen et al. (2020) applies the shift-and-invert formulation to distributed settings and applies distributed Newton methods to solve for the top eigenvector of the covariance matrix. Then, the covariance matrix is deflated to calculate the subsequent eigenvectors. Besides shift-and-invert iterations, manifold optimization is also employed for PCA. Huang and Pan (2020) uses distributed Riemann optimization to find top PCs from homogeneous datasets. To further reduce communication costs, Alimisis et al. (2021) combines quantized distributed optimization and Riemannian gradient descent with an exponential map to calculate the leading eigenvectors of the covariance matrix. These methods usually treat the difference among clients' covariance matrices as errors. Thus, when datasets are heterogeneous, the errors are large, and these algorithms fail to retrieve true PCs.

Gradient descent on manifolds The centralized version of gradient descent on manifolds, or Riemaniann gradient descent, has been well-studied (Absil et al., 2008; Boumal, 2022). Algorithms based on exponential mappings (Edelman et al., 1998) can achieve convergence rates comparable to their Euclidean counterparts. Since exponential mappings are expensive to compute, there are algorithms that replace them with retractions. Tang (2019) presents an elegant framework for analyzing kPCA by Riemannian gradient descent with Cayley retraction. This work proves the local linear convergence of Stiefel gradient descent and also shows that the algorithm can exactly recover the top eigenspaces.

Recent years have also seen advances in distributed manifold optimization. Chen et al. (2021a,b) introduces a simple distributed St-GD algorithm that minimizes a general objective on the manifold. In each round, clients use St-GD on the local objectives and send the updated variables to the server, then the server averages the received update and applies a retraction. The algorithm is guaranteed to converge into stationary points with a sublinear rate.

PerPCA also exploits St-GD to solve PCs. However, our algorithm enhances simple manifold optimization by simultaneously optimizing local and global PCs, while also incorporating orthogonality constraints between the global and local PCs. PerPCA thus introduces a special correction step to handle such constraints. This is done by defining a new retraction measure we name as a *generalized retraction* defined on the entire $\mathbb{R}^{d\times r}$ rather than the tangent bundle of Stiefel manifold $St(d,r)$.

We should note that among all the distributed algorithms discussed, only PerPCA models different or distributed datasets by global and local PCs. Thus, it brings unique advantages in decoupling local and global features from highly heterogeneous datasets. Besides, there are several additional benefits of PerPCA in convergence and computation compared with typical existing models. In terms of convergence, PerPCA converges into stationary points of the empirical reconstruction error and is guaranteed to recover true PCs exactly with proper initialization. The algorithm does not involve a computationally intensive exponential map and can solve k PCs at one time. More importantly, PerPCA is fully federated, and different clients can collaborate by only sharing a few global PCs that encode shared and not unique features. The comparisons of PerPCA and several typical PCA algorithm is summarized in Table 1.

Method	Source	Exact convergence	kPCA	Federated	Personalized
Robust PCA	(Candes et al., 2011)	✓	✓	✗	✗
JIVE	(Lock et al., 2013)	✗	✓	✗	✓
distPCA	(Fan et al., 2019)	✗	✓	✓	✗
Distri-Eigen	(Chen et al., 2020)	✓	✗	✓	✗
CEDRE	(Huang and Pan, 2020)	✓	✗	✓	✗
PCA by St-GD	(Tang, 2019)	✓	✓	✗	✗
PerPCA	**ours**	✓	✓	✓	✓

Table 1: Comparison of related work. Metrics included and their definitions are: (i) Exact convergence: the algorithm can recover top subspaces of sample covariance matrix exactly, (ii) kPCA: the algorithm can calculate the subspace spanned by top k PCs instead of one single component, (iii) Federated: the algorithm can be done in a distributed fashion where raw data remains where it is generated on the edge and only focused updates need to be shared across clients, (iv) Personalized: the algorithm encodes both shared and unique features across all datasets.

2.2 Notations

We first introduce needed notations in this subsection. For a d-dimensional vector $\boldsymbol{x}$, we use $\|\boldsymbol{x}\|$ to denote its 2-norm. The inner product of two vectors is defined as a standard inner product in Euclidean space: $\langle \boldsymbol{x}, \boldsymbol{y}\rangle = \boldsymbol{x}^T\boldsymbol{y}$. We use $\boldsymbol{I}_d$ to denote the identity matrix in $\mathbb{R}^d$. We sometimes omit the subscript d if the dimension is clear from the context. For a real matrix $\boldsymbol{A} \in \mathbb{R}^{m\times n}$, we use $\|\boldsymbol{A}\|_F$ to denote its Frobenius norm $\|\boldsymbol{A}\|_F = \sqrt{\sum_{i=1}^n \sum_{j=1}^m \boldsymbol{A}_{ij}^2}$

and $\|\boldsymbol{A}\|_{op}$ to denote its operator norm $\|\boldsymbol{A}\|_{op} = \max_{\boldsymbol{v}\in\mathbb{R}^n, \|\boldsymbol{v}\|=1} \|\boldsymbol{A}\boldsymbol{v}\|$. For two matrices $A, B \in \mathbb{R}^{m\times n}$, we define their inner product as $\langle \boldsymbol{A}, \boldsymbol{B}\rangle = \sum_{i=1}^{n}\sum_{j=1}^{m} A_{ij}B_{ij} = \mathrm{Tr}\left(\boldsymbol{A}^T\boldsymbol{B}\right)$.

If $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ is symmetric positive definite (PSD), it has an eigendecomposition $\boldsymbol{A} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{U}^T$, where $\boldsymbol{D}$ is n by n diagonal matrix whose diagonal entries are all positive, $\boldsymbol{U}$ is a n by n unitary matrix. Then for $p \in \mathbb{R}$, the p-th power of $\boldsymbol{A}$ is defined as $\boldsymbol{A}^p = \boldsymbol{U}\boldsymbol{D}^p\boldsymbol{U}^T$. For a square matrix $\boldsymbol{A} \in \mathbb{R}^{n\times n}$, we use $\lambda_{min}(\boldsymbol{A})$ and $\lambda_{max}(\boldsymbol{A})$ to denote the minimum and maximum eigenvalue of $\boldsymbol{A}$. Similarly, we use $\lambda_1(\boldsymbol{A})$, $\lambda_2(\boldsymbol{A})$, ... $\lambda_n(\boldsymbol{A})$ to denote the n eigenvalues of $\boldsymbol{A}$ in descending order. We use $\|\boldsymbol{A}\|_{op}$ and $\lambda_{max}(\boldsymbol{A})$ interchangeably when $\boldsymbol{A}$ is symmetric PSD.

For a matrix $\boldsymbol{A} \in \mathbb{R}^{m\times n}$, we use $\mathrm{vec}\,(\boldsymbol{A}) \in \mathbb{R}^{mn}$ to denote its vectorization, i.e., the vector formed by concatenating all the column vectors in $\boldsymbol{A}$. $col(\boldsymbol{A})$ is the linear subspace spanned by all column vectors of $\boldsymbol{A}$. We use $\boldsymbol{A}_{i_1:i_2,j_1:j_2}$ to denote the submatrix of A formed by picking the $i_1, i_1+1 ... i_2$-th row and $j_1, j_1+1 ... j_2$-th column of $\boldsymbol{A}$. For two matrices $\boldsymbol{A} \in \mathbb{R}^{m\times n_1}$ and $\boldsymbol{B} \in \mathbb{R}^{m\times n_2}$, $[\boldsymbol{A}, \boldsymbol{B}] \in \mathbb{R}^{m\times(n_1+n_2)}$ is defined as the concatenation of $\boldsymbol{A}$ and $\boldsymbol{B}$ by column.

Finally, we use the standard $O\,(\cdot)$, $\Omega\,(\cdot)$, and $o\,(\cdot)$ notations throughout the paper.

3. What is PerPCA?

We will establish the formulation of PerPCA in this section.

3.1 Motivation

Suppose we have N clients (i.e. data sources), each with a dataset $\{\boldsymbol{Y}_{(i)}\}_{i=1}^N$, where $\boldsymbol{Y}_{(i)}$ is a d by n_i matrix. d is the dimension of data, and n_i is the the number of datapoints on client i. The datasets $\{\boldsymbol{Y}_{(i)}\}_{i=1}^N$ have commonalities but also possess client-level distinctive features. The task is to find a few low-dimensional common and unique features that best characterize the observations from the high dimensional data $\{\boldsymbol{Y}_{(i)}\}_{i=1}^N$.

Standard PCA uses a small number of principal components (PCs) to explain the variations in $\{\boldsymbol{Y}_{(i)}\}_{i=1}^N$. Such treatment ignores the client-to-client difference in the observations. The present IoT system usually consists of distributed edge devices (clients) that operate in extremely heterogeneous environments. It is thus important to consider the different features of different clients. As a more capacious description of the data, we consider the model where local observations are driven by r_1 global PCs and $r_{2,(i)}$ local PCs. More specifically, from data source i, observation $\boldsymbol{y}_{(i)}$ is generated from

$$\boldsymbol{y}_{(i)} \sim \sum_{q=1}^{r_1} \phi_{(i),q}\boldsymbol{u}_q + \sum_{q=1}^{r_{2,(i)}} \varphi_{(i),q}\boldsymbol{v}_{(i),q} + \boldsymbol{\epsilon}_{(i)} \tag{1}$$

where $\phi_{(i),q}$'s and $\varphi_{(i),q}$'s are coefficients, or scores in PCA terminology. $\boldsymbol{u}_q$'s are global PCs, $\boldsymbol{v}_{(i),q}$'s are local PCs, and $\boldsymbol{\epsilon}_{(i)}$ are i.i.d. noise vectors. r_1 is the number of global PCs, and $r_{2,(i)}$ is the number of local PCs on client i. We allow $\boldsymbol{v}_{(i),q}$'s to be client-dependent while enforcing $\boldsymbol{u}_q$'s to remain the same across all clients. Naturally, $\boldsymbol{u}_q$'s encode the information shared by all participants, while $\boldsymbol{v}_{(i),q}$'s can describe distinctive patterns on each client.

Similar to standard PCA, different principal components need to be orthonormal:

$$\begin{cases} \boldsymbol{u}_{q_1}^T \boldsymbol{u}_{q_2} = \delta_{q_1,q_2} \\ (\boldsymbol{v}_{(i),q_1})^T \boldsymbol{v}_{(i),q_2} = \delta_{q_1,q_2},\ \forall q_1, q_2,\ \forall i = 1, \cdots, N \end{cases} \tag{2}$$

where δ_{q_1,q_2} is the Kronecker delta. In addition to (2), we further require that the global and local features are orthogonal:

$$\boldsymbol{u}_{q_1}^T \boldsymbol{v}_{(i),q_2} = 0,\ \forall i = 1, \cdots, N \tag{3}$$

The orthogonality of PCs implies that the shared and unique features span different subspaces, thus describing independent and decoupled patterns in the data sources.

(1) is an interpretable linear model that naturally incorporates both common and individual features of different clients. It is useful in applications where disentangling global and local features is important. The development of IoT and recent advancements in federated and distributed analytics present numerous such applications, including time series data, image and video data, and language data. We will show the efficacy of (1) on several examples.

3.2 Method

The task of `PerPCA` is to recover global and local PCs from observations $\{\boldsymbol{Y}_{(i)}\}_{i=1}^N$. We can write global and local PCs into matrix form:

$$\begin{cases} \boldsymbol{U} = [\boldsymbol{u}_1, \cdots, \boldsymbol{u}_{r_1}] \\ \boldsymbol{V}_{(i)} = [\boldsymbol{v}_{(i),1}, \cdots, \boldsymbol{v}_{(i),r_{2,(i)}}] \end{cases} \tag{4}$$

and solve for $\boldsymbol{U}$ and $\boldsymbol{V}_{(i)}$'s by minimizing the empirical reconstruction loss:

$$\begin{gathered} \min_{\boldsymbol{U}, \{\boldsymbol{V}_{(i)}\}_{i=1,\cdots,N}} \frac{1}{2} \sum_{i=1}^N \frac{1}{n_i} \left\| \boldsymbol{Y}_{(i)} - \hat{\boldsymbol{Y}}_{(i)} \right\|_F^2 \\ \text{subject to } \boldsymbol{U}^T \boldsymbol{U} = \boldsymbol{I},\ \boldsymbol{V}_{(i)}^T \boldsymbol{V}_{(i)} = \boldsymbol{I},\ \boldsymbol{V}_{(i)}^T \boldsymbol{U} = \boldsymbol{0},\ \forall i \end{gathered} \tag{5}$$

where $\hat{\boldsymbol{Y}}_{(i)}$ is the statistical fit for client i's data given PCs $\boldsymbol{U}$ and $\boldsymbol{V}_{(i)}$:

$$\hat{\boldsymbol{Y}}_{(i)} = \boldsymbol{U}\boldsymbol{U}^T \boldsymbol{Y}_{(i)} + \boldsymbol{V}_{(i)} \boldsymbol{V}_{(i)}^T \boldsymbol{Y}_{(i)} \tag{6}$$

Intuitively, in (5), we look for the PCs so that the predicted $\hat{\boldsymbol{Y}}_{(i)}$ can best fit the distributed datasets. The objective (5) has another interpretation: by some algebra, we can transform the objective (5) into:

$$\begin{gathered} \max_{\boldsymbol{U}, \{\boldsymbol{V}_{(i)}\}_{i=1,\cdots,N}} \frac{1}{2} \sum_{i=1}^N \left[\operatorname{Tr}\left(\boldsymbol{U}^T \boldsymbol{S}_{(i)} \boldsymbol{U}\right) + \operatorname{Tr}\left(\boldsymbol{V}_{(i)}^T \boldsymbol{S}_{(i)} \boldsymbol{V}_{(i)}\right) \right] \\ \text{subject to } \boldsymbol{U}^T \boldsymbol{U} = \boldsymbol{I},\ \boldsymbol{V}_{(i)}^T \boldsymbol{V}_{(i)} = \boldsymbol{I},\ \boldsymbol{V}_{(i)}^T \boldsymbol{U} = \boldsymbol{0},\ \forall i \end{gathered} \tag{7}$$

where $\boldsymbol{S}_{(i)}$ is defined as the data covariance matrix:

$$\boldsymbol{S}_{(i)} = \frac{1}{n_i}\boldsymbol{Y}_{(i)}\boldsymbol{Y}_{(i)}^T$$

From (7), it is clear that `PerPCA` attempts to find global and local low dimensional subspaces that best align with the data covariance matrix. We will study objective (7) from here on.

For simplicity, we introduce

$$f_i(\boldsymbol{U}, \boldsymbol{V}_{(i)}) = \frac{1}{2}\mathrm{Tr}\left(\boldsymbol{U}^T\boldsymbol{S}_{(i)}\boldsymbol{U}\right) + \frac{1}{2}\mathrm{Tr}\left(\boldsymbol{V}_{(i)}^T\boldsymbol{S}_{(i)}\boldsymbol{V}_{(i)}\right) \tag{8}$$

and

$$f(\boldsymbol{U}, \{\boldsymbol{V}_{(i)}\}) = \sum_{i=1}^{N} f_i(\boldsymbol{U}, \boldsymbol{V}_{(i)}) \tag{9}$$

Then (7) transforms to maximizing f under orthonormality constraints. Notice that though f and f_i's are convex, the constraint in (7) is nonconvex. Thus, the problem is nonconvex.

The nonconvex formulation (7) appears difficult to analyze and solve. In the following sections, we will delve into the identifiability and optimization of (7). Fortunately, our results show that under minimal conditions, (7) can be solved efficiently, and the optimal solution can recover the true PCs.

4. Are Global and Local PCs Identifiable?

Given the formulation (7), one may ask whether it is possible to identify the true local and global PCs by solving (7).

Apparently, global and local PCs cannot be decoupled in every case. As a simple counterexample, if all local PCs are the same, then distinguishing local from global PCs is impossible, as there are infinite combinations of them that all can maximize the explained variance in (7). The edifying counterexample poses the fundamental question of model identifiability. Therefore, we need to find out which data instances are identifiable. In the following, we will introduce an identifiability condition, then establish the relationship between the estimated and true PCs.

We restrict our analysis to recovering the subspace spanned by top PCs (Bhatia, 1997). Therefore we introduce the projection matrix notation $\boldsymbol{P}_{\boldsymbol{U}}$: if $\boldsymbol{U}$ is a matrix with orthonormal columns, i.e. $\boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{I}$, then $\boldsymbol{P}_{\boldsymbol{U}}$ is defined as $\boldsymbol{P}_{\boldsymbol{U}} = \boldsymbol{U}\boldsymbol{U}^T$. We use $\boldsymbol{\Pi}_g$ to denote the projection matrix to the true global eigenspace, i.e., $\boldsymbol{\Pi}_g = \boldsymbol{P}_{\boldsymbol{U}_{\text{true}}}$, where $\boldsymbol{U}_{\text{true}}$ are the true top global PCs. Also, we use $\boldsymbol{\Pi}_{(i)}$ to denote the projection matrix to the true local eigenspace, $\boldsymbol{\Pi}_{(i)} = \boldsymbol{P}_{\boldsymbol{V}_{(i),\text{true}}}$, where $\boldsymbol{V}_{(i),\text{true}}$ are the true top local PCs on client i.

Remember, we model global and local PCs as mutually vertical features; such property can be formally characterized by the following assumption.

Assumption 4.1 *(Orthogonality of global and local PCs) Let $\boldsymbol{\Pi}_g$ be the global projection matrix, $\boldsymbol{\Pi}_{(i)}$'s the local projection matrices. We assume that*

$$\boldsymbol{\Pi}_g\boldsymbol{\Pi}_{(i)} = 0 \tag{10}$$

In addition, we consider the case where the subspace corresponding to the projection $\mathbf{\Pi}_g + \mathbf{\Pi}_{(i)}$ *is indeed an invariant subspace of the population covariance matrix on client* i, $\mathbf{\Sigma}_{(i)}$ *, i.e.* $\left(\mathbf{\Pi}_g + \mathbf{\Pi}_{(i)}\right) \mathbf{\Sigma}_{(i)} = \mathbf{\Sigma}_{(i)} \left(\mathbf{\Pi}_g + \mathbf{\Pi}_{(i)}\right)$.

In Assumption 4.1, the requirement $\left(\mathbf{\Pi}_g + \mathbf{\Pi}_{(i)}\right) \mathbf{\Sigma}_{(i)} = \mathbf{\Sigma}_{(i)} \left(\mathbf{\Pi}_g + \mathbf{\Pi}_{(i)}\right)$ essentially assumes that $\boldsymbol{U}_{true}$ and $\boldsymbol{V}_{(i),true}$ are indeed the eigenvectors of the population covariance matrix $\mathbf{\Sigma}_{(i)}$.

As the counterexample suggests, assumption 4.1 alone is insufficient to guarantee the identifiability of global and local PCs. To distinguish them, we need another identifiability condition. To rule out the counterexample, local PCs and accordingly $\mathbf{\Pi}_{(i)}$, should differ from each other. To this end, we introduce the notion of "*misalignment*". Misalignment is quantified by the parameter θ, which represents the maximum eigenvalue of the average of the local projection matrices. Assumption 4.2 is a formal statement of the identifiability condition.

Assumption 4.2 *(Misalignment) Let* $\mathbf{\Pi}_{(i)}$*'s be the local projection matrices. We assume there exists a positive constant* $\theta \in (0, 1)$ *such that:*

$$\lambda_{\max}\left(\frac{1}{N}\sum_{i=1}^{N}\mathbf{\Pi}_{(i)}\right) \leq 1 - \theta \tag{11}$$

The constant θ characterizes the misalignment between local principal spaces. When θ is larger, the local eigenspaces are more heterogeneous. When θ is smaller, the local eigenspaces are more similar. As an extreme case, if all $\mathbf{\Pi}_{(i)}$'s are identical, $\frac{1}{N}\sum_{i=1}^{N}\mathbf{\Pi}_{(i)}$ is still a projection, thus its maximum eigenvalue is 1 and θ becomes zero.

4.1 Statistical error

It turns out that the identifiability Assumption 4.2 is sufficient to ensure identifiability. The following perturbation bound shows that when the sample covariance matrix is close to the population covariance matrix, we can obtain relatively accurate estimates of global and local eigenspaces through solving (7).

Theorem 1 *Under assumption 4.1 and 4.2, and if there exists a constant* $\delta > 0$*, such that* $\lambda_{r_1+r_{2,(i)}}\left(\left(\mathbf{\Pi}_g + \mathbf{\Pi}_{(i)}\right)\mathbf{\Sigma}_{(i)}\right) - \lambda_1\left(\left(\boldsymbol{I} - \mathbf{\Pi}_g - \mathbf{\Pi}_{(i)}\right)\mathbf{\Sigma}_{(i)}\right) \geq \delta$ *for all* i*, we have:*

$$\left\|\boldsymbol{P}_{\hat{\boldsymbol{U}}} - \mathbf{\Pi}_g\right\|_F^2 + \frac{1}{N}\sum_{i=1}^{N}\left\|\boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i)}} - \mathbf{\Pi}_{(i)}\right\|_F^2 \leq \frac{8}{\theta\delta^2}\frac{1}{N}\sum_{i=1}^{N}\left\|\mathbf{\Sigma}_{(i)} - \boldsymbol{S}_{(i)}\right\|_F^2 \tag{12}$$

where $\hat{\boldsymbol{U}}$*, and* $\hat{\boldsymbol{V}}_{(i)}$*'s are the optimal solutions to the objective in* (7)*.*

δ is usually called eigengap in literature (Vu et al., 2013; Huang and Pan, 2020). The δ^{-2} factor on the right-hand side of (12) is standard for matrix perturbation analysis.

Theorem 1 confirms the intuition on identifiability. Specifically, as θ increases, the right-hand side of equation (12) decreases, resulting in a smaller estimation error. Consequently, finding local and global PCs becomes easier. *This result critically highlights that heterogeneity can be a blessing.* For the counterexample, $\theta \to 0$, the right-hand side approaches infinity. Hence, one cannot accurately recover the PCs.

In addition, Theorem 1 *highlights the benefits of collaborative learning* across multiple related clients. The right-hand side of (12) is the average difference between the sample and population covariance matrix on all clients. For clients with a larger dataset, the distance is lower, and for clients with a smaller dataset, the distance can be higher. Through jointly optimizing objective (7), clients learn from each other and obtain PC estimates with statistical error depending on the average distance.

4.2 Minimax statistical lower bound

Though the statistical error bound provided in Theorem 1 is intuitive, it is not apparent whether the bound is sharp. To fully understand the statistical difficulty in recovering shared and unique components from $\{\boldsymbol{S}_{(i)}\}$, we will establish a lower bound on the minimax risk of estimators under the subspace error.

For simplicity, we define the subspace error between $\{\hat{\boldsymbol{U}}, \{\hat{\boldsymbol{V}}_{(i)}\}\}$ and $\{\boldsymbol{U}, \{\boldsymbol{V}_{(i)}\}\}$ as

$$L_{\texttt{subspace}}\left(\{\hat{\boldsymbol{U}}, \{\hat{\boldsymbol{V}}_{(i)}\}\}, \{\boldsymbol{U}, \{\boldsymbol{V}_{(i)}\}\}\right) = \left\|\boldsymbol{P}_{\hat{\boldsymbol{U}}} - \boldsymbol{P}_{\boldsymbol{U}}\right\|_F^2 + \frac{1}{N}\sum_{i=1}^{N}\left\|\boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i)}} - \boldsymbol{P}_{\boldsymbol{V}_{(i)}}\right\|_F^2 \tag{13}$$

Additionally, we use Θ to denote the parameter space specified by Assumption 4.1,

$$\Theta = \left\{\boldsymbol{U}, \{\boldsymbol{V}_{(i)}\} | \boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{I}, \boldsymbol{V}_{(i)}^T\boldsymbol{V}_{(i)} = \boldsymbol{I}, \boldsymbol{U}^T\boldsymbol{V}_{(i)} = 0\right\} \tag{14}$$

The following theorem provides a lower bound for the statistical error.

Theorem 2 *If the data generation process satisfies Assumptions 4.1 and 4.2, the eigengap introduced in Theorem 1 is at least δ, and $\sum_{i=1}^{N}\left\|\boldsymbol{S}_{(i)} - \boldsymbol{\Sigma}_{(i)}\right\|_F^2 = o(1)$, then among data generated by all possible $\{\boldsymbol{U}_{true}, \{\boldsymbol{V}_{(i),true}\}\} \in \Theta$, the supremum of the subspace error between the optimal solution to (7), $\{\hat{\boldsymbol{U}}, \{\hat{\boldsymbol{V}}_{(i)}\}\}$, and the ground truth, $\{\boldsymbol{U}_{true}, \{\boldsymbol{V}_{(i),true}\}\}$, is at least*

$$\sup_{\{\boldsymbol{U}_{true}, \{\boldsymbol{V}_{(i),true}\}\} \in \Theta} \frac{L_{subspace}\left(\{\hat{\boldsymbol{U}}, \{\hat{\boldsymbol{V}}_{(i)}\}\}, \{\boldsymbol{U}_{true}, \{\boldsymbol{V}_{(i),true}\}\}\right)}{\frac{1}{N}\sum_{i=1}^{N}\left\|\boldsymbol{\Sigma}_{(i)} - \boldsymbol{S}_{(i)}\right\|_F^2} = \Omega\left(\frac{1}{\theta} + \frac{1}{\delta^2}\right) \tag{15}$$

Theorem 2 measures the subspace error minimax lower bound in terms of misalignment parameter θ and eigengap δ. Roughly speaking, the lower bound is greater than $\Omega\left(\left(\frac{1}{\theta} + \frac{1}{\delta^2}\right)\frac{1}{N}\sum_{i=1}^{N}\left\|\boldsymbol{\Sigma}_{(i)} - \boldsymbol{S}_{(i)}\right\|_F^2\right)$. This almost matches the upper bound provided in Theorem 1 as the error scales with $\frac{1}{\theta}$ and $\frac{1}{\delta^2}$. Theorem 2 also demonstrates the intrinsic statistical difficulty of separating global and local PCs. When the local PCs are more aligned and noise components grow larger, θ and δ become smaller, and the statistical error of the subspace estimate becomes larger accordingly.

The proof of Theorem 2 is based on a variant of the "spiked population model" (Birnbaum et al., 2013). We use perturbation analysis to calculate the leading order of the subspace error when the sample covariance is close to the population covariance. The full proof is in Appendix C. There is also a comparison between the theoretical statistical error estimate and the statistical error obtained from numerical simulations in Appendix C.

4.3 Sample complexity

In this section, we estimate the statistical error when data are generated by a sub-Gaussian distribution. A random vector $\boldsymbol{y} \in \mathbb{R}^d$ admits a sub-Gaussian distribution with parameter σ if for each fixed vector $\boldsymbol{v} \in \mathbb{S}^{d-1}$, $\mathbb{E}\left[e^{\lambda\langle \boldsymbol{v}, y\rangle}\right] \leq e^{\frac{\lambda^2\sigma^2}{2}}$ for all $\lambda \in \mathbb{R}$. σ is a parameter that denotes the variance level: when σ is larger the data are noisier. As a special case, if $\boldsymbol{y}$ admits a Gaussian distribution with mean zero and covariance $\boldsymbol{\Sigma}_y$, then $\sigma^2 = \|\boldsymbol{\Sigma}_y\|_{op}$ (Wainwright, 2019). The following corollary gives an upper bound of the estimation error.

Corollary 3 *If the dataset on each client i $\{\boldsymbol{Y}_{(i)}\}_{i=1}^N$ admits an i.i.d. sub-Gaussian distribution with parameter σ, and the assumptions in Theorem 1 are satisfied, then with probability at least $1-\widetilde{\delta}$ (over the randomness of the data generation process), we have:*

$$\left\|\boldsymbol{P}_{\hat{\boldsymbol{U}}} - \boldsymbol{\Pi}_g\right\|_F^2 + \frac{1}{N}\sum_{i=1}^N \left\|\boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i)}} - \boldsymbol{\Pi}_{(i)}\right\|_F^2 \leq \frac{1}{\theta\delta^2}\sigma^4 C^2 \frac{d}{N}\sum_{i=1}^N \max\left\{\left(\frac{d+\log\frac{2N}{\widetilde{\delta}}}{n_i}\right)^2, \frac{d+\log\frac{2N}{\widetilde{\delta}}}{n_i}\right\} \tag{16}$$

where C is a constant.

The inequality (16) essentially shows the consistency of the solutions $\hat{\boldsymbol{U}}$ and $\hat{\boldsymbol{V}}$. When the data dimension d is fixed and sample size n_i is relatively large, the right hand side of (16) decreases with $O\left(\sum_{i=1}^N \frac{1}{N\theta\delta^2 n_i}\right)$. As n_i's approach infinity, the subspace error also decreases to 0, and the estimated eigenspaces approach the true values accordingly.

Equation (16) also highlights the benefits of knowledge sharing. If each client only uses their own data to estimate the PCs, the estimation error would be $O\left(\frac{1}{n_i}\right)$. The error can be high for clients with few observations (i.e., small n_i). However, when N clients collaborate in learning global and local PCs, the estimation error becomes the average of individual statistical errors $O\left(\sum_{i=1}^N \frac{1}{N\theta\delta^2 n_i}\right)$. Data-poor clients can thus borrow strength from other clients to improve the estimates of their PCs.

The statistical consistency and knowledge-sharing effect will also be examined by numerical experiments in Section 7.

Here, we note that statistical consistency can not be achieved by existing estimates without personalized modeling. For example, the statistical error of `distPCA` (Fan et al., 2019) depends on $O\left(\frac{1}{N}\sum_{i=1}^N \left\|\boldsymbol{\Sigma}_{(i),l}\right\|_{op}\right)$, which does not decrease with number of observations n_i as long as $\left\|\boldsymbol{\Sigma}_{(i),l}\right\|_{op} > 0$. The comparison highlights the advantages of personalization through our formulation in (7).

Now we present the proof of Corollary 3.

Proof We will adopt the covariance concentration bound in Wainwright (2019) and Rinaldo (2019). Since data on client i admit independent sub-Gaussian distributions, theorem 13.3 in Rinaldo (2019) states that, with probability at least $1-\delta_1$, there exists a constant C such that:

$$\left\|\boldsymbol{\Sigma}_{(i)} - \boldsymbol{S}_{(i)}\right\|_{op} \leq \sigma^2 C \max\left\{\sqrt{\frac{d+\log\frac{2}{\delta_1}}{n_i}}, \frac{d+\log\frac{2}{\delta_1}}{n_i}\right\}$$

We can choose $\delta_1 = \frac{\widetilde{\delta}}{N}$. Then by a union bound, we know that with probability at least $1 - \widetilde{\delta}$:

$$\left\|\boldsymbol{\Sigma}_{(i)} - \boldsymbol{S}_{(i)}\right\|_{op} \leq \sigma^2 C \max\left\{\sqrt{\frac{d + \log\frac{2N}{\widetilde{\delta}}}{n_i}}, \frac{d + \log\frac{2N}{\widetilde{\delta}}}{n_i}\right\}$$

holds for all i.

Combining this and Theorem 1, we can prove the bound in (16). ■

Equation (16) also gives a simple estimate of the sample complexity.

Corollary 4 *Under the assumptions of Theorem 1, and assuming that data on client i admits an i.i.d sub-Gaussian with parameter σ, if each client has at least $O\left(\frac{1}{\epsilon}\frac{\sigma^4 d^2}{\theta\delta^2}\right)$ observations, then with high probability, the estimation error is smaller than ϵ.*

Proof The proof is quite straightforward. Notice that when $n_i \geq d$, the right-hand side of (16) is dominated by $\frac{d}{n_i}$. Thus if we neglect the logarithm factors on the right-hand side of (16) and set $\frac{4}{\theta\delta^2}\sigma^4 C^2 d\frac{1}{N}\sum_{i=1}^{N}\frac{d}{n_i} \leq \epsilon$, the statistical error will also be upper bounded by ϵ.

It is natural to see that the inequality holds when each client has observations no less than $O\left(\frac{1}{\epsilon}\frac{\sigma^4 d^2}{\theta\delta^2}\right)$. ■

5. Recovering Local and Global PCs

The statistical consistency proved in Section 4 dwells on the premise that the objective in (7) can be solved to optimality. An efficient algorithm to solve the problem is not apparent as the constraints in (7) are nonconvex. In this section, we develop a class of algorithms to solve (7).

The major difficulty in optimizing (7) lies in the nonconvex constraints: in addition to the orthonormal constraints on $\boldsymbol{U}$ and $\boldsymbol{V}_{(i)}$'s, the constraints $\boldsymbol{U}^T\boldsymbol{V}_{(i)} = 0$ require global and local PCs to be mutually orthogonal. The later constraints introduce interaction between local and global variables, which deems simple distributed Stiefel manifold descent (Chen et al., 2021b) incompetent.

To handle the orthogonality constraints, we propose a class of algorithms that we call Personalized PCA (`PerPCA`). `PerPCA` adopts Stiefel manifold gradient descent to ensure that all constraints are satisfied during the algorithm. It is worth noting that `PerPCA` is naturally *federated* as the computation is distributed over clients, and only updates of the global PCs need to be shared.

In the following of this section, we will build the `PerPCA` algorithm step by step. But before delving into the technical details of parallel gradients and retractions, to illustrate the essence of `PerPCA`, we will first present a simple instance of `PerPCA` that exploits polar projections to maintain the orthonormality of the updates.

5.1 An instance of PerPCA

The polar projection of a general full-column-rank matrix $\boldsymbol{W} \in \mathbb{R}^{n_1 \times n_2}$ where $n_1 \geq n_2$ returns an orthonormal matrix defined as

$$\texttt{Polar}\left(\boldsymbol{W}\right) = \boldsymbol{W}\left(\boldsymbol{W}^T\boldsymbol{W}\right)^{-\frac{1}{2}} \tag{17}$$

Polar projection can be efficiently implemented via SVD (Breloy et al., 2021). It is shown that among all the orthonormal matrices, $\texttt{Polar}\left(\boldsymbol{W}\right)$ is closest to $\boldsymbol{W}$ (Kahan, 2011). Therefore, we can combine gradient descent with polar projection to solve problem (7). The pseudocode is summarized in Algorithm 1.

Algorithm 1 An instance of PerPCA using Polar Projection

Input client covariance matrices $\{\boldsymbol{S}_{(i)}\}_{i=1}^N$, stepsize η_τ
Initialize $\boldsymbol{U}_1$, and $\boldsymbol{V}_{(1),\frac{1}{2}}, \cdots, \boldsymbol{V}_{(N),\frac{1}{2}}$.
for Communication rounds $\tau = 1, ..., R$ **do**
  for Client $i = 1, \cdots, N$ **do**
    $\boldsymbol{V}_{(i),\tau} = \texttt{Polar}\left(\boldsymbol{V}_{(i),\tau-\frac{1}{2}} - \boldsymbol{U}_\tau\boldsymbol{U}_\tau^T\boldsymbol{V}_{(i),\tau-\frac{1}{2}}\right)$
    $[\boldsymbol{U}_{(i),\tau+1}, \boldsymbol{V}_{(i),\tau+\frac{1}{2}}] = \texttt{Polar}\left(\left[\boldsymbol{U}_\tau, \boldsymbol{V}_{(i),\tau}\right] + \eta_\tau\boldsymbol{S}_{(i)}\left[\boldsymbol{U}_\tau, \boldsymbol{V}_{(i),\tau}\right]\right)$
    Uploads $\boldsymbol{U}_{(i),\tau+1}$ to server.
  end for
  Server calculates $\boldsymbol{U}_{\tau+1} = \texttt{Polar}\left(\frac{1}{N}\sum_{i=1}^N \boldsymbol{U}_{(i),\tau+1}\right)$
  Server broadcasts $\boldsymbol{U}_{\tau+1}$
end for

In Algorithm 1, at each iteration, client i first deflates $\boldsymbol{V}_{(i),\tau-\frac{1}{2}}$ to make it orthogonal to $\boldsymbol{U}_\tau$. This ensures that the updates are feasible as $\boldsymbol{U}_\tau^T\boldsymbol{V}_{(i),\tau} = 0$, $\boldsymbol{U}_\tau^T\boldsymbol{U}_\tau = \boldsymbol{I}$, and $\boldsymbol{V}_{(i),\tau}^T\boldsymbol{V}_{(i),\tau} = \boldsymbol{I}$. Then client i uses gradient ascent and polar projection to update $\boldsymbol{U}_{(i),\tau+1}$ and $\boldsymbol{V}_{(i),\tau+\frac{1}{2}}$. This step increases the objective while respecting the orthonormal constraints on $\boldsymbol{U}$ and $\boldsymbol{V}_{(i)}$. After the updates, client i sends the updated $\boldsymbol{U}_{(i),\tau+1}$ to the server. The server takes the average of all received $\boldsymbol{U}_{(i),\tau+1}$, orthonormalize it, then broadcast the updated $\boldsymbol{U}_{\tau+1}$.

It is intuitively understandable how the iterations in Algorithm 1 maximize the objective while keeping the updates feasible. In the rest of this section, we will study a broader class of algorithms through the lens of manifold optimization and show that Algorithm 1 is actually a special case of such algorithm class. We will begin by reviewing a few concepts from manifold optimization and then provide our definition for a class of operations called "*generalized retraction*". Then, we will use the techniques from Stiefel gradient descent to design a class of algorithms that solves (7).

5.2 Generalized retractions

We begin by introducing the Stiefel manifold commonly used in matrix analysis (Edelman et al., 1998).

The Stiefel manifold $St(d, r)$ is the set of all d by r orthonormal matrices:

$$St(d, r) = \{\boldsymbol{U} \in \mathbb{R}^{d\times r} | \boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{I}\} \tag{18}$$

It is embedded in a $d \times r$ dimensional Euclidean space. One can verify that $St(d,r)$ is not convex in general (Edelman et al., 1998).

For $\boldsymbol{U} \in St(d,r)$, the tangent space of $St(d,r)$ at $\boldsymbol{U}$ is defined as:

$$\mathcal{T}_{\boldsymbol{U}} = \{\boldsymbol{\xi} \in \mathbb{R}^{d\times r} | \boldsymbol{\xi}^T\boldsymbol{U} + \boldsymbol{U}^T\boldsymbol{\xi} = 0\}$$

It can be derived by differentiating $\boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{I}$. The normal space $\mathcal{N}_{\boldsymbol{U}}$ is defined as the orthogonal space of the tangent space at $\boldsymbol{U}$.

Both $\mathcal{T}_{\boldsymbol{U}}$ and $\mathcal{N}_{\boldsymbol{U}}$ are linear subspaces of $\mathbb{R}^{d\times r}$. Therefore, we can define the projection onto them. $\mathcal{P}_{\mathcal{N}_{\boldsymbol{U}}}$ denotes the projection onto the normal space:

$$\mathcal{P}_{\mathcal{N}_{\boldsymbol{U}}}(\boldsymbol{V}) = \frac{1}{2}\boldsymbol{U}\left(\boldsymbol{U}^T\boldsymbol{V} + \boldsymbol{V}^T\boldsymbol{U}\right)$$

Similarly, $\mathcal{P}_{\mathcal{T}_{\boldsymbol{U}}}$ denotes the projection onto the tangent space:

$$\mathcal{P}_{\mathcal{T}_{\boldsymbol{U}}}(\boldsymbol{V}) = \boldsymbol{V} - \mathcal{P}_{\mathcal{N}_{\boldsymbol{U}}}(\boldsymbol{V})$$

One can verify that for any matrix $\boldsymbol{V} \in \mathbb{R}^{d\times r}$, $\mathcal{P}_{\mathcal{T}_{\boldsymbol{U}}}(\boldsymbol{V})^T\boldsymbol{U} + \boldsymbol{U}^T\mathcal{P}_{\mathcal{T}_{\boldsymbol{U}}}(\boldsymbol{V}) = 0$

Next, we introduce the notion of a generalized retraction. The motivation for a generalized retraction is rather straightforward. For an orthogonal matrix $\boldsymbol{U}$ and a general update matrix $\boldsymbol{\xi}$, the matrix $\boldsymbol{U}+\boldsymbol{\xi}$ can probably violate the orthonormal constraint: $(\boldsymbol{U}+\boldsymbol{\xi})^T(\boldsymbol{U}+\boldsymbol{\xi}) \neq \boldsymbol{I}$. The generalized retraction finds an approximation $\boldsymbol{U}+\boldsymbol{\xi}$ that strictly satisfies the orthonormal constraint. Ideally, the best approximation can be found via projection. However, the projection onto a general nonlinear manifold is hard to analyze. Therefore, one can relax this projection to a generalized retraction. More formally, a generalized retraction can be defined as:

Definition 5 *We call a mapping*

$$\mathcal{GR}_{\boldsymbol{U}}(\cdot) : \mathbb{R}^{d\times r} \to St(d,r)$$

a generalized retraction if

1. *(Property 1): $col(\mathcal{GR}_{\boldsymbol{U}}(\boldsymbol{\xi})) = col(\boldsymbol{U}+\boldsymbol{\xi}),\ \forall \boldsymbol{U} \in St(d,r),\ \forall \boldsymbol{\xi} \in \mathbb{R}^{d\times r}$*
2. *(Property 2): There exist constants $M_1, M_2 \geq 0$ and $M_3 > 0$ such that:*

$$\|\mathcal{GR}_{\boldsymbol{U}}(\boldsymbol{\xi}) - (\boldsymbol{U} + \mathcal{P}_{\mathcal{T}_{\boldsymbol{U}}}(\boldsymbol{\xi}))\|_F \leq M_1\|\mathcal{P}_{\mathcal{T}_{\boldsymbol{U}}}(\boldsymbol{\xi})\|_F^2 + M_2\|\boldsymbol{\xi} - \mathcal{P}_{\mathcal{T}_{\boldsymbol{U}}}(\boldsymbol{\xi})\|_F,$$
$$\forall \boldsymbol{U} \in St(d,r),\ \forall \boldsymbol{\xi} \in \mathbb{R}^{d\times r}, \|\boldsymbol{\xi}\|_F \leq M_3$$

Figure 2 is an illustration of the Stiefel manifold, tangent space, and generalized retraction.

Notice that the definition of a generalized retraction extends the definition of retraction in literature (Absil et al., 2008). Retraction is usually defined as a mapping from the tangent bundle $\mathcal{T}_{\boldsymbol{U}}$ to the Stiefel manifold $St(d,r)$ (Edelman et al., 1998). However, a generalized retraction is a mapping from a general $\mathbb{R}^{d\times r}$ to the Stiefel manifold $St(d,r)$. This extension allows us to directly apply the generalized retraction to any matrix, eliminating the need to project it to the tangent space beforehand.

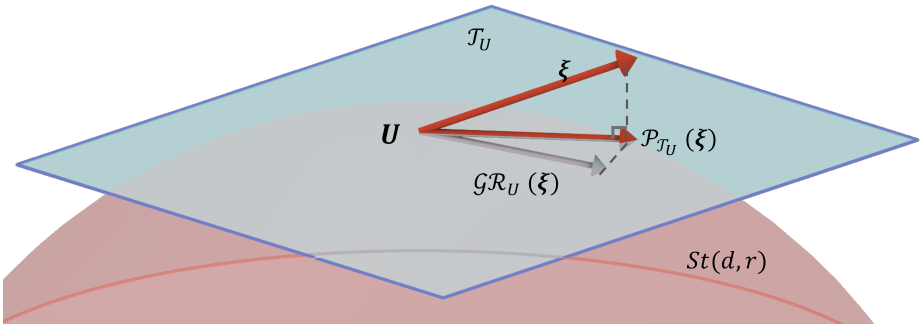


Figure 2: An illustration of the Stiefel manifold, tangent space, and generalized retraction. The red surface represents the Stiefel manifold. The blue plane represents the tangent space at $\boldsymbol{U} \in St(d,r)$. $\boldsymbol{\xi}$ is a general d by r matrix that represents the update direction. $\mathcal{P}_{\mathcal{T}_{\boldsymbol{U}}}(\boldsymbol{\xi})$ projects $\boldsymbol{\xi}$ to the tangent space on $\boldsymbol{U}$. Generalized retraction $\mathcal{GR}_{\boldsymbol{U}}(\boldsymbol{\xi})$ maps $\boldsymbol{U}+\boldsymbol{\xi}$ back to the Stiefel manifold.

Property 1 requires that a generalized retraction preserves column spaces. This property is indispensable in our algorithm development as we use it to ensure the orthogonality of global and local PCs. The second property requires that $\mathcal{GR}_{\boldsymbol{U}}(\boldsymbol{\xi})$ be close to the projection to the tangent space $\boldsymbol{U}+\mathcal{P}_{\mathcal{T}_{\boldsymbol{U}}}(\boldsymbol{\xi})$. In the special case of $\boldsymbol{\xi} \in \mathcal{T}_{\boldsymbol{U}}$, property 2 reduces to $\|\mathcal{GR}_{\boldsymbol{U}}(\boldsymbol{\xi}) - (\boldsymbol{U}+\boldsymbol{\xi})\|_F \leq M_1 \|\boldsymbol{\xi}\|_F^2$, which coincides with the definition of retraction in literature (Chen et al., 2021a). When the norm of $\boldsymbol{\xi}$ is small, the requirement essentially implies that the difference between a generalized retraction and the projection to a tangent space is a higher-order term.

Though Definition 5 looks demanding, we can show that there are several available choices for a generalized retraction.

Proposition 6 *Polar projection is defined as:*

$$\mathcal{GR}_{\boldsymbol{U}}^{polar}(\boldsymbol{\xi}) = (\boldsymbol{U}+\boldsymbol{\xi})\left(\boldsymbol{I}+\boldsymbol{\xi}^T\boldsymbol{U}+\boldsymbol{U}^T\boldsymbol{\xi}+\boldsymbol{\xi}^T\boldsymbol{\xi}\right)^{-\frac{1}{2}} \tag{19}$$

is a generalized retraction. The computation complexity is $O(dr^2+r^3)$.

(19) is consistent with the definition (17), though the notations are slightly different. Notice that polar projection can be equivalently calculated by the SVD of $\boldsymbol{U}+\boldsymbol{\xi}$ (Breloy et al., 2021). We relegate the proof and the implementation details to Appendix F.1. As discussed, an interesting property of the polar projection is that it is equivalent to the projection of $\boldsymbol{U}+\boldsymbol{\xi}$ onto the Stiefel manifold:

$$\mathcal{GR}_{\boldsymbol{U}}^{\text{polar}}(\boldsymbol{\xi}) = \arg\min_{\boldsymbol{V}\in St(d,r)} \|\boldsymbol{U}+\boldsymbol{\xi}-\boldsymbol{V}\|_F \tag{20}$$

The proof of (20) can be found in Kahan (2011).

QR decomposition is another influential algorithm in numerical linear algebra. It also satisfies the requirements of a generalized retraction.

Proposition 7 *For a matrix $\boldsymbol{U}+\boldsymbol{\xi} \in \mathbb{R}^{d\times r}$, QR decomposition finds an orthogonal matrix $\boldsymbol{Q} \in St(d,r)$ and a upper triangular matrix $\boldsymbol{R} \in \mathbb{R}^{r\times r}$, such that $\boldsymbol{QR}=\boldsymbol{U}+\boldsymbol{\xi}$. As such, a QR retraction is defined as:*

$$\mathcal{GR}_{\boldsymbol{U}}^{QR}(\boldsymbol{\xi})=\boldsymbol{Q}$$

is a generalized retraction. The computation complexity is $O(dr^2)$.

We relegate the proof to Appendix F.2.

In all of our experiments, we choose the generalized retraction as a polar decomposition.

5.3 `PerPCA`: The algorithm

Now, we are ready to introduce the personalized PCA algorithm, `PerPCA`. Recall that our algorithm is designed to be federated and requires multiple communication rounds between a client and some central server/entity that orchestrates the collaborative learning process. Suppose at communication round τ, each client has feasible global components $\boldsymbol{U}_\tau$ and local components $\boldsymbol{V}_{(i),\tau}$, i.e., $[\boldsymbol{U}_\tau, \boldsymbol{V}_{(i),\tau}] \in St(d, r_1+r_{2,(i)})$. Then client i calculates the gradient of objective f_i defined in (8):

$$\begin{cases}\nabla_{\boldsymbol{U}} f_i(\boldsymbol{U}_\tau, \boldsymbol{V}_{(i),\tau}) = \boldsymbol{S}_{(i)}\boldsymbol{U}_\tau \\ \nabla_{\boldsymbol{V}_{(i)}} f_i(\boldsymbol{U}_\tau, \boldsymbol{V}_{(i),\tau}) = \boldsymbol{S}_{(i)}\boldsymbol{V}_{(i),\tau}\end{cases}$$

Since the gradient direction generally does not align with the tangent space of $\mathcal{T}_{[\boldsymbol{U}_\tau,\boldsymbol{V}_{(i),\tau}]}$, simple gradient ascent will move $[\boldsymbol{U}_\tau, \boldsymbol{V}_{(i),\tau}]$ out of $St(d, r_1+r_{2,(i)})$. To ensure the iterates move along the manifold, Stiefel optimization first projects the gradient to the tangent space:

$$\mathbf{G}_{(i),\tau} = \mathcal{P}_{\mathcal{T}_{\left[\boldsymbol{U}_\tau,\boldsymbol{V}_{(i),\tau}\right]}}\left(\boldsymbol{S}_{(i)}\left[\boldsymbol{U}_\tau, \boldsymbol{V}_{(i),\tau}\right]\right) \tag{21}$$

In literature, $\mathbf{G}_{(i),\tau}$ is usually referred to as the parallel gradient on the manifold (Edelman et al., 1998). We shall note that $\mathbf{G}_{(i),\tau}$ defined above is a d by $r_1+r_{2,(i)}$ matrix.

Clients then update global and local PCs in the direction of the parallel gradient $\mathbf{G}_{(i),\tau}$. As there is a small difference between the Stiefel manifold and the tangent space, the updated PCs are still not orthonormalized. Therefore, we use a generalized retraction to retract the updated local components to the Stiefel manifold. We use $\boldsymbol{V}_{(i),\tau+\frac{1}{2}}$ to denote the retracted matrix. For the global components, clients first send them to a server. The server then takes the average and uses a generalized retraction to map the average to $St(d,r)$. The updated global PC matrix is denoted as $\boldsymbol{U}_{\tau+1}$.

A major challenge then arises: after the server averages the global PCs, $\boldsymbol{U}_{\tau+1}$ is not orthogonal to $\boldsymbol{V}_{(i),\tau+\frac{1}{2}}$ anymore, i.e., $\boldsymbol{U}_{\tau+1}^T\boldsymbol{V}_{(i),\tau+\frac{1}{2}} \neq 0$ in general. Thus $\boldsymbol{U}_{\tau+1}$ and $\boldsymbol{V}_{(i),\tau+\frac{1}{2}}$'s become infeasible, and the algorithm based on St-GD cannot proceed. One can verify that $\boldsymbol{U}_\tau^T\boldsymbol{V}_{(i),\tau+\frac{1}{2}} = O(\eta_\tau)$, which has the same order as the parallel gradient update. Thus, we cannot resolve the infeasibility issue by decreasing stepsize. This is a fundamental limitation of a simple route that uses distributed St-GD.

Can we resolve the challenge by enforcing the orthogonality between global and local PC estimates? Inspired by Gram-Schmit orthonormalization, we introduce a correction step on the local PCs. We calculate the projection of $\boldsymbol{V}_{(i),\tau+\frac{1}{2}}$ onto the column space of $\boldsymbol{U}_{\tau+1}$, and

subtract the projected matrix from $\boldsymbol{V}_{(i),\tau+\frac{1}{2}}$. The resulting (deflated) matrix is orthogonal to $\boldsymbol{U}_{\tau+1}$. Then, we use a generalized retraction to map the subtracted matrix to the Stiefel manifold. Remember that one key property of a generalized retraction is that it preserves the column space; the retracted matrix is thus still orthogonal to $\boldsymbol{U}_{\tau+1}$. We use $\boldsymbol{V}_{(i),\tau+1}$ to denote the retracted matrix. Now $\boldsymbol{U}_{\tau+1}$ and $\boldsymbol{V}_{(i),\tau+1}$ are feasible, and the updates can repeat over multiple communication rounds until convergence. The pseudocode is summarized in Algorithm 2.

Algorithm 2 `PerPCA` by St-GD

Input client covariance matrices $\{\boldsymbol{S}_{(i)}\}_{i=1}^N$, stepsize η_τ
Initialize $\boldsymbol{U}_1$, and $\boldsymbol{V}_{(1),\frac{1}{2}}, \cdots, \boldsymbol{V}_{(N),\frac{1}{2}}$.
for Communication rounds $\tau = 1, ..., R$ **do**
  for Client $i = 1, \cdots, N$ **do**
    $\boldsymbol{V}_{(i),\tau} = \mathcal{GR}_{\boldsymbol{V}_{(i),\tau-\frac{1}{2}}}\left(-\boldsymbol{U}_\tau \boldsymbol{U}_\tau^T \boldsymbol{V}_{(i),\tau-\frac{1}{2}}\right)$ `// Deflate then retract`
    Choice 1:
      Calculate $\mathbf{G}_{(i),\tau} = \mathcal{P}_{\mathcal{T}_{[\boldsymbol{U}_\tau, \boldsymbol{V}_{(i),\tau}]}}\left(\boldsymbol{S}_{(i)}\left[\boldsymbol{U}_\tau, \boldsymbol{V}_{(i),\tau}\right]\right)$ `// Tangent projection`
      Update $\boldsymbol{U}_{(i),\tau+1} = \boldsymbol{U}_\tau + \eta_\tau (\mathbf{G}_{(i),\tau})_{1:d,1:r_1}$ `// Gradient ascent`
      Update $\boldsymbol{V}_{(i),\tau+\frac{1}{2}} = \mathcal{GR}_{\boldsymbol{V}_{(i),\tau}}\left(\eta_\tau (\mathbf{G}_{(i),\tau})_{1:d,(r_1+1):(r_1+r_{2,(i)})}\right)$ `// Retract`
    Choice 2:
      Update $[\boldsymbol{U}_{(i),\tau+1}, \boldsymbol{V}_{(i),\tau+\frac{1}{2}}] = \mathcal{GR}^{\text{polar}}_{[\boldsymbol{U}_\tau, \boldsymbol{V}_{(i),\tau}]}\left(\eta_\tau \boldsymbol{S}_{(i)}\left[\boldsymbol{U}_\tau, \boldsymbol{V}_{(i),\tau}\right]\right)$
      `// Retract after gradient ascent`
    Send $\boldsymbol{U}_{(i),\tau+1}$ to the server. `// Share global PCs`
  end for
  Server calculates $\boldsymbol{U}_{\tau+1} = \mathcal{GR}_{\boldsymbol{U}_\tau}\left(\frac{1}{N}\sum_{i=1}^N \boldsymbol{U}_{(i),\tau+1} - \boldsymbol{U}_\tau\right)$ `// Average then retract`
  Server broadcasts $\boldsymbol{U}_{\tau+1}$
end for
Return principal components $\boldsymbol{U}_R$ and $\boldsymbol{V}_{(i),R}$'s.

The first line in the client loop $\boldsymbol{V}_{(i),\tau} = \mathcal{GR}_{\boldsymbol{V}_{(i),\tau-\frac{1}{2}}}\left(-\boldsymbol{U}_\tau \boldsymbol{U}_\tau^T \boldsymbol{V}_{(i),\tau-\frac{1}{2}}\right)$ represents the correction on the local PC matrix. Regardless of whether $\boldsymbol{U}_\tau$ and $\boldsymbol{V}_{(i),\tau-\frac{1}{2}}$ are orthogonal, $\boldsymbol{U}_\tau$ and $\boldsymbol{V}_{(i),\tau}$ are always feasible: $[\boldsymbol{U}_\tau, \boldsymbol{V}_{(i),\tau}] \in St(d, r_1 + r_{2,(i)})$. Then each client applies standard St-GD (choice 1) or a variant of St-GD (choice 2) to update $\boldsymbol{U}_{(i),\tau+1}$ and $\boldsymbol{V}_{(i),\tau+\frac{1}{2}}$ simultaneously. The updated global PCs are sent to the server. The server takes the simple average of all received global PCs and retracts the average to $St(d, r_1)$. The obtained $\boldsymbol{U}_{\tau+1}$ is then broadcasted back to the clients and becomes the starting point of the next iteration. The algorithm repeats for a certain number of communication rounds.

In Algorithm 2, we introduce two algorithmic choices on the client side. For choice 1, clients perform standard St-GD: first project the updates to the tangent space, then retract them to the Stiefel manifold. For choice 2, clients use polar projection to replace the St-GD. This update rule is inspired by the Minorization-Maximization algorithm (Breloy et al., 2021). Remember that by (20), polar projection acts as a projection into the nonlinear Stiefel manifold. Hence, it is close to the composition of the projection onto the tangent

space and the retraction from the tangent space onto the nonlinear manifold. We propose two choices to enrich practitioners' toolkits as they have similar performances in most of our case studies. We focus on choice 1 in our theoretical analysis. However, it is observed in the video segmentation task that choice 2 allows us to use larger stepsizes, thus converging faster. Hence, we leave it to practitioners' discretion to make specific algorithmic choices.

In general, the computation complexity per communication round at one client is $O(d^2)$. To see that, we can analyze the update of Algorithm 2. One iteration only involves matrix multiplication and generalized retractions. The computation complexity of matrix multiplication $\boldsymbol{S}_{(i)}\boldsymbol{U}_\tau$ is $O(d^2 r_1)$. The complexity of the tangent projection step is similar. When the rank r_1 and $r_{2,(i)}$ is far smaller than data dimension d, the computation complexity for generalized retractions is only $O(d)$. Thus, the per-iteration computation complexity is $O(d^2)$. It is worth noting that the complexity can be further reduced to $O(d)$ if the covariance matrix $\boldsymbol{S}_{(i)}$ is known to be low rank. More specifically, when $\boldsymbol{S}_{(i)}$ has a low-rank Cholesky decomposition $\boldsymbol{S}_{(i)} = \boldsymbol{Y}_{(i)}\boldsymbol{Y}_{(i)}^T$, where $\boldsymbol{Y}_{(i)} \in \mathbb{R}^{d\times n_{(i)}}$ is a low-rank matrix $n_{(i)} \ll d$, the computation cost of matrix multiplication $\boldsymbol{S}_{(i)}\boldsymbol{U}_\tau = \boldsymbol{Y}_{(i)}\boldsymbol{Y}_{(i)}^T\boldsymbol{U}_\tau$ is reduced to $O(dn_{(i)}r_1)$. As $n_{(i)}$ and r_1 is far smaller than d, this becomes $O(d)$. Hence the per-iteration computation complexity is only $O(d)$.

6. Does Algorithm 2 Recover the Local and Global Truth?

Though the development of Algorithm 2 is intuitive, it is important to understand whether it converges and, if so, what kind of solution it can recover. In this section, we will analyze the convergence of Algorithm 2 and show that, in general, Algorithm 2 converges into stationary points of the objective. In addition, when the local and global components are initialized properly, Algorithm 2 will converge into the global optimal solutions linearly, and the result exactly recovers the true local and global PCs.

6.1 Global convergence

To analyze the convergence, we make an additional assumption that the largest eigenvalues of the sample covariance matrices $\boldsymbol{S}_{(i)}$'s are upper bounded:

Assumption 6.1 *We assume that the operator norms of $\boldsymbol{S}_{(i)}$'s are upper bounded by constants $G_{(i),op}$:*

$$\left\|\boldsymbol{S}_{(i)}\right\|_{op} \leq G_{(i),op} \tag{22}$$

and the Frobenius norms of $\boldsymbol{S}_{(i)}$'s are upper bounded by constants $G_{(i),F}$:

$$\left\|\boldsymbol{S}_{(i)}\right\|_{F} \leq G_{(i),F} \tag{23}$$

We use $G_{max,op}$ to denote $\max_i G_{(i),op}$, and $G_{max,F}$ to denote $\max_i G_{(i),F}$.

Assumption 6.1 is a common assumption in optimization literature, as it essentially assumes the objective is Lipschitz continuous. Also, if we assume the data are independently generated and follow a sub-Gaussian distribution, Assumption 6.1 will hold with high probability (Wainwright, 2019).

The first order condition (KKT condition) to problem (1) is that for the parallel gradients defined in (21), the local parts are zero on each client, and the average of the global parts is zero:

$$\begin{cases} \left(\mathbf{G}_{(i)}\right)_{1:d,(r_1+1):(r_1+r_{2,(i)})} = 0, \quad \forall i \in \{1, 2, , \cdots, N\} \\ \frac{1}{N}\sum_{i=1}^{N}\left(\mathbf{G}_{(i)}\right)_{1:d,1:(r_1+1)} = 0 \end{cases} \tag{24}$$

The proof of KKT conditions (24) is in Appendix B. It is clear from Algorithm 2 that when (24) is satisfied, the global and local PC updates will be stationary. Thus, (24) essentially describes the stationary points of (7).

On non-stationary points, (24) generally does not hold. The below theorem provides an upper bound on the magnitude of the violations to conditions (24). As the violations decrease to zero when the number of communication approaches infinity, the theorem shows that Algorithm 2 will converge into the KKT points. We use r to denote maximum rank $r = \max\{r_1, r_{2,(1)}, \cdots, r_{2,(N)}\}$.

Theorem 8 *Under Assumption 6.1, if we choose a constant stepsize* $\eta_\tau = \eta_1 = O(\frac{1}{G_{max,op}\sqrt{r}})$*, then Algorithm 2 with choice* 1 *will converge into stationary points:*

$$\min_{\tau\in\{1,\dots R\}} \left[\left\|\sum_{i=1}^{N}\left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\boldsymbol{S}_{(i)}\boldsymbol{U}_\tau\right\|^2 + \sum_{i=1}^{N}\left\|\left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\sum_{i=1}^{N}\boldsymbol{S}_{(i)}\boldsymbol{V}_{(i),\tau}\right\|^2\right] = O\left(\frac{1}{R}\right)$$

Despite the nonconvex constraints in (7), Algorithm 2 provably converges to stationary points, regardless of initial conditions. The $\frac{1}{R}$ convergence rate is comparable to the rate in literature (Chen et al., 2021a).

Our algorithm handles global and local PCs at the same time and attains stationary points of both components. In the following section, we will show the proof sketch of Theorem 8. The complete proof is relegated to Appendix D.

6.1.1 Proof sketch for Theorem 8 and key lemmas

As discussed before, one major difficulty in analyzing Algorithm 2 lies in the correction step. The correction step changes local PCs by $O(\eta_\tau)$, which is comparable to that in the gradient ascent step. Therefore, a naïve treatment to the correction step will generate a large error term that cannot be bounded.

To bypass the issue, we exploit one nice structure in objective (8): $f_i(\boldsymbol{U}, \boldsymbol{V}_{(i)})$ is dependent only on the subspace spanned by the concatenated matrix $[\boldsymbol{U}, \boldsymbol{V}_{(i)}]$. Therefore one can make adjustments on $col(\boldsymbol{U})$ and $col(\boldsymbol{V}_{(i)})$ without changing the objective value, as long as $col([\boldsymbol{U}, \boldsymbol{V}_{(i)}])$ are the same.

One major technical novelty of our work is to introduce Lyapunov functions that take this key property into consideration. We define the two following Lyapunov functions:

$$\mathcal{L}_{(i),1}(\boldsymbol{U}, \boldsymbol{V}) = -\frac{1}{2}\mathrm{Tr}\left(\boldsymbol{U}^T\left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{V}}\right)\boldsymbol{S}_{(i)}\left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{V}}\right)\boldsymbol{U}\right) \tag{25}$$

and,

$$\mathcal{L}_{(i),2}(\boldsymbol{U},\boldsymbol{V}) = -\frac{1}{2}\mathrm{Tr}\left(\boldsymbol{V}^T\boldsymbol{S}_{(i)}\boldsymbol{V}\right) \tag{26}$$

It's easy to see that when $\boldsymbol{V}^T\boldsymbol{U} = 0$, we have:

$$\mathcal{L}_{(i),1}(\boldsymbol{U},\boldsymbol{V}) + \mathcal{L}_{(i),2}(\boldsymbol{U},\boldsymbol{V}) = -\frac{1}{2}\mathrm{Tr}\left(\boldsymbol{U}^T\boldsymbol{S}_{(i)}\boldsymbol{U}\right) - \frac{1}{2}\mathrm{Tr}\left(\boldsymbol{V}^T\boldsymbol{S}_{(i)}\boldsymbol{V}\right) = -f_n(\boldsymbol{U},\boldsymbol{V})$$

At each communication step τ, global and local components are indeed orthogonal $\boldsymbol{U}_\tau^T\boldsymbol{V}_{(i),\tau} = 0$, thus $\mathcal{L}_{(i),1}(\boldsymbol{U}_\tau,\boldsymbol{V}_{(i),\tau}) + \mathcal{L}_{(i),2}(\boldsymbol{U}_\tau,\boldsymbol{V}_{(i),\tau}) = -f_n(\boldsymbol{U}_\tau,\boldsymbol{V}_{(i),\tau})$.

$\mathcal{L}_{(i),1}$ explicitly encodes the orthogonality constraint into the objective. Such design enables convenient handling of the correction step: we can prove that the correction step on $\boldsymbol{V}$ changes $\mathcal{L}_{(i),1} + \mathcal{L}_{(i),2}$ only by $O(\eta_\tau^2)$. Therefore only the descent step can change $\mathcal{L}_{(i),1} + \mathcal{L}_{(i),2}$ by $O(\eta_\tau)$. Thus, the change of Lyapunov functions is dominated by the update from the parallel gradient. By calculating the update of $\boldsymbol{U}$ and $\{\boldsymbol{V}_{(i)}\}$ in each communication round, we can have the following informal version of the sufficient descent lemma:

Lemma 9 *(Informal) When we choose a constant stepsize $\eta_\tau = \eta = O\left(\frac{1}{G_{max,op}\sqrt{r}}\right)$, and $\boldsymbol{U}_\tau$ and $\boldsymbol{V}_{(i),\tau}$ satisfy the orthogonality condition $\boldsymbol{U}_\tau^T\boldsymbol{V}_{(i),\tau} = 0$, we have:*

$$\begin{aligned}
&\left\langle \sum_{i=1}^N \nabla_{\boldsymbol{U}}\mathcal{L}_{(i),1}(\boldsymbol{U}_\tau,\boldsymbol{V}_{(i),\tau}), \boldsymbol{U}_{\tau+1} - \boldsymbol{U}_\tau \right\rangle \\
&+ \sum_{i=1}^N \left\langle \nabla_{\boldsymbol{V}_{(i)}}\mathcal{L}_{(i),1}(\boldsymbol{U}_\tau,\boldsymbol{V}_{(i),\tau}) + \nabla_{\boldsymbol{V}_{(i)}}\mathcal{L}_{(i),2}(\boldsymbol{U}_\tau,\boldsymbol{V}_{(i),\tau}), \boldsymbol{V}_{(i),\tau+1} - \boldsymbol{V}_{(i),\tau} \right\rangle \\
&\leq -\eta\left(\frac{1}{N}\left\|\sum_{i=1}^N \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\boldsymbol{S}_{(i)}\boldsymbol{U}_\tau\right\|_F^2 + \sum_{i=1}^N \left\|\left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\sum_{i=1}^N \boldsymbol{S}_{(i)}\boldsymbol{V}_{(i),\tau}\right\|_F^2\right) \\
&+ O(\eta^2)
\end{aligned} \tag{27}$$

When η is small, the $O(\eta)$ terms will dominate $O(\eta^2)$ terms. Thus, Lemma 9 essentially shows that in Algorithm 2, the change of Lyapunov functions is negative semidefinite in one communication round. With the sufficient decrease property, standard analysis on first-order optimization yields a $O\left(\frac{1}{R}\right)$ convergence rate.

Formal proofs of Theorem 8 and Lemma 9 can be found in Appendix D.

6.2 Local convergence

Theorem 8 only shows that Algorithm 2 converges into stationary points but does not provide further information about the property of the final solution. In problems like feature extraction, we want to know whether the stationary point is a globally optimal solution or whether it corresponds to the true PCs.

To this end, we analyze the convergence of global and local PCs. The convergence depends on a Polyak-Lojasiewicz style condition. Similar to Section 6.1, we will introduce another assumption about the eigenvalue distribution of the sample covariance matrix. Without loss of generality, in this section, we assume $r_1 = r_{2,(1)} = \cdots = r_{2,(N)} = r$.

Assumption 6.2 *(Covariance matrix eigenvalue lower bound) We further assume that the population covariance $\mathbf{\Sigma}_{(i)}$ can be entirely explained by $\mathbf{\Pi}_g + \mathbf{\Pi}_{(i)}$, i.e., $\mathbf{\Sigma}_{(i)}\left(\mathbf{\Pi}_g + \mathbf{\Pi}_{(i)}\right) = \mathbf{\Sigma}_{(i)}$, and that the minimum nonzero eigenvalues of $\mathbf{\Sigma}_{(i)}$ is lower bounded by a constant $\mu > 0$:*

$$\mu\left(\mathbf{\Pi}_g + \mathbf{\Pi}_{(i)}\right) \preceq \mathbf{\Sigma}_{(i)} \tag{28}$$

where $\mathbf{\Pi}_g$ and $\mathbf{\Pi}_{(i)}$ are rank-r projection matrices.

Assumption 6.2 assumes that data covariance can be decomposed as noiseless global and local parts with rank r. The noiseless assumption of the population covariance matrices is the standard assumption in the local convergence analysis of many PCA algorithms (e.g., (Tang, 2019)).

The following theorem shows that if Algorithm 2 is initialized within the attractive basin of the global optimum, the iterates will converge to the global optimal solution linearly.

Theorem 10 *(Informal) Under assumptions 4.2, 6.1, and 6.2, if the difference between the population and sample covariance is small, when we initialize close to the global optimum, and choose a constant stepsize $\eta_\tau = \eta = O\left(\frac{1}{G_{op,max}\sqrt{r}}\right)$, then Algorithm 2 with choice 1 will converge into the global optimum:*

$$f(\hat{\boldsymbol{U}}, \{\hat{\boldsymbol{V}}_{(i)}\}) - f(\boldsymbol{U}_R, \{\boldsymbol{V}_{(i),R}\}) = O\left(\left(1 - \eta\frac{\mu\theta}{32}\right)^R\right)$$

where $\{\hat{\boldsymbol{U}}, \{\hat{\boldsymbol{V}}_{(i)}\}\}$ is one set of optimal solutions to problem (7).

Furthermore, we can recover the exact global optimal solutions:

$$\left\|\boldsymbol{P}_{\boldsymbol{U}_R} - \boldsymbol{P}_{\hat{\boldsymbol{U}}_g}\right\|_F^2 + \frac{1}{N}\sum_{i=1}^N \left\|\boldsymbol{P}_{\boldsymbol{V}_{(i),R}} - \boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i)}}\right\|_F^2 = O\left(\left(1 - \eta\frac{\mu\theta}{32}\right)^R\right)$$

It is worthwhile to point out that in Theorem 10, the convergence is faster for a larger misalignment parameter θ. This is intuitively understandable since when local eigenspaces are more heterogeneous, it is easier to identify different eigenspaces. On the other hand, if all the local eigenspaces are similar, it is difficult to distinguish local PCs from global PCs; thus, the convergence is slower. *This result is in striking contrast to standard federated learning (e.g., Li et al. (2020, 2018a)), where data heterogeneity leads to slower convergence.* We will verify this finding in Section 7. A formal version of Theorem 10 and its proof is relegated to the Appendix E.

With the statistical error bound provided by Theorem 1 and the convergence guarantee from Theorem 10, we can derive the following corollary.

Corollary 11 *Under the same assumptions as Theorem 1 and Theorem 10, after $t = \Omega\left(\frac{\sqrt{r}G_{max,op}}{\mu\theta}\log\frac{1}{\varepsilon_{stats}}\right)$ communication rounds, we can obtain estimates of global and local PCs that satisfy,*

$$\|\boldsymbol{P}_{\boldsymbol{U}_t} - \mathbf{\Pi}_g\|_F^2 + \frac{1}{N}\sum_{i=1}^N \left\|\boldsymbol{P}_{\boldsymbol{V}_{(i),t}} - \mathbf{\Pi}_{(i)}\right\|_F^2 = O\left(\varepsilon_{stats}\right)$$

where ε_{stats} is the statistical error $\varepsilon_{stats} = \frac{1}{\theta\delta^2}\sigma^4 C^2\frac{d}{N}\sum_{i=1}^N\frac{1}{n_i}$.

Proof By the triangle inequality, we know

$$\begin{aligned}
&\|\boldsymbol{P}_{\boldsymbol{U}_t} - \boldsymbol{\Pi}_g\|_F^2 + \frac{1}{N}\sum_{i=1}^{N}\left\|\boldsymbol{P}_{\boldsymbol{V}_{(i),t}} - \boldsymbol{\Pi}_{(i)}\right\|_F^2 \\
&\leq 2\left\|\boldsymbol{P}_{\boldsymbol{U}_t} - \boldsymbol{P}_{\hat{\boldsymbol{U}}}\right\|_F^2 + \frac{2}{N}\sum_{i=1}^{N}\left\|\boldsymbol{P}_{\boldsymbol{V}_{(i),t}} - \boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i),\tau}}\right\|_F^2 \\
&+ 2\left\|\boldsymbol{\Pi}_g - \boldsymbol{P}_{\hat{\boldsymbol{U}}}\right\|_F^2 + \frac{2}{N}\sum_{i=1}^{N}\left\|\boldsymbol{\Pi}_{(i)} - \boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i),\tau}}\right\|_F^2
\end{aligned}$$

The first term is bounded by Theorem 10, and the second term is bounded by Theorem 1 ■

6.2.1 Proof sketch of Theorem 10 and key lemmas

To prove the exponential convergence in Theorem 10, we need a stronger version of the sufficient decrease inequality than Lemma 9. We should show that, in each communication round, the change in the Lyapunov functions is negative definite. This requires a careful analysis of the geometry of objective (7) around the global optimum $\boldsymbol{P}_{\hat{\boldsymbol{U}}}$ and $\{\boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i),\tau}}\}$.

The key result is the Polyak-Lojasiewicz (PL) inequality.

Lemma 12 *(Polyak-Lojasiewicz inequality) Under the same conditions as Theorem 10, we have*

$$\begin{aligned}
&\frac{1}{N}\left\|\sum_{i=1}^{N}\left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\boldsymbol{S}_{(i)}\boldsymbol{U}_\tau\right\|_F^2 + \sum_{i=1}^{N}\left\|\left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\sum_{i=1}^{N}\boldsymbol{S}_{(i)}\boldsymbol{V}_{(i),\tau}\right\|_F^2 \\
&\geq \frac{\theta\mu}{32}\left(f(\hat{\boldsymbol{U}}, \{\hat{\boldsymbol{V}}_{(i)}\}) - f(\boldsymbol{U}_R, \{\boldsymbol{V}_{(i),R}\})\right)
\end{aligned}$$

The PL inequality shows that the norm of the parallel gradient is lower bounded, a constant fraction of the optimality gap. It certifies a nice geometric property in objective (7) so that each step of gradient descent can make significant progress. By combining the PL inequality with Lemma 16, we can easily prove Theorem 10.

One of our major technical contributions is to establish the PL inequality for the nonconvex problem (7). We analyze the local geometry of the problem with the help of one special set of optimal solutions $\{\hat{\boldsymbol{U}}_\tau, \{\hat{\boldsymbol{V}}_{(i),\tau}\}\}$. We show that this set of optimal solutions is close to the current iterate $\{\boldsymbol{U}_\tau, \{\boldsymbol{V}_{(i),\tau}\}\}$. Also, the difference $\{\boldsymbol{U}_\tau - \hat{\boldsymbol{U}}_\tau, \{\boldsymbol{V}_{(i),\tau} - \hat{\boldsymbol{V}}_{(i),\tau}\}\}$ is aligned with the parallel gradient. As a result, the parallel gradient can direct the updates to the optimal solutions.

The full proof of Lemma 12 and Theorem 10 is relegated to Appendix E.

7. Numerical Experiments

This section tests our model on a set of datasets across different applications. We start in Section 7.1 with a proof of concept study using a synthetic dataset to verify theoretical findings in Sections 4 and 6.

We also discuss the effects of overparametrization and show an interesting application of `PerPCA` in federated client clustering using local PCs. In Section 7.2, we provide an illustrative example in comparison with `Robust PCA` to shed light on the end goal of our model. Next, we apply `PerPCA` to a real-life heterogeneous distributed dataset FEMNIST and CIFAR10 to show `PerPCA`'s advantages in finding better features in Section 7.3. Finally, we demonstrate how `PerPCA` can separate shared and unique features in video and language data in Section 7.4.

We note that from Theorem 10, a suitable initialization is needed for the best performance of `PerPCA`. We thus employ the standard one-communication round distributed PCA algorithm proposed in Qu et al. (2002) as the initialization of global PCs in Algorithm 2, unless specified otherwise. Local PCs are always randomly initialized. In this section we set $r_{2,(1)} = r_{2,(2)} = \cdots = r_{2,(N)} = r_2$.

7.1 Proof of concept on synthetic datasets

We generate data from model (1). The $\boldsymbol{u}_q$'s and $\boldsymbol{v}_q$'s are set to be orthogonal components. After obtaining $\boldsymbol{u}_q$'s and $\boldsymbol{v}_q$'s, we sample the score coefficients $\phi_{(i),q}$'s and $\varphi_{(i),q}$'s from i.i.d. Gaussian distributions. Noise $\boldsymbol{\epsilon}_{(i)}$ are also sampled from i.i.d. Gaussian distributions.

Under this setting, multiple aspects are tested: in Section 7.1.1, we revisit the example in Figure 1 and examine the convergence behavior of `PerPCA` numerically. In Sections 7.1.2, 7.1.3, and 7.1.4, we demonstrate how the statistical errors change with the (i) number of observations n, (ii) data dimension d, and (iii) number of clients N, and compare the results with our theory. In Section 7.1.5, we show that in `PerPCA`, clients benefit from knowledge sharing to improve their PC estimates. Then we investigate the numerical performance of `PerPCA` when r_1 and r_2 are overparametrized in Section 7.1.6. Finally, in Section 7.1.7, we describe a method that exploits the estimated local PCs for client clustering.

7.1.1 Convergence of `PerPCA`

We first analyze the convergence of `PerPCA`. Theorem 10 predicts that (i) `PerPCA` has local linear convergence, and (ii) a larger θ can expedite convergence. To verify the two theoretical results, we run `PerPCA` on a group of synthetic data. We set $N = 2$, $d = 3$ and $n_{(i)} = 1000$. Each client has exactly one global $\boldsymbol{u}_1$ and one local component $\boldsymbol{v}_{(i),1}$. After setting global PC $\boldsymbol{u}_1$ and local PCs $\boldsymbol{v}_{(1),1}$ and $\boldsymbol{v}_{(2),1}$, we generate the data according to the model (1) where coefficients $\phi_{(i),q}$ and $\varphi_{(i),q}$ are randomly sampled from Gaussian distributions. By changing the direction of local PCs $\boldsymbol{v}_{(1),1}$ and $\boldsymbol{v}_{(2),1}$, we can modify θ:

$$\theta = \sin^2\left(\frac{1}{2}\arccos(\boldsymbol{v}_{(1),1}^T\boldsymbol{v}_{(2),1})\right)$$

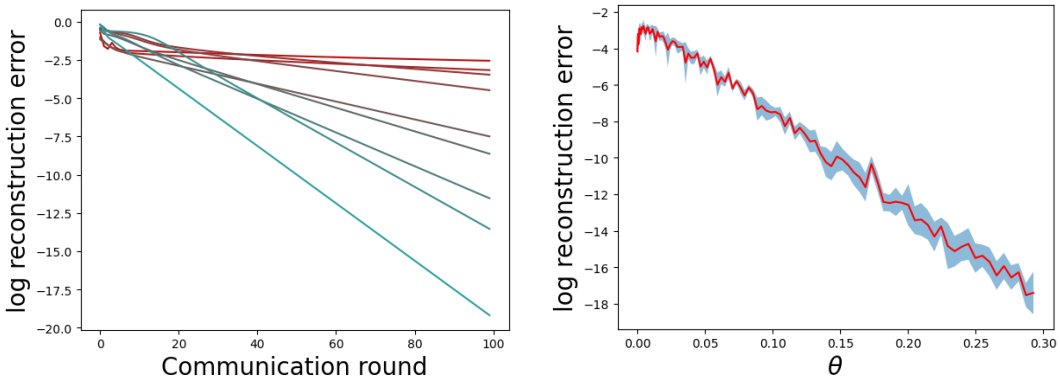
Figure 1, shown in the introduction, is an instance of this analysis where $\theta = 0.127$.

To see the θ's effect on convergence, we generate the data with θ ranging from 0 to 0.3. In this experiment, we initialize global and local PCs to be random Gaussian vectors. We run each experiment with the same stepsize $\eta = 0.1$ but from 10 different random initializations and collect the reconstruction error in each communication round τ. The reconstruction

error is defined as the objective in (5) divided by the number of observations $n_{(i)}$:

$$\text{Reconstruction error} = \frac{1}{N}\sum_{i-1}^{N}\frac{1}{n_{(i)}}\left\|\boldsymbol{Y}_{(i)} - \left(\boldsymbol{P}_{\boldsymbol{U}} + \boldsymbol{P}_{\boldsymbol{V}_{(i)}}\right)\boldsymbol{Y}_{(i)}\right\|_F^2 \tag{29}$$

Results are shown in Figure 3. From Figure 3(left), we can see that `PerPCA` indeed enjoys linear convergence. Furthermore, bluer curves have a larger slope, which indicates that a larger θ leads to faster convergence. Such a finding is corroborated by Figure 3(right), which plots the log error at the 100-th communication round with respect to θ. It is clear that the log error decreases linearly with the increase in θ. These results thus confirm insights from Theorem 10.



(a) Log reconstruction error vs communication round

(b) Final log reconstruction error vs θ

Figure 3: Left: the learning curve of the reconstruction error. Each curve represents one set of experiments with one θ. The bluer the curve is, the larger θ is. Right: log reconstruction error after 100 communication rounds for datasets with different misalignment parameters θ. We run each experiment 10 times, each with a different random initialization. The red line represents the mean log error after 100 communication rounds for the ten experiments, and the blue-shaded region shows the confidence interval.

7.1.2 Dependence of Statistical Error on n

Knowing that `PerPCA` converges rather quickly, we can use the final iterates of Algorithm 2 as an estimate of the optimal solution to problem (7). To show that the estimate can indeed recover the true local and global PCs, we calculate the subspace error between eigenspace estimates and true values defined in (13),

$$\text{Subspace error} = \|\boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{\Pi}_g\|_F^2 + \frac{1}{N}\sum_{i=1}^{N}\left\|\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} - \boldsymbol{\Pi}_{(i)}\right\|_F^2 \tag{30}$$

Remember that Theorem 1 shows that such error should decrease to 0 as the number of observations on each client approaches infinity. Additionally, Corollary 3 gives a finite-sample error bound of the subspace error.

Here, we benchmark with a one-shot approach `distPCA` (Fan et al., 2019). However, we provide a simple variant of `distPCA` to make it amenable for personalization. For standard `distPCA`, each client first calculates the top $r_1 + r_2$ principal components and sends them to the server. The server then concatenates all the received PCs into a $d \times N(r_1 + r_2)$ matrix and calculates the top r_1 principal components of the matrix. To enable personalization in `distPCA`, we take the following route: we use the obtained top r_1 principal components $\boldsymbol{U}_{\texttt{distPCA}}$ as estimates of the global principal components. Then, we estimate local PCs with the help of the global ones. Specifically, the global PCs $\boldsymbol{U}_{\texttt{distPCA}}$ are sent back to clients. Each client then deflates the sample covariance matrix $\boldsymbol{S}_{(i),deflate} = (\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_{\texttt{distPCA}}})\, \boldsymbol{S}_{(i)}\, (\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_{\texttt{distPCA}}})$, and calculates the top r_2 principal components of $\boldsymbol{S}_{(i),deflate}$ as local PCs.

To analyze the statistical consistency, we run `PerPCA` on datasets with varying numbers of observations $n_{(i)}$ and compare with the benchmark algorithm `distPCA`. We set $n_{(1)} = n_{(2)} = \cdots = n$. We fix data dimension $d = 15$ and generate data from 2 global PCs and 10 local PCs. On each client, the variances contributed by local PCs are set to be 100 times larger than those contributed by global PCs to simulate large heterogeneity. This is achieved by setting the standard deviations of $\phi_{(i),q}$ to be 10 times smaller than $\varphi_{(i),q}$ in data-generating model (1). We use 100 clients. Among them 50 clients have n observations, and the rest 50 clients only have $\frac{1}{10}n$ observations. We run both algorithms and estimate the subspace error (30) from 5 different random seeds.

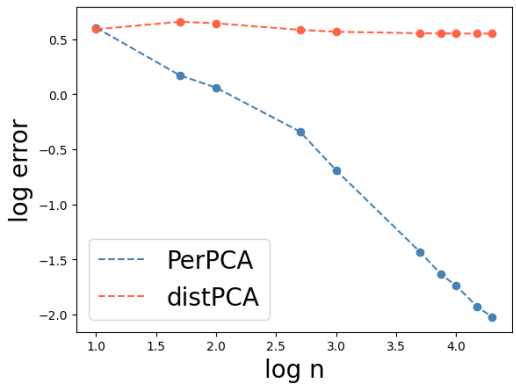


Figure 4: Log subspace error vs local observations n. `PerPCA` is consistent while `distPCA` is not.

Results in Figure 4 show that `PerPCA` achieves smaller statistical error for almost all n, and more importantly, the error decreases with n, which indicates that `PerPCA` gives consistent estimates of global and local PCs. When the error is small, the slope of the curve is approximately -1, which matches the theoretical error upper bound $O\left(\frac{1}{n}\right)$ in Corollary 3.

In comparison, the statistical error of `distPCA` does not decrease even when n is very large, implying that the method is not consistent for heterogeneous datasets. This result also

sheds light on an important insight. **Simply learning global components and using them for personalization in a train-then-personalize philosophy is not optimal, as global components from aggregated data may not contain useful information required for personalization**.

7.1.3 Dependence of Statistical Error on d

We also examine the performance of `PerPCA` on data with different dimensions d. We fix $n = 10000$ and generate data with different d. Other settings are the same as Section 7.1.2. We calculate the subspace error of estimates given by `PerPCA` and `distPCA`. Results are plotted in Figure 5.

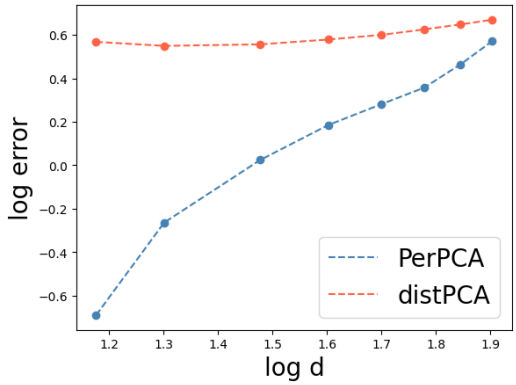


Figure 5: Log error vs data dimension d.

From Figure 5, `PerPCA` still achieves smaller statistical error for all d. Also, the error grows almost quadratically with d, which again matches the upper bound in Corollary 3.

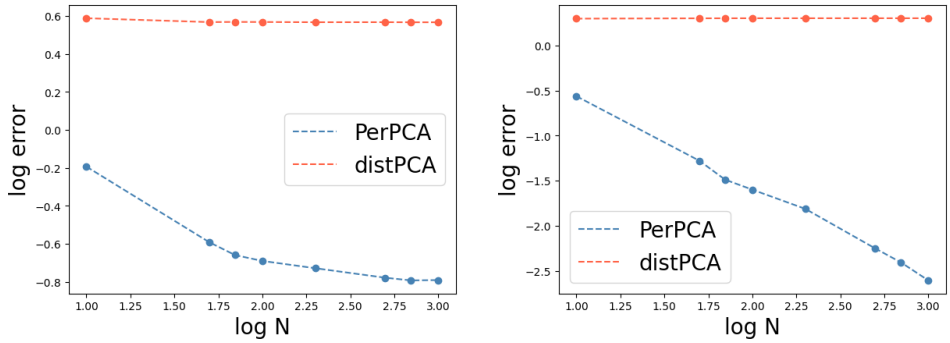
7.1.4 Dependence of Statistical Error on N

Now, we explore whether the number of clients N affects the statistical error. We fix $d = 15$, $n = 10000$, and change N from 10 to 1000. The other settings are also the same as in Section 7.1.2. After obtaining global and local PCs, we calculate the subspace error of both global and local PCs (30) and the subspace error of only global PCs $\|\boldsymbol{P}_{\boldsymbol{U}} - \boldsymbol{\Pi}_g\|_F^2$. Results are plotted in Figure 6.

Figure 6(a) shows that when N increases, the average subspace error decreases slowly. The decreasing trend is more conspicuous for the subspace error of global PCs shown in Figure 6(b). This is understandable as when more clients participate in `PerPCA`, more observations are available. Thus, global PCs can be better estimated.

7.1.5 Shared knowledge

When the PCs on different clients are extremely heterogeneous, it is natural to ask whether clients are sharing knowledge and learning from each other in `PerPCA`. Corollary 3 indicates that clients can benefit from participating in the collaborative learning process from a theoretical perspective. In this section, we show numerical results on how the learned global components improve client-level predictions.



(a) Average of subspace error of global and local PC estimates

(b) Subspace error of global PC estimates

Figure 6: Left: the average of local and global PCs' subspace error. Right: Global PCs' subspace error.

The dataset on client i is split into a training set $\boldsymbol{Y}_{(i),train}$ and testing set $\boldsymbol{Y}_{(i),test}$. We use the training set $\boldsymbol{Y}_{(i),train}$ to find estimates for global and local PCs and the testing set to calculate the testing error. We focus on the reconstruction error defined in (29). As in Section 7.1.2, we simulate two groups of clients with highly unbalanced dataset sizes. One group of clients has n observations. We call them data-rich clients. The other group of clients have only $\frac{1}{10}n$ observations. We call them data-sparse clients. We set $N = 100$ and $n = 100$.

In this experiment, we compare `PerPCA` with 3 benchmarks: `indivPCA`, `CPCA`, and `distPCA`. For `indivPCA`, each client uses their own data to calculate PCs independently without any knowledge sharing. `CPCA` represents PCA on the pooled data from all clients, i.e., all data are uploaded to a central server, and PCA is learned on the aggregated dataset. For fair comparisons, we allow `indivPCA` and `CPCA` to retain $r_1 + r_2$ principal components. The results of testing reconstruction error averaged over the groups are shown in Table 2. `Ground Truth` corresponds to the testing loss by the true PCs.

Client Group	`indivPCA`	`CPCA`	`disPCA`	`PerPCA`	`Ground Truth`
Data sparse	1.87 ± 0.01	2.07 ± 0.01	1.91 ± 0.01	$\mathbf{1.68} \pm 0.02$	1.50 ± 0.01
Data rich	1.80 ± 0.01	2.10 ± 0.01	1.88 ± 0.01	$\mathbf{1.52} \pm 0.01$	1.50 ± 0.01

Table 2: Testing reconstruction error averaged on each group

From Table 2, it is clear that `PerPCA` achieves the smallest testing error in both the data-sparse and the data-rich group, thus having the best predictive performance. As `PerPCA` outperforms `indivPCA`, we can conclude that `PerPCA` learns useful shared knowledge. The results highlight `PerPCA`'s ability to extract common features from heterogeneous datasets. Also, `CPCA` exhibits the worst performance. This again highlights the need for personalized learning when data comes from heterogeneous sources.

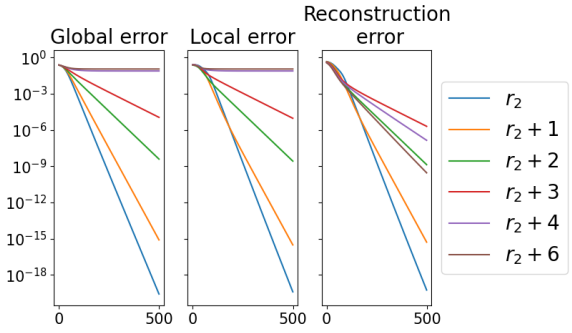


Figure 7: Simulations where we choose $\hat{r}_2 \geq r_2$.

7.1.6 Overpametrization

In practice, when the true rank r_1 and $r_{2,(i)}$'s are unknown, practitioners may choose the rank of $\boldsymbol{U}$ and $\boldsymbol{V}_{(i)}$ to be larger than the ground truth. This is called an overparametrized regime (Zhuo et al., 2021). Overparametrization is a common technique for PCA and matrix factorization. Here we investigate the numerical performance of `PerPCA` in an overparametrized regime.

We use synthetic data to analyze the convergence behavior of `PerPCA`. We set $d = 30$ and $r_1 = 1$, $r_{2,(i)} = r_2 = 1$. Then we randomly generate data $\{\boldsymbol{Y}_{(i)}\}$ for $N = 20$ clients and calculate the corresponding covariance matrix $\{\boldsymbol{S}_{(i)}\}$. The data are generated without noise to better understand the convergence.

We run overparametrized `PerPCA` on the generated data. More specifically, we choose orthonormal matrices $\boldsymbol{U} \in \mathbb{R}^{d\times\hat{r}_1}$ and $\boldsymbol{V}_{(i)} \in \mathbb{R}^{d\times\hat{r}_2}$ with rank $\hat{r}_1 \geq r_1$ and $\hat{r}_2 \geq r_2$ in Algorithm 2. Since, in practice, people may over-parametrize the rank of both global and local PCs differently, we study both cases separately.

Case 1: $\hat{r}_2 > r_2$. We choose the rank of local PCs $\hat{r}_2$ to be higher than the ground truth r_2, while keeping $\hat{r}_1 = r_1$. Then we run `PerPCA` starting from random initializations to obtain iterates $\boldsymbol{U}_\tau$ and $\{\boldsymbol{V}_{(i),\tau}\}$ for different τ. We analyze three metrics:

- Global error: $\frac{1}{N}\sum_{i=1}^{N}\frac{1}{n_{(i)}}\left\|\boldsymbol{P}_{\boldsymbol{U}_\tau}\boldsymbol{Y}_{(i)} - \boldsymbol{\Pi}_g\boldsymbol{Y}_{(i)}\right\|_F^2$
- Local error: $\frac{1}{N}\sum_{i=1}^{N}\frac{1}{n_{(i)}}\left\|\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\boldsymbol{Y}_{(i)} - \boldsymbol{\Pi}_{(i)}\boldsymbol{Y}_{(i)}\right\|_F^2$
- Reconstruction Error: $\frac{1}{N}\sum_{i=1}^{N}\frac{1}{n_{(i)}}\left\|\boldsymbol{Y}_{(i)} - \left(\boldsymbol{\Pi}_{(i)} + \boldsymbol{\Pi}_g\right)\boldsymbol{Y}_{(i)}\right\|_F^2$

Apparently, the reconstruction error is upper bounded by the sum of the local error and global error. We plot these metrics for different $\hat{r}_2$ in Figure 7.

There are a few interesting observations in Figure 7. Firstly, for all $\hat{r}_2$, the reconstruction errors decrease linearly. This is understandable as using a larger rank in local features $\hat{r}_2$ can add more representation power to the model, thus helping model fitting. As the covariance matrices are noiseless, the linear decrease of reconstruction error is also consistent with the standard matrix factorization results in Zhuo et al. (2021). Secondly, when the local features $\{\boldsymbol{V}_{(i)}\}$ are slightly parametrized $r_2 \leq \hat{r}_2 \leq r_2 + 3$, the global error and local error

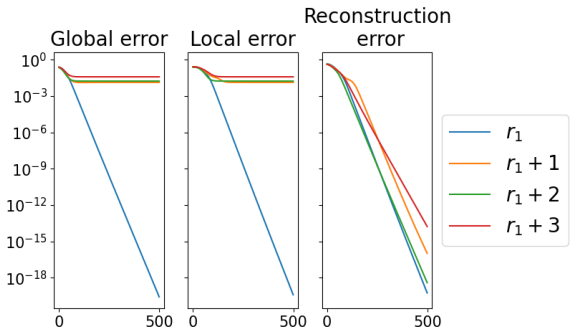


Figure 8: Simulations where we choose $\hat{r}_1 \geq r_1$.

also decrease linearly. Such results show that with slightly overparametrized $\{\boldsymbol{V}_{(i)}\}$, one can still recover the global features and the local ones. Thirdly, when $\hat{r}_2$ is very large, $\hat{r}_2 \geq r_2 + 4$, the global error and local error decrease sublinearly. In this highly overparametrized regime, though the reconstruction error decreases to zero linearly, the learned global and local features do not converge to the ground truth equally fast.

When the ground truth $\boldsymbol{\Pi}_g$ and $\boldsymbol{\Pi}_{(i)}$ are unknown, one cannot evaluate the local and global error. Therefore, we propose the estimated misalignment θ_{est} value as a statistic indicative of the global-local separation:

$$\theta_{est} = 1 - \lambda_{\max}\left(\frac{1}{N}\sum_{i=1}^{N} \boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i)}}\right)$$

where $\hat{\boldsymbol{V}}_{(i)}$ is the recovered local PCs on client i. θ_{est} measures how different the local features are.

We calculate θ_{est} for different ranks of $\boldsymbol{V}_{(i)}$ and show the results in Table 3.

$\hat{r}_2$	r_2	r_2+1	r_2+2	r_2+3	r_2+4	r_2+6
θ_{est}	0.90	0.69	0.36	0.19	1.8×10^{-6}	2.5×10^{-9}

Table 3: Misalignment value θ for different ranks of matrix $\boldsymbol{V}_{(i)}$

From Table 3, when $\hat{r}_2$ increases from r_2+3 to r_2+4, θ_{est} decreases rapidly from 0.19 to almost 0. Such change indicates that the local features are very aligned when $\hat{r}_2 = r_2 + 4$. Thus, local features are not "distinguishable". The abrupt changes θ_{est} echo the results in Figure 7: when θ_{est} is small, local PCs are similar, and the separation between local and global PCs is not clear.

Case 2: $\hat{r}_1 > r_1$. Similarly, we choose the rank of global PCs $\hat{r}_1$ to be higher than the ground truth r_1, while keeping $\hat{r}_2 = r_2$. The results are shown in Figure 8.

Figure 8 demonstrates different qualitative behaviors than Figure 7. Even when $\hat{r}_1$ is slightly overparametrized, $\hat{r}_1 = r_1 + 1$, the global and local errors do not linearly decrease to 0. Yet the fitting error for all cases decreases linearly. The comparison implies that when $\boldsymbol{U}$ is overparametrized, the combined features $\boldsymbol{U}$ and $\{\boldsymbol{V}_{(i)}\}$ can still explain well the data variance, but may not exactly characterize the global and local features.

In light of the insights gained, we recommend that practitioners carefully select small values for r_1 and r_2 in a way that ensures a small reconstruction error while also maintaining a large value for θ_{est} if they suspect heterogeneity among the data sources.

7.1.7 Clustering based on local principal components

Apart from capturing the variance structure in the data, the learned local and global components can reveal high-level information about the client's interrelatedness. Below we present an interesting application of `PerPCA` in client clustering.

An important question in federated and distributed learning is how to cluster clients based on some summary statistics from their data. This is usually done by exploiting some distance metrics over the estimated parameters or gradients (Sattler et al., 2019) from each client. `PerPCA` can pose an alternative approach for client clustering based on local PCs. The intuition is that by focusing on local PCs, differences across clients are more explicit compared to the raw data. More specifically, when $r_{2,(1)} = \cdots = r_{2,(N)} = r_2$, one can calculate the subspace distance between client i and j $\rho_{i,j}$ defined as:

$$\rho_{i,j} = \frac{1}{r_2} \left\| \boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i)}} - \boldsymbol{P}_{\hat{\boldsymbol{V}}_{(j)}} \right\|_F^2 \tag{31}$$

If the column space of $\hat{\boldsymbol{V}}_{(i)}$ and $\hat{\boldsymbol{V}}_{(j)}$ are more similar, $\rho_{i,j}$ will be smaller.

The $\rho_{i,j}$'s measure the closeness of local subspaces, thus revealing a similarity structure among clients. They form an $N \times N$ matrix $\boldsymbol{\rho}$. As such, simple spectral clustering (Hastie et al., 2009) on $\boldsymbol{\rho}$ can be used to analyze the relations among different clients.

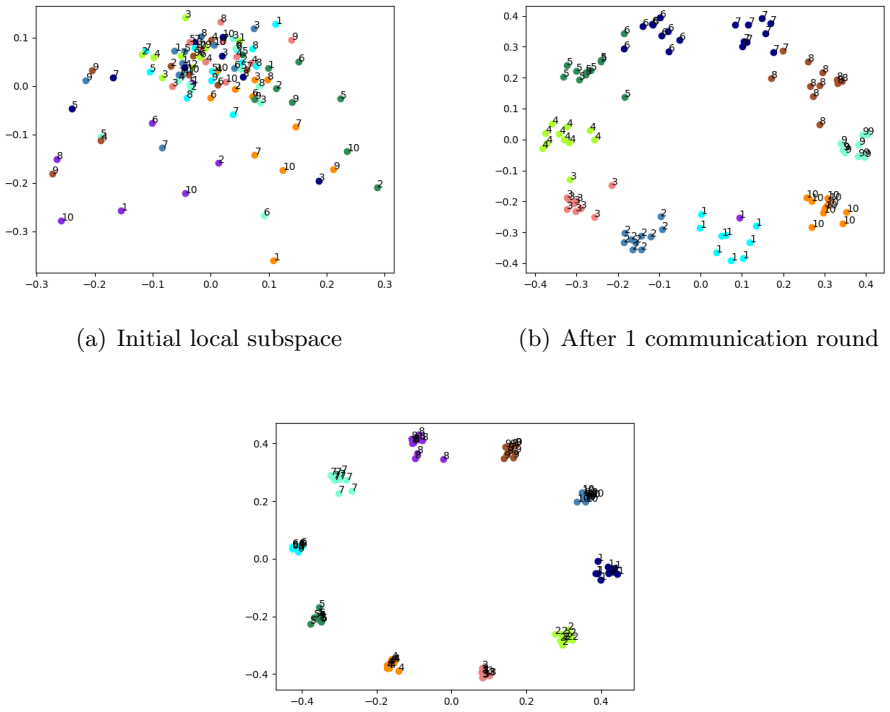
As an example, we generate clients from 10 different client groups. Clients in one group have the same local PCs. Different groups have different local PCs. The data on clients within one group thus have a similar variance structure. We set $r_1 = 2$, $r_2 = 3$, and $d = 15$. We apply `PerPCA` and calculate the matrix $\boldsymbol{\rho}(\tau)$ with each communication round τ. We omit the dependence $\boldsymbol{\rho}(\tau)$ on τ for simplicity. Then, we use multidimensional scaling (MDS) (Hastie et al., 2009) and spectral clustering on $\boldsymbol{\rho}$. Results are shown in Figure 9.

Since local PCs are randomly initialized, it is hard to find meaningful structures from initialization in Figure 9(a). However, after only one communication round, the true structure emerges in Figure 9(b). After 30 communication rounds, clients can be effectively clustered based on their learned local PCs.

7.2 An illustrative example in comparison to `Robust PCA`

The philosophy of finding common and unique features can be applied to other tasks beyond explaining data variance. In this section, we use a simple example to demonstrate how `PerPCA` can separate shared and unique features from image data.

We start by comparing `PerPCA` with `Robust PCA`. Though `Robust PCA` is proposed to learn low-rank and sparse parts, it is also potentially useful in finding irregular and common patterns from a dataset. When data come from different sources $\{\boldsymbol{Y}_{(i)}\}$ and have equal number of columns $n_{(1)} = \cdots = n_{(N)} = n$, one can stack them into one matrix $\boldsymbol{Y}_{\text{stack}} = \left[\text{vec}\left(\boldsymbol{Y}_{(1)}\right), \cdots, \text{vec}\left(\boldsymbol{Y}_{(N)}\right)\right]$. Then `Robust PCA` can be applied on the stacked matrix $\boldsymbol{Y}_{\text{stack}} \in \mathbb{R}^{nd \times N}$ to distinguish low rank and sparse parts. The common wisdom is to



(a) Initial local subspace

(b) After 1 communication round

(c) After 30 communication rounds

Figure 9: MDS of the distance matrix $\boldsymbol{\rho}$. Color denotes the output of the spectral clustering algorithm. Numbers denote the true cluster labels.

use a low-rank part to represent shared patterns and a sparse part to represent irregular trends (Candes et al., 2011).

The underlying assumption of such an approach is that unique features are somewhat sparse among all datasets. However, there are cases where a sparse matrix cannot model unique features. An example is shown in Table 4. We create 4 images of different icons (triangle, disk, cross, and cloud) on similar background textures using PowerPoint and distinguish the icons from the background. As a greyscale image can naturally be represented by an observation matrix with dimensions of its height and width, we can construct 4 datasets representing 4 images. Then we apply `PerPCA` and `Robust PCA` to identify the icons.

From Table 4, it is apparent that `Robust PCA` does not perform well as it cannot recover the icons and always leaves shadows of icons on other images, probably because icons occupy a large space in the image and thus cannot be modeled by sparse noise. `PerPCA` recovers the icons by projecting the images to the subspace spanned by local PCs. The third row in Table 4 shows that `PerPCA` has decent performance as the icons recovered have clear edges and shapes.

This highlights the need for personalized inference in many applications where PCA is utilized.

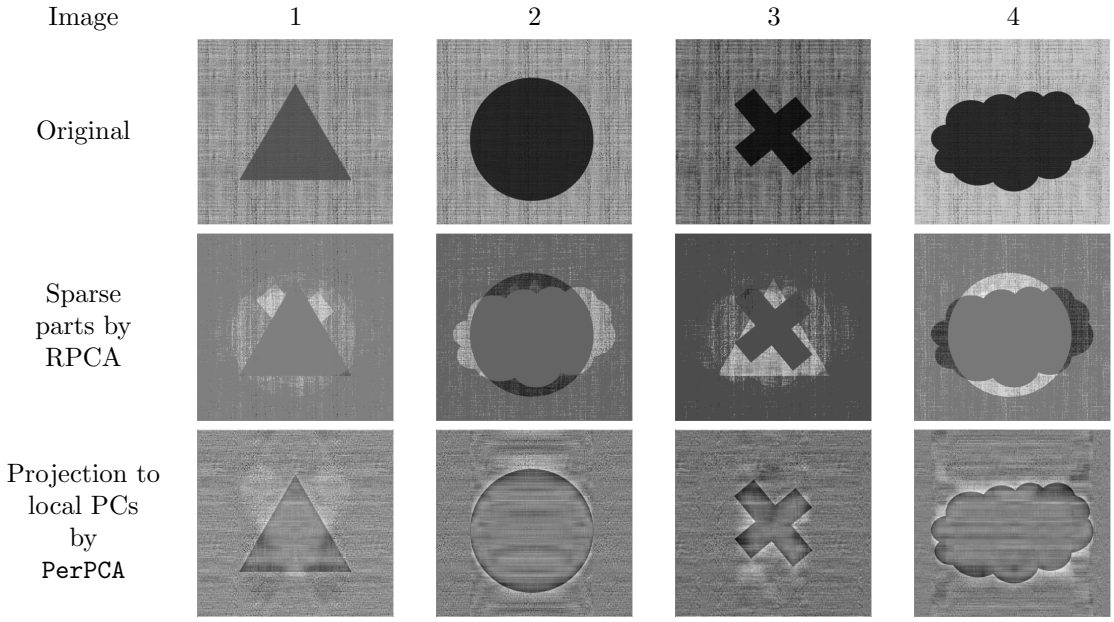


Table 4: A comparison of `PerPCA` and `Robust PCA` on images of icons on background textures.

7.3 Real-life federated dataset

We also apply our algorithm on FEMNIST (Caldas et al., 2019) and CIFAR10 (Krizhevsky et al., 2009).

FEMNIST is a popular dataset in federated analytics. It consists of greyscale images of handwritten digits and English letters contributed by 3550 different writers. Each image has $28 \times 28 = 784$ pixels. Different writers have different writing styles. Thus, the datasets are inherently heterogeneous. Our task is to learn a few PCs that can represent the dataset. On average, each client has 89 images. We represent an image by a vector in $\mathbb{R}^{784}$. For these vectors, we randomly choose 80% of them to form the training set and take the rest as the test set.

CIFAR10 is a multiclass image dataset. It consists of 60000 images from 10 classes. To simulate a heterogeneous setting, we separate the training and testing set of CIFAR10 into 20 parts such that each part contains images from only 2 classes. Then, we assign the separated parts to 20 clients. The data partition scheme is consistent with federated learning literature (McMahan et al., 2017). Then we use similar data preprocessing procedures to vectorize the images on each client and construct the dataset $\{\boldsymbol{Y}_{(i)}\}$.

We use `PerPCA`, `indivPCA`, `CPCA`, and `distPCA` to fit PCs on training sets. Then, we evaluate the reconstruction error (29) on both training and test sets. The experiments are repeated 3 times to calculate the mean and standard deviations . The results are shown in Table 5.

As the reconstruction error represents the difference between the original and reconstructed image, it represents how well the learned PCs can characterize the features in the

Reconstruction error	`indivPCA`	`CPCA`	`disPCA`	`PerPCA`
FEMNIST Training	$\mathbf{0.49} \pm 0.01$	1.72 ± 0.01	1.43 ± 0.01	1.44 ± 0.02
CIFAR10 Training	$\mathbf{105.68} \pm 0.01$	114.79 ± 0.01	113.56 ± 0.01	113.69 ± 0.02
FEMNIST Testing	1.97 ± 0.03	1.73 ± 0.01	1.73 ± 0.01	$\mathbf{1.70} \pm 0.01$
CIFAR10 Testing	120.79 ± 0.02	115.44 ± 0.02	115.43 ± 0.01	$\mathbf{115.33} \pm 0.02$

Table 5: The mean and standard deviations of the training and testing reconstruction error on FEMNIST

image. In Table 5, `indivPCA` achieves the lowest training error but incurs high testing error, suggesting that learned PCs overfit the training sets. `PerPCA` has the lowest testing loss both in FEMNIST and CIFAR10, highlighting `PerPCA`'s ability to leverage common knowledge with unique trends to find better features from data.

7.4 Other Applications

Besides the experiments in the previous sections, `PerPCA` can excel in various tasks that require separating shared and unique features. In this section, we will use video segmentation and topic extraction as two examples to show the applicability of `PerPCA`.

7.4.1 Video segmentation

The task of video segmentation is to separate moving parts (foreground) from stationary backgrounds in a video. For a video with F frames, where each frame is an image with width W and height H, we can model it as F separated datasets. Each dataset has the data of one image frame or H observations from $\mathbb{R}^W$. Therefore, we can naturally apply `PerPCA` to recover local and global PCs from the constructed datasets of all frames. Intuitively, the global PCs should capture shared features across all frames, representing the stationary background. Meanwhile, local PCs capture unique features in each frame corresponding to the moving parts. Hence, after obtaining the global and local PCs, we project the original picture onto the subspace spanned by these components to extract the background and foreground segments.

We use a surveillance video example from Vacavant et al. (2012). We set $r_1 = 50$ and $r_{2,(1)} = \cdots = r_{2,(N)} = 50$ and apply Algorithm 2 with choice 2. Some segmentation results are shown in Table 6. From Table 6, we can see that backgrounds and moving parts are well separated by global and local PCs, validating `PerPCA`'s ability to find common and unique features in image datasets.

7.4.2 Topic extraction

`PerPCA` is also useful in modeling changing topics in language datasets. As a demonstration, we analyze the presidential debate transcriptions from 1960 to 2020 (Asokan, 2022). The goal is to extract key debating topics for each specific election year.

The dataset contains 9135 dialogues in 46 debates from 13 election years, where one dialogue is the speech the speaker makes in the debate before another person speaks. After

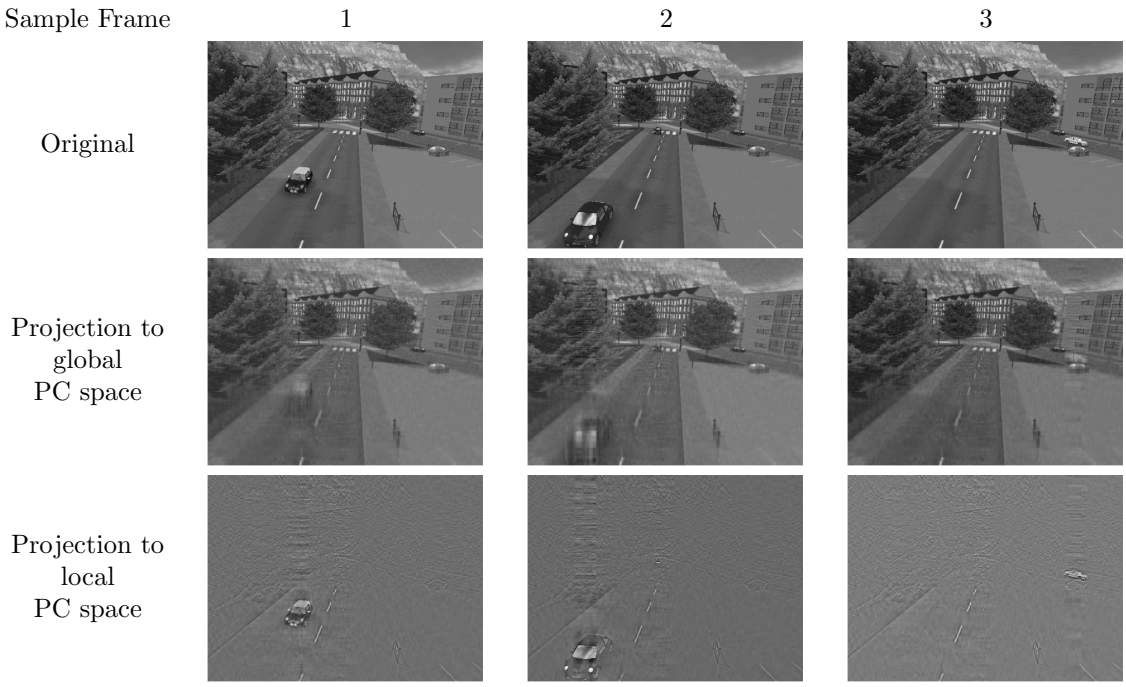


Table 6: Video segmentation. We separate moving cars from the background in a video from (Vacavant et al., 2012).

Table 7: U.S. presidential debate key topics represented by local & global PCs

Year	Top local principal components words
1960	peace, Castro, Africa, Kennedy, now, world, ...
1976	billion, Carter, Governor, Africa, Ford, people, world, ...
1980	coal, oil, money, energy, Social, Security, Reagan, ...
1984	Union, tax, Soviet, arms, leadership, proposal, ...
1988	drug, young, strong, build, future, enforcement, good, ...
1992	Bill, school, children, care, health, taxes, reform, plan, control, ...
1996	Clinton, Security, Medicare, budget, tax, Dole, Bob, ...
2000	school, public, plan, children, money, Social, Security, health, tax, ...
2004	wrong, plan, cost, free, Saddam, troops, Iraq, war, health, tax, ...
2008	nuclear, oil, troops, Iraq, Afghanistan, Pakistan, health, Iran, energy, ...
2012	million, small, business, China, Medicare, Romney, jobs, tax, ...
2016	Russia, Trump, Hillary, companies, taxes, Mosul, Iran, deal, ...
2020	Harris, Pence, Trump, down, Joe, Biden, jobs, Donald, health, ...
Common words	Tax, country, States, make, world, money, people, cut, ...

we remove common English words such as “you”, “I”, “and”, “at”, “that”, from the text corpus, there are 5464 different words used in the dataset.

We model the dataset as a collection of 13 separate datasets, each of which has all the dialogues in one election year. To construct the observation matrix $\boldsymbol{Y}_{(i)}$, we first use one-hot encoding to map an English word into a vector in $\mathbb{R}^{5464}$. Then, we add all vectors corresponding to words that appear in one dialogue. The added vector is one observation in $\mathbb{R}^{5464}$. $\boldsymbol{Y}_{(i)}$ is formed by concatenating observations corresponding to dialogues in the election year.

With the datasets constructed, we run `PerPCA` for 20 communication rounds to extract local PCs. We set $r_1 = 2$ and $r_{2,(1)} = \cdots = r_{2,(N)} = 2$. To show the key topics represented by the two local PCs, we find the words corresponding to the dimensions in each local and global PC that have the top 20 largest absolute values. Table 7 contains the most informative keywords from the top 20 keywords obtained.

From Table 7, one can find different debating key topics for different years. For some years, the key topics are about public finance and domestic economic reform. For others, the key topics are more about international relations. These topics represent the central issues at a specific time in history.

8. Conclusion

This work proposes `PerPCA`, a systematic approach to decouple shared and unique features from heterogeneous datasets. We show that the problem is well formulated, and consistency can be guaranteed under mild conditions. A fully federated algorithm with a convergence guarantee is designed to efficiently obtain global and local PCs from noisy observations. Extensive simulations highlight `PerPCA`'s ability to separate shared and unique features in various applications.

As the first systematic approach to decouple shared and unique features quantitatively, we envision that `PerPCA` can find use across various downstream analytics such as interpretability, clustering, classification, change detection, and transfer/federated learning. Within these areas, one can leverage unique knowledge so that differences become more explicit and leverage shared knowledge to transfer useful information from one source to another. Exploration along these directions may be promising.

In addition, within `PerPCA`, there are several avenues for expansion and exploration. On the optimization front, it is promising to design algorithms that can converge faster, or require lower computation resources, including Grassmannian gradient descent, and adaptive stepsize Stiefel gradient descent. Further, extensions of `PerPCA` that consider missing data, large noise, sparse factors, or malicious intruders are important directions for future work.

9. Acknowledgements and Disclosure of Funding

We thank the anonymous reviewers for their constructive feedback. This research is supported by Raed Al Kontar's National Science Foundation (NSF) CAREER Grant 2144147.

Appendix A. Proof of Theorem 1

In this section, we will show the proof of Theorem 1. We first use standard perturbation analysis on the eigenspaces of $\mathbf{\Sigma}_{(i)}$ (Vu et al., 2013). By assumption, $\mathbf{\Pi}_g$ and $\mathbf{\Pi}_{(i)}$ are the projections onto top eigenspaces of $\mathbf{\Sigma}_{(i)}$, therefore for any orthogonal projection matrix $\boldsymbol{P}_{\hat{\boldsymbol{U}}}$ and $\boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i)}}$, we have:

$$\begin{aligned}
&\left\langle \mathbf{\Sigma}_{(i)}, \mathbf{\Pi}_g + \mathbf{\Pi}_{(i)} - \boldsymbol{P}_{\hat{\boldsymbol{U}}} - \boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i)}} \right\rangle \\
&= \left\langle \left(\mathbf{\Pi}_g + \mathbf{\Pi}_{(i)}\right) \mathbf{\Sigma}_{(i)}, \boldsymbol{I} - \boldsymbol{P}_{\hat{\boldsymbol{U}}} - \boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i)}} \right\rangle - \left\langle \left(\boldsymbol{I} - \mathbf{\Pi}_g - \mathbf{\Pi}_{(i)}\right) \mathbf{\Sigma}_{(i)}, \boldsymbol{P}_{\hat{\boldsymbol{U}}} + \boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i)}} \right\rangle \\
&\geq \lambda_{r_1+r_{2,(i)}} \left(\left(\mathbf{\Pi}_g + \mathbf{\Pi}_{(i)}\right) \mathbf{\Sigma}_{(i)}\right) \left\langle \mathbf{\Pi}_g + \mathbf{\Pi}_{(i)}, \boldsymbol{I} - \boldsymbol{P}_{\hat{\boldsymbol{U}}} - \boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i)}} \right\rangle \\
&- \lambda_1 \left(\left(\boldsymbol{I} - \mathbf{\Pi}_g - \mathbf{\Pi}_{(i)}\right) \mathbf{\Sigma}_{(i)}\right) \left\langle \boldsymbol{I} - \mathbf{\Pi}_g - \mathbf{\Pi}_{(i)}, \boldsymbol{P}_{\hat{\boldsymbol{U}}} + \boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i)}} \right\rangle \\
&\geq \delta \left(r_1 + r_{2,(i)} - \left\langle \mathbf{\Pi}_g + \mathbf{\Pi}_{(i)}, \boldsymbol{P}_{\hat{\boldsymbol{U}}} + \boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i)}} \right\rangle \right)
\end{aligned}$$

Summing both sides for i from 1 to N, we have:

$$\sum_{i=1}^{N} \left\langle \mathbf{\Sigma}_{(i)}, \mathbf{\Pi}_g + \mathbf{\Pi}_{(i)} - \boldsymbol{P}_{\hat{\boldsymbol{U}}} - \boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i)}} \right\rangle \geq \delta \sum_{i=1}^{N} \left(r_1 + r_{2,(i)} - \left\langle \mathbf{\Pi}_g + \mathbf{\Pi}_{(i)}, \boldsymbol{P}_{\hat{\boldsymbol{U}}} + \boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i)}} \right\rangle \right) \quad (32)$$

Since $\boldsymbol{P}_{\hat{\boldsymbol{U}}}$ and $\{\boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i)}}\}$ are the optimal solutions to (7), and $\mathbf{\Pi}_g$ and $\{\mathbf{\Pi}_{(i)}\}$ are feasible, we know that:

$$\sum_{i=1}^{N} \left\langle \boldsymbol{S}_{(i)}, \boldsymbol{P}_{\hat{\boldsymbol{U}}} + \boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i)}} \right\rangle \geq \sum_{i=1}^{N} \left\langle \boldsymbol{S}_{(i)}, \mathbf{\Pi}_g + \mathbf{\Pi}_{(i)} \right\rangle \quad (33)$$

Combining (32) and (33), we can obtain:

$$\sum_{i=1}^{N} \left\langle \boldsymbol{S}_{(i)} - \mathbf{\Sigma}_{(i)}, \boldsymbol{P}_{\hat{\boldsymbol{U}}} + \boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i)}} - \mathbf{\Pi}_g - \mathbf{\Pi}_{(i)} \right\rangle \geq \delta \sum_{i=1}^{N} \left(r_1 + r_{2,(i)} - \left\langle \mathbf{\Pi}_g + \mathbf{\Pi}_{(i)}, \boldsymbol{P}_{\hat{\boldsymbol{U}}} + \boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i)}} \right\rangle \right)$$

We can use the Cauchy-Schwartz inequality to further bound the left-hand side as:

$$\left\langle \boldsymbol{S}_{(i)} - \mathbf{\Sigma}_{(i)}, \boldsymbol{P}_{\hat{\boldsymbol{U}}} + \boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i)}} - \mathbf{\Pi}_g - \mathbf{\Pi}_{(i)} \right\rangle \leq \left\| \boldsymbol{S}_{(i)} - \mathbf{\Sigma}_{(i)} \right\|_F \left\| \boldsymbol{P}_{\hat{\boldsymbol{U}}} + \boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i)}} - \mathbf{\Pi}_g - \mathbf{\Pi}_{(i)} \right\|_F$$

Notice that

$$\begin{aligned}
&\left\| \boldsymbol{P}_{\hat{\boldsymbol{U}}} + \boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i)}} - \mathbf{\Pi}_g - \mathbf{\Pi}_{(i)} \right\|_F \\
&= \sqrt{\left\| \boldsymbol{P}_{\hat{\boldsymbol{U}}} + \boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i)}} \right\|_F^2 + \left\| \mathbf{\Pi}_g + \mathbf{\Pi}_{(i)} \right\|_F^2 - 2 \left\langle \boldsymbol{P}_{\hat{\boldsymbol{U}}} + \boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i)}}, \mathbf{\Pi}_g + \mathbf{\Pi}_{(i)} \right\rangle} \\
&= \sqrt{2} \sqrt{r_1 + r_{2,(i)} - \left\langle \boldsymbol{P}_{\hat{\boldsymbol{U}}} + \boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i)}}, \mathbf{\Pi}_g + \mathbf{\Pi}_{(i)} \right\rangle}
\end{aligned}$$

We thus have:

$$\begin{aligned}
&\sum_{i=1}^{N} \left\langle \boldsymbol{S}_{(i)} - \mathbf{\Sigma}_{(i)}, \boldsymbol{P}_{\hat{\boldsymbol{U}}} + \boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i)}} - \mathbf{\Pi}_g - \mathbf{\Pi}_{(i)} \right\rangle \\
&\leq \sqrt{2} \sqrt{\sum_{i=1}^{N} \left\| \boldsymbol{S}_{(i)} - \mathbf{\Sigma}_{(i)} \right\|_F^2} \sqrt{\sum_{i=1}^{N} \left[r_1 + r_{2,(i)} - \left\langle \boldsymbol{P}_{\hat{\boldsymbol{U}}} + \boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i)}}, \mathbf{\Pi}_g + \mathbf{\Pi}_{(i)} \right\rangle \right]}
\end{aligned}$$

by another application of Cauchy-Schwartz inequality.

Finally:

$$\frac{1}{N}\sum_{i=1}^{N}\left[r_1 + r_{2,(i)} - \left\langle \boldsymbol{P}_{\hat{\boldsymbol{U}}} + \boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i)}}, \boldsymbol{\Pi}_g + \boldsymbol{\Pi}_{(i)} \right\rangle\right] \leq \frac{2}{N\delta^2}\sum_{i=1}^{N}\left\|\boldsymbol{S}_{(i)} - \boldsymbol{\Sigma}_{(i)}\right\|_F^2 \tag{34}$$

The relation (34) slightly extends the standard result from matrix perturbation theory. However, it only shows the summation of $\boldsymbol{P}_{\hat{\boldsymbol{U}}}$ and $\boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i)}}$ is close to the summation of $\boldsymbol{\Pi}_g$ and $\boldsymbol{\Pi}_{(i)}$. One cannot infer additional information about the closeness of $\boldsymbol{P}_{\hat{\boldsymbol{U}}}$ to $\boldsymbol{\Pi}_g$, or $\boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i)}}$ to $\boldsymbol{\Pi}_{(i)}$. In other words, (34) alone does not ensure that the recovered global and local PCs correspond to true PCs.

Such a guarantee is too weak in practice when we want to know if the solved $\boldsymbol{U}$ and $\boldsymbol{V}_{(i)}$'s are close to the ground truth. Fortunately, we can show that this is indeed the case if the problem satisfies the identifiability assumption 4.2. An important finding is the following lemma, which indicates that the closeness in direct sum space can lead to closeness in each global and local subspaces.

Lemma 13 *Suppose for $i = 1, \cdots, N$, $\boldsymbol{P}_{\boldsymbol{U}}$, $\boldsymbol{P}_{\boldsymbol{V}_{(i)}}$ and $\boldsymbol{P}^{\star}_{\boldsymbol{U}}$, $\boldsymbol{P}^{\star}_{\boldsymbol{V}_{(i)}}$ are projection matrices satisfying $\boldsymbol{P}_{\boldsymbol{U}}\boldsymbol{P}_{\boldsymbol{V}_{(i)}} = 0$ and $\boldsymbol{P}^{\star}_{\boldsymbol{U}}\boldsymbol{P}^{\star}_{\boldsymbol{V}_{(i)}} = 0$ for each i. Among them, $\boldsymbol{P}_{\boldsymbol{U}}$ and $\boldsymbol{P}^{\star}_{\boldsymbol{U}}$ have rank r_1, $\boldsymbol{P}_{\boldsymbol{V}_{(i)}}$ and $\boldsymbol{P}^{\star}_{\boldsymbol{V}_{(i)}}$ have rank $r_{2,(i)}$. If there exists a positive constant $\theta > 0$ such that*

$$\lambda_{max}\left(\frac{1}{N}\sum_{i=1}^{N}\boldsymbol{P}^{\star}_{\boldsymbol{V}_{(i)}}\right) \leq 1 - \theta$$

We have the following bound:

$$\sum_{i=1}^{N} r_1 + r_{2,(i)} - \left\langle \boldsymbol{P}_{\boldsymbol{U}} + \boldsymbol{P}_{\boldsymbol{V}_{(i)}}, \boldsymbol{P}^{\star}_{\boldsymbol{U}} + \boldsymbol{P}^{\star}_{\boldsymbol{V}_{(i)}} \right\rangle \leq N\left(r_1 - \langle \boldsymbol{P}^{\star}_{\boldsymbol{U}}, \boldsymbol{P}_{\boldsymbol{U}}\rangle\right) + \sum_{i=1}^{N} r_{2,(i)} - \left\langle \boldsymbol{P}^{\star}_{\boldsymbol{V}_{(i)}}, \boldsymbol{P}_{\boldsymbol{V}_{(i)}} \right\rangle \tag{35}$$

Also:

$$\sum_{i=1}^{N} r_1 + r_{2,(i)} - \left\langle \boldsymbol{P}_{\boldsymbol{U}} + \boldsymbol{P}_{\boldsymbol{V}_{(i)}}, \boldsymbol{P}^{\star}_{\boldsymbol{U}} + \boldsymbol{P}^{\star}_{\boldsymbol{V}_{(i)}} \right\rangle \geq \frac{\theta}{2}\left(N\left(r_1 - \langle \boldsymbol{P}^{\star}_{\boldsymbol{U}}, \boldsymbol{P}_{\boldsymbol{U}}\rangle\right) + \sum_{i=1}^{N} r_{2,(i)} - \left\langle \boldsymbol{P}^{\star}_{\boldsymbol{V}_{(i)}}, \boldsymbol{P}_{\boldsymbol{V}_{(i)}} \right\rangle\right) \tag{36}$$

The proof of Lemma 13 is at the end of Section G. By applying inequality (36) to (34), we can prove the desired error bound in Theorem 1.

Appendix B. KKT condition

We show the KKT conditions (24). The lagrangian to the objective (7) is:

$$\begin{aligned}\mathscr{L} = &\sum_{i=1}^{N}\left[\frac{1}{2}\mathrm{Tr}\left(\boldsymbol{U}^T\boldsymbol{S}_{(i)}\boldsymbol{U}\right) + \frac{1}{2}\mathrm{Tr}\left(\boldsymbol{V}_{(i)}^T\boldsymbol{S}_{(i)}\boldsymbol{V}_{(i)}\right) + \left\langle \boldsymbol{\Lambda}_{2,(i)}, \boldsymbol{V}_{(i)}^T\boldsymbol{V}_{(i)} - \boldsymbol{I}\right\rangle + \left\langle \boldsymbol{\Lambda}_{3,(i)}, \boldsymbol{U}^T\boldsymbol{V}_{(i)}\right\rangle\right] \\ &+ \left\langle \boldsymbol{\Lambda}_1, \boldsymbol{U}^T\boldsymbol{U}\right\rangle\end{aligned} \tag{37}$$

where $\mathbf{\Lambda}_1 \in \mathbb{R}^{r_1 \times r_1}$, $\mathbf{\Lambda}_{2,(i)} \in \mathbb{R}^{r_{2,(i)} \times r_{2,(i)}}$, and $\mathbf{\Lambda}_{3,(i)} \in \mathbb{R}^{r_1 \times r_{2,(i)}}$ are dual variables. The KKT conditions are:

$$\begin{cases} \boldsymbol{S}_{(i)}\boldsymbol{V}_{(i)} + \boldsymbol{V}_{(i)}\left(\mathbf{\Lambda}_{2,(i)} + \mathbf{\Lambda}_{2,(i)}^T\right) + \boldsymbol{U}\mathbf{\Lambda}_{3,(i)} = 0 \\ \sum_{i=1}^{N}\left[\boldsymbol{S}_{(i)}\boldsymbol{U} + \boldsymbol{V}_{(i)}\mathbf{\Lambda}_{3,(i)}^T\right] + \boldsymbol{U}\left(\mathbf{\Lambda}_1 + \mathbf{\Lambda}_1^T\right) = 0 \\ \boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{I}_{r_1},\ \boldsymbol{V}_{(i)}^T\boldsymbol{V}_{(i)} = \boldsymbol{I}_{r_{2,(i)}},\ \boldsymbol{U}^T\boldsymbol{V}_i = 0 \end{cases} \tag{38}$$

By left multiplying the first equation in (38) with $\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}} - \boldsymbol{P}_{\boldsymbol{V}_{(i)}}$, we have $\left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}} - \boldsymbol{P}_{\boldsymbol{V}_{(i)}}\right)\boldsymbol{S}_{(i)}\boldsymbol{V}_{(i)} = 0$, which is the first equation in (24). By left multiplying the first equation in (38) with $\boldsymbol{U}^T$, we have $\mathbf{\Lambda}_{(3,(i))} = -\boldsymbol{U}^T\boldsymbol{S}_{(i)}\boldsymbol{V}_{(i)}$. Plugging this into the second equation in (38), we have $\sum_{i=1}^{N}\left[\boldsymbol{S}_{(i)}\boldsymbol{U} - \boldsymbol{P}_{\boldsymbol{V}_{(i)}}\boldsymbol{S}_{(i)}\boldsymbol{U}\right] + \boldsymbol{U}\left(\mathbf{\Lambda}_1 + \mathbf{\Lambda}_1^T\right) = 0$. We then left multiply both sides again by $\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}}$. The second equation in (24) follows accordingly. One can also infer (38) from (24).

Appendix C. Proof of Theorem 2

Inspired by Birnbaum et al. (2013), in this section, we will use a "spiked population model" to demonstrate the lower bound. We will first use matrix perturbation analysis to estimate the leading order term for the estimation error of global PCs. Then, we verify our results through numerical experiments.

Proof To prove theorem 2, it suffices to find one set of parameters under which the statistical error is indeed $\Omega\left(\frac{1}{\theta} + \frac{1}{\delta^2}\right)$. For simplicity, we consider $N = 2$ and $r_1 = r_2 = 1$, i.e., each client is driven by one global feature and one local feature. We define a few needed signal vectors $\boldsymbol{w}_{1,1}, \boldsymbol{w}_{1,2}, \boldsymbol{w}_{2,1}, \boldsymbol{w}_{2,2} \in \mathbb{R}^4$ and a noise vector $\boldsymbol{w}_3 \in \mathbb{R}^4$ as

$$\begin{aligned} \boldsymbol{w}_{1,1} &= (\cos\gamma\sin\alpha, \sin\alpha\sin\gamma, \cos\alpha, 0)^T \\ \boldsymbol{w}_{1,2} &= (\cos\gamma\cos\alpha, \cos\alpha\sin\gamma, -\sin\alpha, 0)^T \\ \boldsymbol{w}_{2,1} &= (\cos\gamma\sin\alpha, -\sin\alpha\sin\gamma, \cos\alpha, 0)^T \\ \boldsymbol{w}_{2,2} &= (\cos\gamma\cos\alpha, -\cos\alpha\sin\gamma, -\sin\alpha, 0)^T \\ \boldsymbol{w}_3 &= (0, 0, 0, 1)^T \end{aligned} \tag{39}$$

Then, we define the population covariance matrix as,

$$\begin{aligned} \mathbf{\Sigma}_1 &= 2\boldsymbol{w}_{1,1}\boldsymbol{w}_{1,1}^T + \boldsymbol{w}_{1,2}\boldsymbol{w}_{1,2}^T + \varrho\boldsymbol{w}_3\boldsymbol{w}_3^T \\ \mathbf{\Sigma}_2 &= 2\boldsymbol{w}_{2,1}\boldsymbol{w}_{2,1}^T + \boldsymbol{w}_{2,2}\boldsymbol{w}_{2,2}^T + \varrho\boldsymbol{w}_3\boldsymbol{w}_3^T \end{aligned} \tag{40}$$

where ϱ is a constant $\varrho < 1$. In (40), $2\boldsymbol{w}_{1,1}\boldsymbol{w}_{1,1}^T + \boldsymbol{w}_{1,2}\boldsymbol{w}_{1,2}^T$ denotes the signal part in $\mathbf{\Sigma}_1$ and $\varrho\boldsymbol{w}_3\boldsymbol{w}_3^T$ denotes the noise part. Apparently, the model (40) satisfies assumption 4.1 in the main paper. The eigengap δ is $\delta = 1 - \varrho$.

It is easy to check that if we run `PerPCA` directly on the population covariance matrices $\{\mathbf{\Sigma}_{(1)}, \mathbf{\Sigma}_{(2)}\}$ defined in (40), the algorithm would recover the optimal global PC as

$$\boldsymbol{u} = (0, 0, 1, 0)^T$$

and the optimal local PCs as

$$\begin{aligned} \boldsymbol{v}_1 &= (\cos\gamma, \sin\gamma, 0, 0)^T \\ \boldsymbol{v}_2 &= (\cos\gamma, -\sin\gamma, 0, 0)^T \end{aligned} \tag{41}$$

It is also easy to see that the misalignment parameter $\theta = \sin^2\gamma$ when $0 \le \gamma \le \frac{\pi}{4}$.

We further introduce $\boldsymbol{v}^\perp{}_1$ and $\boldsymbol{v}^\perp{}_2$ as,

$$\begin{aligned} \boldsymbol{v}^\perp{}_1 &= (-\sin\gamma, \cos\gamma, 0, 0)^T \\ \boldsymbol{v}^\perp{}_2 &= (\sin\gamma, \cos\gamma, 0, 0)^T \end{aligned}$$

Now we consider the sample covariance matrices $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$. For simplicity, we assume they are only slightly perturbed from the population covariance matrices; $\boldsymbol{S}_1 = \boldsymbol{\Sigma}_1 + \varepsilon\delta\boldsymbol{S}_1$ and $\boldsymbol{S}_2 = \boldsymbol{\Sigma}_2 + \varepsilon\delta\boldsymbol{S}_2$, where $\varepsilon << 1$ is a small number, and $\delta\boldsymbol{S}_1$ and $\delta\boldsymbol{S}_2$ are defined as,

$$\begin{aligned} \delta\boldsymbol{S}_1 &= \boldsymbol{v}_1\boldsymbol{w}_3^T + \boldsymbol{w}_3\boldsymbol{v}_1^T + \boldsymbol{v}_1\boldsymbol{u}^T + \boldsymbol{u}\boldsymbol{v}_1^T + \boldsymbol{v}_1\boldsymbol{v}^\perp{}_2^T + \boldsymbol{v}^\perp{}_2\boldsymbol{v}_1^T + \boldsymbol{w}_3\boldsymbol{u}^T + \boldsymbol{u}\boldsymbol{w}_3^T \\ \delta\boldsymbol{S}_2 &= \boldsymbol{v}_2\boldsymbol{w}_3^T + \boldsymbol{w}_3\boldsymbol{v}_2^T + \boldsymbol{v}_2\boldsymbol{u}^T + \boldsymbol{u}\boldsymbol{v}_2^T + \boldsymbol{v}_2\boldsymbol{v}^\perp{}_1^T + \boldsymbol{v}^\perp{}_1\boldsymbol{v}_2^T \end{aligned} \tag{42}$$

i.e., there are some small perturbations in the sample covariance matrix. $\delta\boldsymbol{S}_1$ and $\delta\boldsymbol{S}_2$ model the small differences between the sample covariance and population covariance matrices. It is easy to calculate that,

$$\frac{1}{2}\left(\left\|\boldsymbol{S}_{(1)} - \boldsymbol{\Sigma}_{(1)}\right\|_F^2 + \left\|\boldsymbol{S}_{(2)} - \boldsymbol{\Sigma}_{(2)}\right\|_F^2\right)^2 = 7\varepsilon^2$$

We can run `PerPCA` on the sample covariance matrices $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$. The optimal global and local optimal PCs are denoted as $\hat{\boldsymbol{u}}$ and $(\hat{\boldsymbol{v}}_1, \hat{\boldsymbol{v}}_2)$. Apparently, $\hat{\boldsymbol{u}}$ and $(\hat{\boldsymbol{v}}_1, \hat{\boldsymbol{v}}_2)$ are a function of ε, and as ε becomes zero, the sample covariance becomes the population covariance, and $(\hat{\boldsymbol{u}}, \hat{\boldsymbol{v}}_1, \hat{\boldsymbol{v}}_2)$ become $(\boldsymbol{u}, \boldsymbol{v}_1, \boldsymbol{v}_2)$.

To estimate $(\hat{\boldsymbol{u}}, \hat{\boldsymbol{v}}_1, \hat{\boldsymbol{v}}_2)$ when ε is nonzero, we can use the KKT conditions to analyze how $(\hat{\boldsymbol{u}}, \hat{\boldsymbol{v}}_1, \hat{\boldsymbol{v}}_2)$ change with respect to ε. Remember that the KKT conditions (38) are,

$$\begin{aligned} \boldsymbol{S}_1\hat{\boldsymbol{v}}_1 &= \hat{\boldsymbol{v}}_1\lambda_{21} + \hat{\boldsymbol{v}}_1\hat{\boldsymbol{v}}_1^T\boldsymbol{S}_1\hat{\boldsymbol{u}} \\ \boldsymbol{S}_2\hat{\boldsymbol{v}}_2 &= \hat{\boldsymbol{v}}_2\lambda_{22} + \hat{\boldsymbol{v}}_2\hat{\boldsymbol{v}}_2^T\boldsymbol{S}_1\hat{\boldsymbol{u}} \\ (\boldsymbol{S}_1 + \boldsymbol{S}_2)\,\hat{\boldsymbol{u}} &= \hat{\boldsymbol{u}}\lambda_1 + \hat{\boldsymbol{u}}\hat{\boldsymbol{u}}^T\boldsymbol{S}_1\hat{\boldsymbol{v}}_1 + \hat{\boldsymbol{u}}\hat{\boldsymbol{u}}^T\boldsymbol{S}_2\hat{\boldsymbol{v}}_2 \end{aligned} \tag{43}$$

Since $\boldsymbol{\Sigma}_{(1)}$ and $\boldsymbol{\Sigma}_{(2)}$ do not have duplicate eigenvalues, from Greenbaum et al. (2020), $(\hat{\boldsymbol{u}}, \hat{\boldsymbol{v}}_1, \hat{\boldsymbol{v}}_2)$ and $(\lambda_1, \lambda_{21}, \lambda_{22})$ are analytic functions of ε when ε is small. We can thus write the Taylor series expansion of $(\hat{\boldsymbol{u}}, \hat{\boldsymbol{v}}_1, \hat{\boldsymbol{v}}_2)$ as,

$$\begin{aligned} \hat{\boldsymbol{u}}(\varepsilon) &= \boldsymbol{u} + \varepsilon\boldsymbol{u}^{(1)} + \varepsilon^2\boldsymbol{u}^{(2)} + \cdots \\ \hat{\boldsymbol{v}}_1(\varepsilon) &= \boldsymbol{v}_1 + \varepsilon\boldsymbol{v}_1^{(1)} + \varepsilon^2\boldsymbol{v}_1^{(2)} + \cdots \\ \hat{\boldsymbol{v}}_2(\varepsilon) &= \boldsymbol{v}_2 + \varepsilon\boldsymbol{v}_2^{(1)} + \varepsilon^2\boldsymbol{v}_2^{(2)} + \cdots \\ \lambda_1(\varepsilon) &= \lambda_1^{(0)} + \varepsilon\lambda_1^{(1)} + \varepsilon^2\lambda_1^{(2)} + \cdots \\ \lambda_{21}(\varepsilon) &= \lambda_{21}^{(0)} + \varepsilon\lambda_{21}^{(1)} + \varepsilon^2\lambda_{21}^{(2)} + \cdots \\ \lambda_{22}(\varepsilon) &= \lambda_{22}^{(0)} + \varepsilon\lambda_{22}^{(1)} + \varepsilon^2\lambda_{22}^{(2)} + \cdots \end{aligned} \tag{44}$$

where $\boldsymbol{u}^{(1)}$ is the first-order coefficient and $\boldsymbol{u}^{(2)}$ is the second-order coefficient for the expansion of $\hat{\boldsymbol{u}}(\varepsilon)$. Similar notations are used for other variables.

Then we can take the expansion (44) into the KKT conditions (43) and match the $O(\varepsilon)$ terms on both sides,

$$\begin{aligned}
\delta \boldsymbol{S}_1 \boldsymbol{v}_1 + \boldsymbol{\Sigma}_1 \boldsymbol{v}_1^{(1)} &= \boldsymbol{v}_1 \lambda_{21}^{(1)} + \boldsymbol{v}_1^{(1)} \lambda_{21}^{(0)} \\
&\quad + \boldsymbol{v}_1^{(1)} \boldsymbol{v}_1^T \boldsymbol{\Sigma}_1 \boldsymbol{u} + \boldsymbol{v}_1 \boldsymbol{v}_1^{(1)T} \boldsymbol{\Sigma}_1 \boldsymbol{u} + \boldsymbol{v}_1 \boldsymbol{v}_1^T \delta \boldsymbol{S}_1 \boldsymbol{u} + \boldsymbol{v}_1 \boldsymbol{v}_1^T \boldsymbol{\Sigma}_1 \boldsymbol{u}^{(1)} \\
\delta \boldsymbol{S}_2 \boldsymbol{v}_2 + \boldsymbol{\Sigma}_2 \boldsymbol{v}_2^{(1)} &= \boldsymbol{v}_2 \lambda_{22}^{(1)} + \boldsymbol{v}_2^{(1)} \lambda_{22}^{(0)} \\
&\quad + \boldsymbol{v}_2^{(1)} \boldsymbol{v}_2^T \boldsymbol{\Sigma}_2 \boldsymbol{u} + \boldsymbol{v}_2 \boldsymbol{v}_2^{(1)T} \boldsymbol{\Sigma}_2 \boldsymbol{u} + \boldsymbol{v}_2 \boldsymbol{v}_2^T \delta \boldsymbol{S}_2 \boldsymbol{u} + \boldsymbol{v}_2 \boldsymbol{v}_2^T \boldsymbol{\Sigma}_2 \boldsymbol{u}^{(1)} \\
\left(\delta \boldsymbol{S}_1 + \delta \boldsymbol{S}_2\right) \boldsymbol{u} + \left(\boldsymbol{\Sigma}_1 + \boldsymbol{\Sigma}_2\right) \boldsymbol{u}^{(1)} &= \boldsymbol{u} \lambda_1^{(1)} + \boldsymbol{u}^{(1)} \lambda_1^{(0)} \\
&\quad + \boldsymbol{u}^{(1)} \boldsymbol{u}^T \boldsymbol{\Sigma}_1 \boldsymbol{v}_1 + \boldsymbol{u} \boldsymbol{u}^{(1)T} \boldsymbol{\Sigma}_1 \boldsymbol{v}_1 + \boldsymbol{u} \boldsymbol{u}^T \delta \boldsymbol{S}_1 \boldsymbol{v}_1 + \boldsymbol{u} \boldsymbol{u}^T \boldsymbol{\Sigma}_1 \boldsymbol{v}_1^{(1)} \\
&\quad + \boldsymbol{u}^{(1)} \boldsymbol{u}^T \boldsymbol{\Sigma}_2 \boldsymbol{v}_2 + \boldsymbol{u} \boldsymbol{u}^{(1)T} \boldsymbol{\Sigma}_2 \boldsymbol{v}_2 + \boldsymbol{u} \boldsymbol{u}^T \delta \boldsymbol{S}_2 \boldsymbol{v}_2 + \boldsymbol{u} \boldsymbol{u}^T \boldsymbol{\Sigma}_2 \boldsymbol{v}_2^{(1)}
\end{aligned} \tag{45}$$

To solve equation (45), we can expand $\boldsymbol{u}^{(1)}$, $\boldsymbol{v}_1^{(1)}$, and $\boldsymbol{v}_2^{(1)}$ over a basis,

$$\begin{aligned}
\boldsymbol{u}^{(1)} &= \varphi_{00} \boldsymbol{u} + \varphi_{01} \boldsymbol{v}_1 + \varphi_{02} \boldsymbol{v}^{\perp}{}_2 + \varphi_{03} \boldsymbol{w}_3 \\
\boldsymbol{v}_1^{(1)} &= \varphi_{10} \boldsymbol{u} + \varphi_{11} \boldsymbol{v}_1 + \varphi_{12} \boldsymbol{v}^{\perp}{}_2 + \varphi_{13} \boldsymbol{w}_3 \\
\boldsymbol{v}_2^{(1)} &= \varphi_{20} \boldsymbol{u} + \varphi_{21} \boldsymbol{v}^{\perp}{}_1 + \varphi_{22} \boldsymbol{v}_2 + \varphi_{23} \boldsymbol{w}_3
\end{aligned} \tag{46}$$

Since $\|\hat{\boldsymbol{u}}\| = \|\hat{\boldsymbol{v}}_1\| = \|\hat{\boldsymbol{v}}_2\| = 1$, we know that $\varphi_{00} = \varphi_{11} = \varphi_{22} = 0$. Then we can take (46) into (45), and solve φ's as,

$$\begin{aligned}
\varphi_{01} &= -\frac{1}{4} \sin(2\alpha) \cot(\gamma) \\
\varphi_{02} &= -\frac{1}{2} \sin(\alpha) \cos(\alpha) \\
\varphi_{03} &= -\frac{2 \sin(2\alpha) + \cos(2\alpha) + 2\varrho - 3}{4\left(\varrho^2 - 3\varrho + 2\right)} \\
\varphi_{10} &= \frac{1}{4} \sin(2\alpha) \cot(\gamma) \\
\varphi_{12} &= \frac{1}{4} (\cos(2\alpha) + 3) \\
\varphi_{13} &= -\frac{\sin(2\alpha) - 2\cos(2\alpha) + 4\varrho - 6}{4\left(\varrho^2 - 3\varrho + 2\right)} \\
\varphi_{20} &= \frac{1}{4} \sin(2\alpha) \cot(\gamma) \\
\varphi_{21} &= \frac{1}{4} (\cos(2\alpha) + 3) \\
\varphi_{23} &= -\frac{\sin(2\alpha) - 2\cos(2\alpha) + 4\varrho - 6}{4\left(\varrho^2 - 3\varrho + 2\right)}
\end{aligned} \tag{47}$$

Now, we obtained the closed-form formula for the first-order perturbation of global and local PCs. It is straightforward to calculate that

$$\begin{aligned}
&\left\|\hat{\boldsymbol{u}}\hat{\boldsymbol{u}}^T - \boldsymbol{u}\boldsymbol{u}^T\right\|_F^2 \\
&\left\|\left(\boldsymbol{u} + \varepsilon\boldsymbol{u}^{(1)} + \varepsilon^2\boldsymbol{u}^{(2)} + \cdots\right)\left(\boldsymbol{u} + \varepsilon\boldsymbol{u}^{(1)} + \varepsilon^2\boldsymbol{u}^{(2)} + \cdots\right)^T - \boldsymbol{u}\boldsymbol{u}^T\right\|_F^2 \\
&= \varepsilon^2 2\left\|\boldsymbol{u}^{(1)}\right\|^2 + O\left(\varepsilon^3\right) \\
&= \frac{\varepsilon^2}{64}\left(3\csc^2(\gamma) + \frac{\left(4\varrho + 2\sqrt{3} - 5\right)^2}{(1-\varrho)^2(2-\varrho)^2}\right) + O(\varepsilon^3)
\end{aligned} \tag{48}$$

Since we know that $\theta = \sin^2(\gamma)$ and $\delta = 1 - \varrho$ when $\gamma \leq \frac{\pi}{4}$, we have,

$$\left\|\hat{\boldsymbol{u}}\hat{\boldsymbol{u}}^T - \boldsymbol{u}\boldsymbol{u}^T\right\|_F^2 = \frac{\varepsilon^2}{64}\left(3\frac{1}{\theta} + \frac{\left(2\sqrt{3} - 1 - 4\delta\right)^2}{\delta^2(\delta+1)^2}\right) + O(\varepsilon^3) \tag{49}$$

When ε is small, the higher order term $O\left(\varepsilon^3\right)$ can be neglected. Thus the error in (49) can be further simplified to $\Omega\left(\varepsilon^2\left(\frac{1}{\theta} + \frac{1}{\delta^2}\right)\right)$ when θ and δ are small. This completes our proof. ■

We also verify the predicted error (49) via numerical simulations. In the simulations, we run `PerPCA` on the sample covariance matrices $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ to obtain the global PC $\hat{\boldsymbol{u}}$. Then we use $\hat{\boldsymbol{u}}$ to calculate the subspace error $\left\|\hat{\boldsymbol{u}}\hat{\boldsymbol{u}}^T - \boldsymbol{u}\boldsymbol{u}^T\right\|$. This is the actual statistical error for the estimates from `PerPCA`. We compare it with the predicted values in (49) under different parameter values of θ and δ. Results are shown in Figure 10 and 11.

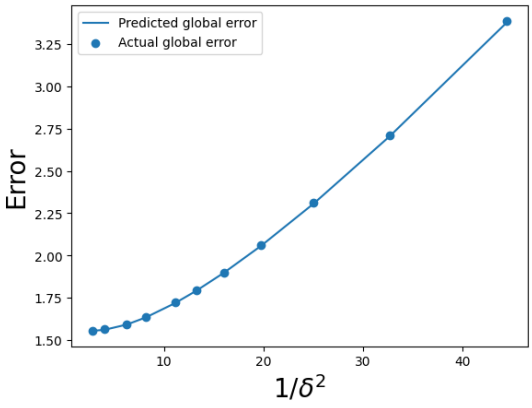


Figure 10: The (rescaled) global PC error $\left\|\hat{\boldsymbol{u}}\hat{\boldsymbol{u}}^T - \boldsymbol{u}\boldsymbol{u}^T\right\|_F^2/\varepsilon^2$ under different eigengap δ.

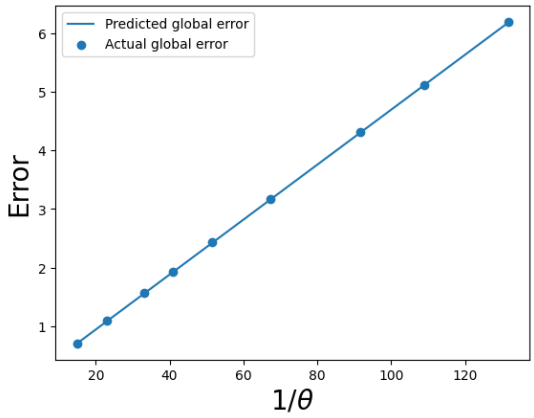


Figure 11: The (rescaled) global PC error $\left\|\hat{\boldsymbol{u}}\hat{\boldsymbol{u}}^T - \boldsymbol{u}\boldsymbol{u}^T\right\|_F^2/\varepsilon^2$ under different misalignment parameter θ.

Figure 10 and 11 demonstrate good matches between the predicted statistical error and the actual statistical error. The two curves vividly show that when $\frac{1}{\theta}$ and $\frac{1}{\delta^2}$ is large, the global subspace error grows linearly with $\frac{1}{\theta}$ and $\frac{1}{\delta^2}$.

Appendix D. Proof of Theorem 8

Now we analyze the global convergence of Algorithm 2. We begin by calculating the derivative of $\mathcal{L}_{(i),1}$ and $\mathcal{L}_{(i),2}$. The derivative of $\mathcal{L}_{(i),1}$ over $\boldsymbol{U}$ is:

$$\nabla_{\boldsymbol{U}}\mathcal{L}_{(i),1}(\boldsymbol{U},\boldsymbol{V}) = -\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{V}}\right)\boldsymbol{S}_{(i)}\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{V}}\right)\boldsymbol{U} \tag{50}$$

When $\boldsymbol{V}^T\boldsymbol{U}=0$, this reduces to:

$$\nabla_{\boldsymbol{U}}\mathcal{L}_{(i),1}(\boldsymbol{U},\boldsymbol{V}) = -\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{V}}\right)\boldsymbol{S}_{(i)}\boldsymbol{U}$$

And the derivative of $\mathcal{L}_{(i),1}$ over $\boldsymbol{V}$ is:

$$\nabla_{\boldsymbol{V}}\mathcal{L}_{(i),1}(\boldsymbol{U},\boldsymbol{V}) = \boldsymbol{P}_{\boldsymbol{U}}\boldsymbol{S}_{(i)}\boldsymbol{V}+\boldsymbol{S}_{(i)}\boldsymbol{P}_{\boldsymbol{U}}\boldsymbol{V}-\boldsymbol{P}_{\boldsymbol{U}}\boldsymbol{P}_{\boldsymbol{V}}\boldsymbol{S}_{(i)}\boldsymbol{V}-\boldsymbol{S}_{(i)}\boldsymbol{P}_{\boldsymbol{V}}\boldsymbol{P}_{\boldsymbol{U}}\boldsymbol{V} \tag{51}$$

When $\boldsymbol{V}^T\boldsymbol{U}=0$, this reduces to:

$$\nabla_{\boldsymbol{V}}\mathcal{L}_{(i),1}(\boldsymbol{U},\boldsymbol{V}) = \boldsymbol{P}_{\boldsymbol{U}}\boldsymbol{S}_{(i)}\boldsymbol{V}$$

Similarly, the derivative of $\mathcal{L}_{(i),2}$ over $\boldsymbol{V}$ is:

$$\nabla_{\boldsymbol{V}}\mathcal{L}_{(i),2}(\boldsymbol{U},\boldsymbol{V}) = -\boldsymbol{S}_{(i)}\boldsymbol{V} \tag{52}$$

The following lemma shows that the function we introduced is Lipschitz continuous.

Lemma 14 *When $\|\boldsymbol{U}\|_{op}$ and $\|\boldsymbol{V}\|_{op}$ are upper bounded by 1, the functions $\mathcal{L}_{(i),1}+\mathcal{L}_{(i),2}$ are Lipschitz continuous with constant L. More formally, for any $\boldsymbol{U}_1,\boldsymbol{U}_2,\boldsymbol{V}_1,\boldsymbol{V}_2\in\mathbb{R}^{d\times r}$, such that $\|\boldsymbol{U}_1\|_{op},\|\boldsymbol{U}_2\|_{op},\|\boldsymbol{V}_1\|_{op},\|\boldsymbol{V}_2\|_{op}\le 1$, we have:*

$$\begin{aligned}&\Big\|\Big[\nabla_{\boldsymbol{U}}\mathcal{L}_{(i),1}(\boldsymbol{U}_2,\boldsymbol{V}_2)-\nabla_{\boldsymbol{U}}\mathcal{L}_{(i),1}(\boldsymbol{U}_1,\boldsymbol{V}_1),\\&\nabla_{\boldsymbol{V}}\mathcal{L}_{(i),1}(\boldsymbol{U}_2,\boldsymbol{V}_2)+\nabla_{\boldsymbol{V}}\mathcal{L}_{(i),2}(\boldsymbol{U}_2,\boldsymbol{V}_2)-\nabla_{\boldsymbol{V}}\mathcal{L}_{(i),1}(\boldsymbol{U}_1,\boldsymbol{V}_1)-\nabla_{\boldsymbol{V}}\mathcal{L}_{(i),2}(\boldsymbol{U}_1,\boldsymbol{V}_1)\Big]\Big\|_F\\&\le L\sqrt{\|\boldsymbol{U}_1-\boldsymbol{U}_2\|_F^2+\|\boldsymbol{V}_1-\boldsymbol{V}_2\|_F^2}\end{aligned} \tag{53}$$

where

$$L = 9\sqrt{2}G_{(i),op} \tag{54}$$

Proof First, we calculate the difference in the gradient of $\boldsymbol{U}$:

$$\begin{aligned}&\left\|\nabla_{\boldsymbol{U}}\mathcal{L}_{(i),1}(\boldsymbol{U}_2,\boldsymbol{V}_2)-\nabla_{\boldsymbol{U}}\mathcal{L}_{(i),1}(\boldsymbol{U}_1,\boldsymbol{V}_1)\right\|_F\\&=\left\|\left(\boldsymbol{I}-\boldsymbol{V}_1\boldsymbol{V}_1^T\right)\boldsymbol{S}_{(i)}\boldsymbol{U}_1-\left(\boldsymbol{I}-\boldsymbol{V}_2\boldsymbol{V}_2^T\right)\boldsymbol{S}_{(i)}\boldsymbol{U}_2\right\|_F\\&\le\left\|\left(\boldsymbol{I}-\boldsymbol{V}_1\boldsymbol{V}_1^T\right)\boldsymbol{S}_{(i)}\boldsymbol{U}_1-\left(\boldsymbol{I}-\boldsymbol{V}_2\boldsymbol{V}_2^T\right)\boldsymbol{S}_{(i)}\boldsymbol{U}_1\right\|_F+\left\|\left(\boldsymbol{I}-\boldsymbol{V}_2\boldsymbol{V}_2^T\right)\boldsymbol{S}_{(i)}\boldsymbol{U}_1-\left(\boldsymbol{I}-\boldsymbol{V}_2\boldsymbol{V}_2^T\right)\boldsymbol{S}_{(i)}\boldsymbol{U}_2\right\|_F\\&\le\left\|\boldsymbol{V}_1\boldsymbol{V}_1^T-\boldsymbol{V}_2\boldsymbol{V}_2^T\right\|_F G_{(i),op}+\|\boldsymbol{U}_1-\boldsymbol{U}_2\|_F G_{(i),op}\\&\le 2\|\boldsymbol{V}_1-\boldsymbol{V}_2\|_F G_{(i),op}+\|\boldsymbol{U}_1-\boldsymbol{U}_2\|_F G_{(i),op}\end{aligned}$$

where we used the triangle inequality for the Frobenius norm for the first inequality, and Lemma 23 for the second and third inequality. Next, we calculate the difference in the gradient of $\boldsymbol{V}$:

$$\begin{aligned}
&\left\|\nabla_{\boldsymbol{V}}\mathcal{L}_{(i),1}(\boldsymbol{U}_2,\boldsymbol{V}_2)+\nabla_{\boldsymbol{V}}\mathcal{L}_{(i),2}(\boldsymbol{U}_2,\boldsymbol{V}_2)-\nabla_{\boldsymbol{V}}\mathcal{L}_{(i),1}(\boldsymbol{U}_1,\boldsymbol{V}_1)-\nabla_{\boldsymbol{V}}\mathcal{L}_{(i),2}(\boldsymbol{U}_1,\boldsymbol{V}_1)\right\|_F\\
&\leq \left\|\left(\boldsymbol{P}_{\boldsymbol{U}_2}-\boldsymbol{I}\right)\boldsymbol{S}_{(i)}\boldsymbol{V}_2-\left(\boldsymbol{P}_{\boldsymbol{U}_1}-\boldsymbol{I}\right)\boldsymbol{S}_{(i)}\boldsymbol{V}_1\right\|_F+\left\|\boldsymbol{S}_{(i)}\boldsymbol{P}_{\boldsymbol{U}_2}\boldsymbol{V}_2-\boldsymbol{S}_{(i)}\boldsymbol{P}_{\boldsymbol{U}_2}\boldsymbol{V}_2\right\|\\
&+\left\|\boldsymbol{P}_{\boldsymbol{U}_2}\boldsymbol{P}_{\boldsymbol{V}_2}\boldsymbol{S}_{(i)}\boldsymbol{V}_2-\boldsymbol{P}_{\boldsymbol{U}_1}\boldsymbol{P}_{\boldsymbol{V}_1}\boldsymbol{S}_{(i)}\boldsymbol{V}_1\right\|_F+\left\|\boldsymbol{S}_{(i)}\boldsymbol{P}_{\boldsymbol{V}_2}\boldsymbol{P}_{\boldsymbol{U}_2}\boldsymbol{V}_2-\boldsymbol{S}_{(i)}\boldsymbol{P}_{\boldsymbol{V}_1}\boldsymbol{P}_{\boldsymbol{U}_1}\boldsymbol{V}_1\right\|_F\\
&\leq 7\left\|\boldsymbol{V}_1-\boldsymbol{V}_2\right\|_F G_{(i),op}+6\left\|\boldsymbol{U}_1-\boldsymbol{U}_2\right\|_F G_{(i),op}
\end{aligned}$$

Summing them up, we know:

$$\begin{aligned}
&\Big\|\Big[\nabla_{\boldsymbol{U}}\mathcal{L}_{(i),1}(\boldsymbol{U}_2,\boldsymbol{V}_2)-\nabla_{\boldsymbol{U}}\mathcal{L}_{(i),1}(\boldsymbol{U}_1,\boldsymbol{V}_1),\\
&\nabla_{\boldsymbol{V}}\mathcal{L}_{(i),1}(\boldsymbol{U}_2,\boldsymbol{V}_2)+\nabla_{\boldsymbol{V}}\mathcal{L}_{(i),2}(\boldsymbol{U}_2,\boldsymbol{V}_2)-\nabla_{\boldsymbol{V}}\mathcal{L}_{(i),1}(\boldsymbol{U}_1,\boldsymbol{V}_1)-\nabla_{\boldsymbol{V}}\mathcal{L}_{(i),2}(\boldsymbol{U}_1,\boldsymbol{V}_1)\Big]\Big\|_F\\
&\leq \left\|\nabla_{\boldsymbol{U}}\mathcal{L}_{(i),1}(\boldsymbol{U}_2,\boldsymbol{V}_2)-\nabla_{\boldsymbol{U}}\mathcal{L}_{(i),1}(\boldsymbol{U}_1,\boldsymbol{V}_1)\right\|_F\\
&+\left\|\nabla_{\boldsymbol{V}}\mathcal{L}_{(i),1}(\boldsymbol{U}_2,\boldsymbol{V}_2)+\nabla_{\boldsymbol{V}}\mathcal{L}_{(i),2}(\boldsymbol{U}_2,\boldsymbol{V}_2)-\nabla_{\boldsymbol{V}}\mathcal{L}_{(i),1}(\boldsymbol{U}_1,\boldsymbol{V}_1)-\nabla_{\boldsymbol{V}}\mathcal{L}_{(i),2}(\boldsymbol{U}_1,\boldsymbol{V}_1)\right\|_F\\
&\leq 9\left\|\boldsymbol{V}_1-\boldsymbol{V}_2\right\|_F G_{(i),op}+7\left\|\boldsymbol{U}_1-\boldsymbol{U}_2\right\|_F G_{(i),op}\\
&\leq 9\sqrt{2}G_{max,op}\sqrt{\left\|\boldsymbol{U}_1-\boldsymbol{U}_2\right\|_F^2+\left\|\boldsymbol{V}_1-\boldsymbol{V}_2\right\|_F^2}
\end{aligned} \tag{55}$$

We thus complete the proof. ■

Now we introduce some notations:

$$\square\boldsymbol{U}_\tau=\frac{1}{N}\sum_{i=1}^{N}\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}_\tau}-\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\boldsymbol{S}_{(i)}\boldsymbol{U}_\tau \tag{56}$$

It is easy to verify $\square\boldsymbol{U}_\tau\in\mathcal{T}_{\boldsymbol{U}_\tau}$ when $\boldsymbol{U}^T\boldsymbol{V}_{(i),\tau}=0$:

$$\boldsymbol{U}_\tau^T\square\boldsymbol{U}_\tau=0$$

The Frobenius norm of $\square\boldsymbol{U}_\tau$ is upper bounded by:

$$\begin{aligned}
&\left\|\square\boldsymbol{U}_\tau\right\|_F\\
&=\left\|\frac{1}{N}\sum_{i=1}^{N}\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}_\tau}-\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\boldsymbol{S}_{(i)}\boldsymbol{U}_\tau\right\|_F\\
&\leq\frac{1}{N}\sum_{i=1}^{N}\left\|\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}_\tau}-\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\right\|_{op}\left\|\boldsymbol{S}_{(i)}\right\|_{op}\left\|\boldsymbol{U}_\tau\right\|_F\\
&\leq\frac{1}{N}\sum_{i=1}^{N}G_{max,op}\sqrt{r}\\
&=G_{max,op}\sqrt{r}
\end{aligned} \tag{57}$$

where we applied Lemma 23 for the first inequality.

By the client update rule, we know that:

$$\boldsymbol{U}_{(i),\tau+1} = \boldsymbol{U}_\tau + \eta_\tau \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \boldsymbol{S}_{(i)} \boldsymbol{U}_\tau \tag{58}$$

Therefore, after the server takes the average of $\boldsymbol{U}_{(i),\tau+1}$ and performs a generalized retraction, the following holds:

$$\boldsymbol{U}_{\tau+1} = \boldsymbol{U}_\tau + \eta_\tau \square \boldsymbol{U}_\tau + \eta_\tau^2 \boldsymbol{e}_{1,\tau} \tag{59}$$

where $\boldsymbol{e}_{1,\tau}$ is an error term defined as:

$$\boldsymbol{e}_{1,\tau} = \frac{1}{\eta_\tau^2} \left(\boldsymbol{U}_{\tau+1} - \boldsymbol{U}_\tau - \eta_\tau \square \boldsymbol{U}_\tau\right)$$

By definition of a generalized retraction, since $\square \boldsymbol{U}_\tau$ is in the tangent space of $\boldsymbol{U}_\tau$, we have:

$$\left\|\boldsymbol{e}_{1,\tau}\right\|_F \leq M_1 \left\|\square \boldsymbol{U}_\tau\right\|_F^2$$

where we applied the condition $\eta_\tau \leq \frac{M_3}{\sqrt{r} G_{max,op}}$ thus $\left\|\eta_\tau \square \boldsymbol{U}_\tau\right\|_F \leq M_3$. Remember that M_3 is a numerical constant in the definition of generalized retraction (Definition 5).

Similarly, we define:

$$\square \boldsymbol{V}_{(i),\tau} = \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \boldsymbol{S}_{(i)} \boldsymbol{V}_{(i),\tau} \tag{60}$$

Also by Lemma 23, the Frobenius norm of $\square \boldsymbol{V}_{(i),\tau}$ is upper bounded by:

$$\begin{aligned}
&\left\|\square \boldsymbol{V}_{(i),\tau}\right\|_F \\
&= \left\|\left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \boldsymbol{S}_{(i)} \boldsymbol{V}_{(i),\tau}\right\|_F \\
&\leq \left\|\left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\right\|_{op} \left\|\boldsymbol{S}_{(i)}\right\|_{op} \left\|\boldsymbol{V}_{(i),\tau}\right\|_F \\
&\leq G_{(i),op} \sqrt{r}
\end{aligned} \tag{61}$$

Now we calculate the update of $\boldsymbol{V}_{(i),\tau}$ in one communication round. We summarize the result in the following lemma.

Lemma 15 *If we choose the stepsize* $\eta_\tau \leq \min\left\{\frac{M_3}{2}, \frac{\sqrt{M_3}}{\sqrt{6+12M_1+M_1^2}}\right\} \frac{1}{G_{max,op}\sqrt{r}}$, *the update of* $\boldsymbol{V}_{(i)}$ *is given by:*

$$\boldsymbol{V}_{(i),\tau+1} = \boldsymbol{V}_{(i),\tau} + \eta_\tau \square \boldsymbol{V}_{(i),\tau} - \eta_\tau \boldsymbol{U}_\tau \square \boldsymbol{U}_\tau^T \boldsymbol{V}_{(i),\tau} + \eta_\tau^2 \boldsymbol{e}_{5,(i),\tau}$$

where $\boldsymbol{e}_{5,(i),\tau}$ *is an error term that satisfies:*

$$\left\|\boldsymbol{e}_{5,(i),\tau}\right\| \leq C_{5,0} \left\|\square \boldsymbol{U}_\tau\right\|^2 + C_{5,1} \left\|\square \boldsymbol{V}_{(i),\tau}\right\|^2$$

where $C_{5,0}$ *and* $C_{5,1}$ *are two constants that only depend on* M_1 *and* M_2 *from the generalized retraction Definition 5.*

Proof We first calculate the projection:

$$\begin{aligned}
&\boldsymbol{U}_{\tau+1}\boldsymbol{U}_{\tau+1}^T \\
&= \left(\boldsymbol{U}_\tau + \eta_\tau \square \boldsymbol{U}_\tau + \eta_\tau^2 \boldsymbol{e}_{1,\tau}\right)\left(\boldsymbol{U}_\tau + \eta_\tau \square \boldsymbol{U}_\tau + \eta_\tau^2 \boldsymbol{e}_{1,\tau}\right)^T \\
&= \boldsymbol{U}_\tau \boldsymbol{U}_\tau^T + \eta_\tau \left(\boldsymbol{U}_\tau \square \boldsymbol{U}_\tau^T + \square \boldsymbol{U}_\tau \boldsymbol{U}_\tau^T\right) + \eta_\tau^2 \boldsymbol{e}_{2,(i),\tau}
\end{aligned}$$

where $\boldsymbol{e}_{2,(i),\tau}$ is defined as:

$$\boldsymbol{e}_{2,(i),\tau} = \square \boldsymbol{U}_\tau \square \boldsymbol{U}_\tau^T + \boldsymbol{U}_\tau \boldsymbol{e}_{1,(i),\tau}^T + \boldsymbol{e}_{1,(i),\tau} \boldsymbol{U}_\tau^T + \eta_\tau \square \boldsymbol{U}_\tau \boldsymbol{e}_{1,(i),\tau}^T + \eta_\tau \boldsymbol{e}_{1,(i),\tau} \square \boldsymbol{U}_\tau^T + \eta_\tau^2 \boldsymbol{e}_{1,(i),\tau} \boldsymbol{e}_{1,(i),\tau}^T$$

Its norm is upper bounded by:

$$\begin{aligned}
&\left\|\boldsymbol{e}_{2,(i),\tau}\right\|_F \\
&\leq \left\|\square \boldsymbol{U}_\tau\right\|_F^2 + 2\left\|\boldsymbol{U}_\tau\right\|_{op}\left\|\boldsymbol{e}_{1,(i),\tau}\right\|_F + 2\eta_\tau \left\|\square \boldsymbol{U}_\tau\right\|_F \left\|\boldsymbol{e}_{1,(i),\tau}\right\|_F + \eta_\tau^2 \left\|\boldsymbol{e}_{1,(i),\tau}\right\|_F^2 \\
&\leq \left\|\square \boldsymbol{U}_\tau\right\|_F^2 + 2M_1 \left\|\square \boldsymbol{U}_\tau\right\|_F^2 + 2\eta_\tau M_1 \left\|\square \boldsymbol{U}_\tau\right\|_F^3 + \eta_\tau^2 M_1^2 \left\|\square \boldsymbol{U}_\tau\right\|_F^4 \\
&\leq (1 + 3M_1 + \frac{1}{4}M_1^2) \left\|\square \boldsymbol{U}_\tau\right\|_F^2
\end{aligned}$$

where the final inequality comes from upper bound (57) and the choice of stepsize η_τ: $\eta_\tau \leq \frac{1}{G_{max,op}\sqrt{r}} \frac{1}{\sqrt{6+12M_1+M_1^2}} \leq \frac{1}{2G_{max,op}\sqrt{r}}$.

Similarly, we define $\boldsymbol{e}_{3,(i),\tau}$ as:

$$\boldsymbol{e}_{3,(i),\tau} = \frac{1}{\eta_\tau^2}\left(\boldsymbol{V}_{(i),\tau+\frac{1}{2}} - \boldsymbol{V}_{(i),\tau} - \eta_\tau \square \boldsymbol{V}_{(i),\tau}\right)$$

By definition of a retraction, the norm of $\boldsymbol{e}_{3,(i),\tau}$ is upper bounded by:

$$\left\|\boldsymbol{e}_{3,(i),\tau}\right\|_F \leq M_1 \left\|\square \boldsymbol{V}_{(i),\tau}\right\|_F^2$$

Then

$$\begin{aligned}
&\boldsymbol{U}_{\tau+1}\boldsymbol{U}_{\tau+1}^T \boldsymbol{V}_{(i),\tau+\frac{1}{2}} \\
&= \boldsymbol{U}_\tau \boldsymbol{U}_\tau^T \boldsymbol{V}_{(i),\tau+\frac{1}{2}} + \eta_\tau \left(\boldsymbol{U}_\tau \square \boldsymbol{U}_\tau^T + \square \boldsymbol{U}_\tau \boldsymbol{U}_\tau^T\right) \boldsymbol{V}_{(i),\tau+\frac{1}{2}} + \eta_\tau^2 \boldsymbol{e}_{2,(i),\tau} \boldsymbol{V}_{(i),\tau+\frac{1}{2}} \\
&= \boldsymbol{U}_\tau \boldsymbol{U}_\tau^T \left(\boldsymbol{V}_{(i),\tau} + \eta_\tau \square \boldsymbol{V}_{(i),\tau} + \eta_\tau^2 \boldsymbol{e}_{3,(i),\tau}\right) \\
&+ \eta_\tau \left(\boldsymbol{U}_\tau \square \boldsymbol{U}_\tau^T + \square \boldsymbol{U}_\tau \boldsymbol{U}_\tau^T\right)\left(\boldsymbol{V}_{(i),\tau} + \eta_\tau \square \boldsymbol{V}_{(i),\tau} + \eta_\tau^2 \boldsymbol{e}_{3,(i),\tau}\right) + \eta_\tau^2 \boldsymbol{e}_{2,(i),\tau} \boldsymbol{V}_{(i),\tau+\frac{1}{2}} \\
&= \eta_\tau \boldsymbol{U}_\tau \square \boldsymbol{U}_\tau^T \boldsymbol{V}_{(i),\tau} + \eta_\tau^2 \boldsymbol{U}_\tau \boldsymbol{U}_\tau^T \boldsymbol{e}_{3,(i),\tau} + \eta_\tau^2 \boldsymbol{e}_{2,(i),\tau} \boldsymbol{V}_{(i),\tau+\frac{1}{2}} + \eta_\tau^2 \boldsymbol{U}_\tau \square \boldsymbol{U}_\tau^T \square \boldsymbol{V}_{(i),\tau} \\
&+ \eta_\tau^3 \left(\boldsymbol{U}_\tau \square \boldsymbol{U}_\tau^T + \square \boldsymbol{U}_\tau \boldsymbol{U}_\tau^T\right) \boldsymbol{e}_{3,(i),\tau} \\
&= \eta_\tau \boldsymbol{U}_\tau \square \boldsymbol{U}_\tau^T \boldsymbol{V}_{(i),\tau} + \eta_\tau^2 \boldsymbol{e}_{4,(i),\tau}
\end{aligned}$$

where we use $\boldsymbol{e}_{4,(i),\tau}$ to denote:

$$\boldsymbol{e}_{4,(i),\tau} = \boldsymbol{U}_\tau \boldsymbol{U}_\tau^T \boldsymbol{e}_{3,(i),\tau} + \boldsymbol{e}_{2,(i),\tau} \boldsymbol{V}_{(i),\tau+\frac{1}{2}} + \boldsymbol{U}_\tau \square \boldsymbol{U}_\tau^T \square \boldsymbol{V}_{(i),\tau} + \eta_\tau \left(\boldsymbol{U}_\tau \square \boldsymbol{U}_\tau^T + \square \boldsymbol{U}_\tau \boldsymbol{U}_\tau^T\right) \boldsymbol{e}_{3,(i),\tau}$$

Its norm is upper bounded as:

$$\begin{aligned}
&\left\|\boldsymbol{e}_{4,(i),\tau}\right\|_F \\
&\leq \left\|\boldsymbol{e}_{3,(i),\tau}\right\|_F + \left\|\boldsymbol{e}_{2,(i),\tau}\right\|_F + \left\|\square \boldsymbol{U}_\tau\right\|_F \left\|\square \boldsymbol{V}_{(i),\tau}\right\|_F + \eta_\tau \left(\left\|\square \boldsymbol{U}_\tau\right\|_F + \left\|\square \boldsymbol{V}_{(i),\tau}\right\|_F\right) \left\|\boldsymbol{e}_{3,(i),\tau}\right\|_F \\
&\leq C_{4,0} \left\|\square \boldsymbol{U}_\tau\right\|_F^2 + C_{4,1} \left\|\square \boldsymbol{V}_{(i),\tau}\right\|_F^2
\end{aligned}$$

where

$$C_{4,0} = \frac{3}{2} + 3M_1 + \frac{1}{4}M_1^2$$

and

$$C_{4,1} = \frac{1}{2} + 2M_1$$

Thus we know when $\eta_\tau \leq \min\{\frac{M_3}{2}, \sqrt{\frac{M_3}{4C_{4,0}}}\} \frac{1}{G_{max,op}\sqrt{r}}$, $\left\|\boldsymbol{U}_{\tau+1}\boldsymbol{U}_{\tau+1}^T\boldsymbol{V}_{(i),\tau+\frac{1}{2}}\right\|_F \leq M_3$

Next we calculate the projection $\mathcal{P}_{\mathcal{N}_{\boldsymbol{V}_{(i),\tau+\frac{1}{2}}}}\left(-\boldsymbol{U}_{\tau+1}\boldsymbol{U}_{\tau+1}^T\boldsymbol{V}_{(i),\tau+\frac{1}{2}}\right)$:

$$\begin{aligned}
&\mathcal{P}_{\mathcal{N}_{\boldsymbol{V}_{(i),\tau+\frac{1}{2}}}}\left(-\boldsymbol{U}_{\tau+1}\boldsymbol{U}_{\tau+1}^T\boldsymbol{V}_{(i),\tau+\frac{1}{2}}\right) = -\boldsymbol{V}_{(i),\tau+\frac{1}{2}}\left(\boldsymbol{V}_{(i),\tau+\frac{1}{2}}^T\boldsymbol{U}_{\tau+1}\boldsymbol{U}_{\tau+1}^T\boldsymbol{V}_{(i),\tau+\frac{1}{2}}\right) \\
&= \eta_\tau^2 \boldsymbol{V}_{(i),\tau+\frac{1}{2}}\boldsymbol{V}_{(i),\tau+\frac{1}{2}}^T\boldsymbol{e}_{4,(i),\tau}
\end{aligned}$$

We use $\boldsymbol{e}_{5,(i),\tau}$ to denote the difference between $\mathcal{GR}_{\boldsymbol{V}_{(i),\tau+\frac{1}{2}}}\left(-\boldsymbol{U}_{\tau+1}\boldsymbol{U}_{\tau+1}^T\boldsymbol{V}_{(i),\tau+\frac{1}{2}}\right)$ and $\boldsymbol{V}_{(i),\tau} + \eta_\tau\square\boldsymbol{V}_{(i),\tau} - \eta_\tau\boldsymbol{U}_\tau\square\boldsymbol{U}_\tau^T\boldsymbol{V}_{(i),\tau}$, then its norm is upper bounded by:

$$\begin{aligned}
&\eta_\tau^2\left\|\boldsymbol{e}_{5,(i),\tau}\right\|_F \\
&= \left\|\mathcal{GR}_{\boldsymbol{V}_{(i),\tau+\frac{1}{2}}}\left(-\boldsymbol{U}_{\tau+1}\boldsymbol{U}_{\tau+1}^T\boldsymbol{V}_{(i),\tau+\frac{1}{2}}\right) - \boldsymbol{V}_{(i),\tau} - \eta_\tau\square\boldsymbol{V}_{(i),\tau} + \eta_\tau\boldsymbol{U}_\tau\square\boldsymbol{U}_\tau^T\boldsymbol{V}_{(i),\tau}\right\|_F \\
&\leq \left\|\mathcal{GR}_{\boldsymbol{V}_{(i),\tau+\frac{1}{2}}}\left(-\boldsymbol{U}_{\tau+1}\boldsymbol{U}_{\tau+1}^T\boldsymbol{V}_{(i),\tau+\frac{1}{2}}\right) - \boldsymbol{V}_{(i),\tau+\frac{1}{2}} + \boldsymbol{U}_{\tau+1}\boldsymbol{U}_{\tau+1}^T\boldsymbol{V}_{(i),\tau+\frac{1}{2}}\right\|_F \\
&+ \left\|\boldsymbol{V}_{(i),\tau+\frac{1}{2}} - \boldsymbol{U}_{\tau+1}\boldsymbol{U}_{\tau+1}^T\boldsymbol{V}_{(i),\tau+\frac{1}{2}} - \boldsymbol{V}_{(i),\tau} - \eta_\tau\square\boldsymbol{V}_{(i),\tau} + \eta_\tau\boldsymbol{U}_\tau\square\boldsymbol{U}_\tau^T\boldsymbol{V}_{(i),\tau}\right\|_F
\end{aligned}$$

By property 2 of the generalized retraction in Definition 5, we have:

$$\begin{aligned}
&\left\|\mathcal{GR}_{\boldsymbol{V}_{(i),\tau+\frac{1}{2}}}\left(-\boldsymbol{U}_{\tau+1}\boldsymbol{U}_{\tau+1}^T\boldsymbol{V}_{(i),\tau+\frac{1}{2}}\right) - \left(\boldsymbol{V}_{(i),\tau+\frac{1}{2}} - \boldsymbol{U}_{\tau+1}\boldsymbol{U}_{\tau+1}^T\boldsymbol{V}_{(i),\tau+\frac{1}{2}}\right)\right\|_F \\
&\leq M_1\left\|\mathcal{P}_{\mathcal{T}_{\boldsymbol{V}_{(i),\tau+\frac{1}{2}}}}\left(-\boldsymbol{U}_{\tau+1}\boldsymbol{U}_{\tau+1}^T\boldsymbol{V}_{(i),\tau+\frac{1}{2}}\right)\right\|_F^2 + (M_2+1)\left\|\mathcal{P}_{\mathcal{N}_{\boldsymbol{V}_{(i),\tau+\frac{1}{2}}}}\left(-\boldsymbol{U}_{\tau+1}\boldsymbol{U}_{\tau+1}^T\boldsymbol{V}_{(i),\tau+\frac{1}{2}}\right)\right\|_F \\
&\leq M_1\left\|-\boldsymbol{U}_{\tau+1}\boldsymbol{U}_{\tau+1}^T\boldsymbol{V}_{(i),\tau+\frac{1}{2}}\right\|_F^2 + (M_2+1)\eta_\tau^2\left\|\boldsymbol{V}_{(i),\tau+\frac{1}{2}}\boldsymbol{V}_{(i),\tau+\frac{1}{2}}^T\boldsymbol{e}_{4,(i),\tau}\right\|_F \\
&= M_1\left\|\eta_\tau\boldsymbol{U}_\tau\square\boldsymbol{U}_\tau^T\boldsymbol{V}_{(i),\tau} + \eta_\tau^2\boldsymbol{e}_{4,(i),\tau}\right\|_F^2 + (M_2+1)\eta_\tau^2\left\|\boldsymbol{V}_{(i),\tau+\frac{1}{2}}\boldsymbol{V}_{(i),\tau+\frac{1}{2}}^T\boldsymbol{e}_{4,(i),\tau}\right\|_F \\
&\leq 2M_1\eta_\tau^2\left(\left\|\boldsymbol{U}_\tau\square\boldsymbol{U}_\tau^T\boldsymbol{V}_{(i),\tau}\right\|_F^2 + \eta_\tau^4\left\|\boldsymbol{e}_{4,(i),\tau}\right\|_F^2\right) + (M_2+1)\eta_\tau^2\left\|\boldsymbol{V}_{(i),\tau+\frac{1}{2}}\boldsymbol{V}_{(i),\tau+\frac{1}{2}}^T\boldsymbol{e}_{4,(i),\tau}\right\|_F \\
&\leq \eta_\tau^2\left\|\square\boldsymbol{U}_\tau\right\|_F^2(2M_1(C_{4,0}+C_{4,1})C_{4,0} + 2M_1 + M_2C_{4,0}) \\
&+ \eta_\tau^2\left\|\square\boldsymbol{V}_{(i),\tau}\right\|_F^2(2M_1(C_{4,0}+C_{4,1})C_{4,1} + M_2C_{4,1})
\end{aligned}$$

For the second part:

$$\begin{aligned}
&\left\|\boldsymbol{V}_{(i),\tau+\frac{1}{2}} - \boldsymbol{U}_{\tau+1}\boldsymbol{U}_{\tau+1}^T\boldsymbol{V}_{(i),\tau+\frac{1}{2}} - \boldsymbol{V}_{(i),\tau} - \eta_\tau \square \boldsymbol{V}_{(i),\tau} + \eta_\tau \boldsymbol{U}_\tau \square \boldsymbol{U}_\tau^T \boldsymbol{V}_{(i),\tau}\right\|_F \\
&= \eta_\tau^2 \left\|e_{3,(n),\tau} - e_{4,(n),\tau}\right\|_F \\
&\leq \eta_\tau^2 \left\|e_{3,(n),\tau}\right\|_F + \eta_\tau^2 \left\|e_{4,(n),\tau}\right\|_F
\end{aligned}$$

Therefore, the norm of $\boldsymbol{e}_{5,(i),\tau}$ is upper bounded as:

$$\begin{aligned}
&\left\|\boldsymbol{e}_{5,(i),\tau}\right\|_F \\
&\leq \|\square \boldsymbol{U}_\tau\|_F^2 \left(2M_1(C_{4,0}+C_{4,1})C_{4,0} + 2M_1 + M_2 C_{4,0} + C_{4,0}\right) \\
&+ \left\|\square \boldsymbol{V}_{(i),\tau}\right\|_F^2 \left(2M_1(C_{4,0}+C_{4,1})C_{4,1} + M_2 C_{4,1} + M_1 + C_{4,1}\right)
\end{aligned}$$

This completes our proof, with

$$C_{5,0} = \frac{1}{8}\left(12\left(M_2+1\right) + M_1\left(24M_2 + M_1\left(M_1\left(M_1+32\right)+254\right) + 2\left(M_2+109\right)\right) + 88\right))$$

and

$$C_{5,1} = M_1^4 + \frac{81M_1^3}{4} + 13M_1^2 + \left(2M_2+5\right)M_1 + \frac{1}{2}\left(M_2+1\right)$$

■

The following lemma shows the sufficient decrease property:

Lemma 16 *(Formal version of Lemma 9) When we choose the stepsize* $\eta_\tau \leq \frac{1}{G_{max,op}\sqrt{r}} \min\left\{\frac{M_3}{2}, \frac{\sqrt{M_3}}{\sqrt{6+12M_1+M_1^2}}\right\}$*, and* $\boldsymbol{U}_\tau$ *and* $\boldsymbol{V}_{(i),\tau}$ *satisfy the orthogonality condition* $\boldsymbol{U}_\tau^T\boldsymbol{V}_{(i),\tau} = 0$*, we have:*

$$\begin{aligned}
&\left\langle \sum_{i=1}^N \nabla_{\boldsymbol{U}} \mathcal{L}_{(i),1}(\boldsymbol{U}_\tau, \boldsymbol{V}_{(i),\tau}), \boldsymbol{U}_{\tau+1} - \boldsymbol{U}_\tau \right\rangle \\
&+ \sum_{i=1}^N \left\langle \nabla_{\boldsymbol{V}_{(i)}} \mathcal{L}_{(i),1}(\boldsymbol{U}_\tau, \boldsymbol{V}_{(i),\tau}) + \nabla_{\boldsymbol{V}_{(i)}} \mathcal{L}_{(i),2}(\boldsymbol{U}_\tau, \boldsymbol{V}_{(i),\tau}), \boldsymbol{V}_{(i),\tau+1} - \boldsymbol{V}_{(i),\tau} \right\rangle \qquad (62) \\
&\leq -\eta_\tau N \|\square \boldsymbol{U}_\tau\|_F^2 - \eta_\tau \sum_{i=1}^N \left\|\square \boldsymbol{V}_{(i),\tau}\right\|_F^2 + \eta_\tau^2 \left(C_{6,0} N \|\square \boldsymbol{U}_\tau\|_F^2 + C_{6,1} \sum_{i=1}^N \left\|\square \boldsymbol{V}_{(i),\tau}\right\|_F^2\right)
\end{aligned}$$

where $C_{6,0}$ *and* $C_{6,1}$ *are constants dependent only on* M_1*,* M_2*,* r*, and* $G_{max,op}$*:*

$$C_{6,0} = G_{max,op}\sqrt{r}(M_1 + C_{5,0})$$

and

$$C_{6,1} = G_{max,op}\sqrt{r} C_{5,1}$$

Proof We firstly calculate the sufficient decrease of $\boldsymbol{U}$:

$$
\begin{aligned}
&\left\langle \sum_{i=1}^{N} \nabla_{\boldsymbol{U}} \mathcal{L}_{(i),1}(\boldsymbol{U}_\tau, \boldsymbol{V}_{(i),\tau}), \boldsymbol{U}_{\tau+1} - \boldsymbol{U}_\tau \right\rangle \\
&= \left\langle -\sum_{i=1}^{N} \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} \right) \boldsymbol{S}_{(i)} \boldsymbol{U}_\tau, \boldsymbol{U}_{\tau+1} - \boldsymbol{U}_\tau \right\rangle \\
&= -\left\langle \sum_{i=1}^{N} \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} \right) \boldsymbol{S}_{(i)} \boldsymbol{U}_\tau, \frac{\eta_\tau}{N} \sum_{i=1}^{N} \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} - \boldsymbol{P}_{\boldsymbol{U}_\tau} \right) \boldsymbol{S}_{(i)} \boldsymbol{U}_\tau + \eta_\tau^2 \boldsymbol{e}_{1,\tau} \right\rangle \\
&= -\left\langle \sum_{i=1}^{N} \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} \right) \boldsymbol{S}_{(i)} \boldsymbol{U}_\tau, \frac{\eta_\tau}{N} \sum_{i=1}^{N} \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} - \boldsymbol{P}_{\boldsymbol{U}_\tau} \right) \boldsymbol{S}_{(i)} \boldsymbol{U}_\tau \right\rangle \\
&- \left\langle \sum_{i=1}^{N} \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} \right) \boldsymbol{S}_{(i)} \boldsymbol{U}_\tau, \eta_\tau^2 \boldsymbol{e}_{1,\tau} \right\rangle \\
&= -\left\langle \sum_{i=1}^{N} \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} - \boldsymbol{P}_{\boldsymbol{U}_\tau} \right) \boldsymbol{S}_{(i)} \boldsymbol{U}_\tau, \frac{\eta_\tau}{N} \sum_{i=1}^{N} \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} - \boldsymbol{P}_{\boldsymbol{U}_\tau} \right) \boldsymbol{S}_{(i)} \boldsymbol{U}_\tau \right\rangle \\
&- \left\langle \sum_{i=1}^{N} \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} \right) \boldsymbol{S}_{(i)} \boldsymbol{U}_\tau, \eta_\tau^2 \boldsymbol{e}_{1,\tau} \right\rangle \\
&\leq -\eta_\tau N \left\| \square \boldsymbol{U}_\tau \right\|_F^2 + \eta_\tau^2 \left\| \boldsymbol{e}_{1,\tau} \right\|_F \sum_{i=1}^{N} \left\| \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} \right) \boldsymbol{S}_{(i)} \boldsymbol{U}_\tau \right\|_F \\
&\leq -\eta_\tau N \left\| \square \boldsymbol{U}_\tau \right\|_F^2 + M_1 \eta_\tau^2 \left\| \square \boldsymbol{U}_\tau \right\|_F^2 \sum_{i=1}^{N} G_{(i),op} \sqrt{r}
\end{aligned}
$$

Next, we calculate:

$$
\begin{aligned}
&\left\langle \nabla_{\boldsymbol{V}_{(i)}} \mathcal{L}_{(i),1}(\boldsymbol{U}_\tau, \boldsymbol{V}_{(i),\tau}) + \nabla_{\boldsymbol{V}_{(i)}} \mathcal{L}_{(i),2}(\boldsymbol{U}_\tau, \boldsymbol{V}_{(i),\tau}), \boldsymbol{V}_{(i),\tau+1} - \boldsymbol{V}_{(i),\tau} \right\rangle \\
&= \left\langle -\left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} \right) \boldsymbol{S}_{(i)} \boldsymbol{V}_{(i),\tau}, \eta_\tau \square \boldsymbol{V}_{(i),\tau} + \eta_\tau \boldsymbol{U}_\tau \square \boldsymbol{U}_\tau^T \boldsymbol{V}_{(i),\tau} + \eta_\tau^2 \boldsymbol{e}_{5,(i),\tau} \right\rangle \\
&= \left\langle -\left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} \right) \boldsymbol{S}_{(i)} \boldsymbol{V}_{(i),\tau}, \eta_\tau \square \boldsymbol{V}_{(i),\tau} \right\rangle + \left\langle -\left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} \right) \boldsymbol{S}_{(i)} \boldsymbol{V}_{(i),\tau}, \eta_\tau \boldsymbol{U}_\tau \square \boldsymbol{U}_\tau^T \boldsymbol{V}_{(i),\tau} \right\rangle \\
&+ \left\langle -\left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} \right) \boldsymbol{S}_{(i)} \boldsymbol{V}_{(i),\tau}, \eta_\tau^2 \boldsymbol{e}_{5,(i),\tau} \right\rangle \\
&= \left\langle -\left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} \right) \boldsymbol{S}_{(i)} \boldsymbol{V}_{(i),\tau}, \eta_\tau \square \boldsymbol{V}_{(i),\tau} \right\rangle + \left\langle -\left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} \right) \boldsymbol{S}_{(i)} \boldsymbol{V}_{(i),\tau}, \eta_\tau^2 \boldsymbol{e}_{5,(i),\tau} \right\rangle \\
&\leq -\eta_\tau \left\| \square \boldsymbol{V}_{(i),\tau} \right\|_F^2 + \eta_\tau^2 \left\| \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} \right) \boldsymbol{S}_{(i)} \boldsymbol{V}_{(i),\tau} \right\|_F \left\| \boldsymbol{e}_{5,(i),\tau} \right\|_F \\
&\leq -\eta_\tau \left\| \square \boldsymbol{V}_{(i),\tau} \right\|_F^2 + \eta_\tau^2 G_{(i),op} \sqrt{r} \left(C_{5,0} \left\| \square \boldsymbol{U}_\tau \right\|_F^2 + C_{5,1} \left\| \square \boldsymbol{V}_{(i),\tau} \right\|^2 \right)
\end{aligned}
$$

Adding them, we have:

$$\begin{aligned}
&\left\langle \sum_{i=1}^{N} \nabla_{\boldsymbol{U}} \mathcal{L}_{(i),1}(\boldsymbol{U}_\tau, \boldsymbol{V}_{(i),\tau}), \boldsymbol{U}_{\tau+1} - \boldsymbol{U}_\tau \right\rangle \\
&+ \left\langle \nabla_{\boldsymbol{V}_{(i)}} \mathcal{L}_{(i),1}(\boldsymbol{U}_\tau, \boldsymbol{V}_{(i),\tau}) + \nabla_{\boldsymbol{V}_{(i)}} \mathcal{L}_{(i),2}(\boldsymbol{U}_\tau, \boldsymbol{V}_{(i),\tau}), \boldsymbol{V}_{(i),\tau+1} - \boldsymbol{V}_{(i),\tau} \right\rangle \\
&\leq -\eta_\tau N \left\| \square \boldsymbol{U}_\tau \right\|_F^2 - \sum_{i=1}^{N} \eta_\tau \left\| \square \boldsymbol{V}_{(i),\tau} \right\|_F^2 + \eta_\tau^2 \left(N C_{6,0} \left\| \square \boldsymbol{U}_\tau \right\|_F^2 + C_{6,1} \sum_{i=1}^{N} \left\| \square \boldsymbol{V}_{(i),\tau} \right\|_F^2 \right)
\end{aligned}$$

where the constants are:

$$C_{6,0} = G_{max,op} \sqrt{r} (M_1 + C_{5,0})$$

and

$$C_{6,1} = G_{max,op} \sqrt{r} C_{5,1}$$

■

Finally, we come to the proof of Theorem 8.

Proof We choose constant a stepsize $\eta_\tau = \eta_1$ small enough:

$$\begin{aligned}
\eta_1 \leq \eta_c = \min \Bigg\{ & \frac{1}{2C_{6,0} + L\left(\left(\left(1 + \frac{M_2}{2}\right)^2 + 2\left(1 + \frac{C_{5,0}}{2}\right)^2 \right) \right)}, \frac{1}{2C_{6,1} + 2L\left(1 + \frac{C_{5,1}}{2}\right)^2}, \\
& \frac{1}{G_{max,op}\sqrt{r}} \frac{M_3}{2}, \frac{1}{G_{max,op}\sqrt{r}} \frac{\sqrt{M_3}}{\sqrt{6 + 12M_1 + M_1^2}} \Bigg\}
\end{aligned} \tag{63}$$

Obviously, η_1 satisfies the requirement in Lemma 15 and 16.

By the property of Lipschitz continuity, we have:

$$\begin{aligned}
&\mathcal{L}_{(i)}\left(\boldsymbol{U}_{\tau+1}, \boldsymbol{V}_{(i),\tau+1}\right) \leq \mathcal{L}_{(i)}\left(\boldsymbol{U}_\tau, \boldsymbol{V}_{(i),\tau}\right) \\
&+ \left\langle \nabla_{\boldsymbol{U}} \mathcal{L}_{(i)}\left(\boldsymbol{U}_\tau, \boldsymbol{V}_{(i),\tau}\right), \boldsymbol{U}_{\tau+1} - \boldsymbol{U}_\tau \right\rangle + \left\langle \nabla_{\boldsymbol{U}} \mathcal{L}_{(i)}\left(\boldsymbol{U}_\tau, \boldsymbol{V}_{(i),\tau}\right), \boldsymbol{V}_{(i),\tau+1} - \boldsymbol{V}_{(i),\tau} \right\rangle \\
&+ \frac{L}{2}\left(\left\| \boldsymbol{V}_{(i),\tau+1} - \boldsymbol{V}_{(i),\tau} \right\|_F^2 + \left\| \boldsymbol{U}_{\tau+1} - \boldsymbol{U}_\tau \right\|_F^2 \right)
\end{aligned}$$

where L is defined in (54). Since $\boldsymbol{U}_{\tau+1}^T \boldsymbol{V}_{(i),\tau+1} = 0$ and $\boldsymbol{U}_\tau^T \boldsymbol{V}_{(i),\tau} = 0$, we know that

$$\mathcal{L}_{(i)}\left(\boldsymbol{U}_{\tau+1}, \boldsymbol{V}_{(i),\tau+1}\right) = -f_i\left(\boldsymbol{U}_{\tau+1}, \boldsymbol{V}_{(i),\tau+1}\right)$$

and that

$$\mathcal{L}_{(i)}\left(\boldsymbol{U}_\tau, \boldsymbol{V}_{(i),\tau}\right) = -f_i\left(\boldsymbol{U}_\tau, \boldsymbol{V}_{(i),\tau}\right)$$

Then, summing up both sides for n from 1 to N, we have:

$$\begin{aligned}
&- f\left(\boldsymbol{U}_{\tau+1}, \{\boldsymbol{V}_{(i),\tau+1}\}\right) \leq -f\left(\boldsymbol{U}_\tau, \{\boldsymbol{V}_{(i),\tau}\}\right) \\
&+ \left\langle \sum_{i=1}^{N} \nabla_{\boldsymbol{U}} \mathcal{L}_{(i)}\left(\boldsymbol{U}_\tau, \boldsymbol{V}_{(i),\tau}\right), \boldsymbol{U}_{\tau+1} - \boldsymbol{U}_\tau \right\rangle + \sum_{i=1}^{N} \left\langle \nabla_{\boldsymbol{U}} \mathcal{L}_{(i)}\left(\boldsymbol{U}_\tau, \boldsymbol{V}_{(i),\tau}\right), \boldsymbol{V}_{(i),\tau+1} - \boldsymbol{V}_{(i),\tau} \right\rangle \\
&+ \sum_{i=1}^{N} \frac{L}{2}\left(\left\| \boldsymbol{V}_{(i),\tau+1} - \boldsymbol{V}_{(i),\tau} \right\|_F^2 + \left\| \boldsymbol{U}_{\tau+1} - \boldsymbol{U}_\tau \right\|_F^2 \right)
\end{aligned}$$

From equation (59), we know

$$\begin{aligned}
&\|\boldsymbol{U}_{\tau+1} - \boldsymbol{U}_\tau\|_F \\
&= \left\|\eta_\tau \square \boldsymbol{U}_\tau + \eta_\tau^2 \boldsymbol{e}_{1,\tau}\right\|_F \\
&\leq \eta_\tau \|\square \boldsymbol{U}_\tau\|_F + \left\|\eta_\tau^2 \boldsymbol{e}_{1,\tau}\right\|_F \\
&\leq \eta_\tau \|\square \boldsymbol{U}_\tau\|_F + \eta_\tau^2 M_1 \|\square \boldsymbol{U}_\tau\|_F^2 \\
&\leq \eta_\tau (1 + \frac{M_1}{2}) \|\square \boldsymbol{U}_\tau\|_F
\end{aligned}$$

Similarly, from Lemma 15, we have:

$$\begin{aligned}
&\left\|\boldsymbol{V}_{(i),\tau+1} - \boldsymbol{V}_{(i),\tau}\right\|_F \\
&= \left\|\eta_\tau \square \boldsymbol{V}_{(i),\tau} + \eta_\tau \boldsymbol{U}_\tau \square \boldsymbol{U}_\tau^T \boldsymbol{V}_{(i),\tau} + \eta_\tau^2 \boldsymbol{e}_{5,(i),\tau}\right\|_F \\
&\leq \eta_\tau \left\|\square \boldsymbol{V}_{(i),\tau}\right\|_F + \eta_\tau \left\|\boldsymbol{U}_\tau \square \boldsymbol{U}_\tau^T \boldsymbol{V}_{(i),\tau}\right\|_F + \eta_\tau^2 \left\|\boldsymbol{e}_{5,(i),\tau}\right\|_F \\
&\leq \eta_\tau \left\|\square \boldsymbol{V}_{(i),\tau}\right\|_F + \eta_\tau \|\square \boldsymbol{U}_\tau\|_F + \eta_\tau^2 \left\|\boldsymbol{e}_{5,(i),\tau}\right\|_F \\
&\leq \eta_\tau \|\square \boldsymbol{U}_\tau\|_F (1 + C_{5,0} \eta_\tau \|\square \boldsymbol{U}_\tau\|_F) + \eta_\tau \left\|\square \boldsymbol{V}_{(i),\tau}\right\|_F \left(1 + C_{5,1} \eta_\tau \left\|\square \boldsymbol{V}_{(i),\tau}\right\|_F\right) \\
&\leq \eta_\tau \left\|\square \boldsymbol{V}_{(i),\tau}\right\|_F \left(1 + \frac{1}{2} C_{5,1}\right) + \eta_\tau \|\square \boldsymbol{U}_\tau\|_F \left(1 + \frac{1}{2} C_{5,0}\right)
\end{aligned}$$

Combining the two inequalities and Lemma 16, we have:

$$\begin{aligned}
&- f\left(\boldsymbol{U}_{\tau+1}, \{\boldsymbol{V}_{(i),\tau+1}\}\right) \\
&\leq -f\left(\boldsymbol{U}_\tau, \{\boldsymbol{V}_{(i),\tau}\}\right) - \eta_\tau \left(N \|\square \boldsymbol{U}_\tau\|_F^2 + \sum_{i=1}^N \left\|\square \boldsymbol{V}_{(i),\tau}\right\|_F^2\right) \\
&+ \eta_\tau^2 N C_{6,0} \|\square \boldsymbol{U}_\tau\|_F^2 + \eta_\tau^2 \sum_{i=1}^N C_{6,1} \left\|\square \boldsymbol{V}_{(i),\tau}\right\|_F^2 \\
&+ \eta_\tau^2 \frac{L}{2} \sum_{i=1}^N \left(\left(\left(1 + \frac{M_2}{2}\right)^2 + 2\left(1 + \frac{C_{5,0}}{2}\right)^2\right) \|\square \boldsymbol{U}_\tau\|_F^2 + 2\left(1 + \frac{C_{5,1}}{2}\right)^2 \left\|\square \boldsymbol{V}_{(i),\tau}\right\|_F^2\right) \\
&\leq -f\left(\boldsymbol{U}_\tau, \{\boldsymbol{V}_{(i),\tau}\}\right) - \frac{\eta_\tau}{2} \left(N \|\square \boldsymbol{U}_\tau\|_F^2 + \sum_{i=1}^N \left\|\square \boldsymbol{V}_{(i),\tau}\right\|_F^2\right)
\end{aligned} \tag{64}$$

Summing up both sides for τ from 1 to R and rearranging terms, we have:

$$\frac{\eta_1}{2} \sum_{\tau=1}^R \left(N \|\square \boldsymbol{U}_\tau\|_F^2 + \sum_{i=1}^N \left\|\square \boldsymbol{V}_{(i),\tau}\right\|_F^2\right) \leq -f(\boldsymbol{U}_1, \{\boldsymbol{V}_{(i),1}\}) + f(\boldsymbol{U}_{R+1}, \{\boldsymbol{V}_{(i),R+1}\})$$

As a result,

$$\min_{\tau \in \{1,\cdots,N\}} \sum_{\tau=1}^R \left(N \|\square \boldsymbol{U}_\tau\|_F^2 + \sum_{i=1}^N \left\|\square \boldsymbol{V}_{(i),\tau}\right\|_F^2\right) \leq \frac{2\left(f(\boldsymbol{U}_{R+1}, \{\boldsymbol{V}_{(i),R+1}\}) - f(\boldsymbol{U}_1, \{\boldsymbol{V}_{(i),1}\})\right)}{R\eta_1}$$

This completes the proof of Theorem 8. Notice that $C_{6,0}$, $C_{6,1}$, and L are of the order $G_{max,op}\sqrt{r}$, thus the requirement on η_c in equation (63) becomes:

$$\eta_c = C_\eta \frac{1}{G_{max,op}\sqrt{r}}$$

where C_η is a constant that only depends on M_1, M_2, and M_3 from the generalized retraction Definition 5. ■

Appendix E. Proof for local linear convergence

In this section, we will show the full proof of Theorem 10. A formal theorem is stated below.

Theorem 17 *(Formal version of theorem 10) Under assumptions 4.2, 6.1, and 6.2, if the difference between the population and sample covariance is small* $\sqrt{\sum_{i=1}^N \left\|\boldsymbol{S}_{(i)} - \boldsymbol{\Sigma}_{(i)}\right\|_F^2} \leq \min\{\frac{\sqrt{2}-1}{4}\mu\theta^{3/2}, \frac{\mu^2\theta^2}{128^2\times 2G_{max,op}}\}$ *and* $\left\|\boldsymbol{S}_{(i)} - \boldsymbol{\Sigma}_{(i)}\right\| \leq G_{max,op}$*, when we initialize close to the global optimum* $\phi_0 \leq \phi_\tau \leq \frac{\mu^3\theta^3}{411041792G_{max,op}^2}$*, and choose a constant stepsize* $\eta_t = \eta = O\left(\frac{1}{G_{op,max}\sqrt{r}}\right)$*, then Algorithm 2 with choice 1 will converge into the global optimum:*

$$f(\hat{\boldsymbol{U}}, \{\hat{\boldsymbol{V}}_{(i)}\}) - f(\boldsymbol{U}_R, \{\boldsymbol{V}_{(i),R}\}) = O\left(\left(1 - \eta\frac{\mu\theta}{32}\right)^R\right)$$

where $\{\hat{\boldsymbol{U}}, \{\hat{\boldsymbol{V}}_{(i)}\}\}$ *is a set of optimal solutions to problem* (7)*.*

Furthermore, we can recover the exact global optimal solutions:

$$\left\|\boldsymbol{P}_{\boldsymbol{U}_R} - \boldsymbol{P}_{\hat{\boldsymbol{U}}_g}\right\|_F^2 + \frac{1}{N}\sum_{i=1}^N \left\|\boldsymbol{P}_{\boldsymbol{V}_{(i),R}} - \boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i)}}\right\|_F^2 = O\left(\left(1 - \eta\frac{\mu\theta}{32}\right)^R\right)$$

We will start by introducing needed notations, then proceed to establish some lemmas that characterize the local geometry of the optimization objective, then prove Theorem 10 at the end.

At communication round τ, remember that we use $\boldsymbol{U}_\tau$ and $\boldsymbol{V}_{(i),\tau}$ to denote the updated variables. We use $\left(\hat{\boldsymbol{U}}, \{\hat{\boldsymbol{V}}_{(i)}\}\right)$ to denote one set of optimal solutions to (7). For simplicity, we use $\hat{\boldsymbol{\Pi}}_g$ to denote the projection $\hat{\boldsymbol{\Pi}}_g = \hat{\boldsymbol{U}}\hat{\boldsymbol{U}}^T$ and $\hat{\boldsymbol{\Pi}}_{(i)}$ to denote the projection $\hat{\boldsymbol{\Pi}}_{(i)} = \hat{\boldsymbol{V}}_{(i)}\hat{\boldsymbol{V}}_{(i)}^T$.

Since each covariance matrix $\boldsymbol{S}_{(i)}$ is symmetric positive semidefinite, we can find matrix $\boldsymbol{F}_{(i)} \in \mathbb{R}^{d\times d}$ such that $\boldsymbol{F}_{(i)}\boldsymbol{F}_{(i)}^T = \boldsymbol{S}_{(i)}$ by Cholesky factorization. Furthermore, we can define $\boldsymbol{F}_{(i),g}$, $\boldsymbol{F}_{(i),l}$, and $\boldsymbol{R}_{(i)}$ as,

$$\begin{aligned}
\boldsymbol{F}_{(i),g} &= \hat{\boldsymbol{\Pi}}_g\boldsymbol{F}_{(i)} \\
\boldsymbol{F}_{(i),l} &= \hat{\boldsymbol{\Pi}}_{(i)}\boldsymbol{F}_{(i)} \\
\hat{\boldsymbol{F}}_{(i),l} &= \boldsymbol{F}_{(i),g} + \boldsymbol{F}_{(i),l} \\
\boldsymbol{R}_{(i)} &= \boldsymbol{F}_{(i)} - \hat{\boldsymbol{\Pi}}_g\boldsymbol{F}_{(i)} - \hat{\boldsymbol{\Pi}}_{(i)}\boldsymbol{F}_{(i)}
\end{aligned} \tag{65}$$

Apparently, $\boldsymbol{F}_{(i),g}^T \boldsymbol{R}_{(i)} = \boldsymbol{F}_{(i),l}^T \boldsymbol{R}_{(i)} = 0$.

Next we will introduce a set of optimal solutions $\left(\hat{\boldsymbol{U}}_\tau, \{\hat{\boldsymbol{V}}_{(i),\tau}\}\right)$ that is close to the current updates $(\boldsymbol{U}_\tau, \{\boldsymbol{V}_{(i),\tau}\}$. The variables $\hat{\boldsymbol{U}}_\tau$ and $\hat{\boldsymbol{V}}_{(i),\tau}$'s are defined as

$$\hat{\boldsymbol{U}}_\tau = \hat{\boldsymbol{\Pi}}_g \boldsymbol{U}_\tau \left((\boldsymbol{U}_\tau)^T \hat{\boldsymbol{\Pi}}_g \boldsymbol{U}_\tau\right)^{-1/2} \tag{66}$$

and

$$\hat{\boldsymbol{V}}_{(i),\tau} = \hat{\boldsymbol{\Pi}}_{(i)} \boldsymbol{V}_{(i),\tau} \left(\boldsymbol{V}_{(i),\tau}^T \hat{\boldsymbol{\Pi}}_{(i)} \boldsymbol{V}_{(i),\tau}\right)^{-1/2} \tag{67}$$

for each $n = 1, ...N$.

It's easy to verify that

$$\hat{\boldsymbol{U}}_\tau (\hat{\boldsymbol{U}}_\tau)^T = \hat{\boldsymbol{\Pi}}_g$$

and that

$$\hat{\boldsymbol{V}}_{(i),\tau} (\hat{\boldsymbol{V}}_{(i),\tau})^T = \hat{\boldsymbol{\Pi}}_{(i)}$$

Notice that $\{\hat{\boldsymbol{U}}_\tau, \{\hat{\boldsymbol{V}}_{(i),\tau}\}\}$ is one set of global optimal solutions that is dependent on the iteration index τ. The $\hat{\boldsymbol{U}}_\tau$ and $\hat{\boldsymbol{V}}_{(i),\tau}$'s are dependent on the communication round τ. We use $\Delta \boldsymbol{U}_\tau$ to denote the difference between $\boldsymbol{U}_\tau$ and $\hat{\boldsymbol{U}}_\tau$:

$$\Delta \boldsymbol{U}_\tau = \boldsymbol{U}_\tau - \hat{\boldsymbol{U}}_\tau \tag{68}$$

and similarly:

$$\Delta \boldsymbol{V}_{(i),\tau} = \boldsymbol{V}_{(i),\tau} - \hat{\boldsymbol{V}}_{(i),\tau} \tag{69}$$

Since $\hat{\boldsymbol{U}}_\tau$ and $\hat{\boldsymbol{V}}_{(i),\tau}$'s are optimal, we can simplify the KKT conditions in (24) as

$$\begin{aligned} &\boldsymbol{R}_{(i)} \boldsymbol{F}_{(i)}^T \hat{\boldsymbol{V}}_{(i),\tau} = 0, \quad \forall i \in [N] \\ &\sum_{i=1}^N \boldsymbol{R}_{(i)} \boldsymbol{F}_{(i)}^T \hat{\boldsymbol{U}}_\tau = 0 \end{aligned} \tag{70}$$

We can replace $\boldsymbol{F}_{(i)}$ by $\hat{\boldsymbol{F}}_{(i)}$ in (70) since $\boldsymbol{F}_{(i)}^T \hat{\boldsymbol{V}}_{(i),\tau} = \hat{\boldsymbol{F}}_{(i)}^T \hat{\boldsymbol{V}}_{(i),\tau}$ and $\boldsymbol{F}_{(i)}^T \hat{\boldsymbol{U}}_\tau = \hat{\boldsymbol{F}}_{(i)}^T \hat{\boldsymbol{U}}_\tau$.

We will first show some properties of the introduced variables.

Lemma 18 *Under the same conditions as Theorem 10, there exists constants* $\hat{\theta} = \frac{\theta}{\sqrt{2}}$, $\hat{\mu} = \frac{\mu}{\sqrt{2}}$, *such that the following holds,*

1. $\left\|\hat{\boldsymbol{F}}_{(i)}\right\| \leq \sqrt{2G_{max,op}}$.
2. *The smallest nonzero eigenvalue of* $\hat{\boldsymbol{F}}_{(i)} \hat{\boldsymbol{F}}_{(i)}^T$ *is lower bounded by* $\hat{\mu}$.
3. $\left\|\sum_{i=1}^N \frac{1}{N} \hat{\boldsymbol{\Pi}}_{(i)}\right\| \leq 1 - \hat{\theta}$.
4. $\left\|\boldsymbol{R}_{(i)}\right\| \leq \frac{\hat{\mu}\hat{\theta}}{64\sqrt{2G_{max,op}}} = \frac{\mu\theta}{128\sqrt{2G_{max,op}}}$

We will use the $\hat{\theta}$ and $\hat{\mu}$ notations in the remaining parts of the section.

Proof We will prove the claims one by one.

From (34), we know,

$$\frac{1}{N}\sum_{i=1}^{N}\left\|\boldsymbol{\Pi}_g+\boldsymbol{\Pi}_{(i)}-\hat{\boldsymbol{\Pi}}_g-\hat{\boldsymbol{\Pi}}_{(i)}\right\|_F^2\leq\frac{4}{N\delta^2}\sum_{i=1}^{N}\left\|\boldsymbol{S}_{(i)}-\boldsymbol{\Sigma}_{(i)}\right\|_F^2$$

where we replace δ by μ since $\left(\boldsymbol{I}-\boldsymbol{\Pi}_g-\boldsymbol{\Pi}_{(i)}\right)\boldsymbol{\Sigma}_{(i)}=0$.

Therefore, the difference between $\hat{\boldsymbol{F}}_{(i)}\hat{\boldsymbol{F}}_{(i)}^T$ and $\boldsymbol{\Sigma}_{(i)}$ is upper bounded by,

$$\begin{aligned}
&\left\|\hat{\boldsymbol{F}}_{(i)}\hat{\boldsymbol{F}}_{(i)}^T-\boldsymbol{\Sigma}_{(i)}\right\|\\
&=\left\|\left(\hat{\boldsymbol{\Pi}}_g+\hat{\boldsymbol{\Pi}}_{(i)}\right)\boldsymbol{S}_{(i)}\left(\hat{\boldsymbol{\Pi}}_g+\hat{\boldsymbol{\Pi}}_{(i)}\right)-\boldsymbol{\Sigma}_{(i)}\right\|\\
&\leq\left\|\left(\hat{\boldsymbol{\Pi}}_g+\hat{\boldsymbol{\Pi}}_{(i)}\right)\boldsymbol{\Sigma}_{(i)}\left(\hat{\boldsymbol{\Pi}}_g+\hat{\boldsymbol{\Pi}}_{(i)}\right)-\boldsymbol{\Sigma}_{(i)}\right\|+\left\|\left(\hat{\boldsymbol{\Pi}}_g+\hat{\boldsymbol{\Pi}}_{(i)}\right)\left(\boldsymbol{\Sigma}_{(i)}-\boldsymbol{S}_{(i)}\right)\left(\hat{\boldsymbol{\Pi}}_g+\hat{\boldsymbol{\Pi}}_{(i)}\right)\right\|\\
&\leq 2\left\|\hat{\boldsymbol{\Pi}}_g+\hat{\boldsymbol{\Pi}}_{(i)}-\boldsymbol{\Pi}_g-\boldsymbol{\Pi}_{(i)}\right\|_F\left\|\boldsymbol{\Sigma}_{(i)}\right\|+\left\|\boldsymbol{\Sigma}_{(i)}-\boldsymbol{S}_{(i)}\right\|\\
&\leq\frac{8G_{max,op}}{\mu}\sqrt{\sum_{j=1}^{N}\left\|\boldsymbol{\Sigma}_{(j)}-\boldsymbol{S}_{(j)}\right\|_F^2}+\left\|\boldsymbol{\Sigma}_{(i)}-\boldsymbol{S}_{(i)}\right\|\\
&\leq\frac{\mu^2\theta^2}{128^2\times 2G_{max,op}}
\end{aligned}$$

From Weyl's theorem, we know,

$$\begin{aligned}
\left\|\hat{\boldsymbol{F}}_{(i)}\right\|^2&=\left\|\hat{\boldsymbol{F}}_{(i)}\hat{\boldsymbol{F}}_{(i)}^T\right\|\\
&\leq\left\|\hat{\boldsymbol{F}}_{(i)}\hat{\boldsymbol{F}}_{(i)}^T-\boldsymbol{\Sigma}_{(i)}\right\|+\left\|\boldsymbol{\Sigma}_{(i)}\right\|\\
&\leq 2G_{max,op}
\end{aligned}$$

This proves Claim 1.

Also by Weyl's theorem, we know,

$$\begin{aligned}
&\lambda_{2r}\left(\hat{\boldsymbol{F}}_{(i)}\hat{\boldsymbol{F}}_{(i)}^T\right)\\
&\geq\lambda_{2r}\left(\hat{\boldsymbol{F}}_{(i)}\hat{\boldsymbol{F}}_{(i)}^T-\boldsymbol{\Sigma}_{(i)}\right)-\left\|\hat{\boldsymbol{F}}_{(i)}\hat{\boldsymbol{F}}_{(i)}^T-\boldsymbol{\Sigma}_{(i)}\right\|\\
&\geq\mu-\frac{\mu^2\theta^2}{32768G_{max,op}}\\
&\geq\mu\frac{1}{\sqrt{2}}
\end{aligned}$$

This proves Claim 2.

Next, we consider the results from Theorem 1,

$$\left\|\boldsymbol{P}_{\hat{\boldsymbol{U}}}-\boldsymbol{\Pi}_g\right\|_F^2+\frac{1}{N}\sum_{i=1}^{N}\left\|\boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i)}}-\boldsymbol{\Pi}_{(i)}\right\|_F^2\leq\frac{8}{\theta\mu^2}\frac{1}{N}\sum_{i=1}^{N}\left\|\boldsymbol{\Sigma}_{(i)}-\boldsymbol{S}_{(i)}\right\|_F^2$$

Therefore, an upper bound for $\left\|\boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i)}}-\boldsymbol{\Pi}_{(i)}\right\|_F$ is

$$\begin{aligned}&\left\|\boldsymbol{P}_{\hat{\boldsymbol{V}}_{(i)}}-\boldsymbol{\Pi}_{(i)}\right\|_F\\&\leq 2\sqrt{2}\frac{1}{\sqrt{\theta}\mu}\sqrt{\sum_{i=1}^{N}\left\|\boldsymbol{\Sigma}_{(i)}-\boldsymbol{S}_{(i)}\right\|_F^2}\\&\leq\theta\frac{2-\sqrt{2}}{2}\end{aligned}$$

where we applied the condition that $\sqrt{\sum_{i=1}^{N}\left\|\boldsymbol{\Sigma}_{(i)}-\boldsymbol{S}_{(i)}\right\|_F^2}\leq\mu\theta^{1.5}\frac{\sqrt{2}-1}{4}$ in the last inequality.

As a result, we have,

$$\begin{aligned}&\left\|\frac{1}{N}\sum_{i=1}^{N}\hat{\boldsymbol{\Pi}}_{(i)}\right\|\\&\leq\left\|\frac{1}{N}\sum_{i=1}^{N}\boldsymbol{\Pi}_{(i)}\right\|+\frac{1}{N}\sum_{i=1}^{N}\left\|\boldsymbol{\Pi}_{(i)}-\hat{\boldsymbol{\Pi}}_{(i)}\right\|\\&\leq 1-\theta+\theta\left(1-\frac{1}{\sqrt{2}}\right)\\&=\frac{\theta}{\sqrt{2}}\end{aligned}$$

This proves Claim 3.

Then we analyze the norm of $\boldsymbol{R}_{(i)}\boldsymbol{R}_{(i)}^T$,

$$\begin{aligned}&\left\|\boldsymbol{R}_{(i)}\right\|^2=\left\|\boldsymbol{R}_{(i)}^T\boldsymbol{R}_{(i)}\right\|\\&=\left\|\left(\boldsymbol{I}-\hat{\boldsymbol{\Pi}}_g-\hat{\boldsymbol{\Pi}}_{(i)}\right)\boldsymbol{S}_{(i)}\left(\boldsymbol{I}-\hat{\boldsymbol{\Pi}}_g-\hat{\boldsymbol{\Pi}}_{(i)}\right)\right\|\\&\leq\left\|\left(\boldsymbol{I}-\hat{\boldsymbol{\Pi}}_g-\hat{\boldsymbol{\Pi}}_{(i)}\right)\boldsymbol{\Sigma}_{(i)}\left(\boldsymbol{I}-\hat{\boldsymbol{\Pi}}_g-\hat{\boldsymbol{\Pi}}_{(i)}\right)\right\|\\&+\left\|\left(\boldsymbol{I}-\hat{\boldsymbol{\Pi}}_g-\hat{\boldsymbol{\Pi}}_{(i)}\right)\left(\boldsymbol{S}_{(i)}-\boldsymbol{\Sigma}_{(i)}\right)\left(\boldsymbol{I}-\hat{\boldsymbol{\Pi}}_g-\hat{\boldsymbol{\Pi}}_{(i)}\right)\right\|\\&\leq\left\|\boldsymbol{\Pi}_g+\boldsymbol{\Pi}_{(i)}-\hat{\boldsymbol{\Pi}}_g-\hat{\boldsymbol{\Pi}}_{(i)}\right\|_F\left\|\boldsymbol{\Sigma}_{(i)}\right\|+\left\|\boldsymbol{S}_{(i)}-\boldsymbol{\Sigma}_{(i)}\right\|\\&\leq\sqrt{\sum_{i=1}^{N}\left\|\boldsymbol{\Sigma}_{(i)}-\boldsymbol{S}_{(i)}\right\|_F^2}\left(1+4\frac{G_{max,op}}{\delta}\right)\frac{\mu\theta}{128\sqrt{2G_{max,op}}}\end{aligned}$$

where the last inequality comes from the fact that $\sum_{i=1}^{N}\left\|\boldsymbol{\Sigma}_{(i)}-\boldsymbol{S}_{(i)}\right\|_F^2\leq\frac{1}{\left(1+\frac{8G_{max,op}}{\mu}\right)^2}\left(\frac{\mu^2\theta^2}{32768G_{max,op}}\right)^2$.

■

As discussed in Section 6.1.1, different $\boldsymbol{U}$ and $\boldsymbol{V}_{(i)}$'s may have the same objective value, as long as they span the same column space. We introduce a variable ζ to denote the subspace distance between the estimate and ground truth:

$$\zeta_{(i),\tau} = r - \left\langle \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}, \boldsymbol{\Pi}_{(i)} \right\rangle \tag{71}$$

for each $i = 1, ...N$, and,

$$\zeta_{(0),\tau} = r - \langle \boldsymbol{P}_{\boldsymbol{U}_\tau}, \boldsymbol{\Pi}_g \rangle \tag{72}$$

We use ζ_τ to denote:

$$\zeta_\tau = \zeta_{(0),\tau} + \frac{1}{N}\sum_{i=1}^{N} \zeta_{(i),\tau} \tag{73}$$

The $\zeta_{(0),\tau}$ and $\zeta_{(i),\tau}$'s defined represent how far away the iterates are from the ground truth, measured by subspace distance.

We can also define,

$$\widetilde{\zeta}_{(i),\tau} = 2r - \left\langle \boldsymbol{P}_{\boldsymbol{U}_\tau} + \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}, \hat{\boldsymbol{\Pi}}_g + \hat{\boldsymbol{\Pi}}_{(i)} \right\rangle \tag{74}$$

for each $i = 1, ...N$. We use $\widetilde{\zeta}_\tau$ to denote:

$$\widetilde{\zeta}_\tau = \sum_{i=1}^{N} \widetilde{\zeta}_{(i),\tau} \tag{75}$$

From Lemma 13, since $\hat{\boldsymbol{\Pi}}_{(i)}$'s are $\hat{\theta}$-misaligned, there exists a relation between ζ_τ and $\widetilde{\zeta}_\tau$:

$$\frac{\hat{\theta}}{2} N\zeta_\tau \leq \widetilde{\zeta}_\tau \leq N\zeta_\tau \tag{76}$$

For simplicity, we also define the optimality gap ϕ_τ as,

$$\phi_\tau = \frac{1}{2}\sum_{i=1}^{N} \left(\mathrm{Tr}\left(\hat{\boldsymbol{\Pi}}_g \boldsymbol{S}_{(i)}\right) + \mathrm{Tr}\left(\hat{\boldsymbol{\Pi}}_{(i)} \boldsymbol{S}_{(i)}\right) - \mathrm{Tr}\left(\boldsymbol{P}_{\boldsymbol{U}_\tau} \boldsymbol{S}_{(i)}\right) - \mathrm{Tr}\left(\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} \boldsymbol{S}_{(i)}\right)\right) \tag{77}$$

We then use the optimality gap ϕ_τ to upper bound the norm of $\Delta\boldsymbol{U}_\tau$ and $\Delta\boldsymbol{V}_{(i),\tau}$.

Lemma 19 *Under the same conditions as Theorem 10, we have,*

$$\phi_\tau \geq \frac{\hat{\theta}\hat{\mu}}{16}\left(N \left\|\Delta\boldsymbol{U}_\tau\right\|_F^2 + \sum_{i=1}^{N} \left\|\Delta\boldsymbol{V}_{(i),\tau}\right\|_F^2 \right) \tag{78}$$

Proof By definition of the optimality gap ϕ_τ, we have,

$$
\begin{aligned}
2\phi_\tau &= \sum_{i=1}^{N} \left(\mathrm{Tr}\left(\hat{\boldsymbol{\Pi}}_g \boldsymbol{S}_{(i)}\right) + \mathrm{Tr}\left(\hat{\boldsymbol{\Pi}}_{(i)} \boldsymbol{S}_{(i)}\right) - \mathrm{Tr}\left(\boldsymbol{P}_{\boldsymbol{U}_\tau} \boldsymbol{S}_{(i)}\right) - \mathrm{Tr}\left(\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} \boldsymbol{S}_{(i)}\right)\right) \\
&= \sum_{i=1}^{N} \left(\left\| \boldsymbol{F}_{(i)} - \left(\boldsymbol{P}_{\boldsymbol{U}_\tau} + \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \boldsymbol{F}_{(i)} \right\|_F^2 - \left\| \boldsymbol{F}_{(i)} - \left(\hat{\boldsymbol{\Pi}}_g + \hat{\boldsymbol{\Pi}}_{(i)}\right) \boldsymbol{F}_{(i)} \right\|_F^2 \right) \\
&= \sum_{i=1}^{N} \left(\left\| \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \hat{\boldsymbol{F}}_{(i)} + \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \boldsymbol{R}_{(i)} \right\|_F^2 - \left\| \boldsymbol{R}_{(i)} \right\|_F^2 \right) \\
&= \sum_{i=1}^{N} \left\| \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \hat{\boldsymbol{F}}_{(i)} \right\|_F^2 + \underbrace{2 \left\langle \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \hat{\boldsymbol{F}}_{(i)}, \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \boldsymbol{R}_{(i)} \right\rangle}_{\texttt{Term I}} \\
&+ \underbrace{\left\| \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \boldsymbol{R}_{(i)} \right\|_F^2 - \left\| \boldsymbol{R}_{(i)} \right\|_F^2}_{\texttt{Term II}}
\end{aligned}
$$

The above can be further simplified. For `Term I`, we have,

$$
\begin{aligned}
&\sum_{i=1}^{N} 2 \left\langle \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \hat{\boldsymbol{F}}_{(i)}, \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \boldsymbol{R}_{(i)} \right\rangle \\
&= \sum_{i=1}^{N} 2\mathrm{Tr}\left(\boldsymbol{R}_{(i)}^T \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \hat{\boldsymbol{F}}_{(i)} \right) \\
&= \sum_{i=1}^{N} 2\mathrm{Tr}\left(\boldsymbol{R}_{(i)}^T \left(\hat{\boldsymbol{\Pi}}_g + \hat{\boldsymbol{\Pi}}_{(i)} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \hat{\boldsymbol{F}}_{(i)} \right) \\
&= \sum_{i=1}^{N} 2\mathrm{Tr}\left(\boldsymbol{R}_{(i)}^T \left(-\Delta \boldsymbol{U}_\tau \Delta \boldsymbol{U}_\tau^T - \Delta \boldsymbol{V}_{(i),\tau} \Delta \boldsymbol{V}_{(i),\tau}^T\right) \hat{\boldsymbol{F}}_{(i)} \right)
\end{aligned}
$$

where we have applied the KKT conditions (70) that $\hat{\boldsymbol{V}}_{(i),\tau}^T \hat{\boldsymbol{F}}_{(i)} \boldsymbol{R}_{(i)}^T = 0$ and $\sum_{i=1}^{N} \hat{\boldsymbol{U}}_\tau^T \hat{\boldsymbol{F}}_{(i)} \boldsymbol{R}_{(i)}^T = 0$ in the third equality.

For term `Term II`, we can also derive

$$
\begin{aligned}
&\left\| \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \boldsymbol{R}_{(i)} \right\|_F^2 - \left\| \boldsymbol{R}_{(i)} \right\|_F^2 \\
&= \mathrm{Tr}\left(\boldsymbol{R}_{(i)}^T \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \boldsymbol{R}_{(i)} \right) - \mathrm{Tr}\left(\boldsymbol{R}_{(i)}^T \boldsymbol{R}_{(i)} \right) \\
&= \mathrm{Tr}\left(\boldsymbol{R}_{(i)}^T \left(\hat{\boldsymbol{\Pi}}_g + \hat{\boldsymbol{\Pi}}_{(i)} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \boldsymbol{R}_{(i)} \right) \\
&= -\mathrm{Tr}\left(\boldsymbol{R}_{(i)}^T \left(\Delta \boldsymbol{U}_\tau \Delta \boldsymbol{U}_\tau^T + \Delta \boldsymbol{V}_{(i),\tau} \Delta \boldsymbol{V}_{(i),\tau}^T\right) \boldsymbol{R}_{(i)} \right)
\end{aligned}
$$

where we used the condition $\hat{\boldsymbol{U}}_\tau^T \boldsymbol{R}_{(i)} = \hat{\boldsymbol{V}}_{(i),\tau}^T \boldsymbol{R}_{(i)} = 0$ in the last equality.

Combining these, we have,

$$
\begin{aligned}
&2\phi_\tau \\
&= \sum_{i=1}^{N} \Big(\Big\| \Big(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} \Big) \hat{\boldsymbol{F}}_{(i)} \Big\|_F^2 - \mathrm{Tr} \left(\boldsymbol{R}_{(i)}^T \left(\Delta \boldsymbol{U}_\tau \Delta \boldsymbol{U}_\tau^T + \Delta \boldsymbol{V}_{(i),\tau} \Delta \boldsymbol{V}_{(i),\tau}^T \right) \boldsymbol{R}_{(i)} \right) \\
&- 2\mathrm{Tr} \left(\boldsymbol{R}_{(i)}^T \left(\Delta \boldsymbol{U}_\tau \Delta \boldsymbol{U}_\tau^T + \Delta \boldsymbol{V}_{(i),\tau} \Delta \boldsymbol{V}_{(i),\tau}^T \right) \hat{\boldsymbol{F}}_{(i)} \right) \Big) \qquad (79)
\end{aligned}
$$

From Lemma 18, we know that $\hat{\boldsymbol{F}}_{(i)} \hat{\boldsymbol{F}}_{(i)}^T \succeq \hat{\mu} \left(\hat{\boldsymbol{\Pi}}_g + \hat{\boldsymbol{\Pi}}_{(i)} \right)$, thus

$$
\begin{aligned}
&\sum_{i=1}^{N} \mathrm{Tr} \left(\left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} \right) \hat{\boldsymbol{F}}_{(i)} \hat{\boldsymbol{F}}_{(i)}^T \right) \geq \sum_{i=1}^{N} \mathrm{Tr} \left(\left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} \right) \left(\hat{\boldsymbol{\Pi}}_g + \hat{\boldsymbol{\Pi}}_{(i)} \right) \right) \hat{\mu} \\
&\geq \frac{\hat{\mu}\hat{\theta}}{2} \sum_{i=1}^{N} \left(r - \mathrm{Tr} \left(\hat{\boldsymbol{\Pi}}_g \boldsymbol{P}_{\boldsymbol{U}_\tau} \right) + r - \mathrm{Tr} \left(\hat{\boldsymbol{\Pi}}_{(i)} \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} \right) \right) \\
&\geq \frac{\hat{\mu}\hat{\theta}}{4} \sum_{i=1}^{N} \left(\|\Delta \boldsymbol{U}_\tau\|_F^2 + \left\|\Delta \boldsymbol{V}_{(i),\tau}\right\|_F^2 \right)
\end{aligned}
$$

By Cauchy-Schwartz inequality, we have,

$$
\begin{aligned}
&\mathrm{Tr} \left(\boldsymbol{R}_{(i)}^T \left(\Delta \boldsymbol{U}_\tau \Delta \boldsymbol{U}_\tau^T + \Delta \boldsymbol{V}_{(i),\tau} \Delta \boldsymbol{V}_{(i),\tau}^T \right) \boldsymbol{R}_{(i)} \right) \\
&= \mathrm{Tr} \left(\boldsymbol{R}_{(i)}^T \Delta \boldsymbol{U}_\tau \Delta \boldsymbol{U}_\tau^T \boldsymbol{R}_{(i)} \right) + \mathrm{Tr} \left(\boldsymbol{R}_{(i)}^T \Delta \boldsymbol{V}_{(i),\tau} \Delta \boldsymbol{V}_{(i),\tau}^T \boldsymbol{R}_{(i)} \right) \\
&\leq \|\Delta \boldsymbol{U}_\tau\|_F^2 \left\|\boldsymbol{R}_{(i)}\right\|^2 + \left\|\Delta \boldsymbol{V}_{(i),\tau}\right\|_F^2 \left\|\boldsymbol{R}_{(i)}\right\|^2
\end{aligned}
$$

and

$$
\begin{aligned}
&2\mathrm{Tr} \left(\boldsymbol{R}_{(i)}^T \left(\Delta \boldsymbol{U}_\tau \Delta \boldsymbol{U}_\tau^T + \Delta \boldsymbol{V}_{(i),\tau} \Delta \boldsymbol{V}_{(i),\tau}^T \right) \hat{\boldsymbol{F}}_{(i)} \right) \\
&\leq 2 \|\Delta \boldsymbol{U}_\tau\|_F^2 \left\|\boldsymbol{R}_{(i)}\right\| \left\|\hat{\boldsymbol{F}}_{(i)}\right\| + 2 \left\|\Delta \boldsymbol{V}_{(i),\tau}\right\|_F^2 \left\|\boldsymbol{R}_{(i)}\right\| \left\|\hat{\boldsymbol{F}}_{(i)}\right\|
\end{aligned}
$$

Since $\left\|\boldsymbol{R}_{(i)}\right\| \leq \frac{\hat{\mu}\hat{\theta}}{64\sqrt{2G_{max,op}}}$ and $\left\|\hat{\boldsymbol{F}}_{(i)}\right\| \leq \sqrt{2G_{max,op}}$, we have

$$
\begin{aligned}
&\left\|\boldsymbol{R}_{(i)}\right\|^2 + 2 \left\|\boldsymbol{R}_{(i)}\right\| \left\|\hat{\boldsymbol{F}}_{(i)}\right\| \\
&\leq 3 \left\|\boldsymbol{R}_{(i)}\right\| \left\|\hat{\boldsymbol{F}}_{(i)}\right\| \\
&\leq \frac{\hat{\mu}\hat{\theta}}{8}
\end{aligned}
$$

Thus we have,

$$
2\phi_\tau \geq \frac{\hat{\mu}\hat{\theta}}{8} \sum_{i=1}^{N} \left(\|\Delta \boldsymbol{U}_\tau\|_F^2 + \left\|\Delta \boldsymbol{V}_{(i),\tau}\right\|_F^2 \right)
$$

This completes our proof. ■

Next we will provide a lemma that characterizes the landscape of the objective.

Lemma 20 *Under the same conditions as Theorem 10, we have,*

$$
\begin{aligned}
&-\left\langle \sum_{i=1}^{N}\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}_\tau}-\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\boldsymbol{S}_{(i)}\boldsymbol{U}_\tau, \Delta\boldsymbol{U}_\tau\right\rangle - \sum_{i=1}^{N}\left\langle \left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}_\tau}-\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\boldsymbol{S}_{(i)}\boldsymbol{V}_{(i),\tau}, \Delta\boldsymbol{V}_{(i),\tau}\right\rangle \\
&\geq \phi_\tau
\end{aligned}
$$

Proof

We first consider the inner product term,

$$
\begin{aligned}
&\left\langle \left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}_\tau}-\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\boldsymbol{S}_{(i)}\boldsymbol{V}_{(i),\tau}, \Delta\boldsymbol{V}_{(i),\tau}\right\rangle \\
&= \left\langle \left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}_\tau}-\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\boldsymbol{F}_{(i)}, \Delta\boldsymbol{V}_{(i),\tau}\boldsymbol{V}_{(i),\tau}^T\boldsymbol{F}_{(i)}\right\rangle \\
&= \underbrace{\left\langle \left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}_\tau}-\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\hat{\boldsymbol{F}}_{(i)}, \Delta\boldsymbol{V}_{(i),\tau}\boldsymbol{V}_{(i),\tau}^T\hat{\boldsymbol{F}}_{(i)}\right\rangle}_{\texttt{Term III}} + \underbrace{\left\langle \left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}_\tau}-\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\boldsymbol{R}_{(i)}, \Delta\boldsymbol{V}_{(i),\tau}\boldsymbol{V}_{(i),\tau}^T\boldsymbol{R}_{(i)}\right\rangle}_{\texttt{Term IV}} \\
&+ \underbrace{\left\langle \left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}_\tau}-\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\hat{\boldsymbol{F}}_{(i)}, \Delta\boldsymbol{V}_{(i),\tau}\boldsymbol{V}_{(i),\tau}^T\boldsymbol{R}_{(i)}\right\rangle}_{\texttt{Term V}} + \underbrace{\left\langle \left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}_\tau}-\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\boldsymbol{R}_{(i)}, \Delta\boldsymbol{V}_{(i),\tau}\boldsymbol{V}_{(i),\tau}^T\hat{\boldsymbol{F}}_{(i)}\right\rangle}_{\texttt{Term VI}}
\end{aligned}
$$

We will analyze each term separately. For `Term III`, we know that $\Delta\boldsymbol{V}_{(i),\tau}\boldsymbol{V}_{(i),\tau}^T = \boldsymbol{V}_{(i),\tau}\boldsymbol{V}_{(i),\tau}^T - \hat{\boldsymbol{V}}_{(i),\tau}\hat{\boldsymbol{V}}_{(i),\tau} - \hat{\boldsymbol{V}}_{(i),\tau}\Delta\boldsymbol{V}_{(i),\tau}^T$. Therefore,

$$
\begin{aligned}
&\left\langle \left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}_\tau}-\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\hat{\boldsymbol{F}}_{(i)}, \Delta\boldsymbol{V}_{(i),\tau}\boldsymbol{V}_{(i),\tau}^T\hat{\boldsymbol{F}}_{(i)}\right\rangle \\
&= \left\langle \left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}_\tau}-\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\hat{\boldsymbol{F}}_{(i)}, \left(\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}-\hat{\boldsymbol{\Pi}}_{(i)}-\hat{\boldsymbol{V}}_{(i),\tau}\Delta\boldsymbol{V}_{(i),\tau}^T\right)\hat{\boldsymbol{F}}_{(i)}\right\rangle \\
&= \left\langle \left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}_\tau}-\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\hat{\boldsymbol{F}}_{(i)}, \left(\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}-\hat{\boldsymbol{\Pi}}_{(i)}\right)\hat{\boldsymbol{F}}_{(i)}\right\rangle \\
&- \left\langle \left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}_\tau}-\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\hat{\boldsymbol{F}}_{(i)}, \left(\hat{\boldsymbol{V}}_{(i),\tau}\Delta\boldsymbol{V}_{(i),\tau}^T\right)\hat{\boldsymbol{F}}_{(i)}\right\rangle \\
&= \left\langle \left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}_\tau}-\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\hat{\boldsymbol{F}}_{(i)}, \left(\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}-\hat{\boldsymbol{\Pi}}_{(i)}\right)\hat{\boldsymbol{F}}_{(i)}\right\rangle + \boldsymbol{\epsilon}_{1,(i),\tau}
\end{aligned}
$$

where $\boldsymbol{\epsilon}_{1,(i),\tau}$ is defined as $\boldsymbol{\epsilon}_{1,(i),\tau} = -\left\langle \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\hat{\boldsymbol{F}}_{(i)}, \left(\hat{\boldsymbol{V}}_{(i),\tau}\Delta\boldsymbol{V}_{(i),\tau}^T\right)\hat{\boldsymbol{F}}_{(i)}\right\rangle$. Its norm is upper bounded by

$$
\begin{aligned}
&\left|\boldsymbol{\epsilon}_{1,(i),\tau}\right| \\
&= \left|\mathrm{Tr}\left(\hat{\boldsymbol{F}}_{(i)}^T\left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\hat{\boldsymbol{V}}_{(i),\tau}\Delta\boldsymbol{V}_{(i),\tau}^T\hat{\boldsymbol{F}}_{(i)}\right)\right| \\
&= \left|\mathrm{Tr}\left(\hat{\boldsymbol{F}}_{(i)}^T\left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\Delta\boldsymbol{V}_{(i),\tau}\Delta\boldsymbol{V}_{(i),\tau}^T\hat{\boldsymbol{F}}_{(i)}\right)\right| \\
&= \left|\mathrm{Tr}\left(\hat{\boldsymbol{F}}_{(i)}^T\left(\hat{\boldsymbol{\Pi}}_g + \hat{\boldsymbol{\Pi}}_{(i)} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\Delta\boldsymbol{V}_{(i),\tau}\Delta\boldsymbol{V}_{(i),\tau}^T\hat{\boldsymbol{F}}_{(i)}\right)\right| \\
&\leq \left\|\hat{\boldsymbol{F}}_{(i)}^T\left(\hat{\boldsymbol{\Pi}}_g + \hat{\boldsymbol{\Pi}}_{(i)} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\right\|_F \left\|\Delta\boldsymbol{V}_{(i),\tau}\Delta\boldsymbol{V}_{(i),\tau}^T\hat{\boldsymbol{F}}_{(i)}\right\|_F \\
&\leq \left\|\hat{\boldsymbol{\Pi}}_g + \hat{\boldsymbol{\Pi}}_{(i)} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right\|_F \left\|\Delta\boldsymbol{V}_{(i),\tau}\right\|_F^2 \left\|\hat{\boldsymbol{F}}_{(i)}\right\|^2 \\
&\leq 4\sqrt{2}\sqrt{\widetilde{\zeta}_{(i),\tau}}\zeta_{(i),\tau}G_{max,op}
\end{aligned}
$$

For `Term IV`, we have,

$$
\begin{aligned}
&\left\langle \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\boldsymbol{R}_{(i)}, \Delta\boldsymbol{V}_{(i),\tau}\boldsymbol{V}_{(i),\tau}^T\boldsymbol{R}_{(i)}\right\rangle \\
&= \left\langle \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\boldsymbol{R}_{(i)}, \Delta\boldsymbol{V}_{(i),\tau}\Delta\boldsymbol{V}_{(i),\tau}^T\boldsymbol{R}_{(i)}\right\rangle \\
&= \left\langle \boldsymbol{R}_{(i)}, \Delta\boldsymbol{V}_{(i),\tau}\Delta\boldsymbol{V}_{(i),\tau}^T\boldsymbol{R}_{(i)}\right\rangle + \boldsymbol{\epsilon}_{2,(i),\tau}
\end{aligned}
$$

where $\boldsymbol{\epsilon}_{2,(i),\tau}$ is defined as

$$
\boldsymbol{\epsilon}_{2,(i),\tau} = -\left\langle \left(\boldsymbol{P}_{\boldsymbol{U}_\tau} + \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\boldsymbol{R}_{(i)}, \Delta\boldsymbol{V}_{(i),\tau}\Delta\boldsymbol{V}_{(i),\tau}^T\boldsymbol{R}_{(i)}\right\rangle
$$

and its norm is upper bounded by,

$$
\begin{aligned}
&\left|\boldsymbol{\epsilon}_{2,(i),\tau}\right|_F \\
&= \left|\mathrm{Tr}\left(\boldsymbol{R}_{(i)}^T\left(\boldsymbol{P}_{\boldsymbol{U}_\tau} + \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\Delta\boldsymbol{V}_{(i),\tau}\Delta\boldsymbol{V}_{(i),\tau}^T\boldsymbol{R}_{(i)}\right)\right| \\
&= \left|\mathrm{Tr}\left(\boldsymbol{R}_{(i)}^T\left(\boldsymbol{P}_{\boldsymbol{U}_\tau} + \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} - \hat{\boldsymbol{\Pi}}_g - \hat{\boldsymbol{\Pi}}_{(i)}\right)\left(\boldsymbol{P}_{\boldsymbol{U}_\tau} + \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\Delta\boldsymbol{V}_{(i),\tau}\Delta\boldsymbol{V}_{(i),\tau}^T\boldsymbol{R}_{(i)}\right)\right| \\
&\leq \left\|\boldsymbol{R}_{(i)}^T\left(\boldsymbol{P}_{\boldsymbol{U}_\tau} + \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} - \hat{\boldsymbol{\Pi}}_g - \hat{\boldsymbol{\Pi}}_{(i)}\right)\right\|_F \left\|\left(\boldsymbol{P}_{\boldsymbol{U}_\tau} + \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\Delta\boldsymbol{V}_{(i),\tau}\Delta\boldsymbol{V}_{(i),\tau}^T\boldsymbol{R}_{(i)}\right\|_F \\
&\leq \left\|\boldsymbol{P}_{\boldsymbol{U}_\tau} + \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} - \hat{\boldsymbol{\Pi}}_g - \hat{\boldsymbol{\Pi}}_{(i)}\right\|_F \left\|\Delta\boldsymbol{V}_{(i),\tau}\right\|_F^2 \left\|\boldsymbol{R}_{(i)}\right\|^2 \\
&\leq 2\sqrt{2}\sqrt{\widetilde{\zeta}_{(i),\tau}}\zeta_{(i),\tau}G_{max,op}
\end{aligned}
$$

For the `Term V`,

$$
\begin{aligned}
&\left\langle \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\hat{\boldsymbol{F}}_{(i)}, \Delta\boldsymbol{V}_{(i),\tau}\boldsymbol{V}_{(i),\tau}^T\boldsymbol{R}_{(i)}\right\rangle \\
&= \left\langle \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\hat{\boldsymbol{F}}_{(i)}, \Delta\boldsymbol{V}_{(i),\tau}\Delta\boldsymbol{V}_{(i),\tau}^T\boldsymbol{R}_{(i)}\right\rangle \\
&= \boldsymbol{\epsilon}_{3,(i),\tau}
\end{aligned}
$$

where the norm of $\boldsymbol{\epsilon}_{3,(i),\tau}$ is upper bounded by,

$$\begin{aligned}
&\left|\boldsymbol{\epsilon}_{3,(i),\tau}\right| \\
&= \left|\mathrm{Tr}\left(\hat{\boldsymbol{F}}_{(i)}^T\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}_\tau}-\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\Delta\boldsymbol{V}_{(i),\tau}\Delta\boldsymbol{V}_{(i),\tau}^T\boldsymbol{R}_{(i)}\right)\right| \\
&= \left|\mathrm{Tr}\left(\hat{\boldsymbol{F}}_{(i)}^T\left(\hat{\boldsymbol{\Pi}}_g+\hat{\boldsymbol{\Pi}}_{(i)}-\boldsymbol{P}_{\boldsymbol{U}_\tau}-\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\Delta\boldsymbol{V}_{(i),\tau}\Delta\boldsymbol{V}_{(i),\tau}^T\boldsymbol{R}_{(i)}\right)\right| \\
&\leq \left\|\hat{\boldsymbol{F}}_{(i)}^T\left(\hat{\boldsymbol{\Pi}}_g+\hat{\boldsymbol{\Pi}}_{(i)}-\boldsymbol{P}_{\boldsymbol{U}_\tau}-\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\right\|_F\left\|\Delta\boldsymbol{V}_{(i),\tau}\Delta\boldsymbol{V}_{(i),\tau}^T\boldsymbol{R}_{(i)}\right\|_F \\
&\leq \left\|\boldsymbol{P}_{\boldsymbol{U}_\tau}+\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}-\hat{\boldsymbol{\Pi}}_g-\hat{\boldsymbol{\Pi}}_{(i)}\right\|_F\left\|\Delta\boldsymbol{V}_{(i),\tau}\right\|_F^2\left\|\boldsymbol{R}_{(i)}\right\|\left\|\hat{\boldsymbol{F}}_{(i)}\right\| \\
&\leq 4\sqrt{2}\sqrt{\widetilde{\zeta}_{(i),\tau}}\zeta_{(i),\tau}G_{max,op}
\end{aligned}$$

For `Term VI`,

$$\begin{aligned}
&\left\langle\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}_\tau}-\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\boldsymbol{R}_{(i)},\Delta\boldsymbol{V}_{(i),\tau}\boldsymbol{V}_{(i),\tau}^T\hat{\boldsymbol{F}}_{(i)}\right\rangle \\
&=\left\langle\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}_\tau}-\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\boldsymbol{R}_{(i)},\Delta\boldsymbol{V}_{(i),\tau}\Delta\boldsymbol{V}_{(i),\tau}^T\hat{\boldsymbol{F}}_{(i)}\right\rangle \\
&=\left\langle\boldsymbol{R}_{(i)},\Delta\boldsymbol{V}_{(i),\tau}\Delta\boldsymbol{V}_{(i),\tau}^T\hat{\boldsymbol{F}}_{(i)}\right\rangle+\boldsymbol{\epsilon}_{4,(i),\tau}
\end{aligned}$$

where $\boldsymbol{\epsilon}_{4,(i),\tau}$ is defined as

$$\boldsymbol{\epsilon}_{4,(i),\tau}=-\left\langle\left(\boldsymbol{P}_{\boldsymbol{U}_\tau}+\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\boldsymbol{R}_{(i)},\Delta\boldsymbol{V}_{(i),\tau}\Delta\boldsymbol{V}_{(i),\tau}^T\hat{\boldsymbol{F}}_{(i)}\right\rangle$$

Its norm is upper bounded by,

$$\begin{aligned}
&\left|\boldsymbol{\epsilon}_{4,(i),\tau}\right| \\
&= \left|\mathrm{Tr}\left(\boldsymbol{R}_{(i)}^T\left(\boldsymbol{P}_{\boldsymbol{U}_\tau}+\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\Delta\boldsymbol{V}_{(i),\tau}\Delta\boldsymbol{V}_{(i),\tau}^T\hat{\boldsymbol{F}}_{(i)}\right)\right| \\
&= \left|\mathrm{Tr}\left(\boldsymbol{R}_{(i)}^T\left(-\boldsymbol{\Pi}_g-\boldsymbol{\Pi}_{(i)}+\boldsymbol{P}_{\boldsymbol{U}_\tau}+\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\Delta\boldsymbol{V}_{(i),\tau}\Delta\boldsymbol{V}_{(i),\tau}^T\hat{\boldsymbol{F}}_{(i)}\right)\right| \\
&\leq \left\|\Delta\boldsymbol{V}_{(i),\tau}\right\|_F^2\left\|\boldsymbol{R}_{(i)}\right\|\left\|\hat{\boldsymbol{F}}_{(i)}\right\|\left\|-\boldsymbol{\Pi}_g-\boldsymbol{\Pi}_{(i)}+\boldsymbol{P}_{\boldsymbol{U}_\tau}+\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right\|_F \\
&\leq 4\sqrt{2}\sqrt{\widetilde{\zeta}_{(i),\tau}}\zeta_{(i),\tau}G_{max,op}
\end{aligned}$$

Combining these terms, we have,

$$\begin{aligned}
&\left\langle\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}_\tau}-\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\boldsymbol{S}_{(i)}\boldsymbol{V}_{(i),\tau},\Delta\boldsymbol{V}_{(i),\tau}\right\rangle \\
&=\left\langle\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}_\tau}-\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\hat{\boldsymbol{F}}_{(i)},\left(\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}-\hat{\boldsymbol{\Pi}}_{(i)}\right)\hat{\boldsymbol{F}}_{(i)}\right\rangle+\left\langle\boldsymbol{R}_{(i)},\Delta\boldsymbol{V}_{(i),\tau}\Delta\boldsymbol{V}_{(i),\tau}^T\boldsymbol{R}_{(i)}\right\rangle \\
&+\left\langle\boldsymbol{R}_{(i)},\Delta\boldsymbol{V}_{(i),\tau}\Delta\boldsymbol{V}_{(i),\tau}^T\hat{\boldsymbol{F}}_{(i)}\right\rangle \\
&+\boldsymbol{\epsilon}_{1,(i),\tau}+\boldsymbol{\epsilon}_{2,(i),\tau}+\boldsymbol{\epsilon}_{3,(i),\tau}+\boldsymbol{\epsilon}_{4,(i),\tau}
\end{aligned}\tag{80}$$

Similarly, we can calculate the inner product term,

$$\begin{aligned}
&\sum_{i=1}^{N} \left\langle \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} \right) \boldsymbol{S}_{(i)} \boldsymbol{U}_\tau, \Delta \boldsymbol{U}_\tau \right\rangle \\
&= \sum_{i=1}^{N} \left\langle \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} \right) \boldsymbol{F}_{(i)}, \Delta \boldsymbol{U}_\tau \boldsymbol{U}_\tau^T \boldsymbol{F}_{(i)} \right\rangle \\
&= \sum_{i=1}^{N} \underbrace{\left\langle \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} \right) \hat{\boldsymbol{F}}_{(i)}, \Delta \boldsymbol{U}_\tau \boldsymbol{U}_\tau^T \hat{\boldsymbol{F}}_{(i)} \right\rangle}_{\texttt{Term VII}} + \underbrace{\left\langle \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} \right) \boldsymbol{R}_{(i)}, \Delta \boldsymbol{U}_\tau \boldsymbol{U}_\tau^T \boldsymbol{R}_{(i)} \right\rangle}_{\texttt{Term VIII}} \\
&+ \underbrace{\left\langle \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} \right) \hat{\boldsymbol{F}}_{(i)}, \Delta \boldsymbol{U}_\tau \boldsymbol{U}_\tau^T \boldsymbol{R}_{(i)} \right\rangle}_{\texttt{Term IX}} + \underbrace{\left\langle \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} \right) \boldsymbol{R}_{(i)}, \Delta \boldsymbol{U}_\tau \boldsymbol{U}_\tau^T \hat{\boldsymbol{F}}_{(i)} \right\rangle}_{\texttt{Term X}}
\end{aligned}$$

For the `Term VII`, we can simplify it as,

$$\begin{aligned}
&\left\langle \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} \right) \hat{\boldsymbol{F}}_{(i)}, \Delta \boldsymbol{U}_\tau \boldsymbol{U}_\tau^T \hat{\boldsymbol{F}}_{(i)} \right\rangle \\
&= \left\langle \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} \right) \hat{\boldsymbol{F}}_{(i)}, \left(\boldsymbol{P}_{\boldsymbol{U}_\tau} - \hat{\boldsymbol{\Pi}}_g \right) \hat{\boldsymbol{F}}_{(i)} \right\rangle + \boldsymbol{\epsilon}_{5,(i),\tau}
\end{aligned}$$

where $\boldsymbol{\epsilon}_{5,(i),\tau}$ is defined as,

$$\begin{aligned}
\boldsymbol{\epsilon}_{5,(i),\tau} &= -\left\langle \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} \right) \hat{\boldsymbol{F}}_{(i)}, \hat{\boldsymbol{U}}_\tau \Delta \boldsymbol{U}_\tau^T \hat{\boldsymbol{F}}_{(i)} \right\rangle \\
&= \left\langle \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} \right) \hat{\boldsymbol{F}}_{(i)}, \Delta \boldsymbol{U}_\tau \Delta \boldsymbol{U}_\tau^T \hat{\boldsymbol{F}}_{(i)} \right\rangle
\end{aligned}$$

Its norm is upper bounded by

$$\begin{aligned}
&\left| \boldsymbol{\epsilon}_{5,(i),\tau} \right|_F \\
&= \left| \mathrm{Tr} \left(\boldsymbol{F}_{(i)}^T \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} \right) \Delta \boldsymbol{U}_\tau \Delta \boldsymbol{U}_\tau^T \hat{\boldsymbol{F}}_{(i)} \right) \right| \\
&= \left| \mathrm{Tr} \left(\hat{\boldsymbol{F}}_{(i)}^T \left(\hat{\boldsymbol{\Pi}}_g + \hat{\boldsymbol{\Pi}}_{(i)} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} \right) \Delta \boldsymbol{U}_\tau \Delta \boldsymbol{U}_\tau^T \hat{\boldsymbol{F}}_{(i)} \right) \right| \\
&\leq \left\| \hat{\boldsymbol{F}}_{(i)}^T \left(\hat{\boldsymbol{\Pi}}_g + \hat{\boldsymbol{\Pi}}_{(i)} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} \right) \right\|_F \left\| \Delta \boldsymbol{U}_\tau \Delta \boldsymbol{U}_\tau^T \hat{\boldsymbol{F}}_{(i)} \right\|_F \\
&\leq \left\| \hat{\boldsymbol{\Pi}}_g + \hat{\boldsymbol{\Pi}}_{(i)} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} \right\|_F \| \Delta \boldsymbol{U}_\tau \|_F^2 \left\| \hat{\boldsymbol{F}}_{(i)} \right\|^2 \\
&\leq 4\sqrt{2} \sqrt{\widetilde{\zeta}_{(i),\tau}} \zeta_{(0),\tau} G_{max,op}
\end{aligned}$$

For `Term VIII`, also we have,

$$\begin{aligned}
&\left\langle \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} \right) \boldsymbol{R}_{(i)}, \Delta \boldsymbol{U}_\tau \boldsymbol{U}_\tau^T \boldsymbol{R}_{(i)} \right\rangle \\
&\left\langle \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} \right) \boldsymbol{R}_{(i)}, \Delta \boldsymbol{U}_\tau \Delta \boldsymbol{U}_\tau^T \boldsymbol{R}_{(i)} \right\rangle \\
&= \left\langle \boldsymbol{R}_{(i)}, \Delta \boldsymbol{U}_\tau \Delta \boldsymbol{U}_\tau^T \boldsymbol{R}_{(i)} \right\rangle + \boldsymbol{\epsilon}_{6,(i),\tau}
\end{aligned}$$

where $\boldsymbol{\epsilon}_{6,(i),\tau}$ is defined as,

$$\boldsymbol{\epsilon}_{6,(i),\tau} = -\left\langle \left(\boldsymbol{P}_{\boldsymbol{U}_\tau} + \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \boldsymbol{R}_{(i)}, \Delta\boldsymbol{U}_\tau \Delta\boldsymbol{U}_\tau^T \boldsymbol{R}_{(i)} \right\rangle$$

and its norm is upper bounded by,

$$\begin{aligned}
&\left|\boldsymbol{\epsilon}_{6,(i),\tau}\right|_F \\
&= \left|\mathrm{Tr}\left(\boldsymbol{R}_{(i)}^T \left(\boldsymbol{P}_{\boldsymbol{U}_\tau} + \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \Delta\boldsymbol{U}_\tau \Delta\boldsymbol{U}_\tau^T \boldsymbol{R}_{(i)}\right)\right| \\
&= \left|\mathrm{Tr}\left(\boldsymbol{R}_{(i)}^T \left(\boldsymbol{P}_{\boldsymbol{U}_\tau} + \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} - \hat{\boldsymbol{\Pi}}_g - \hat{\boldsymbol{\Pi}}_{(i)}\right) \left(\boldsymbol{P}_{\boldsymbol{U}_\tau} + \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \Delta\boldsymbol{U}_\tau \Delta\boldsymbol{U}_\tau^T \boldsymbol{R}_{(i)}\right)\right| \\
&\leq \left\|\boldsymbol{R}_{(i)}^T \left(\boldsymbol{P}_{\boldsymbol{U}_\tau} + \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} - \hat{\boldsymbol{\Pi}}_g - \hat{\boldsymbol{\Pi}}_{(i)}\right)\right\|_F \left\|\left(\boldsymbol{P}_{\boldsymbol{U}_\tau} + \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \Delta\boldsymbol{U}_\tau \Delta\boldsymbol{U}_\tau^T \boldsymbol{R}_{(i)}\right\|_F \\
&\leq \left\|\boldsymbol{P}_{\boldsymbol{U}_\tau} + \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} - \hat{\boldsymbol{\Pi}}_g - \hat{\boldsymbol{\Pi}}_{(i)}\right\|_F \left\|\Delta\boldsymbol{U}_\tau\right\|_F^2 \left\|\boldsymbol{R}_{(i)}\right\|^2 \\
&\leq 2\sqrt{2}\sqrt{\widetilde{\zeta}_{(i),\tau}}\zeta_{(0),\tau} G_{max,op}
\end{aligned}$$

For `Term IX`, we have,

$$\begin{aligned}
&\left\langle \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \hat{\boldsymbol{F}}_{(i)}, \Delta\boldsymbol{U}_\tau \boldsymbol{U}_\tau^T \boldsymbol{R}_{(i)} \right\rangle \\
&= \left\langle \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \hat{\boldsymbol{F}}_{(i)}, \Delta\boldsymbol{U}_\tau \Delta\boldsymbol{U}_\tau^T \boldsymbol{R}_{(i)} \right\rangle \\
&= \boldsymbol{\epsilon}_{7,(i),\tau}
\end{aligned}$$

where the norm of $\boldsymbol{\epsilon}_{7,(i),\tau}$ is upper bounded by,

$$\begin{aligned}
&\left|\boldsymbol{\epsilon}_{7,(i),\tau}\right| \\
&= \left|\mathrm{Tr}\left(\hat{\boldsymbol{F}}_{(i)}^T \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \Delta\boldsymbol{U}_\tau \Delta\boldsymbol{U}_\tau^T \boldsymbol{R}_{(i)}\right)\right| \\
&= \left|\mathrm{Tr}\left(\hat{\boldsymbol{F}}_{(i)}^T \left(\hat{\boldsymbol{\Pi}}_g + \hat{\boldsymbol{\Pi}}_{(i)} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \Delta\boldsymbol{U}_\tau \Delta\boldsymbol{U}_\tau^T \boldsymbol{R}_{(i)}\right)\right| \\
&\leq \left\|\hat{\boldsymbol{F}}_{(i)}^T \left(\hat{\boldsymbol{\Pi}}_g + \hat{\boldsymbol{\Pi}}_{(i)} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right)\right\|_F \left\|\Delta\boldsymbol{U}_\tau \Delta\boldsymbol{U}_\tau^T \boldsymbol{R}_{(i)}\right\|_F \\
&\leq \left\|\hat{\boldsymbol{\Pi}}_g + \hat{\boldsymbol{\Pi}}_{(i)} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right\|_F \left\|\Delta\boldsymbol{U}_\tau\right\|_F^2 \left\|\boldsymbol{R}_{(i)}\right\| \left\|\hat{\boldsymbol{F}}_{(i)}\right\| \\
&\leq 4\sqrt{2}\sqrt{\widetilde{\zeta}_{(i),\tau}}\zeta_{(0),\tau} G_{max,op}
\end{aligned}$$

Finally, for `Term X`, we have,

$$\begin{aligned}
&\sum_{i=1}^N \left\langle \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \boldsymbol{R}_{(i)}, \Delta\boldsymbol{U}_\tau \boldsymbol{U}_\tau^T \hat{\boldsymbol{F}}_{(i)} \right\rangle \\
&= \sum_{i=1}^N \left\langle \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \boldsymbol{R}_{(i)}, \Delta\boldsymbol{U}_\tau \hat{\boldsymbol{U}}_\tau^T \hat{\boldsymbol{F}}_{(i)} \right\rangle + \left\langle \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \boldsymbol{R}_{(i)}, \Delta\boldsymbol{U}_\tau \Delta\boldsymbol{U}_\tau^T \hat{\boldsymbol{F}}_{(i)} \right\rangle \\
&= \sum_{i=1}^N \left\langle \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \boldsymbol{R}_{(i)}, \Delta\boldsymbol{U}_\tau \Delta\boldsymbol{U}_\tau^T \hat{\boldsymbol{F}}_{(i)} \right\rangle - \left\langle \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} \boldsymbol{R}_{(i)}, \Delta\boldsymbol{U}_\tau \hat{\boldsymbol{U}}_\tau \hat{\boldsymbol{F}}_{(i)} \right\rangle \\
&= \sum_{i=1}^N \left\langle \boldsymbol{R}_{(i)}, \Delta\boldsymbol{U}_\tau \Delta\boldsymbol{U}_\tau^T \hat{\boldsymbol{F}}_{(i)} \right\rangle + \boldsymbol{\epsilon}_{8,(i),\tau}
\end{aligned}$$

where $\boldsymbol{\epsilon}_{8,(i),\tau}$ is defined as,

$$\boldsymbol{\epsilon}_{8,(i),\tau} = \left\langle \left(-\boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \boldsymbol{R}_{(i)}, \Delta\boldsymbol{U}_\tau \Delta\boldsymbol{U}_\tau^T \hat{\boldsymbol{F}}_{(i)} \right\rangle - \left\langle \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} \boldsymbol{R}_{(i)}, \Delta\boldsymbol{U}_\tau \hat{\boldsymbol{U}}_\tau \hat{\boldsymbol{F}}_{(i)} \right\rangle$$

Its norm is upper bounded by

$$\begin{aligned}
&\left|\boldsymbol{\epsilon}_{8,(i),\tau}\right| \\
&\leq \left|\mathrm{Tr}\left(\boldsymbol{R}_{(i)}^T \left(-\boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \Delta\boldsymbol{U}_\tau \Delta\boldsymbol{U}_\tau^T \hat{\boldsymbol{F}}_{(i)}\right)\right| + \left|\mathrm{Tr}\left(\boldsymbol{R}_{(i)}^T \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} \Delta\boldsymbol{U}_\tau \hat{\boldsymbol{U}}_\tau \hat{\boldsymbol{F}}_{(i)}\right)\right| \\
&\leq \left|\mathrm{Tr}\left(\boldsymbol{R}_{(i)}^T \left(-\boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \Delta\boldsymbol{U}_\tau \Delta\boldsymbol{U}_\tau^T \hat{\boldsymbol{F}}_{(i)}\right)\right| \\
&+ \left|\mathrm{Tr}\left(\boldsymbol{R}_{(i)}^T \left(\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} - \hat{\boldsymbol{\Pi}}_{(i)}\right) \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} \Delta\boldsymbol{U}_\tau \hat{\boldsymbol{U}}_\tau \hat{\boldsymbol{F}}_{(i)}\right)\right| \\
&\leq 4\sqrt{2}\sqrt{\widetilde{\zeta}_{(i),\tau}}\zeta_{(0),\tau} G_{max,op} + \left\|\boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} - \hat{\boldsymbol{\Pi}}_{(i)}\right\|_F \left\|\Delta\boldsymbol{U}_\tau\right\|_F \left\|\boldsymbol{R}_{(i)}\right\| \left\|\hat{\boldsymbol{F}}_{(i)}\right\|
\end{aligned}$$

Combining them, we have,

$$\begin{aligned}
&\sum_{i=1}^N \left\langle \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \boldsymbol{S}_{(i)} \boldsymbol{U}_\tau, \Delta\boldsymbol{U}_\tau \right\rangle \\
&= \sum_{i=1}^N \left\langle \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \hat{\boldsymbol{F}}_{(i)}, \left(\boldsymbol{P}_{\boldsymbol{U}_\tau} - \hat{\boldsymbol{\Pi}}_g\right) \hat{\boldsymbol{F}}_{(i)} \right\rangle + \left\langle \boldsymbol{R}_{(i)}, \Delta\boldsymbol{U}_\tau \Delta\boldsymbol{U}_\tau^T \boldsymbol{R}_{(i)} \right\rangle \\
&+ \left\langle \boldsymbol{R}_{(i)}, \Delta\boldsymbol{U}_\tau \Delta\boldsymbol{U}_\tau^T \hat{\boldsymbol{F}}_{(i)} \right\rangle \\
&+ \boldsymbol{\epsilon}_{5,(i),\tau} + \boldsymbol{\epsilon}_{6,(i),\tau} + \boldsymbol{\epsilon}_{7,(i),\tau} + \boldsymbol{\epsilon}_{8,(i),\tau}
\end{aligned} \tag{81}$$

Comparing (79), (80), and (81), we know that,

$$\begin{aligned}
&-\left\langle \sum_{i=1}^N \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \boldsymbol{S}_{(i)} \boldsymbol{U}_\tau, \Delta\boldsymbol{U}_\tau \right\rangle - \sum_{i=1}^N \left\langle \left(\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{U}_\tau} - \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}}\right) \boldsymbol{S}_{(i)} \boldsymbol{V}_{(i),\tau}, \Delta\boldsymbol{V}_{(i),\tau} \right\rangle \\
&= 2\phi_\tau - \sum_{i=1}^N \mathrm{Tr}\left(\boldsymbol{R}_{(i)}^T \left(\Delta\boldsymbol{U}_\tau \Delta\boldsymbol{U}_\tau^T + \Delta\boldsymbol{V}_{(i),\tau} \Delta\boldsymbol{V}_{(i),\tau}^T\right) \boldsymbol{F}_{(i)}\right) - \sum_{i=1}^N \sum_{\alpha=1}^8 \boldsymbol{\epsilon}_{\alpha,(i),\tau}
\end{aligned}$$

From the estimated upper bounds of $\left|\boldsymbol{\epsilon}_{1,(i),\tau}\right|$ to $\left|\boldsymbol{\epsilon}_{8,(i),\tau}\right|$, we know that,

$$\begin{aligned}
&\left|\sum_{i=1}^N \mathrm{Tr}\left(\boldsymbol{R}_{(i)}^T \left(\Delta\boldsymbol{U}_\tau \Delta\boldsymbol{U}_\tau^T + \Delta\boldsymbol{V}_{(i),\tau} \Delta\boldsymbol{V}_{(i),\tau}^T\right) \boldsymbol{F}_{(i)}\right)\right| + \left|\sum_{i=1}^N \sum_{\alpha=1}^8 \boldsymbol{\epsilon}_{\alpha,(i),\tau}\right|_F \\
&\leq 2\sum_{i=1}^N \left(\left\|\Delta\boldsymbol{U}_\tau\right\|_F^2 + \left\|\Delta\boldsymbol{V}_{(i),\tau}\right\|_F^2\right) \left\|\boldsymbol{R}_{(i)}\right\| \left\|\hat{\boldsymbol{F}}_{(i)}\right\| + 14\sqrt{2} G_{max,op} \sqrt{\widetilde{\zeta}_{(i),\tau}} \left(\zeta_{(i),\tau} + \zeta_{(0),\tau}\right)
\end{aligned}$$

From Lemma 18, we know that $\left\|\boldsymbol{R}_{(i)}\right\|\left\|\hat{\boldsymbol{F}}_{(i)}\right\| \leq \frac{\hat{\mu}\hat{\theta}}{64}$, we can thus upper bound the first term as,

$$\begin{aligned}
&2\sum_{i=1}^{N}\left(\|\Delta \boldsymbol{U}_\tau\|_F^2+\left\|\Delta \boldsymbol{V}_{(i),\tau}\right\|_F^2\right)\left\|\boldsymbol{R}_{(i)}\right\|\left\|\hat{\boldsymbol{F}}_{(i)}\right\| \\
&\leq \sum_{i=1}^{N}\left(\|\Delta \boldsymbol{U}_\tau\|_F^2+\left\|\Delta \boldsymbol{V}_{(i),\tau}\right\|_F^2\right)\frac{\hat{\mu}\hat{\theta}}{32} \\
&\leq \phi_\tau/2
\end{aligned}$$

where the last inequality comes from Lemma 19.

Also, the second term can be bounded as,

$$\begin{aligned}
&14\sqrt{2}G_{max,op}\sum_{i=1}^{N}\sqrt{\widetilde{\zeta}_{(i),\tau}}\left(\zeta_{(i),\tau}+\zeta_{(0),\tau}\right) \\
&\leq 14\sqrt{2}G_{max,op}\sum_{i=1}^{N}\sqrt{\widetilde{\zeta}_{(i),\tau}}\sqrt{\zeta_{(i),\tau}+\zeta_{(0),\tau}}\sqrt{\sum_{j=1}^{N}\zeta_{(j),\tau}+\zeta_{(0),\tau}} \\
&\leq 14\sqrt{2}G_{max,op}\sqrt{\sum_{i=1}^{N}\widetilde{\zeta}_{(i),\tau}}\sqrt{\sum_{i=1}^{N}\zeta_{(i),\tau}+\zeta_{(0),\tau}}\sqrt{\sum_{j=1}^{N}\zeta_{(j),\tau}+\zeta_{(0),\tau}} \\
&\leq 14\sqrt{2}G_{max,op}\left(\sum_{i=1}^{N}\zeta_{(i),\tau}+\zeta_{(0),\tau}\right)^{1.5} \\
&\leq 14\sqrt{2}G_{max,op}\left(2\sum_{i=1}^{N}\|\Delta \boldsymbol{U}_\tau\|_F^2+\left\|\Delta \boldsymbol{V}_{(i),\tau}\right\|_F^2\right)^{1.5} \\
&\leq 56G_{max,op}\left(\frac{\phi_\tau}{\frac{\hat{\mu}\hat{\theta}}{16}}\right)^{3/2} \leq \phi_\tau/2
\end{aligned}$$

where the second inequality comes from Cauchy-Schwrtz inequality, the third inequality comes from Lemma 13, the fourth inequality comes from Lemma 26, the fifth inequality comes from Lemma 19, and the last inequality comes from the fact that $\phi_\tau \leq \frac{\hat{\mu}^3\hat{\theta}^3}{51380224G_{max,op}^2}$.

This completes the proof.

■

Combining Lemma 19 with Lemma 20, we can prove the following PL-inequality.

Lemma 21 *(Lemma 12 in the main paper) Under the same conditions as Theorem 10, we have*

$$N\|\square \boldsymbol{U}_\tau\|_F^2+\sum_{i=1}^{N}\left\|\square \boldsymbol{V}_{(i),\tau}\right\|_F^2 \geq \frac{\hat{\theta}\hat{\mu}}{16}\phi_\tau$$

Proof From Cauchy-Schwartz inequality, we know that,

$$\begin{aligned}
&- N \langle \square \boldsymbol{U}_\tau, \Delta \boldsymbol{U}_\tau \rangle - \sum_{i=1}^{N} \left\langle \square \boldsymbol{V}_{(i),\tau}, \Delta \boldsymbol{V}_{(i),\tau} \right\rangle \\
&\leq N \left\| \square \boldsymbol{U}_\tau \right\|_F \left\| \Delta \boldsymbol{U}_\tau \right\| + \sum_{i=1}^{N} \left\| \square \boldsymbol{V}_{(i),\tau} \right\|_F \left\| \Delta \boldsymbol{V}_{(i),\tau} \right\|_F \\
&\leq \sqrt{N \left\| \square \boldsymbol{U}_\tau \right\|_F^2 + \sum_{i=1}^{N} \left\| \square \boldsymbol{V}_{(i),\tau} \right\|_F^2} \sqrt{N \left\| \Delta \boldsymbol{U}_\tau \right\|_F^2 + \sum_{i=1}^{N} \left\| \Delta \boldsymbol{V}_{(i),\tau} \right\|_F^2} \\
&\leq \sqrt{N \left\| \square \boldsymbol{U}_\tau \right\|_F^2 + \sum_{i=1}^{N} \left\| \square \boldsymbol{V}_{(i),\tau} \right\|_F^2} \sqrt{\frac{\phi_\tau}{\frac{\hat{\mu}\hat{\theta}}{16}}}
\end{aligned}$$

where the last inequality comes from Lemma 19.

From Lemma 20, we know,

$$\begin{aligned}
&- N \langle \square \boldsymbol{U}_\tau, \Delta \boldsymbol{U}_\tau \rangle - \sum_{i=1}^{N} \left\langle \square \boldsymbol{V}_{(i),\tau}, \Delta \boldsymbol{V}_{(i),\tau} \right\rangle \\
&\geq \phi_\tau
\end{aligned}$$

Combining them, we have,

$$N \left\| \square \boldsymbol{U}_\tau \right\|_F^2 + \sum_{i=1}^{N} \left\| \square \boldsymbol{V}_{(i),\tau} \right\|_F^2 \geq \frac{\hat{\theta}\hat{\mu}}{16} \phi_\tau$$

■

Finally, we come to the proof of Theorem 10:

Proof Combining Lemma 21 with equation (64), we know:

$$-f\left(\boldsymbol{U}_{\tau+1}, \{\boldsymbol{V}_{(i),\tau+1}\}\right) \leq -f\left(\boldsymbol{U}_\tau, \{\boldsymbol{V}_{(i),\tau}\}\right) - \frac{\eta}{2} \frac{\hat{\mu}\hat{\theta}}{16}$$

We add $f^\star$ on both sides. Since $\phi_\tau = f^\star - -f\left(\boldsymbol{U}_\tau, \{\boldsymbol{V}_{(i),\tau}\}\right)$, we have:

$$\begin{aligned}
&\phi_{\tau+1} \\
&\leq \phi_\tau - \frac{\eta}{2} \frac{\hat{\mu}\hat{\theta}}{16} \phi_\tau \\
&= \left(1 - \frac{\eta\theta\mu}{64}\right) \phi_\tau
\end{aligned}$$

Thus ϕ_τ decreases linearly with τ. From Lemma 19 and Lemma 26, we can show $\left\| \boldsymbol{P}_{\boldsymbol{U}_\tau} - \hat{\boldsymbol{\Pi}}_g \right\|_F$ and $\left\| \boldsymbol{P}_{\boldsymbol{V}_{(i),\tau}} - \hat{\boldsymbol{\Pi}}_{(i)} \right\|_F$ decrease linearly to zero as well.

This completes the proof of Theorem 10. ■

Appendix F. Some examples of generalized retraction

In this section, we discuss two popular normalization schemes: polar projection and QR decomposition. We prove that both fit the Definition 5 of a generalized retraction. The analysis in this section is inspired by Liu et al. (2019). However, Liu et al. (2019) only considers conventional retraction operations, while we consider generalized retractions.

F.1 Polar projection

Polar projection is defined as:

$$\mathcal{GR}_{\boldsymbol{U}}^{\text{polar}}(\boldsymbol{\xi}) = (\boldsymbol{U}+\boldsymbol{\xi})\left(\boldsymbol{I}+\boldsymbol{U}^T\boldsymbol{\xi}+\boldsymbol{\xi}^T\boldsymbol{U}+\boldsymbol{\xi}^T\boldsymbol{\xi}\right)^{-\frac{1}{2}}$$

Then obviously,

$$col(\mathcal{GR}_{\boldsymbol{U}}(\boldsymbol{\xi})) = col(\boldsymbol{U}+\boldsymbol{\xi})$$

To verify the second property, we can calculate the difference between $\mathcal{GR}_{\boldsymbol{U}}(\boldsymbol{\xi})$ and $\boldsymbol{U}+\mathcal{P}_{\mathcal{T}_{\boldsymbol{U}}}(\boldsymbol{\xi})$.

Notice that

$$\begin{aligned}
&\left(\boldsymbol{I}+\boldsymbol{U}^T\boldsymbol{\xi}+\boldsymbol{\xi}^T\boldsymbol{U}+\boldsymbol{\xi}^T\boldsymbol{\xi}\right)^{-\frac{1}{2}}\\
&=\boldsymbol{I}-\frac{1}{2}\boldsymbol{U}^T\boldsymbol{\xi}-\frac{1}{2}\boldsymbol{\xi}^T\boldsymbol{U}-\frac{1}{2}\boldsymbol{\xi}^T\boldsymbol{\xi}+\sum_{n=2}^{\infty}\left(\boldsymbol{U}^T\boldsymbol{\xi}+\boldsymbol{\xi}^T\boldsymbol{U}+\boldsymbol{\xi}^T\boldsymbol{\xi}\right)^n\frac{(2n-1)!!(-1)^n}{2^n n!}
\end{aligned}$$

We have

$$\begin{aligned}
&\mathcal{GR}_{\boldsymbol{U}}(\boldsymbol{\xi})-(\boldsymbol{U}+\mathcal{P}_{\mathcal{T}_{\boldsymbol{U}}}(\boldsymbol{\xi}))\\
&=(\boldsymbol{U}+\boldsymbol{\xi})\left(\boldsymbol{I}-\frac{1}{2}\boldsymbol{U}^T\boldsymbol{\xi}-\frac{1}{2}\boldsymbol{\xi}^T\boldsymbol{U}-\frac{1}{2}\boldsymbol{\xi}^T\boldsymbol{\xi}+\sum_{n=2}^{\infty}\left(\boldsymbol{U}^T\boldsymbol{\xi}+\boldsymbol{\xi}^T\boldsymbol{U}+\boldsymbol{\xi}^T\boldsymbol{\xi}\right)^n\frac{(2n-1)!!(-1)^n}{2^n n!}\right)\\
&-\left(\boldsymbol{U}-\boldsymbol{\xi}+\frac{1}{2}\boldsymbol{U}^T\boldsymbol{\xi}+\frac{1}{2}\boldsymbol{\xi}^T\boldsymbol{U}\right)\\
&=\left(-\frac{1}{2}\boldsymbol{\xi}^T\boldsymbol{U}^T\boldsymbol{\xi}-\frac{1}{2}\boldsymbol{\xi}^T\boldsymbol{\xi}^T\boldsymbol{U}-\frac{1}{2}\boldsymbol{\xi}^T\boldsymbol{\xi}^T\boldsymbol{\xi}+(\boldsymbol{U}+\boldsymbol{\xi})\sum_{n=2}^{\infty}\left(\boldsymbol{U}^T\boldsymbol{\xi}+\boldsymbol{\xi}^T\boldsymbol{U}+\boldsymbol{\xi}^T\boldsymbol{\xi}\right)^n\frac{(2n-1)!!(-1)^n}{2^n n!}\right)
\end{aligned}\tag{82}$$

By the property of Frobinius norm:

$$\begin{aligned}
&\left\|\boldsymbol{U}^T\boldsymbol{\xi}+\boldsymbol{\xi}^T\boldsymbol{U}+\boldsymbol{\xi}^T\boldsymbol{\xi}\right\|_F\\
&\leq 2\left\|\boldsymbol{\xi}\right\|_F\left\|\boldsymbol{U}^T\right\|_{op}+\left\|\boldsymbol{\xi}\right\|_F^2\\
&=2\left\|\boldsymbol{\xi}\right\|_F+\left\|\boldsymbol{\xi}\right\|_F^2\\
&\leq 3\left\|\boldsymbol{\xi}\right\|_F
\end{aligned}$$

Therefore,

$$\begin{aligned}
&\left\|\mathcal{GR}_{\boldsymbol{U}}\left(\boldsymbol{\xi}\right)-\left(\boldsymbol{U}+\mathcal{P}_{\mathcal{T}_{\boldsymbol{U}}}(\boldsymbol{\xi})\right)\right\|_F \\
&\leq \frac{1}{2}\left\|\boldsymbol{\xi}^T\boldsymbol{U}^T\boldsymbol{\xi}\right\|_F+\frac{1}{2}\left\|\boldsymbol{\xi}^T\boldsymbol{\xi}^T\boldsymbol{U}\right\|_F+\frac{1}{2}\left\|\boldsymbol{\xi}^T\boldsymbol{\xi}^T\boldsymbol{\xi}\right\|_F \\
&+\left\|\boldsymbol{U}+\boldsymbol{\xi}\right\|_F\sum_{n=2}^{\infty}\left\|\boldsymbol{U}^T\boldsymbol{\xi}+\boldsymbol{\xi}^T\boldsymbol{U}+\boldsymbol{\xi}^T\boldsymbol{\xi}\right\|_F^n\frac{(2n-1)!!}{2^n n!} \\
&\leq \left\|\boldsymbol{\xi}\right\|_F^2+\frac{1}{2}\left\|\boldsymbol{\xi}\right\|_F^3+\left(1+\left\|\boldsymbol{\xi}\right\|_F\right)\sum_{n=2}^{\infty}\left(3\left\|\boldsymbol{\xi}\right\|_F\right)^n\frac{(2n-1)!!}{2^n n!} \\
&=\left\|\boldsymbol{\xi}\right\|_F^2+\frac{1}{2}\left\|\boldsymbol{\xi}\right\|_F^3+\left(1+\left\|\boldsymbol{\xi}\right\|_F\right)\frac{3\left(3\left\|\boldsymbol{\xi}\right\|_F\right)^2+\left(3\left\|\boldsymbol{\xi}\right\|_F\right)^3}{2} \\
&\leq M_{polar}\left\|\boldsymbol{\xi}\right\|_F^2
\end{aligned}$$

where $M_{polar}=\frac{253}{8}$. We applied the following summation in the derivation:

$$\begin{aligned}
&\sum_{n=2}^{\infty}x^n\frac{(2n-1)!!}{2^n n!} \\
&=(1-x)^{-1/2}-(1+\frac{x}{2}) \\
&=\frac{3x^2+x^3}{\sqrt{1-x}+(1-x)(1+\frac{x}{2})}
\end{aligned}$$

and the fact that $x\leq\frac{1}{2}$ in the third inequality.

Since

$$\begin{aligned}
&\left\|\boldsymbol{\xi}\right\|_F^2 \\
&=\left\|\mathcal{P}_{\mathcal{T}_{\boldsymbol{U}}}(\boldsymbol{\xi})+\mathcal{P}_{\mathcal{N}_{\boldsymbol{U}}}(\boldsymbol{\xi})\right\|_F^2 \\
&\leq 2\left\|\mathcal{P}_{\mathcal{T}_{\boldsymbol{U}}}(\boldsymbol{\xi})\right\|_F^2+2\left\|\mathcal{P}_{\mathcal{N}_{\boldsymbol{U}}}(\boldsymbol{\xi})\right\|_F^2 \\
&\leq 2\left\|\mathcal{P}_{\mathcal{T}_{\boldsymbol{U}}}(\boldsymbol{\xi})\right\|_F^2+\left\|\mathcal{P}_{\mathcal{N}_{\boldsymbol{U}}}(\boldsymbol{\xi})\right\|_F
\end{aligned}$$

We prove that polar projection is a generalized retraction with $M_1=\frac{253}{4}$ and $M_2=\frac{253}{8}$.

Polar projection can be implemented via singular value decomposition of $\boldsymbol{U}+\boldsymbol{\xi}$, whose computational complexity is $O(dr^2+r^3)$ (Breloy et al., 2021).

F.2 QR decomposition

QR decomposition is an extension of Gram-Schmidt orthonormalization. For a matrix $\boldsymbol{U}+\boldsymbol{\xi}\in\mathbb{R}^{d\times r}$, the method finds a orthogonal matrix $\boldsymbol{Q}\in\mathbb{R}^{d\times r}$ and an upper triangular matrix $\boldsymbol{R}\in\mathbb{R}^{r\times r}$, such that $\boldsymbol{QR}=\boldsymbol{U}+\boldsymbol{\xi}$. Then $\mathcal{GR}_{\boldsymbol{U}}^{\mathrm{QR}}\left(\boldsymbol{\xi}\right)=\boldsymbol{Q}$.

In this section, we will prove that QR decomposition is a generalized retraction for $\left\|\boldsymbol{\xi}\right\|\leq\frac{1}{4}$. Our proof in this section extends that in Liu et al. (2019).

Notice that $col(\boldsymbol{U}+\boldsymbol{\xi})=col(\boldsymbol{Q})$, thus the first property of generalized retraction in Definition 5 is satisfied. We will prove the second in the case $M_3=\frac{1}{4}$

Similar to Liu et al. (2019), we define $\boldsymbol{U}(t) = \boldsymbol{U} + t\boldsymbol{\xi}$, for $t \in [0,1]$, and use $\boldsymbol{Q}(t)\boldsymbol{R}(t)$ to denote the QR decomposition of $\boldsymbol{U}(t)$. Then:

$$\begin{aligned}
&\left\| \mathcal{GR}_{\boldsymbol{U}}^{\mathrm{QR}}(\boldsymbol{\xi}) - (\boldsymbol{U}+\boldsymbol{\xi}) \right\|_F \\
&= \|\boldsymbol{Q}(1) - \boldsymbol{Q}(1)\boldsymbol{R}(1)\|_F = \|\boldsymbol{Q}(1)(\boldsymbol{I} - \boldsymbol{R}(1))\|_F \\
&\leq \|\boldsymbol{R}(1) - \boldsymbol{R}(0)\|_F \\
&= \left\| \int_0^1 \boldsymbol{R}^{'}(t)dt \right\|_F \\
&\leq \int_0^1 \left\| \boldsymbol{R}^{'}(t) \right\|_F dt
\end{aligned}$$

Since $\boldsymbol{Q}(t)\boldsymbol{R}(t)$ is the QR decomposition of $\boldsymbol{U}(t)$, we have:

$$\boldsymbol{R}^T(t)\boldsymbol{R}(t) = \boldsymbol{U}^T(t)\boldsymbol{U}(t) = \boldsymbol{U}^T\boldsymbol{U} + t\boldsymbol{\xi}^T\boldsymbol{U} + t\boldsymbol{U}^T\boldsymbol{\xi} + t^2\boldsymbol{\xi}^T\boldsymbol{\xi} \tag{83}$$

Taking the derivative with respect to t on both sides, we have:

$$\begin{aligned}
&\left(\boldsymbol{R}^{'}\right)^T(t)\boldsymbol{R}(t) + \boldsymbol{R}^T(t)\boldsymbol{R}^{'}(t) \\
&= \boldsymbol{\xi}^T\boldsymbol{U} + \boldsymbol{U}^T\boldsymbol{\xi} + 2t\boldsymbol{\xi}^T\boldsymbol{\xi}
\end{aligned}$$

We can left multiply both sides by $\left(\boldsymbol{R}^{-1}\right)^T(t)$, and right multiply both sides by $\boldsymbol{R}^{-1}(t)$, to obtain:

$$\left(\boldsymbol{R}^{-1}\right)^T(t)\left(\boldsymbol{R}^{'}\right)^T(t) + \boldsymbol{R}^{'}(t)\boldsymbol{R}^{-1}(t) = \left(\boldsymbol{R}^{-1}\right)^T(t)\left(\boldsymbol{\xi}^T\boldsymbol{U} + \boldsymbol{U}^T\boldsymbol{\xi} + 2t\boldsymbol{\xi}^T\boldsymbol{\xi}\right)\boldsymbol{R}^{-1}(t)$$

Since on the left hand side, $\boldsymbol{R}^{'}(t)\boldsymbol{R}^{-1}(t)$ is an upper triangular matrix, its transpose $\left(\boldsymbol{R}^{-1}\right)^T(t)\left(\boldsymbol{R}^{'}\right)^T(t)$ is a lower triangular matrix, we have:

$$\boldsymbol{R}^{'}(t)\boldsymbol{R}^{-1}(t) = \mathrm{up}\left[\left(\boldsymbol{R}^{-1}\right)^T(t)\left(\boldsymbol{\xi}^T\boldsymbol{U} + \boldsymbol{U}^T\boldsymbol{\xi} + 2t\boldsymbol{\xi}^T\boldsymbol{\xi}\right)\boldsymbol{R}^{-1}(t)\right]$$

where for $\boldsymbol{C} \in \mathbb{R}^{d\times d}$, $\mathrm{up}[\cdot]$ is defined as:

$$\mathrm{up}[\boldsymbol{C}]_{ij} = \begin{cases} C_{ij}, \text{ if } j > i \\ \frac{1}{2}C_{ii}, \text{ if } j = i \\ 0, \text{ if } j < i \end{cases}$$

Therefore,

$$\boldsymbol{R}^{'}(t) = \mathrm{up}\left[\left(\boldsymbol{R}^{-1}\right)^T(t)\left(\boldsymbol{\xi}^T\boldsymbol{U} + \boldsymbol{U}^T\boldsymbol{\xi} + 2t\boldsymbol{\xi}^T\boldsymbol{\xi}\right)\boldsymbol{R}^{-1}(t)\right]\boldsymbol{R}(t)$$

and accordingly:

$$\begin{aligned}
\left\|\boldsymbol{R}^{'}(t)\right\|_F &= \left\|\mathrm{up}\left[\left(\boldsymbol{R}^{-1}\right)^T(t)\left(\boldsymbol{\xi}^T\boldsymbol{U} + \boldsymbol{U}^T\boldsymbol{\xi} + 2t\boldsymbol{\xi}^T\boldsymbol{\xi}\right)\boldsymbol{R}^{-1}(t)\right]\boldsymbol{R}(t)\right\|_F \\
&\leq \left\|\mathrm{up}\left[\left(\boldsymbol{R}^{-1}\right)^T(t)\left(\boldsymbol{\xi}^T\boldsymbol{U} + \boldsymbol{U}^T\boldsymbol{\xi} + 2t\boldsymbol{\xi}^T\boldsymbol{\xi}\right)\boldsymbol{R}^{-1}(t)\right]\right\|_F \|\boldsymbol{R}(t)\|_{op} \\
&\leq \left\|\left(\boldsymbol{R}^{-1}\right)^T(t)\left(\boldsymbol{\xi}^T\boldsymbol{U} + \boldsymbol{U}^T\boldsymbol{\xi} + 2t\boldsymbol{\xi}^T\boldsymbol{\xi}\right)\boldsymbol{R}^{-1}(t)\right\|_F \|\boldsymbol{R}(t)\|_{op}
\end{aligned}$$

where we used Lemma 23 for the first inequality.

From (83), we know that:

$$\begin{aligned}
&\|\boldsymbol{R}(t)\|_{op}^2 = \left\|\boldsymbol{R}(t)^T\boldsymbol{R}(t)\right\|_{op} \\
&= \left\|\boldsymbol{I} + t\left(\boldsymbol{\xi}^T\boldsymbol{U} + \boldsymbol{U}^T\boldsymbol{\xi}\right) + t^2\boldsymbol{\xi}^T\boldsymbol{\xi}\right\|_{op} \\
&\geq 1 - t\left\|\boldsymbol{\xi}^T\boldsymbol{U} + \boldsymbol{U}^T\boldsymbol{\xi}\right\|_{op} - t^2\left\|\boldsymbol{\xi}^T\boldsymbol{\xi}\right\|_{op} \\
&\geq 1 - 2\|\boldsymbol{\xi}\|_F - \left\|\boldsymbol{\xi}^T\boldsymbol{\xi}\right\|_F \\
&\geq \frac{7}{16}
\end{aligned}$$

where the first inequality comes from the triangle inequality, the second comes from the fact that $\|\cdot\|_F \geq \|\cdot\|_{op}$, and the third comes from the requirement $\|\boldsymbol{\xi}\|_F \leq \frac{1}{4}$.

Similarly, we can derive:

$$\|\boldsymbol{R}(t)\|_{op}^2 = \left\|\boldsymbol{R}(t)^T\boldsymbol{R}(t)\right\|_{op} \leq 1 + 2\|\boldsymbol{\xi}\|_F + \left\|\boldsymbol{\xi}^T\boldsymbol{\xi}\right\|_F \leq \frac{25}{16}$$

As a result,

$$\begin{aligned}
&\left\|\boldsymbol{R}'(t)\right\|_F \leq \left\|\left(\boldsymbol{R}^{-1}\right)^T(t)\left(\boldsymbol{\xi}^T\boldsymbol{U} + \boldsymbol{U}^T\boldsymbol{\xi} + 2t\boldsymbol{\xi}^T\boldsymbol{\xi}\right)\boldsymbol{R}^{-1}(t)\right\|_F \|\boldsymbol{R}(t)\|_{op} \\
&\leq \left(\left\|\left(\boldsymbol{R}^{-1}\right)^T(t)\left(\boldsymbol{\xi}^T\boldsymbol{U} + \boldsymbol{U}^T\boldsymbol{\xi}\right)\boldsymbol{R}^{-1}(t)\right\|_F + \left\|\left(\boldsymbol{R}^{-1}\right)^T(t)\left(2t\boldsymbol{\xi}^T\boldsymbol{\xi}\right)\boldsymbol{R}^{-1}(t)\right\|_F\right)\frac{5}{4} \\
&\leq \frac{5}{4}\left(\left\|\boldsymbol{\xi}^T\boldsymbol{U} + \boldsymbol{U}^T\boldsymbol{\xi}\right\|_F\left\|\left(\boldsymbol{R}^{-1}\right)^T(t)\boldsymbol{R}^{-1}(t)\right\|_{op} + 2t\left\|\boldsymbol{\xi}^T\boldsymbol{\xi}\right\|_F\left\|\left(\boldsymbol{R}^{-1}\right)^T(t)\boldsymbol{R}^{-1}(t)\right\|_{op}\right) \\
&\leq \frac{20}{7}\left(\left\|\boldsymbol{\xi}^T\boldsymbol{U} + \boldsymbol{U}^T\boldsymbol{\xi}\right\|_F + 2t\left\|\boldsymbol{\xi}^T\boldsymbol{\xi}\right\|_F\right)
\end{aligned}$$

Hence,

$$\begin{aligned}
&\left\|\mathcal{GR}_{\boldsymbol{U}}^{\mathrm{QR}}(\boldsymbol{\xi}) - (\boldsymbol{U} + \boldsymbol{\xi})\right\|_F \\
&\leq \frac{20}{7}\left(\left\|\boldsymbol{\xi}^T\boldsymbol{U} + \boldsymbol{U}^T\boldsymbol{\xi}\right\|_F + \left\|\boldsymbol{\xi}^T\boldsymbol{\xi}\right\|_F\right)
\end{aligned}$$

Since $\boldsymbol{U}$ is an orthogonal matrix, $\|\mathcal{P}_{\mathcal{N}_{\boldsymbol{U}}}(\boldsymbol{\xi})\|_F = \frac{1}{2}\left\|\boldsymbol{U}\left(\boldsymbol{\xi}^T\boldsymbol{U} + \boldsymbol{U}^T\boldsymbol{\xi}\right)\right\|_F = \frac{1}{2}\left\|\boldsymbol{\xi}^T\boldsymbol{U} + \boldsymbol{U}^T\boldsymbol{\xi}\right\|_F$. By Cauchy-Schwartz inequality, $\left\|\boldsymbol{\xi}^T\boldsymbol{\xi}\right\|_F = \left\|\left(\mathcal{P}_{\mathcal{N}_{\boldsymbol{U}}}(\boldsymbol{\xi}) + \mathcal{P}_{\mathcal{T}_{\boldsymbol{U}}}(\boldsymbol{\xi})\right)^T\left(\mathcal{P}_{\mathcal{N}_{\boldsymbol{U}}}(\boldsymbol{\xi}) + \mathcal{P}_{\mathcal{T}_{\boldsymbol{U}}}(\boldsymbol{\xi})\right)\right\|_F \leq 2\|\mathcal{P}_{\mathcal{N}_{\boldsymbol{U}}}(\boldsymbol{\xi})\|_F^2 + 2\|\mathcal{P}_{\mathcal{T}_{\boldsymbol{U}}}(\boldsymbol{\xi})\|_F^2$.

Thus we have:

$$\begin{aligned}
&\left\|\mathcal{GR}_{\boldsymbol{U}}^{\mathrm{QR}}(\boldsymbol{\xi}) - (\boldsymbol{U} + \boldsymbol{\xi})\right\|_F \\
&\leq \frac{20}{7}\left(2\|\mathcal{P}_{\mathcal{N}_{\boldsymbol{U}}}(\boldsymbol{\xi})\|_F + 2\|\mathcal{P}_{\mathcal{N}_{\boldsymbol{U}}}(\boldsymbol{\xi})\|_F^2 + 2\|\mathcal{P}_{\mathcal{T}_{\boldsymbol{U}}}(\boldsymbol{\xi})\|_F^2\right) \\
&\leq \frac{80}{7}\|\mathcal{P}_{\mathcal{N}_{\boldsymbol{U}}}(\boldsymbol{\xi})\|_F + \frac{40}{7}\|\mathcal{P}_{\mathcal{T}_{\boldsymbol{U}}}(\boldsymbol{\xi})\|_F^2
\end{aligned}$$

Hence the second property of definition holds with $M_1 = \frac{40}{7}$ and $M_2 = \frac{80}{7}$.

QR decomposition can be implemented by Gram-Schmidt or Householder algorithm with computation complexity of $O(dr^2)$

Appendix G. Auxiliary lemmas

In this section, we show some auxiliary lemmas needed for the proof in earlier Sections. Most lemmas are derived from basic facts in linear algebra.

We begin with some general inequalities related to matrix trace norms.

Lemma 22 *For two matrices $\boldsymbol{A}, \boldsymbol{B} \in \mathbb{R}^{d\times d}$, if both $\boldsymbol{A}, \boldsymbol{B}$ are symmetric positive definite, then:*

$$Tr(\boldsymbol{AB}) \geq 0$$

A simple corollary is that if $\boldsymbol{A}_1, \boldsymbol{A}_2, \boldsymbol{B} \in \mathbb{R}^{d\times d}$ are symmetric and $\boldsymbol{B}$ is positive semi-definite, and $\boldsymbol{A}_1 \succeq \boldsymbol{A}_2$, then

$$\operatorname{Tr}(\boldsymbol{A}_1\boldsymbol{B}) \geq \operatorname{Tr}(\boldsymbol{A}_2\boldsymbol{B})$$

Proof Since both $\boldsymbol{A}$ and $\boldsymbol{B}$ are positive symmetric, there exists $\boldsymbol{X}, \boldsymbol{Y} \in \mathbb{R}^{d\times d}$, such that $A = X^T X$ and $\boldsymbol{B} = \boldsymbol{Y}^T\boldsymbol{Y}$, therefore:

$$\begin{aligned}
&\operatorname{Tr}(\boldsymbol{AB}) \\
&= \operatorname{Tr}\left(\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{Y}^T\boldsymbol{Y}\right) \\
&= \operatorname{Tr}\left((\boldsymbol{Y}\boldsymbol{X}^T)^T\boldsymbol{Y}\boldsymbol{X}^T\right) \\
&\geq 0
\end{aligned}$$

■

The following lemma presents an upper bound of the Frobenius norm of the product of two matrices.

Lemma 23 *For two matrices $\boldsymbol{A} \in \mathbb{R}^{m\times n}$, and $\boldsymbol{B} \in \mathbb{R}^{n\times k}$, we have:*

$$\|\boldsymbol{AB}\|_F \leq \|\boldsymbol{A}\|_{op} \|\boldsymbol{B}\|_F$$

and:

$$\|\boldsymbol{AB}\|_F \leq \|\boldsymbol{A}\|_F \|\boldsymbol{B}\|_{op}$$

The proof of the lemma can be found in Sun and Luo (2015).

The following lemma introduces a simple upper bound on the Frobenius norm of $\boldsymbol{I}_r - \boldsymbol{U}^T\boldsymbol{P}\boldsymbol{U}$.

Lemma 24 *For any rank-r orthonormal matrix $\boldsymbol{U} \in \mathbb{R}^{d\times r}$, and rank-r projection matrix $\boldsymbol{P} \in \mathbb{R}^{d\times d}$, we have:*

$$\left\|\boldsymbol{I}_r - \boldsymbol{U}^T\boldsymbol{P}\boldsymbol{U}\right\|_F \leq r - Tr\left(\boldsymbol{U}^T\boldsymbol{P}\boldsymbol{U}\right) \tag{84}$$

Proof It is easy to see that $\boldsymbol{I}_r - \boldsymbol{U}^T\boldsymbol{P}\boldsymbol{U}$ is positive semidefinite. Also, for a positive semidefinite matrix, its Frobenius norm is upper bounded by its trace. Inequality (84) follows accordingly. ■

We can proceed to the following lemma that upper bounds the trace of the k-th power of $\boldsymbol{I}_r - \boldsymbol{U}^T\boldsymbol{P}\boldsymbol{U}$.

Lemma 25 *For any rank-r orthonormal matrix $\boldsymbol{U} \in \mathbb{R}^{d\times r}$, and rank-r projection matrix $\boldsymbol{P} \in \mathbb{R}^{d\times r}$, $k = 1, 2, ...$, $\left(\boldsymbol{I}_r - \boldsymbol{U}^T\boldsymbol{P}\boldsymbol{U}\right)^k$ is positive semi-definite and:*

$$0 \leq Tr\left(\left(\boldsymbol{I} - \boldsymbol{U}^T\boldsymbol{P}\boldsymbol{U}\right)^k\right) \leq \left(r - Tr\left(\boldsymbol{U}^T\boldsymbol{P}\boldsymbol{U}\right)\right)^k \tag{85}$$

Proof Since $\boldsymbol{I}_r - \boldsymbol{U}^T\boldsymbol{P}\boldsymbol{U}$ is symmetric positive semidefinite, $\left(\boldsymbol{I}_r - \boldsymbol{U}^T\boldsymbol{P}\boldsymbol{U}\right)^k$ is also symmetric positive semidefinite. Assume eigenvalues of $\boldsymbol{I}_r - \boldsymbol{U}^T\boldsymbol{P}\boldsymbol{U}$ are $\lambda_1, \lambda_2, \cdots, \lambda_d$, with $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d$, we know that $Tr\left(\left(\boldsymbol{I}_r - \boldsymbol{U}^T\boldsymbol{P}\boldsymbol{U}\right)^k\right) = \sum_{i=1}^d \lambda_i^k \leq \lambda_1^{k-1}\sum_{i=1}^d \lambda_i$.

By Lemma 24, we know that

$$\lambda_1^{k-1} \leq \left\|\boldsymbol{I}_r - \boldsymbol{U}^T\boldsymbol{P}\boldsymbol{U}\right\|_F^{k-1} \leq \left(r - \mathrm{Tr}\left(\boldsymbol{U}^T\boldsymbol{P}\boldsymbol{U}\right)\right)^{k-1}$$

This completes our proof. ■

Based on the above results, we can discuss some properties of the projection of a matrix onto a subspace. Suppose we know the column space of $\boldsymbol{U} \in \mathbb{R}^{d\times r}$ is close to that of $\boldsymbol{P} \in \mathbb{R}^{d\times d}$, can we find a matrix $\boldsymbol{U}^*$ close to $\boldsymbol{U}$ with column vectors in $col(\boldsymbol{P})$? The following two lemmas give affirmative answers.

Lemma 26 *For any rank-r orthonormal matrix $\boldsymbol{U} \in \mathbb{R}^{d\times r}$, and rank-r projection matrix $\boldsymbol{P} \in \mathbb{R}^{d\times d}$, we define:*

$$\boldsymbol{U}^\star = \boldsymbol{P}\boldsymbol{U}\left(\boldsymbol{U}^T\boldsymbol{P}\boldsymbol{U}\right)^{-1/2}$$

If $r - Tr\left(\boldsymbol{U}^T\boldsymbol{P}\boldsymbol{U}\right) \leq 1$, we have:

$$\|\boldsymbol{U} - \boldsymbol{U}^\star\|_F^2 \geq r - Tr\left(\boldsymbol{U}^T\boldsymbol{P}\boldsymbol{U}\right) \tag{86}$$

and,

$$\|\boldsymbol{U} - \boldsymbol{U}^\star\|_F^2 \leq 2\left(r - Tr\left(\boldsymbol{U}^T\boldsymbol{P}\boldsymbol{U}\right)\right) \tag{87}$$

Proof To prove the lower bound (86) and upper bound (87), we can write $\|\boldsymbol{U} - \boldsymbol{U}^\star\|_F^2$ as,

$$\begin{aligned}\|\boldsymbol{U} - \boldsymbol{U}^\star\|_F^2 &= \langle\boldsymbol{U}, \boldsymbol{U}\rangle + \langle\boldsymbol{U}^\star, \boldsymbol{U}^\star\rangle - 2\langle\boldsymbol{U}, \boldsymbol{U}^\star\rangle \\ &= 2r - 2\langle\boldsymbol{U}, \boldsymbol{U}^\star\rangle\end{aligned}$$

We first find an upper bound for $\langle\boldsymbol{U}, \boldsymbol{U}^\star\rangle$.

Notice that:

$$\begin{aligned}
&\langle \boldsymbol{U}, \boldsymbol{U}^\star \rangle \\
&= \mathrm{Tr}\left(\boldsymbol{U}^T \boldsymbol{U}^\star\right) \\
&= \mathrm{Tr}\left(\boldsymbol{U}^T \boldsymbol{P} \boldsymbol{U} \left(\boldsymbol{U}^T \boldsymbol{P} \boldsymbol{U}\right)^{-1/2}\right) \\
&= \mathrm{Tr}\left(\left(\boldsymbol{U}^T \boldsymbol{P} \boldsymbol{U}\right)^{1/2}\right) \\
&= \mathrm{Tr}\left(\left(\boldsymbol{I}_r - \left(\boldsymbol{I}_r - \boldsymbol{U}^T \boldsymbol{P} \boldsymbol{U}\right)\right)^{1/2}\right) \\
&= \mathrm{Tr}\left(\boldsymbol{I}_r - \frac{1}{2}\left(\boldsymbol{I}_r - \boldsymbol{U}^T \boldsymbol{P} \boldsymbol{U}\right) - \sum_{n=2}^{\infty} \frac{(2n-3)!!}{2^n n!} \left(\boldsymbol{I}_r - \boldsymbol{U}^T \boldsymbol{P} \boldsymbol{U}\right)^n\right) \\
&= r - \frac{1}{2}\mathrm{Tr}\left(\boldsymbol{I}_r - \boldsymbol{U}^T \boldsymbol{P} \boldsymbol{U}\right) - \sum_{n=2}^{\infty} \frac{(2n-3)!!}{2^n n!} \mathrm{Tr}\left(\left(\boldsymbol{I}_r - \boldsymbol{U}^T \boldsymbol{P} \boldsymbol{U}\right)^n\right) \\
&\leq r - \frac{1}{2}\mathrm{Tr}\left(\boldsymbol{I}_r - \boldsymbol{U}^T \boldsymbol{P} \boldsymbol{U}\right)
\end{aligned}$$

We used the series $(1-x)^{\frac{1}{2}} = 1 - \frac{1}{2}x - \sum_{n=2}^{\infty} \frac{(2n-3)!!}{2^n n!} x^n$, and the result $\mathrm{Tr}\left(\left(\boldsymbol{I}_r - \boldsymbol{U}^T \boldsymbol{P} \boldsymbol{U}\right)^n\right) \geq 0$ from Lemma 25.

As a result:

$$\|\boldsymbol{U} - \boldsymbol{U}^\star\|_F^2 \geq \mathrm{Tr}\left(\boldsymbol{I}_r - \boldsymbol{U}^T \boldsymbol{P} \boldsymbol{U}\right) = r - \mathrm{Tr}\left(\boldsymbol{U}^T \boldsymbol{P} \boldsymbol{U}\right)$$

Similarly, from Lemma 25, $\mathrm{Tr}\left(\left(\boldsymbol{I}_r - \boldsymbol{U}^T \boldsymbol{P} \boldsymbol{U}\right)^n\right) \leq \mathrm{Tr}\left(\boldsymbol{I}_r - \boldsymbol{U}^T \boldsymbol{P} \boldsymbol{U}\right)^n$, thus:

$$\begin{aligned}
&\langle \boldsymbol{U}, \boldsymbol{U}^\star \rangle \\
&= r - \frac{1}{2}\mathrm{Tr}\left(\boldsymbol{I}_r - \boldsymbol{U}^T \boldsymbol{P} \boldsymbol{U}\right) - \sum_{n=2}^{\infty} \frac{(2n-3)!!}{2^n n!} \mathrm{Tr}\left(\left(\boldsymbol{I}_r - \boldsymbol{U}^T \boldsymbol{P} \boldsymbol{U}\right)^n\right) \\
&\geq r - \frac{1}{2}\mathrm{Tr}\left(\boldsymbol{I}_r - \boldsymbol{U}^T \boldsymbol{P} \boldsymbol{U}\right) - \sum_{n=2}^{\infty} \frac{(2n-3)!!}{2^n n!} \mathrm{Tr}\left(\boldsymbol{I}_r - \boldsymbol{U}^T \boldsymbol{P} \boldsymbol{U}\right)^n \\
&= r + \left(1 - \mathrm{Tr}\left(\boldsymbol{I}_r - \boldsymbol{U}^T \boldsymbol{P} \boldsymbol{U}\right)\right)^{\frac{1}{2}} - 1 \\
&\geq -\mathrm{Tr}\left(\boldsymbol{I}_r - \boldsymbol{U}^T \boldsymbol{P} \boldsymbol{U}\right)
\end{aligned}$$

where we used the relation $\sqrt{1-x} - 1 \geq -x, \forall x \in [0,1]$, in the last inequality.

Thus

$$\|\boldsymbol{U} - \boldsymbol{U}^\star\|_F^2 \leq 2\mathrm{Tr}\left(\boldsymbol{I}_r - \boldsymbol{U}^T \boldsymbol{P} \boldsymbol{U}\right) = 2\left(r - \mathrm{Tr}\left(\boldsymbol{U}^T \boldsymbol{P} \boldsymbol{U}\right)\right)$$

This completes our proof. ■

The following lemma shows that we can identify global PCs from local PCs.

Lemma 27 *Suppose for $i = 1, \cdots, N$, $\boldsymbol{P}_{\boldsymbol{U}}$, $\boldsymbol{P}_{\boldsymbol{V}_{(i)}}$ and $\boldsymbol{P}^\star_{\boldsymbol{U}}$, $\boldsymbol{P}^\star_{\boldsymbol{V}_{(i)}}$ are projection matrices satisfying $\boldsymbol{P}_{\boldsymbol{U}} \boldsymbol{P}_{\boldsymbol{V}_{(i)}} = 0$ and $\boldsymbol{P}^\star_{\boldsymbol{U}} \boldsymbol{P}^\star_{\boldsymbol{V}_{(i)}} = 0$ for each i. Among them, $\boldsymbol{P}_{\boldsymbol{U}}$ and $\boldsymbol{P}^\star_{\boldsymbol{U}}$ have rank r_1,*

$\boldsymbol{P}_{\boldsymbol{V}_{(i)}}$ *and* $\boldsymbol{P}^{\star}_{\boldsymbol{V}_{(i)}}$ *have rank* $r_{2,(i)}$*. If there exists a positive constant* $\theta > 0$ *such that*

$$\lambda_{max}(\frac{1}{N}\sum_{i=1}^{N}\boldsymbol{P}^{\star}_{\boldsymbol{V}_{(i)}}) \leq 1-\theta$$

we have the following bound:

$$\begin{aligned}&\sum_{i=1}^{N} r_1 + r_{2,(i)} - Tr\left(\left(\boldsymbol{P}_{\boldsymbol{U}} + \boldsymbol{P}_{\boldsymbol{V}_{(i)}}\right)\left(\boldsymbol{P}^{\star}_{\boldsymbol{U}} + \boldsymbol{P}^{\star}_{\boldsymbol{V}_{(i)}}\right)\right)\\ &\qquad \leq N\left(r_1 - Tr(\boldsymbol{P}^{\star}_{\boldsymbol{U}}\boldsymbol{P}_{\boldsymbol{U}})\right) + \sum_{i=1}^{N} r_{2,(i)} - Tr(\boldsymbol{P}^{\star}_{\boldsymbol{V}_{(i)}}\boldsymbol{P}_{\boldsymbol{V}_{(i)}})\end{aligned} \tag{88}$$

And also:

$$\begin{aligned}&\sum_{i=1}^{N} r_1 + r_{2,(i)} - Tr\left(\left(\boldsymbol{P}_{\boldsymbol{U}} + \boldsymbol{P}_{\boldsymbol{V}_{(i)}}\right)\left(\boldsymbol{P}^{\star}_{\boldsymbol{U}} + \boldsymbol{P}^{\star}_{\boldsymbol{V}_{(i)}}\right)\right)\\ &\geq \frac{\theta}{2}\left(N\left(r_1 - Tr(\boldsymbol{P}^{\star}_{\boldsymbol{U}}\boldsymbol{P}_{\boldsymbol{U}})\right) + \sum_{i=1}^{N} r_{2,(i)} - Tr(\boldsymbol{P}^{\star}_{\boldsymbol{V}_{(i)}}\boldsymbol{P}_{\boldsymbol{V}_{(i)}})\right)\end{aligned} \tag{89}$$

Notice that we can replace + by $\oplus$ on the left hand side of (88) and (89)

Proof We first calculate the upper bound.

Since $\boldsymbol{P}_{\boldsymbol{U}}\boldsymbol{P}^{\star}_{\boldsymbol{V}_{(i)}}\boldsymbol{P}_{\boldsymbol{U}}$ is positive semidefinite, we know that:

$$\mathrm{Tr}\left(\boldsymbol{P}_{\boldsymbol{U}}\boldsymbol{P}^{\star}_{\boldsymbol{V}_{(i)}}\boldsymbol{P}_{\boldsymbol{U}}\right) \geq 0$$

Thus

$$\mathrm{Tr}\left(\boldsymbol{P}_{\boldsymbol{U}}\boldsymbol{P}^{\star}_{\boldsymbol{V}_{(i)}}\boldsymbol{P}_{\boldsymbol{U}}\right) = \mathrm{Tr}\left(\boldsymbol{P}_{\boldsymbol{U}}\boldsymbol{P}^{\star}_{\boldsymbol{V}_{(i)}}\right) \geq 0$$

Similarly, we have:

$$\mathrm{Tr}\left(\boldsymbol{P}^{\star}_{\boldsymbol{U}}\boldsymbol{P}_{\boldsymbol{V}_{(i)}}\right) \geq 0$$

Combining them, we have:

$$\begin{aligned}&\mathrm{Tr}\left(\left(\boldsymbol{P}_{\boldsymbol{U}} + \boldsymbol{P}_{\boldsymbol{V}_{(i)}}\right)\left(\boldsymbol{P}^{\star}_{\boldsymbol{U}} + \boldsymbol{P}^{\star}_{\boldsymbol{V}_{(i)}}\right)\right)\\ &= \mathrm{Tr}\left(\boldsymbol{P}_{\boldsymbol{U}}\boldsymbol{P}^{\star}_{\boldsymbol{U}}\right) + \mathrm{Tr}\left(\boldsymbol{P}_{\boldsymbol{U}}\boldsymbol{P}^{\star}_{\boldsymbol{V}_{(i)}}\right) + \mathrm{Tr}\left(\boldsymbol{P}_{\boldsymbol{V}_{(i)}}\boldsymbol{P}^{\star}_{\boldsymbol{U}}\right) + \mathrm{Tr}\left(\boldsymbol{P}_{\boldsymbol{V}_{(i)}}\boldsymbol{P}^{\star}_{\boldsymbol{V}_{(i)}}\right)\\ &\geq \mathrm{Tr}\left(\boldsymbol{P}_{\boldsymbol{U}}\boldsymbol{P}^{\star}_{\boldsymbol{U}}\right) + \mathrm{Tr}\left(\boldsymbol{P}_{\boldsymbol{V}_{(i)}}\boldsymbol{P}^{\star}_{\boldsymbol{V}_{(i)}}\right)\end{aligned}$$

This proves inequality (88).

Next, we calculate the lower bound.

$$
\begin{aligned}
&\sum_{i=1}^{N} \operatorname{Tr}\left(\left(\boldsymbol{P}_{\boldsymbol{U}}+\boldsymbol{P}_{\boldsymbol{V}_{(i)}}\right)\left(\boldsymbol{P}_{\boldsymbol{U}}^{\star}+\boldsymbol{P}_{\boldsymbol{V}_{(i)}}^{\star}\right)\right) \\
&=\sum_{i=1}^{N} \operatorname{Tr}\left(\boldsymbol{P}_{\boldsymbol{U}} \boldsymbol{P}_{\boldsymbol{U}}^{\star}\right)+\operatorname{Tr}\left(\boldsymbol{P}_{\boldsymbol{U}} \boldsymbol{P}_{\boldsymbol{V}_{(i)}}^{\star})\right)+\operatorname{Tr}\left(\boldsymbol{P}_{\boldsymbol{V}_{(i)}} \boldsymbol{P}_{\boldsymbol{U}}^{\star}\right)+\operatorname{Tr}\left(\boldsymbol{P}_{\boldsymbol{V}_{(i)}} \boldsymbol{P}_{\boldsymbol{V}_{(i)}}^{\star}\right) \\
&=\sum_{i=1}^{N} \operatorname{Tr}\left(\boldsymbol{P}_{\boldsymbol{U}} \boldsymbol{P}_{\boldsymbol{U}}^{\star}\right)+\operatorname{Tr}\left(\boldsymbol{P}_{\boldsymbol{U}}\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}}^{\star}\right) \boldsymbol{P}_{\boldsymbol{V}_{(i)}}^{\star}\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}}^{\star}\right)\right)+\operatorname{Tr}\left(\boldsymbol{P}_{\boldsymbol{V}_{(i)}} \boldsymbol{P}_{\boldsymbol{U}}^{\star}\right)+\operatorname{Tr}\left(\boldsymbol{P}_{\boldsymbol{V}_{(i)}} \boldsymbol{P}_{\boldsymbol{V}_{(i)}}^{\star}\right) \\
&=\sum_{i=1}^{N} \operatorname{Tr}\left(\boldsymbol{P}_{\boldsymbol{U}} \boldsymbol{P}_{\boldsymbol{U}}^{\star}\right)+\operatorname{Tr}\left(\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}}^{\star}\right) \boldsymbol{P}_{\boldsymbol{U}}\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}}^{\star}\right) \boldsymbol{P}_{\boldsymbol{V}_{(i)}}^{\star}\right)+\operatorname{Tr}\left(\boldsymbol{P}_{\boldsymbol{V}_{(i)}} \boldsymbol{P}_{\boldsymbol{U}}^{\star}\right)+\operatorname{Tr}\left(\boldsymbol{P}_{\boldsymbol{V}_{(i)}} \boldsymbol{P}_{\boldsymbol{V}_{(i)}}^{\star}\right)
\end{aligned}
$$

Since $\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}}^{\star}\right) \boldsymbol{P}_{\boldsymbol{U}}\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}}^{\star}\right)$ and $\boldsymbol{P}_{\boldsymbol{V}_{(i)}}^{\star}$ are both symmetric positive semidefinite, we have:

$$
\begin{aligned}
&\operatorname{Tr}\left(\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}}^{\star}\right) \boldsymbol{P}_{\boldsymbol{U}}\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}}^{\star}\right) \frac{1}{N} \sum_{i=1}^{N} \boldsymbol{P}_{\boldsymbol{V}_{(i)}}^{\star}\right) \\
&\leq \operatorname{Tr}\left(\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}}^{\star}\right) \boldsymbol{P}_{\boldsymbol{U}}\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}}^{\star}\right)\right) \lambda_{max}\left(\frac{1}{N} \sum_{i=1}^{N} \boldsymbol{P}_{\boldsymbol{V}_{(i)}}^{\star}\right) \\
&\leq \operatorname{Tr}\left(\boldsymbol{P}_{\boldsymbol{U}}-\boldsymbol{P}_{\boldsymbol{U}} \boldsymbol{P}_{\boldsymbol{U}}^{\star}\right)(1-\theta) \\
&=\left(r_1-\operatorname{Tr}\left(\boldsymbol{P}_{\boldsymbol{U}} \boldsymbol{P}_{\boldsymbol{U}}^{\star}\right)\right)(1-\theta)
\end{aligned}
$$

For notation simplicity, we define $z_0=r_1-\operatorname{Tr}\left(\boldsymbol{P}_{\boldsymbol{U}} \boldsymbol{P}_{\boldsymbol{U}}^{\star}\right)$ and $z_i=r_{2,(i)}-\operatorname{Tr}\left(\boldsymbol{P}_{\boldsymbol{V}_{(i)}} \boldsymbol{P}_{\boldsymbol{V}_{(i)}}^{\star}\right)$.

From the orthogonality, we have:

$$
\begin{aligned}
&\operatorname{Tr}\left(\boldsymbol{P}_{\boldsymbol{V}_{(i)}} \boldsymbol{P}_{\boldsymbol{U}}^{\star}\right) \\
&=\operatorname{Tr}\left(\boldsymbol{P}_{\boldsymbol{V}_{(i)}}\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{V}_{(i)}}^{\star}\right) \boldsymbol{P}_{\boldsymbol{U}}^{\star}\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{V}_{(i)}}^{\star}\right)\right) \\
&=\operatorname{Tr}\left(\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{V}_{(i)}}^{\star}\right) \boldsymbol{P}_{\boldsymbol{V}_{(i)}}\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{V}_{(i)}}^{\star}\right) \boldsymbol{P}_{\boldsymbol{U}}^{\star}\right) \\
&\leq \operatorname{Tr}\left(\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{V}_{(i)}}^{\star}\right) \boldsymbol{P}_{\boldsymbol{V}_{(i)}}\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{V}_{(i)}}^{\star}\right)\right) \lambda_{max}\left(\boldsymbol{P}_{\boldsymbol{U}}^{\star}\right) \\
&\leq \operatorname{Tr}\left(\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{V}_{(i)}}^{\star}\right) \boldsymbol{P}_{\boldsymbol{V}_{(i)}}\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{V}_{(i)}}^{\star}\right)\right) \\
&=\operatorname{Tr}\left(\boldsymbol{P}_{\boldsymbol{V}_{(i)}}-\boldsymbol{P}_{\boldsymbol{V}_{(i)}} \boldsymbol{P}_{\boldsymbol{V}_{(i)}}^{\star}\right) \\
&=z_i
\end{aligned}
$$

Also, from the orthogonality, we have:

$$\begin{aligned}
&\operatorname{Tr}\left(\boldsymbol{P}_{\boldsymbol{V}_{(i)}}\boldsymbol{P}_{\boldsymbol{U}}^{\star}\right)\\
&=\operatorname{Tr}\left(\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}}\right)\boldsymbol{P}_{\boldsymbol{V}_{(i)}}\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}}\right)\boldsymbol{P}_{\boldsymbol{U}}^{\star}\right)\\
&=\operatorname{Tr}\left(\boldsymbol{P}_{\boldsymbol{V}_{(i)}}\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}}\right)\boldsymbol{P}_{\boldsymbol{U}}^{\star}\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}}\right)\right)\\
&\leq\operatorname{Tr}\left(\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}}\right)\boldsymbol{P}_{\boldsymbol{U}}^{\star}\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}}\right)\right)\lambda_{max}\left(\boldsymbol{P}_{\boldsymbol{V}_{(i)}}\right)\\
&\leq\operatorname{Tr}\left(\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}}\right)\boldsymbol{P}_{\boldsymbol{U}}^{\star}\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}}\right)\right)\\
&=\operatorname{Tr}\left(\boldsymbol{P}_{\boldsymbol{U}}^{\star}-\boldsymbol{P}_{\boldsymbol{U}}\boldsymbol{P}_{\boldsymbol{U}}^{\star}\right)\\
&=z_0
\end{aligned}$$

Combining the two:

$$\operatorname{Tr}\left(\boldsymbol{P}_{\boldsymbol{V}_{(i)}}\boldsymbol{P}_{\boldsymbol{U}}^{\star}\right)\leq\min\{z_0,z_i\}$$

As a result:

$$\begin{aligned}
&\sum_{i=1}^{N}r_1+r_{2,(i)}\\
&-\left[\operatorname{Tr}\left(\boldsymbol{P}_{\boldsymbol{U}}\boldsymbol{P}_{\boldsymbol{U}}^{\star}\right)+\operatorname{Tr}\left(\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}}^{\star}\right)\boldsymbol{P}_{\boldsymbol{U}}\left(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{U}}^{\star}\right)\boldsymbol{P}_{\boldsymbol{V}_{(i)}}^{\star}\right)+\operatorname{Tr}\left(\boldsymbol{P}_{\boldsymbol{V}_{(i)}}\boldsymbol{P}_{\boldsymbol{U}}^{\star}\right)+\operatorname{Tr}\left(\boldsymbol{P}_{\boldsymbol{V}_{(i)}}\boldsymbol{P}_{\boldsymbol{V}_{(i)}}^{\star}\right)\right]\\
&\geq\sum_{i=1}^{N}z_0-\left(1-\theta\right)z_0+z_i-\min\{z_0,z_i\}
\end{aligned}$$

Since for any number $\nu\in(0,1)$, we know:

$$z_i-\min\{z_0,z_i\}\geq\nu\left(z_i-z_0\right)$$

We can set $\nu=\frac{\theta}{2}$, then

$$\begin{aligned}
&\sum_{i=1}^{N}z_0-\left(1-\theta\right)z_0+z_i-\min\{z_0,z_i\}\\
&\geq\sum_{i=1}^{N}\theta z_0+\frac{\theta}{2}\left(z_i-z_0\right)\\
&=\frac{\theta}{2}\sum_{i=1}^{N}z_0+z_i
\end{aligned}$$

This proves inequality (89). ■

References

P.-A. Absil, R. Mahony, and R. Sepulchre. *Optimization algorithms on matrix manifolds*. Princeton University Press, 2008.

Ana M Aguilera, Francisco A Ocaña, and Mariano J Valderrama. Forecasting time series by functional pca. discussion of several weighted approaches. *Computational Statistics*, 14(3): 443–467, 1999.

Foivos Alimisis, Peter Davies, Bart Vandereycken, and Dan Alistarh. Distributed principal component analysis with limited communication. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=edCFRvlWqV.

Rohan Asokan. Us presidential debate transcripts 1960-2020. In *Kaggle dataset*, 2022. doi: 10.34740/KAGGLE/DSV/3690532. URL https://www.kaggle.com/datasets/arenagrenade/us-presidential-debate-transcripts-19602020.

Rajendra Bhatia. *Matrix Analysis*. Springer, New York, NY, 1997.

Aharon Birnbaum, Iain M Johnstone, Boaz Nadler, and Debashis Paul. Minimax bounds for sparse pca with noisy high-dimensional data. *Annals of statistics*, 41(3):1055, 2013.

Nicolas Boumal. An introduction to optimization on smooth manifolds. To appear with Cambridge University Press, Apr 2022. URL http://www.nicolasboumal.net/book.

Thierry Bouwmans, Sajid Javed, Hongyang Zhang, Zhouchen Lin, and Ricardo Otazo. On the applications of robust pca in image and video processing. *Proceedings of the IEEE*, 106(8):1427–1457, 2018. doi: 10.1109/JPROC.2018.2853589.

Arnaud Breloy, Sandeep Kumar, Ying Sun, and Daniel P. Palomar. Majorization-minimization on the stiefel manifold with application to robust sparse pca. *IEEE Transactions on Signal Processing*, 69:1507–1520, 2021. doi: 10.1109/TSP.2021.3058442.

Sebastian Caldas, Sai Meher Karthik Duddu, Peter Wu, Tian Li, Jakub Konecny, H. Brendan McMahan, Virginia Smith, and Ameet Talwalkar. Leaf: A benchmark for federated settings. In *NeurIPS*, 2019.

Emmanuel J. Candes, Xiaodong Li, Yi Ma, and John Wright. Robust principal component analysis? *Journal of the ACM (JACM)*, 2011.

Shixiang Chen, Alfredo Garcia, Mingyi Hong, and Shahin Shahrampour. On the local linear rate of consensus on the stiefel manifold. In *Arxiv*, 2021a. URL https://arxiv.org/pdf/2101.09346.pdf.

Shixiang Chen, Alfredo Garcia, Mingyi Hong, and Shahin Shahrampour. Decentralized riemannian gradient descent on the stiefel manifold. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 1594–1605. PMLR, 18–24 Jul 2021b. URL https://proceedings.mlr.press/v139/chen21g.html.

Xi Chen, Jason D. Lee, He Li, and Yun Yang. Distributed estimation for principal component analysis: a gap-free approach. *CoRR*, abs/2004.02336, 2020. URL `https://arxiv.org/abs/2004.02336`.

Charles-Alban Deledalle, Joseph Salmon, and Arnak Dalalyan. Image denoising with patch-based pca: local versus global. In *The 22nd British Machine Vision Conference*, 2011.

Alan Edelman, Tomás A. Arias, and Steven T. Smith. The geometry of algorithms with orthogonality constraints. *SIAM Journal on Matrix Analysis and Applications*, 20(2): 303–353, 1998. doi: 10.1137/S0895479895290954.

Jianqing Fan, Dong Wang, Kaizheng Wang, and Ziwei Zhu. Distributed estimation of principal eigenspaces. *Annals of statistics*, 47,6:3009–3031, 2019. doi: 10.1214/18-AOS1713.

Dan Feldman, Melanie Schmidt, and Christian Sohler. Turning big data into tiny data: Constant-size coresets for k-means, pca and projective clustering. In *Proceedings of the Twenty-Fourth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '13, pages 1434–1453, USA, 2013. Society for Industrial and Applied Mathematics. ISBN 9781611972511.

Karl Pearson F.R.S. Liii. on lines and planes of closest fit to systems of points in space. *Philosophical Magazine Series 1*, 2:559–572, 1901.

Dan Garber and Elad Hazan. Fast and simple pca via convex optimization. *ArXiv*, abs/1509.05647, 2015. URL `https://arxiv.org/abs/1509.05647`.

Dan Garber, Ohad Shamir, and Nathan Srebro. Communication-efficient algorithms for distributed stochastic principal component analysis. In *ICML*, pages 1203–1212, 2017. URL `http://proceedings.mlr.press/v70/garber17a.html`.

Andreas Grammenos, Rodrigo Mendoza Smith, Jon Crowcroft, and Cecilia Mascolo. Federated principal component analysis. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 6453–6464. Curran Associates, Inc., 2020. URL `https://proceedings.neurips.cc/paper/2020/file/47a658229eb2368a99f1d032c8848542-Paper.pdf`.

Anne Greenbaum, Ren-Cang Li, and Michael L. Overton. First-order perturbation theory for eigenvalues and eigenvectors. *SIAM Review*, 62(2):463–482, 2020. doi: 10.1137/19M124784X.

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics, 2009.

David Hong, Fan Yang, Jeffrey A. Fessler, and Laura Balzano. Optimally weighted pca for high-dimensional heteroscedastic data. In *Arxiv*, 2021. URL `https://arxiv.org/pdf/1810.12862.pdf`.

H. Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24:417–441, 1933. doi: http://dx.doi.org/10.1037/h0071325.

Long-Kai Huang and Sinno Pan. Communication-efficient distributed PCA by Riemannian optimization. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 4465–4474. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/huang20e.html.

Hervé Jégou and Ondřej Chum. Negative evidences and co-occurences in image retrieval: The benefit of pca and whitening. In Andrew Fitzgibbon, Svetlana Lazebnik, Pietro Perona, Yoichi Sato, and Cordelia Schmid, editors, *Computer Vision – ECCV 2012*, pages 774–787, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg. ISBN 978-3-642-33709-3.

William Kahan. The nearest orthogonal or unitary matrix. *W. Kahan's Supplementary Notes for Math. 128*, 2011.

Raed Kontar, Shiyu Zhou, Chaitanya Sankavaram, Xinyu Du, and Yilu Zhang. Nonparametric-condition-based remaining useful life prediction incorporating external factors. *IEEE Transactions on Reliability*, 67(1):41–52, 2017.

Raed Kontar, Shiyu Zhou, Chaitanya Sankavaram, Xinyu Du, and Yilu Zhang. Nonparametric modeling and prognosis of condition monitoring signals using multivariate gaussian convolution processes. *Technometrics*, 60(4):484–496, 2018.

Raed Kontar, Naichen Shi, Xubo Yue, Seokhyun Chung, Eunshin Byon, Mosharaf Chowdhury, Jionghua Jin, Wissam Kontar, Neda Masoud, Maher Nouiehed, et al. The internet of federated things (ioft). *IEEE Access*, 9:156071–156113, 2021.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

V. Kulkarni, M. Kulkarni, and A. Pant. Survey of personalization techniques for federated learning. In *2020 Fourth World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4)*, pages 794–797, 2020. doi: 10.1109/WorldS450073.2020.9210355.

Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, and Virginia Smith Ameet Talwalkar. Federated optimization in heterogeneous networks. *Proceedings of the 3rd MLSys Conference*, 2018a.

Wei Li, Minjun Peng, and Qingzhong Wang. Fault detectability analysis in pca method during condition monitoring of sensors in a nuclear power plant. *Annals of Nuclear Energy*, 119:342–351, 2018b.

Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. On the convergence of fedavg on non-iid data. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=HJxNAnVtDS.

Yingyu Liang, Maria-Florina F Balcan, Vandana Kanchanapally, and David Woodruff. Improved distributed principal component analysis. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014. URL https://proceedings.neurips.cc/paper/2014/file/52947e0ade57a09e4a1386d08f17b656-Paper.pdf.

Huikang Liu, Anthony Man-Cho So, and Weijie Wu. Quadratic optimization with orthogonality constraint: Explicit lojasiewicz exponent and linear convergence of retraction-based line-search and stochastic variance-reduced gradient methods. In *Proceedings of the 33rd International Conference on Machine Learning*, 2019.

Eric F Lock, Katherine A Hoadley, James Stephen Marron, and Andrew B Nobel. Joint and individual variation explained (jive) for integrated analysis of multiple data types. *The annals of applied statistics*, 7(1):523, 2013.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017.

John Novembre and Matthew Stephens. Interpreting principal component analyses of spatial population genetic variation. *Nature genetics*, 40(5):646–649, 2008.

Shigeyuki Oba, Motoaki Kawanabe, Klaus-Robert Müller, and Shin Ishii. Heterogeneous component analysis. In J. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems*, volume 20. Curran Associates, Inc., 2007. URL `https://proceedings.neurips.cc/paper/2007/file/a8abb4bb284b5b27aa7cb790dc20f80b-Paper.pdf`.

Francesc Pozo, Yolanda Vidal, and Óscar Salgado. Wind turbine condition monitoring strategy through multiway pca and multivariate inference. *Energies*, 11(4):749, 2018.

Yongming Qu, George Ostrouchov, Nagiza Samatova, and Al Geist. Principal component analysis for dimension reduction in massive distributed data sets. In *Knowledge and Information Systems - KAIS*, 04 2002.

David Reich, Alkes L Price, and Nick Patterson. Principal component analysis of genetic data. *Nature genetics*, 40(5):491–492, 2008.

Alessandro Rinaldo. Lecture notes in advanced statistical theory, Fall 2019.

Felix Sattler, Klaus-Robert Müller, and Wojciech Samek. Clustered federated learning: Model-agnostic distributed multi-task optimization under privacy constraints. *arXiv preprint arXiv:1910.01991*, 2019.

Ruoyu Sun and Ziquan Luo. Guaranteed matrix completion via non-convex factorization. *FOCS*, 2015.

Cheng Tang. Exponentially convergent stochastic k-pca without variance reduction. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Antoine Vacavant, Thierry Chateau, Alexis Wilhelm, and Laurent Lequievre. A benchmark dataset for outdoor foreground/background extraction. In *ACCV Workshops*, 2012. URL `https://api.semanticscholar.org/CorpusID:10634625`.

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*, volume 26. Curran Associates, Inc., 2013. URL https://proceedings.neurips.cc/paper/2013/file/81e5f81db77c596492e6f1a5a792ed53-Paper.pdf.

Martin J. Wainwright. *High-Dimensional Statistics: A Non-Asymptotic Viewpoint*. Cambridge University Press, 2019. doi: 10.1017/9781108627771.

Huan Xu, Constantine Caramanis, and Sujay Sanghavi. Robust pca via outlier pursuit. *IEEE Transactions on Information Theory*, 58(5):3047–3064, 2012. doi: 10.1109/TIT.2011.2173156.

Kiyoung Yang and Cyrus Shahabi. A pca-based similarity measure for multivariate time series. In *Proceedings of the 2nd ACM international workshop on Multimedia databases*, pages 65–74, 2004.

Guoxu Zhou, Andrzej Cichocki, Yu Zhang, and Danilo P Mandic. Group component analysis for multiblock data: Common and individual feature extraction. *IEEE transactions on neural networks and learning systems*, 27(11):2426–2439, 2015.

Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. A comprehensive survey on transfer learning. *Proceedings of the IEEE*, 109(1):43–76, 2020.

Jiacheng Zhuo, Jeongyeol Kwon, Nhat Ho, and Constantine Caramanis. On the computational and statistical complexity of over-parameterized matrix sensing. *arXiv preprint arXiv:2102.02756*, 2021.

Hui Zou, Trevor Hastie, and Robert Tibshirani. Sparse principal component analysis. *Journal of Computational and Graphical Statistics*, 15(2):265–286, 2006. doi: 10.1198/106186006X113430.