Journal of Machine Learning Research 23 (2022) 1-41 Submitted 4/21; Revised 7/22; Published 8/22

Projected Robust PCA with Application to Smooth Image Recovery

Long Feng LFENG@HKU.HK
Department of Statistics & Actuarial Science
The University of Hong Kong
Pokfulam, Hong Kong

Junhui Wang JUNHUIWANG@CUHK.EDU.HK
Department of Statistics
The Chinese University of Hong Kong
Shatin, Hong Kong

Editor: Genevera Allen

Abstract

Most high-dimensional matrix recovery problems are studied under the assumption that the target matrix has certain intrinsic structures. For image data related matrix recovery problems, approximate low-rankness and smoothness are the two most commonly imposed structures. For approximately low-rank matrix recovery, the robust principal component analysis (PCA) is well-studied and proved to be effective. For smooth matrix problem, 2d fused Lasso and other total variation based approaches have played a fundamental role. Although both low-rankness and smoothness are key assumptions for image data analysis, the two lines of research, however, have very limited interaction. Motivated by taking advantage of both features, we in this paper develop a framework named projected robust PCA (PRPCA), under which the low-rank matrices are projected onto a space of smooth matrices. Consequently, a large class of image matrices can be decomposed as a low-rank and smooth component plus a sparse component. A key advantage of this decomposition is that the dimension of the core low-rank component can be significantly reduced. Consequently, our framework is able to address a problematic bottleneck of many low-rank matrix problems: singular value decomposition (SVD) on large matrices. Theoretically, we provide explicit statistical recovery guarantees of PRPCA and include classical robust PCA as a special case.

Keywords: Image analysis, Robust PCA, Low-rankness, smoothness, Interpolation matrices.

1. Introduction

In the past decade, high-dimensional matrix recovery problems have drawn numerous attentions in the communities of statistics, computer science and electrical engineering due to its wide application, particularly in image and video data analysis. Notable problems include face recognition (Parkhi et al., 2015), motion detection in surveillance video (Candès et al., 2011), brain structure study through fMRI (Maldjian et al., 2003), etc. In general, most studies on high-dimensional matrix recovery problems are built upon the assumption

that the target matrix has certain intrinsic structures. For image data related problems, the two most commonly imposed structures are 1) approximate low-rankness and 2) smoothness.

The approximate low-rankness refers to the property that the target matrix can be decomposed as a low-rank component plus a sparse component. Such a decomposition has been intensively studied in the literature, for example, Candès et al. (2011) who proposed the *robust principal component analysis* (RPCA), Chandrasekaran et al. (2011) who developed a notion of rank-sparsity incoherence, etc. The RPCA was originally studied under the noiseless setting and has been extended to various other settings. For instance, Zhou et al. (2010) studied RPCA in a noise case, Vaswani et al. (2018) considered RPCA for robust subspace tracking, etc. We refer to Bouwmans et al. (2018) for a survey of applications of RPCA in computer vision. Moreover, the low-rank and sparse decomposition has also been intensively studied for matrix completion problems with partially observed entries, e.g., Jain and Dhillon (2013); Wright et al. (2013); Chen (2015); Klopp et al. (2017); Chen et al. (2020).

On the other hand, when a matrix is believed to be smooth, *Total Variation* (TV) based approach has played a fundamental role since the pioneering work of Rudin et al. (1992) and Rudin and Osher (1994). The TV has been proven to be effective in preserving image boundaries/edges. In statistics community, a well-studied TV approach is the *2d fused Lasso* (Tibshirani et al., 2005), which penalizes the total absolute difference of adjacent matrix entries using ℓ_1 norm. The 2d fused Lasso has been shown to be efficient when the target matrix is piecewise smooth. More recently, a TV based approach was also used in image-on-scalar regression to promote the piecewise smoothness of image coefficients (Wang et al., 2017).

Although both low-rankness and smoothness are key assumptions for image data analysis, the two lines of study, however, have very limited interaction. On the other hand, the matrices that are both approximately low-rank and smooth not only commonly exist in image data, but also exist in video analysis. Consider a stacked video surveillance matrix—obtained by stacking each video frame into a matrix column. Candès et al. (2011) demonstrates that this matrix is approximately low-rank: the low-rank component corresponding to the stationary background and the sparse component corresponding to the moving objects. However, a critical but often neglected fact is that the low-rank component is roughly column-wise smooth. In other words, each column of the low-rank matrix is roughly the same—because they all represent the same background. In this case, the original matrix is the superposition of a low-rank and smooth component and a sparse component. How to effectively take advantage of both assumptions? This motivates our study in this paper.

1.1 This paper

We propose the following model to build a bridge between the approximate low-rankness and smoothness in high-dimensional matrix recovery problem,

$$\begin{aligned} \boldsymbol{Z} &= \boldsymbol{\Theta} + \boldsymbol{E}, \\ \boldsymbol{\Theta} &= \boldsymbol{P}\boldsymbol{X}_0\boldsymbol{Q}^\top + \boldsymbol{Y}_0. \end{aligned} \tag{1}$$

Here $\boldsymbol{Z} \in \mathbb{R}^{M\times N}$ is the observed matrix with unknown mean matrix $\boldsymbol{\Theta}$ and noise $\boldsymbol{E}$, $\boldsymbol{X}_0 \in \mathbb{R}^{m\times n}$ is an unknown **low-rank** matrix, $\boldsymbol{Y}_0 \in \mathbb{R}^{M\times N}$ is an unknown **sparse** matrix,

$\boldsymbol{P} \in \mathbb{R}^{M\times m}$ and $\boldsymbol{Q} \in \mathbb{R}^{N\times n}$ are respectively certain "row-smoother" and "column-smoother" matrix that will be discussed in detail later. The target is to recover the unknown matrices $\boldsymbol{X}_0$, $\boldsymbol{Y}_0$ and the resulting $\boldsymbol{\Theta}$. We refer to model (1) as the Projected Robust Principal Component Analysis (PRPCA) since the low-rank component in model (1) is projected onto a constrained domain.

We study the following convex optimization problem to estimate the pair $(\boldsymbol{X}_0, \boldsymbol{Y}_0)$ and account for the low-rankness of $\boldsymbol{X}_0$ and sparseness of $\boldsymbol{Y}_0$ in PRPCA,

$$(\widehat{\boldsymbol{X}}, \widehat{\boldsymbol{Y}}) \in \underset{\substack{\boldsymbol{X}\in\mathbb{R}^{n\times m}\\ \boldsymbol{Y}\in\mathbb{R}^{N\times M}}}{\arg\min} \frac{1}{2}\left\|\boldsymbol{Z} - \boldsymbol{P}\boldsymbol{X}\boldsymbol{Q}^\top - \boldsymbol{Y}\right\|_F^2 + \lambda_1\|\boldsymbol{X}\|_* + \lambda_2\|\boldsymbol{Y}\|_{\mathrm{vec}(1)}, \tag{2}$$

where λ_1 and λ_0 are the regularization parameters, $\|\cdot\|_*$ is the nuclear norm (sum of singular values) and $\|\cdot\|_{\mathrm{vec}(1)}$ is the entrywise ℓ_1-norm.

With different pairs of $(\boldsymbol{P}, \boldsymbol{Q})$ and sparsity assumption on $\boldsymbol{Y}_0$, model (1) includes many popular existing models. For example, when $\boldsymbol{P} = \boldsymbol{I}_N$ and $\boldsymbol{Q} = \boldsymbol{I}_M$ are identity matrices and $\boldsymbol{Y}_0$ is entrywise sparse, model (1) reduces to the classical RPCA and the convex optimization problem (2) reduces to the noisy version of principal component pursuit (PCP, Candès et al. 2011). When $\boldsymbol{P}$ is a general matrix, $\boldsymbol{Q}$ is the identity matrix and $\boldsymbol{Y}_0$ is columnwise sparse, model (1) reduces to the robust reduced rank regression studied by She and Chen (2017). Under such case, the $\|\cdot\|_{\mathrm{vec}(1)}$ norm in (2) can be replaced by a mixed $\|\cdot\|_{2,1}$ to account for the columnwise sparsity of $\boldsymbol{Y}_0$ and our analysis below can be rephrased easily. Here for any matrix $\boldsymbol{M}$, $\|\boldsymbol{M}\|_{2,1} = \sum_j \|M_{\cdot,j}\|_2$.

As mentioned before, our study of PRPCA is motivated by taking advantage of both low-rankness and smoothness features of image data. In this paper, we show that the recovery accuracy of RPCA can be improved significantly when introducing the "row-smoother" and "column-smoother" matrices $\boldsymbol{P}$ and $\boldsymbol{Q}$. Beyond recovery accuracy, the PRPCA also brings computational advantages compared to RPCA. Indeed, the computation of RPCA or other low-rank matrix related problems usually involves iterations of singular value decomposition (SVD), which could be a problematic bottleneck for large matrices (Hastie et al., 2015). For smooth matrix recovery, the TV based approaches also pose great computational challenges. On the contrary, when we are able to combine the low-rankness with smoothness, problem (2) allows us to find a low-rank matrix of dimension $n \times m$, rather than the original matrix with dimension $N \times M$. As to be demonstrated in Section 2.1, the "smoother" matrices we considered are mostly "tall and thin" matrices, i.e., $N \geqslant n$ and $M \geqslant m$. That is to say, we are allowed to find a much smaller low-rank matrix and thus the computational cost is reduced. A real image data analysis in Section 7 shows that the computation of PRPCA with $n = N/2$ and $m = M/2$ could be more than 10 times faster than RPCA while also achieving better recovery accuracy.

More specifically, we in this paper study the theoretical properties of model (1) and the convex optimization problem (2) with general matrices $\boldsymbol{P}$ and $\boldsymbol{Q}$. Specifically, we provide explicit theoretical error bounds for the estimation of the sparse component $\boldsymbol{Y}_0$ and low-rank component $\boldsymbol{P}\boldsymbol{X}_0\boldsymbol{Q}^\top$ with general noise matrix $\boldsymbol{E}$. Our results include Hsu et al. (2011) as a special case, where the statistical properties of classical RPCA is studied. The key in our analysis of (2) is a careful construction of a dual certificate through a least-squares method. In addition, a proximal gradient algorithm and its accelerated version are

developed to implement (2). Furthermore, a comprehensive simulation study along with a real image data analysis further demonstrate the superior performance of PRPCA in terms of both recovery accuracy and computational advantage.

1.2 Notations and organizations

A variety of matrix and vector norms are used in this paper. For a vector $\boldsymbol{v} = (v_1, ...v_p)^\top$, $\|\boldsymbol{v}\|_q = \sum_{1\leqslant j\leqslant p}(\|v_j\|^q)^{1/q}$ is the ℓ_q norm, $\|\boldsymbol{v}\|_0$ is the ℓ_0 norm (number of nonzero entries). For a matrix $\boldsymbol{M} = \{M_{i,j}, 1 \leqslant i \leqslant n, 1 \leqslant j \leqslant m\}$, $\|\boldsymbol{M}\|_{\mathrm{vec}(q)} = \left(\sum_{i,j}|M_{i,j}|^q\right)^{1/q}$ is the entry-wise ℓ_q-norm. In particular, $\|\boldsymbol{M}\|_{\mathrm{vec}(2)}$ is the Frobenius norm and also denoted as $\|\boldsymbol{M}\|_F$, $\|\boldsymbol{M}\|_{\mathrm{vec}(0)}$ is the number of non-zero entries in $\boldsymbol{M}$. Moreover, $\|\boldsymbol{M}\|_q = [\sum_i \sigma_i^q(\boldsymbol{M})]^{1/q}$ is the Schatten q-norm, where $\sigma_i(\boldsymbol{M})$ are the singular values. In particular, $\|\boldsymbol{M}\|_1$ is the nuclear norm (sum of the singular values) and also denoted as $\|\boldsymbol{M}\|_*$. Furthermore, $\|\boldsymbol{M}\|_{2,1} = \sum_j \|M_{\cdot,j}\|_2$ is a mixed $\ell_{2,1}$ norm. In addition, $\boldsymbol{M}_{i,\cdot}$ is the i-th row of $\boldsymbol{M}$, $\boldsymbol{M}_{\cdot,j}$ is the j-th column, $\mathrm{vec}(\boldsymbol{M})$ is the vectorization of $\boldsymbol{M}$, $\sigma_{\min}(\boldsymbol{M})$ and $\sigma_{\max}(\boldsymbol{M})$ are the smallest and largest singular values, respectively, $\boldsymbol{M}^+$ is the Moore-Penrose inverse of $\boldsymbol{M}$, $\boldsymbol{M}^* = \boldsymbol{M}(\boldsymbol{M}^\top\boldsymbol{M})^{-1}\boldsymbol{M}^\top$ is the projection matrix (for full column rank matrix $\boldsymbol{M}$). Finally, we use $\boldsymbol{I}_n$ to denote an identity matrix of dimension $n \times n$, and $\otimes$ to denote the Kronecker product.

The rest of the paper is organized as follows. Section 2.1 introduces the interpolation matrices based PRPCA. In Section 3 we discuss the computation of (2) with proximal gradient algorithm. Section 4 provides our main theoretical results for PRPCA. In Section 5 we generalize PRPCA to a noiseless setting. Section 6 includes a comprehensive simulation study. In Section 7, we conduct two real data analysis: one image data and one surveillance video data. Section 8 includes conclusions and future directions.

2. Projected Robust PCA with Smoothing Matrices

In this section, we consider two types of smoothing mechanisms on the low-rank matrix: data-independent smoothing and data-dependent smoothing. Moreover, we provide general assumptions of the "smoother matrices" $\boldsymbol{P}$ and $\boldsymbol{Q}$ that our analysis can be applied.

2.1 The data-independent smoothing and interpolation matrix

Definition 1 *Let N be an even integer and $n = N/2$ [1] . We define the normalized interpolation matrix $\boldsymbol{J}_N$ of dimension $N \times n$ as*

$$\boldsymbol{J}_N = \frac{1}{2}\begin{pmatrix} 2 & 0 & 0 & 0 & \cdots & 0 & 0 \\ 2 & 0 & 0 & 0 & \cdots & 0 & 0 \\ 1 & 1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 2 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 1 & 1 & 0 & \cdots & 0 & 0 \\ & & & & \cdots & & \\ 0 & 0 & 0 & 0 & \cdots & 0 & 2 \end{pmatrix} \in \mathbb{R}^{N\times n}, \tag{3}$$

1. When N is odd, we can let $n = (N+1)/2$ and a slightly different interpolation matrix can be defined in a similar way.

i.e., the j-th column of $\boldsymbol{J}_N$ is

$$\{\boldsymbol{J}_N\}_{\cdot,j} = (\underbrace{0,\ldots,0}_{2(j-1)}, \frac{1}{2}, 1, \frac{1}{2}, \underbrace{0,\ldots,0}_{N-2j-1})^\top, \quad j = 2, ..., n-1,$$

$\{\boldsymbol{J}_N\}_{\cdot,1} = (1, 1, \frac{1}{2}, 0, ..., 0)$ *and* $\{\boldsymbol{J}_N\}_{\cdot,n} = (0, ..., 0, \frac{1}{2}, 1)$.

The interpolation matrices $\boldsymbol{J}_N$ and $\boldsymbol{J}_M$ play the role of "row smoother" and "column smoother", respectively. That is to say, when $\boldsymbol{P} = \boldsymbol{J}_N$, $\boldsymbol{PU} \in \mathbb{R}^{N\times M}$ is a row-wisely smooth matrix for any matrix $\boldsymbol{U} \in \mathbb{R}^{n\times M}$, i.e., except the first row (boundary effect), any odd row of $\boldsymbol{PU}$ is the average of adjacent two rows,

$$(\boldsymbol{PU})_{i,\cdot} = \frac{1}{2}\left\{(\boldsymbol{PU})_{i-1,\cdot} + (\boldsymbol{PU})_{i+1,\cdot}\right\}, \tag{4}$$

for row $i = 3, 5, \ldots, N-1$, and for the boundary row,

$$(\boldsymbol{PU})_{1,\cdot} = (\boldsymbol{PU})_{2,\cdot}. \tag{5}$$

Also, when $\boldsymbol{Q} = \boldsymbol{J}_M$, $\boldsymbol{V}\boldsymbol{J}_M^\top \in \mathbb{R}^{N\times M}$ is a column-wisely smooth matrix for any matrix $\boldsymbol{V} \in \mathbb{R}^{N\times m}$:

$$(\boldsymbol{V}\boldsymbol{Q}^\top)_{\cdot,j} = \frac{1}{2}\left\{(\boldsymbol{V}\boldsymbol{Q}^\top)_{\cdot,j-1} + (\boldsymbol{V}\boldsymbol{Q}^\top)_{\cdot,j+1}\right\},$$

for column $j = 3, 5, \ldots, M-1$, and for the boundary column,

$$(\boldsymbol{V}\boldsymbol{Q}^\top)_{\cdot,1} = (\boldsymbol{V}\boldsymbol{Q}^\top)_{\cdot,2}.$$

As a consequence, $\boldsymbol{P}\boldsymbol{X}_0\boldsymbol{Q}^\top$ in model (1) is a smooth matrix both row-wisely and column-wisely.

We note that the interpolation matrix has also been used in other image analysis literature. For example, when implementing RPCA, Hovhannisyan et al. (2019) used interpolation matrix to build connections between the original "fine" model and a smaller "coarse" model and reduce the computational burden of RPCA. Their work is mainly from a computational perspective, but the principle behind is the same: by applying SVD in models of lower-dimension, the computational burden can be significantly reduced.

Indeed, by introducing the smoothing matrices $\boldsymbol{P}$ and $\boldsymbol{Q}$, the PRPCA enjoys significant computational advantages compared to the standard RPCA. When $\boldsymbol{P}$ and $\boldsymbol{Q}$ are interpolation matrices, we are allowed to find a low-rank matrix of dimension $N/2 \times M/2$, rather than the original matrix with much higher-dimension $N \times M$. Considering that computing a low-rank matrix usually involves SVD, the computational advantage that (2) brings is even more significant. More aggressively, we may further interpolate the low-rank matrix by using a double interpolation matrix

$$(\boldsymbol{P}, \boldsymbol{Q}) = (\boldsymbol{J}_N \times \boldsymbol{J}_{N/2},\ \boldsymbol{J}_M \times \boldsymbol{J}_{M/2}).$$

This would allow us to find a low-rank matrix of even lower dimension, $N/4 \times M/4$, and further reduce the computational burden.

The smoothing mechanism here is data-independent in the sense that the odd row (column) of the low-rank matrix is an equal-weight average of the adjacent two rows (columns). Such mechanism can be modified to a data-dependent smoothing approach, which will be introduced in the next subsection. On the other hand, we observe that the data-independent averaging performs consistently well across a large range of image recovery problems in our simulation and real data analysis. We refer to Sections 6 and 7 for more details.

2.2 The data-dependent smoothing with robust linear regression

In this subsection, we introduce a data dependent smoothing mechanism based on robust linear models. We start with introducing the data-dependent interpolation matrix.

Definition 2 *Let N be an even integer and $n = N/2$. Let w_{ur}^i, w_{lr}^i, $i = 1, \ldots, n-1$ represent the i-th unknown upper-row weight and lower-row weight, respectively. We define the generalized interpolation matrix $\boldsymbol{K}_N$ of dimension $N \times n$ as*

$$\boldsymbol{K} = \boldsymbol{K}_N(\boldsymbol{W}) = \begin{pmatrix} 1 & 0 & 0 & 0 & \cdots & 0 & 0 \\ 1 & 0 & 0 & 0 & \cdots & 0 & 0 \\ w_{ur}^1 & w_{lr}^1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & w_{ur}^2 & w_{lr}^2 & 0 & \cdots & 0 & 0 \\ & & & & \cdots & & \\ 0 & 0 & 0 & 0 & \cdots & 0 & 1 \end{pmatrix} \in \mathbb{R}^{N \times n}, \tag{6}$$

i.e., the j-th column of $\boldsymbol{K}_N$ is

$$\{\boldsymbol{K}_N\}_{\cdot,j} = (\underbrace{0, \ldots, 0}_{2(j-1)}, w_l^{j-1},\ 1,\ w_u^j,\ \underbrace{0, \ldots, 0}_{N-2j-1})^\top, \quad j = 2, ..., n-1,$$

$\{\boldsymbol{K}_N\}_{\cdot,1} = (1, 1, w_u^1, 0, ..., 0)$ *and* $\{\boldsymbol{K}_N\}_{\cdot,n} = (0, ..., 0, w_{lr}^{n-1}, 1)$.

The weights w_{ur}^i, w_{lr}^i are unknown parameters and need to be estimated. Clearly, the normalized interpolation matrices $\boldsymbol{J}_N$ defined in the previous section is a special case of $\boldsymbol{K}_N$ with the weights $w_{ur}^i = 0.5$ and $w_{lr}^i = 0.5$. As in (4) and (5), when $\boldsymbol{P} = \boldsymbol{K}_N$, $\boldsymbol{PU}$ is a weighted smoothing matrix for any matrix $\boldsymbol{U}$ of dimension $n \times M$:

$$(\boldsymbol{PU})_{i',\cdot} = w_{ur}^i (\boldsymbol{PU})_{i'-1,\cdot} + w_{lr}^i (\boldsymbol{PU})_{i'+1,\cdot},$$

and for the boundary row,

$$(\boldsymbol{PU})_{1,\cdot} = (\boldsymbol{PU})_{2,\cdot}.$$

Similarly, we may define $\boldsymbol{K}_M$ in the same way as $\boldsymbol{K}_N$, but with w_{ur}^i, w_{lr}^i, $i = 1, \ldots, n-1$ replaced by w_{lc}^j, w_{rc}^j, $j = 1, \ldots, m-1$, representing left-column and right-column weights. Then, when $\boldsymbol{Q} = \boldsymbol{K}_M$, $\boldsymbol{VQ}^\top \in \mathbb{R}^{N \times M}$ is a column-wisely smooth matrix for any $\boldsymbol{V} \in \mathbb{R}^{N \times m}$.

Now we present the estimation procedure of w_{ur}^i and w_{lr}^i, or the row smoothing matrix $\boldsymbol{K}_N$. We only present the estimation of $\boldsymbol{K}_N$ as the column smoothing matrix $\boldsymbol{K}_M$, or w_{lc}^j,

w_{rc}^j, can be estimated in the same way. Let $\boldsymbol{\Delta} = \boldsymbol{P}\boldsymbol{X}_0\boldsymbol{Q}^\top$ denote the unknown low-rank and smooth matrix. Intuitively, if we treat the odd rows in $\boldsymbol{\Delta}$ as missing, then the values of w_{ur}^i and w_{lr}^i can be viewed as the linear weights when inserting the odd rows based on their neighborhood. Thus, it is ideal to estimate w_{ur}^i and w_{lr}^i based on $\boldsymbol{\Delta}$, in particular, the $(i'-1)$-th, i'-th and $(i'+1)$-th row of the matrix $\boldsymbol{\Delta}$, or $\boldsymbol{\Delta}_{(i'-1):(i'+1),}$, with $i' = 2i+1$. However, as $\boldsymbol{\Delta}$ is unobserved, w_{ur}^i and w_{lr}^i are unable to be estimated directly from $\boldsymbol{\Delta}$. While on the other hand, we note that $\boldsymbol{\Delta} = \boldsymbol{Z} - \boldsymbol{Y} - \boldsymbol{E}$. Considering that 1) $\boldsymbol{Y}$ is a sparse matrix and can be viewed as outliers from $\boldsymbol{\Delta}$, and 2) $\boldsymbol{E}$ is a noise matrix with small entry-wise magnitude, we propose to estimate w_{ur}^i and w_{lr}^i based on the observable $\boldsymbol{Z}$ through robust linear regression to account for the outliers $\boldsymbol{Y}$. Specifically, we consider the following minimization problem:

$$(\hat{w}_{ur}^i, \hat{w}_{lr}^i) = \underset{w_{ur}^i, w_{lr}^i}{\arg\min} \sum_{j=1}^{M} \rho(w_{ur}^i Z_{i'-1,j} + w_{lr}^i Z_{i'+1,j} - Z_{i',j}), \quad i' = 2i+1, \quad i = 1, \ldots, n-1.$$

Here $\rho(\cdot)$ is certain robust loss function. For example, we may take $\rho(\cdot)$ as the Huber loss (Huber, 1992), where

$$\rho(x) = \begin{cases} x^2, & \text{if } |x| \leqslant \kappa, \\ 2\kappa|x| - \kappa^2, & \text{if } |x| > \kappa. \end{cases}$$

The tuning parameter $\kappa > 0$ is assumed to be given in our estimation of w_{ur}^i and w_{lr}^i. When $\boldsymbol{P} = \boldsymbol{K}_N$ and $\boldsymbol{Q} = \boldsymbol{K}_M$, we have the following property.

Proposition 3 *Let $\boldsymbol{P} = \boldsymbol{K}_N \in \mathbb{R}^{N\times n}$ and $\boldsymbol{Q} = \boldsymbol{K}_M \in \mathbb{R}^{M\times m}$ be the matrices in Definition 2 with $N \geqslant 4$, $M \geqslant 4$ and any weights w_{ur}^i, w_{lr}^i, w_{lc}^j and w_{rc}^j. Then for any nonzero matrix $\boldsymbol{X}_0 \in \mathbb{R}^{n\times m}$,*

$$\|\boldsymbol{X}_0\|_* < \|\boldsymbol{P}\boldsymbol{X}_0\boldsymbol{Q}^\top\|_*. \tag{7}$$

When $\boldsymbol{\Theta}$ can be decomposed as in model (1), the RPCA is the optimization problem (2) with the nuclear penalty on $\boldsymbol{X}$ replaced by that on $\boldsymbol{P}\boldsymbol{X}\boldsymbol{Q}^\top$. Proposition 3 suggests that smaller penalty is applied in (2) compared to that of RPCA with the same λ_1.

2.3 PRPCA with general $\boldsymbol{P}$ and $\boldsymbol{Q}$

Although our study of PRPCA is motivated by smooth matrix analysis and resulting interpolation matrices, our results in fact work for general matrices $\boldsymbol{P}$ and $\boldsymbol{Q}$.

Specifically, we proceed our analysis to consider $\boldsymbol{P}$ and $\boldsymbol{Q}$ of full-column rank. Indeed, when the target is to recover $\boldsymbol{P}\boldsymbol{X}_0\boldsymbol{Q}^\top$ as a whole instead of $\boldsymbol{X}_0$, it is sufficient to consider $\boldsymbol{P}$ and $\boldsymbol{Q}$ of full column rank. This can be seen from the following arguments. For any $\boldsymbol{P} \in \mathbb{R}^{N\times n}$ and $\boldsymbol{Q} \in \mathbb{R}^{M\times m}$, there exist $r_1 \leqslant \min(N, n)$, $r_2 \leqslant \min(M, m)$ and full column rank matrices $\boldsymbol{P}_0 \in \mathbb{R}^{N\times r_1}$, $\boldsymbol{Q}_0 \in \mathbb{R}^{M\times r_2}$ such that

$$\boldsymbol{P} = \boldsymbol{P}_0\boldsymbol{\Lambda}, \quad \boldsymbol{Q} = \boldsymbol{Q}_0\boldsymbol{\Omega}$$

hold for some $\boldsymbol{\Lambda} \in \mathbb{R}^{r_1 \times n}$ and $\boldsymbol{\Omega} \in \mathbb{R}^{r_2 \times m}$. As a result, an alternative representation of model (1) with full column rank matrices $\boldsymbol{P}_0$ and $\boldsymbol{Q}_0$ is

$$\boldsymbol{Z} = \boldsymbol{P}_0(\boldsymbol{\Lambda}\boldsymbol{X}_0\boldsymbol{\Omega}^\top)\boldsymbol{Q}_0^\top + \boldsymbol{Y}_0 + \boldsymbol{E}.$$

Here the columns of $\boldsymbol{P}_0$ (or $\boldsymbol{Q}_0$) can be viewed as the "factors" of $\boldsymbol{P}$ (or $\boldsymbol{Q}$). This confirms the sufficiency of considering PRPCA with full-column rank matrices $\boldsymbol{P}$ and $\boldsymbol{Q}$. In the following sections, we derive properties of PRPCA with general $\boldsymbol{P}$ and $\boldsymbol{Q}$ of full-column rank.

3. Computation with Proximal Gradient Algorithm

Given $\boldsymbol{P}$ and $\boldsymbol{Q}$, the problem (2) is a convex optimization problem. In this section, we show that it can be solved easily through a proximal gradient algorithm.

We first denote the loss and penalty function in problem (2) as

$$\mathcal{L}(\boldsymbol{X}, \boldsymbol{Y}) = \frac{1}{2} \left\| \boldsymbol{Z} - \boldsymbol{P}\boldsymbol{X}\boldsymbol{Q}^\top - \boldsymbol{Y} \right\|_F^2, \tag{8}$$

and

$$\mathcal{P}(\boldsymbol{X}, \boldsymbol{Y}) = \lambda_1 \|\boldsymbol{X}\|_* + \lambda_2 \|\boldsymbol{Y}\|_{\mathrm{vec}(1)}, \tag{9}$$

respectively. Also, note that if we let $\boldsymbol{A} = \boldsymbol{Q} \otimes \boldsymbol{P}$, the loss function (8) could be written as

$$\mathcal{L}(\boldsymbol{X}, \boldsymbol{Y}) = \frac{1}{2} \left\| \boldsymbol{A}\mathrm{vec}(\boldsymbol{X}) + \mathrm{vec}(\boldsymbol{Y}) - \mathrm{vec}(\boldsymbol{Z}) \right\|_2^2. \tag{10}$$

To minimize $\mathcal{L}(\boldsymbol{X}, \boldsymbol{Y}) + \mathcal{P}(\boldsymbol{X}, \boldsymbol{Y})$, we utilize a variant of Nesterov's proximal-gradient method (Nesterov, 2013), which iteratively updates

$$(\widehat{\boldsymbol{X}}_{k+1}, \widehat{\boldsymbol{Y}}_{k+1}) \leftarrow \underset{\boldsymbol{X}, \boldsymbol{Y}}{\arg\min}\, \psi(\boldsymbol{X}, \boldsymbol{Y} | \widehat{\boldsymbol{X}}_k, \widehat{\boldsymbol{Y}}_k), \tag{11}$$

where

$$\begin{aligned}
&\psi(\boldsymbol{X}, \boldsymbol{Y} | \widehat{\boldsymbol{X}}_k, \widehat{\boldsymbol{Y}}_k) \\
=&\mathcal{L}(\widehat{\boldsymbol{X}}_k, \widehat{\boldsymbol{Y}}_k) + \langle \nabla_{\boldsymbol{X}} \mathcal{L}(\widehat{\boldsymbol{X}}_k, \widehat{\boldsymbol{Y}}_k), \boldsymbol{X} - \widehat{\boldsymbol{X}}_k \rangle + \langle \nabla_{\boldsymbol{Y}} \mathcal{L}(\widehat{\boldsymbol{X}}_k, \widehat{\boldsymbol{Y}}_k), \boldsymbol{Y} - \widehat{\boldsymbol{Y}}_k \rangle \\
&+ \frac{L_k}{2} \left(\|\boldsymbol{X} - \widehat{\boldsymbol{X}}_k\|_F^2 + \|\boldsymbol{Y} - \widehat{\boldsymbol{Y}}_k\|_F^2 \right) + \mathcal{P}(\boldsymbol{X}, \boldsymbol{Y}),
\end{aligned}$$

L_k is the step size parameter at step k, $\nabla_{\boldsymbol{X}} \mathcal{L}(\widehat{\boldsymbol{X}}_k, \widehat{\boldsymbol{Y}}_k)$ and $\nabla_{\boldsymbol{Y}} \mathcal{L}(\widehat{\boldsymbol{X}}_k, \widehat{\boldsymbol{Y}}_k)$ are the gradients

$$\begin{aligned}
\nabla_{\boldsymbol{X}} \mathcal{L}(\widehat{\boldsymbol{X}}_k, \widehat{\boldsymbol{Y}}_k) &= \boldsymbol{P}^\top(\boldsymbol{P}\widehat{\boldsymbol{X}}_k\boldsymbol{Q}^\top + \widehat{\boldsymbol{Y}}_k - \boldsymbol{Z})\boldsymbol{Q}, \\
\nabla_{\boldsymbol{Y}} \mathcal{L}(\widehat{\boldsymbol{X}}_k, \widehat{\boldsymbol{Y}}_k) &= \boldsymbol{P}\widehat{\boldsymbol{X}}_k\boldsymbol{Q}^\top + \widehat{\boldsymbol{Y}}_k - \boldsymbol{Z}.
\end{aligned}$$

The proximal function $\psi(\boldsymbol{X}, \boldsymbol{Y} | \widehat{\boldsymbol{X}}_k, \widehat{\boldsymbol{Y}}_k)$ is much easier to optimize compared to $\mathcal{L}(\boldsymbol{X}, \boldsymbol{Y}) + \mathcal{P}(\boldsymbol{X}, \boldsymbol{Y})$. In fact, a closed-form expression is available for the updates.

$$\widehat{\boldsymbol{X}}_{k+1} = \mathcal{SVT}\left(\widehat{\boldsymbol{X}}_k - \frac{1}{L_k} \nabla_{\boldsymbol{X}} \mathcal{L}(\widehat{\boldsymbol{X}}_k, \widehat{\boldsymbol{Y}}_k);\ \frac{\lambda_1}{L_k} \right),$$

$$\widehat{\boldsymbol{Y}}_{k+1} \;=\; \mathcal{ST}\left(\widehat{\boldsymbol{Y}}_k - \frac{1}{L_k}\nabla_{\boldsymbol{Y}}\mathcal{L}(\widehat{\boldsymbol{X}}_k, \widehat{\boldsymbol{Y}}_k);\; \frac{\lambda_2}{L_k}\right),$$

where $\mathcal{SVT}$ and $\mathcal{ST}$ are the Singular Value Thresholding and Soft Thresholding operators with specifications below.

Given any non-negative number $\tau_1 \geqslant 0$ and any matrix $\boldsymbol{M}_1 \in \mathbb{R}^{n\times m}$ with singular value decomposition $\boldsymbol{M}_1 = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top$, where $\boldsymbol{\Sigma} = \mathrm{diag}(\{\sigma_i\}_{1\leqslant i\leqslant r})$, $\sigma_i \geqslant 0$, the SVT operater $\mathcal{SVT}(\cdot;\cdot)$, which was first introduced by Cai et al. (2010), is defined as

$$\begin{aligned}\mathcal{SVT}(\boldsymbol{M}_1, \tau_1) &= \underset{\boldsymbol{X}\in\mathbb{R}^{n\times m}}{\arg\min}\, \frac{1}{2}\|\boldsymbol{X} - \boldsymbol{M}_1\|_F^2 + \tau_1\|\boldsymbol{X}\|_* \\ &= \boldsymbol{U}\mathcal{D}_{\tau_1}(\boldsymbol{\Sigma})\boldsymbol{V}^\top,\end{aligned}$$

where $\mathcal{D}_{\tau_1}(\boldsymbol{\Sigma}) = \mathrm{diag}(\{\sigma_i - \tau_1\}_+)$. For any $\tau_2 \geqslant 0$ and any matrix $\boldsymbol{M}_2 \in \mathbb{R}^{N\times M}$, the ST operator $\mathcal{ST}(\cdot;\cdot)$ is defined as

$$\begin{aligned}\mathcal{ST}(\boldsymbol{M}_2; \tau_2) &= \underset{\boldsymbol{Y}\in\mathbb{R}^{N\times M}}{\arg\min}\, \frac{1}{2}\|\boldsymbol{Y} - \boldsymbol{M}_2\|_F^2 + \tau_2\|\boldsymbol{Y}\|_{\mathrm{vec}(1)} \\ &= \mathrm{sgn}(\boldsymbol{M}_2) \circ \left(|\boldsymbol{M}_2| - \tau_2 \mathbf{1_N}\mathbf{1_M}^\top\right)_+.\end{aligned}$$

We summarize the proximal gradient algorithm for PRPCA in Algorithm 1 below.

Algorithm 1: Proximal gradient for PRPCA

Given:	$\boldsymbol{Z} \in \mathbb{R}^{N\times M}$, $\boldsymbol{P} \in \mathbb{R}^{N\times n}$, $\boldsymbol{Q} \in \mathbb{R}^{M\times m}$, λ_1 and λ_2
Initialization:	$\widehat{\boldsymbol{X}}_0 = \widehat{\boldsymbol{X}}_{-1} = \mathbf{0_{n\times m}}$, $\widehat{\boldsymbol{Y}}_0 = \widehat{\boldsymbol{Y}}_{-1} = \mathbf{0_{N\times M}}$
Iteration:	$\boldsymbol{G}_k^Y = \boldsymbol{P}\widehat{\boldsymbol{X}}_k\boldsymbol{Q}^\top + \widehat{\boldsymbol{Y}}_k - \boldsymbol{Z}$
	$\boldsymbol{G}_k^X = \boldsymbol{P}^\top\boldsymbol{G}_k^Y\boldsymbol{Q}$
	$\widehat{\boldsymbol{X}}_{k+1} = \mathcal{SVT}\left(\widehat{\boldsymbol{X}}_k - (1/L^k)\boldsymbol{G}_k^X;\; (1/L_k)\lambda_1\right)$,
	$\widehat{\boldsymbol{Y}}_{k+1} = \mathcal{ST}\left(\widehat{\boldsymbol{Y}}_k - (1/L^k)\boldsymbol{G}_k^Y;\; (1/L_k)\lambda_2\right)$,

Note: L_k can be taken as the reciprocal of a Lipschitz constant for $\nabla\mathcal{L}(\boldsymbol{X}, \boldsymbol{Y})$ or determined by backtracking.

The proximal gradient algorithm for PRPCA iteratively implements SVT and ST. Note that in the SVT step, the singular value decomposition is implemented on $\widehat{\boldsymbol{X}}_k - (1/L_k)\nabla_{\boldsymbol{X}}\mathcal{L}(\widehat{\boldsymbol{X}}_k, \widehat{\boldsymbol{Y}}_k)$, which is of dimension $n \times m$. Compared to the RPCA problem which requires singular value decomposition on matrices of much larger dimension $N \times M$, the PRPCA greatly reduces the computational cost.

Algorithm 2: Accelerated proximal gradient for PRPCA

Given: $\boldsymbol{Z} \in \mathbb{R}^{N\times M}$, $\boldsymbol{P} \in \mathbb{R}^{N\times n}$, $\boldsymbol{Q} \in \mathbb{R}^{M\times m}$, λ_1 and λ_2

Initialization: $\widehat{\boldsymbol{X}}_0 = \widehat{\boldsymbol{X}}_{-1} = \mathbf{0}_{\mathbf{n}\times\mathbf{m}}$, $\widehat{\boldsymbol{Y}}_0 = \widehat{\boldsymbol{Y}}_{-1} = \mathbf{0}_{\mathbf{N}\times\mathbf{M}}$, $t_0 = t_1 = 1$

Iteration:

$$\begin{aligned}
&\boldsymbol{F}_k^X = \widehat{\boldsymbol{X}}_k + t_k^{-1}(t_{k-1}-1)(\widehat{\boldsymbol{X}}_k - \widehat{\boldsymbol{X}}_{k-1})\\
&\boldsymbol{F}_k^Y = \widehat{\boldsymbol{Y}}_k + t_k^{-1}(t_{k-1}-1)(\widehat{\boldsymbol{Y}}_k - \widehat{\boldsymbol{Y}}_{k-1})\\
&\boldsymbol{G}_k^Y = \boldsymbol{P}\widehat{\boldsymbol{X}}_k\boldsymbol{Q}^\top + \widehat{\boldsymbol{Y}}_k - \boldsymbol{Z}\\
&\boldsymbol{G}_k^X = \boldsymbol{P}^\top\boldsymbol{G}_k^Y\boldsymbol{Q}\\
&\widehat{\boldsymbol{X}}_{k+1} = \mathcal{SVT}\left(\boldsymbol{F}_k^X - (1/L^k)\boldsymbol{G}_k^X;\ (1/L_k)\lambda_1\right),\\
&\widehat{\boldsymbol{Y}}_{k+1} = \mathcal{ST}\left(\boldsymbol{F}_k^Y - (1/L^k)\boldsymbol{G}_k^Y;\ (1/L_k)\lambda_2\right),\\
&t_{k+1} = \{1 + (1+4t_k^2)^{1/2}\}/2
\end{aligned}$$

We shall note that the Algorithm 1 can be further accelerated in different ways. First, our approach is an Iterative Shrinkage-Thresholding Algorithm (ISTA), and it can be accelarated in a Fast ISTA (Beck and Teboulle, 2009, FISTA) style as in Algorithm 2. For all the simulation studies and real image data analysis in Section 6 and 7, we adopt the accelerated proximal gradient algorithm.

Second, we realize that various online versions of singular value decomposition are available, and thus it would be interesting to incorporate some online SVD in each update to further accelerate the computation of FISTA. As we discussed earlier, a problematic bottleneck of many low-rank matrix related problem is the SVD computation. Our algorithm allows us to implement SVD on a smaller matrix and reduce the computation burden. To combine the online strategy into our algorithm, a key issue is how to process the input matrix part-by-part. This would be an intersting direction for our future work.

4. Main Theoretical Results

In this section, we present our main theoretical results for recovering the PRPCA. Specifically, we provide sharp theoretical error bounds for the estimation of the low-rank and smooth component $\boldsymbol{P}\boldsymbol{X}_0\boldsymbol{Q}^\top$ and the sparse component $\boldsymbol{Y}_0$ when $\boldsymbol{P}$ and $\boldsymbol{Q}$ are correctly specified. Note that Hsu et al. (2011) studied the theoretical properties of RPCA, i.e., PRPCA with $\boldsymbol{P} = \boldsymbol{I}_N$ and $\boldsymbol{Q} = \boldsymbol{I}_M$ being identity matrices. Our results can be viewed as a generalization of theirs.

4.1 Technique preparations

For a target decomposition of $\boldsymbol{\Theta} = \boldsymbol{P}\boldsymbol{X}_0\boldsymbol{Q}^\top + \boldsymbol{Y}_0$, we consider the following spaces and projections related to $\boldsymbol{X}_0$ and $\boldsymbol{Y}_0$. We start with considering the low-rank component $\boldsymbol{X}_0$. Let $\mathcal{T}_0$ be the span of matrices with either the row space of $\boldsymbol{X}$ contained in that of $\boldsymbol{X}_0$ or

the column space of $\boldsymbol{X}$x contained in that of $\boldsymbol{X}_0$:

$$\begin{aligned}\mathcal{T}_0 =& T_0(\boldsymbol{X}_0)\\ =&\Big\{\boldsymbol{X}_1+\boldsymbol{X}_2 :\in \mathbb{R}^{n\times m}:\ \text{range}(\boldsymbol{X}_1)\subseteq \text{range}(\boldsymbol{X}_0),\\ &\quad \text{range}(\boldsymbol{X}_2^\top)\subseteq \text{range}(\boldsymbol{X}_0^\top)\Big\}. \end{aligned} \tag{12}$$

Let $\mathcal{P}_{\mathcal{T}_0}$ be the orthogonal projector to $\mathcal{T}_0$. Under the inner product $\langle \boldsymbol{A},\boldsymbol{B}\rangle = \text{tr}(\boldsymbol{A}^\top\boldsymbol{B})$, the projection is given by

$$\mathcal{P}_{\mathcal{T}_0}(\boldsymbol{M}) = \boldsymbol{U}_0\boldsymbol{U}_0^T\boldsymbol{M} + \boldsymbol{M}\boldsymbol{V}_0\boldsymbol{V}_0^\top - \boldsymbol{U}_0\boldsymbol{U}_0^\top\boldsymbol{M}\boldsymbol{V}_0\boldsymbol{V}_0^\top. \tag{13}$$

where $\boldsymbol{U}_0\in\mathbb{R}^{n\times r}$ and $\boldsymbol{V}_0\in\mathbb{R}^{m\times r}$ are the matrices of left and right orthogonal singular vectors corresponding to the nonzero singular values of $\boldsymbol{X}_0$, and r is the rank of $\boldsymbol{X}_0$.

Furthermore, let $\mathcal{T}$ be the span of matrices taking the form of $\boldsymbol{P}\boldsymbol{X}\boldsymbol{Q}^\top$, with either the row space of $\boldsymbol{X}$ contained in that of $\boldsymbol{X}_0$, or the column space of $\boldsymbol{X}$ contained in that of $\boldsymbol{X}_0$:

$$\begin{aligned}\mathcal{T} &= T(\boldsymbol{X}_0;\boldsymbol{P},\boldsymbol{Q})\\ &= \Big\{\boldsymbol{P}(\boldsymbol{X}_1+\boldsymbol{X}_2)\boldsymbol{Q}^\top :\in \mathbb{R}^{N\times M}:\ \text{range}(\boldsymbol{X}_1)\subseteq \text{range}(\boldsymbol{X}_0),\\ &\quad \text{range}(\boldsymbol{X}_2^\top)\subseteq \text{range}(\boldsymbol{X}_0^\top)\Big\}.\end{aligned}$$

Apparently, $\mathcal{T}$ reduces to $\mathcal{T}_0$ when $\boldsymbol{P}$ and $\boldsymbol{Q}$ are identity matrices. We further define the orthogonal projector onto $\mathcal{T}$ as $\mathcal{P}_T$:

$$\mathcal{P}_{\mathcal{T}}(\boldsymbol{M}) = \widetilde{\boldsymbol{U}}_0\widetilde{\boldsymbol{U}}_0^T\boldsymbol{M} + \boldsymbol{M}\widetilde{\boldsymbol{V}}_0\widetilde{\boldsymbol{V}}_0^\top - \widetilde{\boldsymbol{U}}_0\widetilde{\boldsymbol{U}}_0^\top\boldsymbol{M}\widetilde{\boldsymbol{V}}_0\widetilde{\boldsymbol{V}}_0^\top, \tag{14}$$

where $\widetilde{\boldsymbol{U}}_0\in\mathbb{R}^{N\times r}$ and $\widetilde{\boldsymbol{V}}_0\in\mathbb{R}^{M\times r}$ are the left singular matrices of $\boldsymbol{P}\boldsymbol{U}_0$ and $\boldsymbol{Q}\boldsymbol{V}_0$, respectively. Given such projections, we introduce a property that measures the sparseness of the singular vectors of $\boldsymbol{P}\boldsymbol{X}_0\boldsymbol{Q}^\top$:

$$\beta(\rho) = \rho^{-1}\|\widetilde{\boldsymbol{U}}_0\widetilde{\boldsymbol{U}}_0^\top\|_{\text{vec}(\infty)} + \rho\|\widetilde{\boldsymbol{V}}_0\widetilde{\boldsymbol{V}}_0^\top\|_{\text{vec}(\infty)} + \|\widetilde{\boldsymbol{U}}_0\|_{2\to\infty}\|\widetilde{\boldsymbol{V}}_0\|_{2\to\infty}. \tag{15}$$

We shall note that the projection (14) is equivalent to the following form:

$$\mathcal{P}_{\mathcal{T}}(\boldsymbol{M}) = \bar{\boldsymbol{U}}_0\bar{\boldsymbol{U}}_0^T\boldsymbol{M} + \boldsymbol{M}\bar{\boldsymbol{V}}_0\bar{\boldsymbol{V}}_0^\top - \bar{\boldsymbol{U}}_0\bar{\boldsymbol{U}}_0^T\boldsymbol{M}\bar{\boldsymbol{V}}_0\bar{\boldsymbol{V}}_0^\top, \tag{16}$$

where $\bar{\boldsymbol{U}}_0\in\mathbb{R}^{N\times r}$ and $\bar{\boldsymbol{V}}_0\in\mathbb{R}^{M\times r}$ are, respectively, matrices of left and right orthogonal singular vectors corresponding to $\widetilde{\boldsymbol{X}}_0 = \boldsymbol{P}\boldsymbol{X}_0\boldsymbol{Q}^\top$. In other words, (14) and (16) are equivalent in the sense that

$$\widetilde{\boldsymbol{U}}_0\widetilde{\boldsymbol{U}}_0^\top = \bar{\boldsymbol{U}}_0\bar{\boldsymbol{U}}_0^T,\quad \widetilde{\boldsymbol{V}}_0\widetilde{\boldsymbol{V}}_0^\top = \bar{\boldsymbol{V}}_0\bar{\boldsymbol{V}}_0^T. \tag{17}$$

Note that $\widetilde{\boldsymbol{U}}_0$ and $\bar{\boldsymbol{U}}_0$ (or $\widetilde{\boldsymbol{V}}_0$ and $\bar{\boldsymbol{V}}_0$) are not necessarily the same to hold (17). Built on $\bar{\boldsymbol{U}}_0$ and $\bar{\boldsymbol{V}}_0$, $\beta(\rho)$ could also be defined as

$$\beta(\rho) = \rho^{-1}\|\bar{\boldsymbol{U}}_0\bar{\boldsymbol{U}}_0^\top\|_{\text{vec}(\infty)} + \rho\|\bar{\boldsymbol{V}}_0\bar{\boldsymbol{V}}_0^\top\|_{\text{vec}(\infty)} + \|\bar{\boldsymbol{U}}_0\|_{2\to\infty}\|\bar{\boldsymbol{V}}_0\|_{2\to\infty}. \tag{18}$$

Here we note that $\|\widetilde{\boldsymbol{U}}_0\|_{2\to\infty} = \|\bar{\boldsymbol{U}}_0\|_{2\to\infty}$ and $\|\widetilde{\boldsymbol{V}}_0\|_{2\to\infty} = \|\bar{\boldsymbol{V}}_0\|_{2\to\infty}$ due to (17). We will mainly use the definition (14) for the projection $\mathcal{P}_{\mathcal{T}}(\boldsymbol{M})$ in our following analysis as it allows us to "separate" the construction of $\widetilde{\boldsymbol{U}}_0$ and $\widetilde{\boldsymbol{V}}_0$ and brings us a lot of benefits when we bound the estimation errors later.

Let $\mathcal{P}_{\mathcal{T}^\perp}(\cdot)$ and $\mathcal{P}_{\mathcal{T}_0^\perp}(\cdot)$ be the projections to the orthogonal complement of $\mathcal{T}$ and $\mathcal{T}_0$ respectively, i.e., $\mathcal{P}_{\mathcal{T}^\perp}(\boldsymbol{M}') = \boldsymbol{M}' - \mathcal{P}_{\mathcal{T}}(\boldsymbol{M}')$ and $\mathcal{P}_{\mathcal{T}_0^\perp}(\boldsymbol{M}'') = \boldsymbol{M}'' - \mathcal{P}_{\mathcal{T}_0}(\boldsymbol{M}'')$ for any matrix $\boldsymbol{M}' \in \mathbb{R}^{N\times M}$, $\boldsymbol{M}'' \in \mathbb{R}^{n\times m}$. We define the following quantity to link the projections $\mathcal{P}_{\mathcal{T}^\perp}(\cdot)$ and $\mathcal{P}_{\mathcal{T}_0^\perp}(\cdot)$,

$$\begin{aligned}\eta_1 = \max\Big\{\eta:\ \eta>0,\\ \eta\|\mathcal{P}_{\mathcal{T}^\perp}(\boldsymbol{P}\boldsymbol{X}\boldsymbol{Q}^\top)\|_* \leqslant \|\mathcal{P}_{\mathcal{T}_0^\perp}(\boldsymbol{X})\|_*,\ \forall \boldsymbol{X}\in\mathbb{R}^{n\times m}\Big\}.\end{aligned} \tag{19}$$

The existence of η_1 can be guaranteed through Proposition 4 below.

Proposition 4 *Let $\mathcal{P}_{\mathcal{T}}(\cdot)$ and $\mathcal{P}_{\mathcal{T}_0}(\cdot)$ be as in (14) and (13). Then, for any $\boldsymbol{X}\in\mathbb{R}^{n\times m}$,*

$$\|\mathcal{P}_{\mathcal{T}_0^\perp}(\boldsymbol{X})\|_* = 0 \quad\Rightarrow\quad \|\mathcal{P}_{\mathcal{T}^\perp}(\boldsymbol{P}\boldsymbol{X}\boldsymbol{Q}^\top)\|_* = 0.$$

Now we consider the sparse component $\boldsymbol{Y}_0$. Define the space of matrices whose supports are subsets of the supports of $\boldsymbol{Y}_0$:

$$\mathcal{S} = S(\boldsymbol{Y}_0) := \{\boldsymbol{Y}\in\mathcal{R}^{N\times M}, \mathrm{supp}(\boldsymbol{Y})\subseteq \mathrm{supp}(\boldsymbol{Y}_0)\}.$$

Define the orthogonal projector to $\mathcal{S}$ as $\mathcal{P}_{\mathcal{S}}$. Under the inner product $\langle A, B\rangle = \mathrm{tr}(A^\top B)$, this projection is given by

$$[\mathcal{P}_{\mathcal{S}}(\boldsymbol{X})]_{i,j} = \begin{cases} X_{i,j}, & (i,j)\in \mathrm{supp}(\boldsymbol{X}_0),\\ 0, & \text{otherwise,}\end{cases} \tag{20}$$

for $i = 1,\cdots,N$ and $j = 1,\cdots,M$. Furthermore, for any matrix $\boldsymbol{M}$, define a $\|\cdot\|_{p\to q}$ transformation norm as

$$\|\boldsymbol{M}\|_{p\to q} = \max\{\|\boldsymbol{M}\nu\|_q : \nu\in\mathbb{R}^n, \|\nu\|_p\leqslant 1\}.$$

Then we define the following quantity that measures the sparseness of $\boldsymbol{Y}_0$:

$$\alpha(\rho) = \max\big\{\rho\|\mathrm{sgn}(\boldsymbol{Y}_0)\|_{1\to1},\ \rho^{-1}\|\mathrm{sgn}(\boldsymbol{Y}_0)\|_{\infty\to\infty}\big\}, \tag{21}$$

where $\{\mathrm{sgn}(\boldsymbol{M})\}_{i,j} = \mathrm{sgn}(\boldsymbol{M}_{i,j})$ is the sign of $M_{i,j}$, and $\rho>0$ is a parameter to accommodate disparity between the number of rows and columns with a natural choice of ρ being $\rho = \sqrt{M/N}$. As $\|\boldsymbol{M}\|_{1\to1} = \max_j\|\boldsymbol{M}e_j\|_1$ and $\|\boldsymbol{M}\|_{\infty\to\infty} = \max_i\|\boldsymbol{M}^\top e_i\|_1$, $\|\mathrm{sgn}(\boldsymbol{Y}_0)\|_{1\to1}$ and $\|\mathrm{sgn}(\boldsymbol{Y}_0)\|_{\infty\to\infty}$ respectively measures the maximum number of nonzero entries in any row and any column of $\boldsymbol{Y}_0$. This explains why $\alpha(\rho)$ is a quantity that measures the sparseness of $\boldsymbol{Y}_0$.

Now we introduce a quantity related to the projection $\mathcal{P}_{\mathcal{S}}(\cdot)$ in (20) and $\mathcal{P}_{\mathcal{S}^\perp}(\cdot)$,

$$\begin{aligned}\eta_2 = \max\Big\{\eta:\ \eta>0,\ \eta\|\boldsymbol{P}^*\boldsymbol{Y}\boldsymbol{Q}^*\|_{\mathrm{vec}(1)}\\ \leqslant \eta\|\mathcal{P}_{\mathcal{S}}(\boldsymbol{Y})\|_{\mathrm{vec}(1)} + \|\mathcal{P}_{\mathcal{S}^\perp}(\boldsymbol{Y})\|_{\mathrm{vec}(1)}, \forall \boldsymbol{Y}\in\mathbb{R}^{N\times M}\Big\}.\end{aligned} \tag{22}$$

The existence of η_2 is obvious. By replacing the vec(1)-norm in (22) with the vec(2)-norm, we could have an idea about the scale of η_2. As $\boldsymbol{P}^*$ and $\boldsymbol{Q}^*$ are projection matrices, we have

$$\|\boldsymbol{P}^*\boldsymbol{Y}\boldsymbol{Q}^*\|_{\mathrm{vec}(2)} \leqslant \|\boldsymbol{Y}\|_{\mathrm{vec}(2)} \leqslant \|\mathcal{P}_{\mathcal{S}}(\boldsymbol{Y})\|_{\mathrm{vec}(2)} + \|\mathcal{P}_{\mathcal{S}^\perp}(\boldsymbol{Y})\|_{\mathrm{vec}(2)},\ \forall \boldsymbol{Y} \in \mathbb{R}^{N\times M}.$$

As a consequence,

$$\begin{aligned} 1 \leqslant \max\Big\{\eta:\ \eta>0,\ &\eta\|\boldsymbol{P}^*\boldsymbol{Y}\boldsymbol{Q}^*\|_{\mathrm{vec}(2)} \\ &\leqslant \eta\|\mathcal{P}_{\mathcal{S}}(\boldsymbol{Y})\|_{\mathrm{vec}(2)} + \|\mathcal{P}_{\mathcal{S}^\perp}(\boldsymbol{Y})\|_{\mathrm{vec}(2)}, \forall \boldsymbol{Y}\in\mathbb{R}^{N\times M}\Big\}. \end{aligned}$$

Although vec(1)-norm is used in (22), an η_2 close to 1 can be expected for many combinations of $\boldsymbol{P}^*$, $\boldsymbol{Q}^*$ and $\mathcal{P}_{\mathcal{S}}(\cdot)$.

4.2 Main results

Define the rank of $\boldsymbol{X}_0$ as r and the sparisty of $\boldsymbol{Y}_0$ as s. To introduce our main results on recovering $\boldsymbol{P}\boldsymbol{X}_0\boldsymbol{Q}^\top$ and $\boldsymbol{Y}_0$, we need the following properties related to $(\boldsymbol{U}_0, \boldsymbol{V}_0)$ and $(\boldsymbol{P}, \boldsymbol{Q})$,

$$\begin{aligned} \boldsymbol{\Gamma} =& \big((\boldsymbol{P}\boldsymbol{U}_0)^+\big)^\top \boldsymbol{V}_0^\top \boldsymbol{Q}^+ + (\boldsymbol{P}^+)^\top \boldsymbol{U}_0 (\boldsymbol{Q}\boldsymbol{V}_0)^+ \\ &- \big((\boldsymbol{P}\boldsymbol{U}_0)^+\big)^\top (\boldsymbol{Q}\boldsymbol{V}_0)^+, \\ \gamma_1 =& \|\boldsymbol{\Gamma}\|_{\mathrm{vec}(\infty)},\quad \gamma_2 = \|\boldsymbol{\Gamma}\|_{2\to 2}. \end{aligned} \tag{23}$$

The quantity $\boldsymbol{\Gamma}$ plays a key role in our analysis below. Note that when $\boldsymbol{P}$ and $\boldsymbol{Q}$ are identity matrices, $\boldsymbol{\Gamma} = \boldsymbol{U}_0\boldsymbol{V}_0^\top$ and $\gamma_2 = \|\boldsymbol{U}_0\boldsymbol{V}_0^\top\|_{2\to 2} = 1$.

Furthermore, define the following random error terms related to the noise matrix $\boldsymbol{E}$ introduced in model (1).

$$\begin{aligned} \epsilon_{2\to 2} &= \|\boldsymbol{E}\|_{2\to 2}, \\ \epsilon_\infty &= \|\mathcal{P}_{\mathcal{T}}(\boldsymbol{E})\|_{\mathrm{vec}(\infty)} + \|\boldsymbol{E}\|_{\mathrm{vec}(\infty)}, \\ \epsilon'_\infty &= \|\mathcal{P}_{\mathcal{T}}(\boldsymbol{P}^*\boldsymbol{E}\boldsymbol{Q}^*)\|_{\mathrm{vec}(\infty)} + \|\boldsymbol{P}^*\boldsymbol{E}\boldsymbol{Q}^*\|_{\mathrm{vec}(\infty)}, \\ \epsilon_* &= \|\mathcal{P}_{\mathcal{T}}(\boldsymbol{P}^*\boldsymbol{E}\boldsymbol{Q}^*)\|_*, \end{aligned} \tag{24}$$

where for any matrix $\boldsymbol{M}$, $\boldsymbol{M}^*$ is the projection matrix onto the column space of $\boldsymbol{M}$. When $\boldsymbol{M}$ is of full-column rank, $\boldsymbol{M}^* = \boldsymbol{M}(\boldsymbol{M}^\top\boldsymbol{M})^{-1}\boldsymbol{M}^\top$. Given these error terms, we suppose that the penalty levels λ_1 and λ_2 satisfy the condition below for certain $c > 1$ and $\rho > 0$,

$$\alpha(\rho)\beta(\rho) < 1 \tag{25}$$

$$\begin{aligned} &\left[\sigma_{\min}^{-1}(\boldsymbol{P})\sigma_{\min}^{-1}(\boldsymbol{Q}) - \frac{c\gamma_1\alpha(\rho)}{1-\alpha(\rho)\beta(\rho)}\right]\lambda_1 \\ &\geqslant c\left(\frac{\alpha(\rho)}{1-\alpha(\rho)\beta(\rho)}\lambda_2 + \frac{\alpha(\rho)}{1-\alpha(\rho)\beta(\rho)}\epsilon_\infty + \epsilon_{2\to 2}\right), \end{aligned} \tag{26}$$

$$[1-(1+c)\alpha(\rho)\beta(\rho)]\,\lambda_2 \geqslant c\left(\gamma_1\lambda_1 + (2-\alpha(\rho)\beta(\rho))\epsilon_\infty\right), \tag{27}$$

We note that when $\boldsymbol{P}$ and $\boldsymbol{Q}$ are interpolation matrices with appropriate dimensions, e.g., $N \geqslant 20$, $M \geqslant 20$, we have $\sigma_{\max}(\boldsymbol{P}) \approx \sigma_{\max}(\boldsymbol{Q}) \approx 1.53$, and $\sigma_{\min}(\boldsymbol{P}) \approx \sigma_{\min}(\boldsymbol{Q}) \approx 1.00$.

Finally, we define δ_1, δ_2 and δ as functions of r, s, γ_1, γ_2, $\alpha(\rho)$, $\beta(\rho)$, λ_1, λ_2 and the error terms. These quantities will be used in Theorem 5 below.

$$\begin{aligned}
\delta_1 =& r\Big(\frac{2\alpha(\rho)}{1-\alpha(\rho)\beta(\rho)}(\lambda_2+\gamma_1\lambda_1+\epsilon_\infty)+2\epsilon_{2\to 2}+\lambda_1\gamma_2\Big),\\
\delta_2 =& \frac{s}{1-\alpha(\rho)\beta(\rho)}(\lambda_2+\lambda_1\gamma_1+\epsilon_\infty),\\
\delta =& (\lambda_1\gamma_2+\epsilon_{2\to 2})\delta_1+(\lambda_2+\epsilon_\infty)\delta_2.
\end{aligned} \tag{28}$$

Now we are ready to state our main results.

Theorem 5 *Let $r = |\mathrm{rank}(\boldsymbol{X}_0)|$ and $s = |\mathrm{supp}(\boldsymbol{Y}_0)|$. Let error terms $\epsilon_{2\to 2}$, ϵ_∞, ϵ'_∞, ϵ_* be as in (24), γ_1 and γ_2 be as in (23) and δ be as in (28). Further let η_1, η_2 be as in (19), (22) and $\eta_0 = \min(\eta_2,\ \eta_1\sigma_{\max}(\boldsymbol{P})\sigma_{\max}(\boldsymbol{Q}))$. Assume that $\boldsymbol{P}$ and $\boldsymbol{Q}$ are of full column rank. Then, when (25) to (27) hold for some $\rho > 0$ and $c > 1$, we have*

$$\begin{aligned}
&(1-\alpha(\rho)\beta(\rho))\|\boldsymbol{P}^*(\widehat{\boldsymbol{Y}}-\boldsymbol{Y}_0)\boldsymbol{Q}^*\|_{vec(1)}\\
\leqslant\ & [\lambda_2(1-1/c)\eta_0]^{-1}\delta+5\lambda_2 s+2s\epsilon_\infty+3s\epsilon'_\infty\\
&+2\sigma_{\min}^{-1}(\boldsymbol{P})\sigma_{\min}^{-1}(\boldsymbol{Q})\lambda_1\sqrt{sr},
\end{aligned} \tag{29}$$

$$\begin{aligned}
&(1-\alpha(\rho)\beta(\rho))\|\widehat{\boldsymbol{Y}}-\boldsymbol{Y}_0\|_{vec(1)}\\
\leqslant\ & [2(1-1/c)\lambda_2]^{-1}(1+\eta_0^{-1})\delta+5\lambda_2 s+2s\epsilon_\infty\\
&+3s\epsilon'_\infty+2\sigma_{\min}^{-1}(\boldsymbol{P})\sigma_{\min}^{-1}(\boldsymbol{Q})\lambda_1\sqrt{sr},
\end{aligned} \tag{30}$$

and

$$\begin{aligned}
&\|\boldsymbol{P}(\widehat{\boldsymbol{X}}-\boldsymbol{X}_0)\boldsymbol{Q}^\top\|_*\\
\leqslant\ & [2(1-1/c)\lambda_1\eta_1]^{-1}\delta+\epsilon_*+2\sigma_{\min}^{-1}(\boldsymbol{P})\sigma_{\min}^{-1}(\boldsymbol{Q})\lambda_1 r\\
&+\sqrt{2r}\|\boldsymbol{P}^*(\widehat{\boldsymbol{Y}}-\boldsymbol{Y}_0)\boldsymbol{Q}^*\|_{vec(2)}.
\end{aligned} \tag{31}$$

We note that the last term in the RHS of (31) can be easily bounded by $\sqrt{2r}\|\boldsymbol{P}^*(\widehat{\boldsymbol{Y}}-\boldsymbol{Y}_0)\boldsymbol{Q}^*\|_{\mathrm{vec}(1)}$ and then (29) can be applied. To understand the derived bounds in Theorem 5, we first recall that the matrices $\boldsymbol{P}$ and $\boldsymbol{Q}$ are of full column rank. When $\sigma_{\min}(\boldsymbol{P}) \asymp \sigma_{\max}(\boldsymbol{P}) \asymp \sigma_{\min}(\boldsymbol{Q}) \asymp \sigma_{\max}(\boldsymbol{Q}) \asymp \mathcal{O}(1)$ as of interpolation matrices, $\gamma_1\alpha(\rho) \asymp \gamma_2 \asymp \mathcal{O}(1)$, and the penalty levels λ_1 and λ_2 are of order

$$\begin{aligned}
\lambda_1 =& \mathcal{O}\Big([\alpha(\rho)\epsilon_\infty]\vee\epsilon_{2\to 2}\Big),\\
\lambda_2 =& \mathcal{O}([1/\alpha(\rho)]\lambda_1),
\end{aligned} \tag{32}$$

the conditions (26) and (27) could be satisfied. As a consequence, the error bounds in Theorem 5 are of order

$$\begin{aligned}
\|\widehat{\boldsymbol{Y}}-\boldsymbol{Y}_0\|_{\mathrm{vec}(1)} \asymp& \|\boldsymbol{P}^*(\widehat{\boldsymbol{Y}}-\boldsymbol{Y}_0)\boldsymbol{Q}^*\|_{\mathrm{vec}(1)}\\
=& \mathcal{O}\Big(r\alpha(\rho)\big\{[\alpha(\rho)(\epsilon_\infty\vee\epsilon'_\infty)]\vee\epsilon_{2\to 2}\big\}\Big),
\end{aligned} \tag{33}$$

and

$$\begin{aligned}&\|\boldsymbol{P}(\widehat{\boldsymbol{X}}-\boldsymbol{X}_0)\boldsymbol{Q}^\top\|_{\mathrm{vec}(1)}\\ =&\mathcal{O}\Big(r^{3/2}\alpha(\rho)\big\{[\alpha(\rho)(\epsilon_\infty\vee\epsilon'_\infty)]\vee\epsilon_{2\to2}\big\}+\epsilon_*\Big).\end{aligned}\tag{34}$$

Hsu et al. (2011) derived the upper bounds for $\|\widehat{\boldsymbol{Y}}-\boldsymbol{Y}_0\|_{\mathrm{vec}(1)}$ and $\|\widehat{\boldsymbol{X}}-\boldsymbol{X}_0\|_*$ under the classical RPCA setup, i.e., $(\boldsymbol{P},\boldsymbol{Q})=(\boldsymbol{I}_N,\boldsymbol{I}_M)$. They imposed the constraint $\|\widehat{\boldsymbol{Y}}-\boldsymbol{Z}\|_{\mathrm{vec}(\infty)}\leqslant b$ in the optimization for some $b\geqslant\|\boldsymbol{Y}_0-\boldsymbol{Z}\|_{\mathrm{vec}(\infty)}$, while also allowing b to go to infinity. We note that the error bounds in (33) and (34) are of the same order as their results when no knowledge of b is imposed, i.e., $b=\infty$. In fact, Theorem 5 can be viewed as a generalization of Hsu et al. (2011) for arbitrary full column rank matrices $\boldsymbol{P}$ and $\boldsymbol{Q}$.

We still need to understand the random error terms in the bound. When the noise matrix $\boldsymbol{E}$ has i.i.d. Gaussian entries, $E_{i,j}\sim\mathcal{N}(0,\sigma^2)$, by Davidson and Szarek (2001), we have the following probabilistic upper bound,

$$\begin{aligned}\|\boldsymbol{E}\|_{2\to2}&\leqslant\sigma\sqrt{N}+\sigma\sqrt{M}+\mathcal{O}(\sigma),\\ \|\boldsymbol{P}^*\boldsymbol{E}\boldsymbol{Q}^*\|_{2\to2}&\leqslant\sigma\sqrt{N}+\sigma\sqrt{M}+\mathcal{O}(\sigma).\end{aligned}$$

In addition, for the terms with vec(∞)-norm, we have the following inequalities hold with high probability

$$\begin{aligned}\|\boldsymbol{E}\|_{\mathrm{vec}(\infty)}&\leqslant\mathcal{O}(\sigma\log(MN)),\\ \|\mathcal{P}_{\mathcal{T}}(\boldsymbol{E})\|_{\mathrm{vec}(\infty)}&\leqslant\mathcal{O}(\sigma\log(MN)),\\ \|\boldsymbol{P}^*\boldsymbol{E}\boldsymbol{Q}^*\|_{\mathrm{vec}(\infty)}&\leqslant\mathcal{O}(\sigma\log(MN)),\\ \|\mathcal{P}_{\mathcal{T}}(\boldsymbol{P}^*\boldsymbol{E}\boldsymbol{Q}^*)\|_{\mathrm{vec}(\infty)}&\leqslant\mathcal{O}(\sigma\log(MN)).\end{aligned}$$

Finally, for the nuclear-normed error term,

$$\|\mathcal{P}_{\mathcal{T}}(\boldsymbol{P}^*\boldsymbol{E}\boldsymbol{Q}^*)\|_*\leqslant2r\|\boldsymbol{P}^*\boldsymbol{E}\boldsymbol{Q}^*\|_{2\to2}\leqslant2r\sigma\sqrt{N}+2r\sigma\sqrt{M}+\mathcal{O}(r\sigma)$$

holds with high probability, where the first inequality holds by Lemma 14 in the supplementary material. We summarize above analyses into the following Corollary.

Corollary 6 *Let $r=|\mathrm{rank}(\boldsymbol{X}_0)|$ and $s=|\mathrm{supp}(\boldsymbol{Y}_0)|$. Assume that the noise matrix $\boldsymbol{E}$ has i.i.d. Gaussian entries. Assume that $\boldsymbol{P}$ and $\boldsymbol{Q}$ are of full column rank with singular values $\sigma_{\min}(\boldsymbol{P})\asymp\sigma_{\max}(\boldsymbol{P})\asymp\sigma_{\min}(\boldsymbol{Q})\asymp\sigma_{\max}(\boldsymbol{Q})\asymp\mathcal{O}(1)$. Assume that $\inf_{\rho>0}\alpha(\rho)\beta(\rho)<1$. Let γ_1 and γ_2 defined as in (23). When $\gamma_1\alpha(\rho)\asymp\gamma_2\asymp\mathcal{O}(1)$ and*

$$\begin{aligned}\lambda_1&\asymp\mathcal{O}\left([\alpha(\rho)\log(MN)]\vee[\sqrt{N}+\sqrt{M}]\right),\\ \lambda_2&\asymp\mathcal{O}\left([1/\alpha(\rho)]\lambda_1\right),\end{aligned}$$

we have

$$\begin{aligned}\|\widehat{\boldsymbol{Y}}-\boldsymbol{Y}_0\|_{vec(1)}\asymp&\|\boldsymbol{P}^*(\widehat{\boldsymbol{Y}}-\boldsymbol{Y}_0)\boldsymbol{Q}^*\|_{vec(1)}\\ =&\mathcal{O}\Big(r\alpha(\rho)\big\{\,[\alpha(\rho)\log(MN)]\vee[\sqrt{N}+\sqrt{M}]\big\}\Big),\\ \|\boldsymbol{P}(\widehat{\boldsymbol{X}}-\boldsymbol{X}_0)\boldsymbol{Q}^\top\|_*=&\mathcal{O}\Big(r^{3/2}\alpha(\rho)\big\{\,[\alpha(\rho)\log(MN)]\vee[\sqrt{N}+\sqrt{M}]\big\}\Big).\end{aligned}$$

We note that such an order of estimation error matches that in Hsu et al. (2011), but, significant computational benefits could be gained through imposing the smoothing matrices $\boldsymbol{P}$ and $\boldsymbol{Q}$. Moreover, the bound on $\|\boldsymbol{P}(\widehat{\boldsymbol{X}} - \boldsymbol{X}_0)\boldsymbol{Q}^\top\|_*$ can be improved if prior knowledge is known on the upper bound of $\|\boldsymbol{Y}_0\|_\infty$.

4.3 Outline of proof

The key to prove Theorem 5 is the following two theorems. In Theorem 7, we provide a transfer property between the two projections $\mathcal{P}_{\mathcal{T}^\perp}(\cdot)$ and $\mathcal{P}_{\mathcal{T}_0^\perp}(\cdot)$ through $\boldsymbol{\Gamma}$. Building on the transfer property, we in Theorem 8 construct a dual certificate $(\boldsymbol{D}_{\mathcal{S}}, \boldsymbol{D}_{\mathcal{T}})$ such that (1) $\boldsymbol{D}_{\mathcal{S}} + \boldsymbol{D}_{\mathcal{T}} + \boldsymbol{E}$ is a subgradient of $\lambda_2\|\boldsymbol{Y}\|_{\mathrm{vec}(1)}$ at $\boldsymbol{Y} = \boldsymbol{Y}_0$, and (2) $\boldsymbol{P}^\top(\boldsymbol{D}_{\mathcal{S}} + \boldsymbol{D}_{\mathcal{T}} + \boldsymbol{E})\boldsymbol{Q}$ is a subgradient of $\lambda_1\|\boldsymbol{X}\|_*$ at $\boldsymbol{X} = \boldsymbol{X}_0$.

Theorem 7 (Transfer Property) *Suppose $\boldsymbol{P}$ and $\boldsymbol{Q}$ are of full column rank. Let $\boldsymbol{\Gamma}$ be as in (23). Let $\boldsymbol{D} \in \mathbb{R}^{N\times M}$ be any matrix satisfying*

$$\mathcal{P}_{\mathcal{T}}(\boldsymbol{D}) = \boldsymbol{\Gamma}.$$

Then, $\boldsymbol{P}^T\boldsymbol{D}\boldsymbol{Q}$ is a sub-gradient of $\|\boldsymbol{X}\|_*$ at $\boldsymbol{X}_0$, in other words,

$$\mathcal{P}_{\mathcal{T}_0}(\boldsymbol{P}^\top\boldsymbol{D}\boldsymbol{Q}) = \boldsymbol{U}_0\boldsymbol{V}_0^\top.$$

Theorem 8 (Dual Certificate) *Let $r = \mathrm{rank}(\boldsymbol{X}_0)$, $s = \|\boldsymbol{Y}\|_0$ and $\rho > 0$. Let error terms $\epsilon_{2\to2}$, ϵ_∞, ϵ'_∞, ϵ_* be as in (24) and η_1, η_2 be as in (19), (22), respectively. Assume that $\inf_{\rho>0}\alpha(\rho)\beta(\rho) < 1$ and the penalty level λ_1 and λ_2 satisfy (26) and (27) for some $c > 1$. Suppose $\boldsymbol{P}$ and $\boldsymbol{Q}$ are of full column rank. Then, the following quantity $\boldsymbol{D}_{\mathcal{S}}$ and $\boldsymbol{D}_{\mathcal{T}}$ are well defined,*

$$\begin{aligned}
\boldsymbol{D}_{\mathcal{S}} &= (\mathcal{I} - \mathcal{P}_{\mathcal{S}}\circ\mathcal{P}_{\mathcal{T}})^{-1}\left(\lambda_2\mathrm{sgn}(\boldsymbol{Y}_0) - \lambda_1\mathcal{P}_{\mathcal{S}}(\boldsymbol{\Gamma}) - (\mathcal{P}_{\mathcal{S}}\circ\mathcal{P}_{\mathcal{T}^\perp})(\boldsymbol{E})\right),\\
\boldsymbol{D}_{\mathcal{T}} &= (\mathcal{I} - \mathcal{P}_{\mathcal{T}}\circ\mathcal{P}_{\mathcal{S}})^{-1}\left(\lambda_1\boldsymbol{\Gamma} - \lambda_2\mathcal{P}_{\mathcal{T}}\big(\mathrm{sgn}(\boldsymbol{Y}_0)\big) - (\mathcal{P}_{\mathcal{T}}\circ\mathcal{P}_{\mathcal{S}^\perp})(\boldsymbol{E})\right).
\end{aligned}$$

They satisfy

$$\begin{aligned}
&\mathcal{P}_{\mathcal{S}}(\boldsymbol{D}_{\mathcal{S}} + \boldsymbol{D}_{\mathcal{T}} + \boldsymbol{E}) = \lambda_2\mathrm{sgn}(\boldsymbol{Y}_0),\\
&\mathcal{P}_{\mathcal{T}}(\boldsymbol{D}_{\mathcal{S}} + \boldsymbol{D}_{\mathcal{T}} + \boldsymbol{E}) = \lambda_1\boldsymbol{\Gamma},\\
&\mathcal{P}_{\mathcal{T}_0}\left(\boldsymbol{P}^\top(\boldsymbol{D}_{\mathcal{S}} + \boldsymbol{D}_{\mathcal{T}} + \boldsymbol{E})\boldsymbol{Q}\right) = \lambda_1\boldsymbol{U}_0\boldsymbol{V}_0^\top,
\end{aligned} \tag{35}$$

and

$$\begin{aligned}
\|\mathcal{P}_{\mathcal{S}^\perp}(\boldsymbol{D}_{\mathcal{S}} + \boldsymbol{D}_{\mathcal{T}} + \boldsymbol{E})\|_{vec(\infty)} &\leqslant \lambda_2/c,\\
\|\mathcal{P}_{\mathcal{T}_0^\perp}\left(\boldsymbol{P}^\top(\boldsymbol{D}_{\mathcal{S}} + \boldsymbol{D}_{\mathcal{T}} + \boldsymbol{E})\boldsymbol{Q}\right)\|_{2\to2} &\leqslant \lambda_1/c.
\end{aligned} \tag{36}$$

Moreover,

$$\begin{aligned}
&\|\boldsymbol{D}_{\mathcal{S}}\|_{2\to 2} \leqslant \frac{\alpha(\rho)}{1-\alpha(\rho)\beta(\rho)}(\lambda_2+\gamma_1\lambda_1+\epsilon_\infty),\\
&\|\boldsymbol{D}_{\mathcal{T}}\|_{vec(\infty)} \leqslant \frac{1}{1-\alpha(\rho)\beta(\rho)}\left(\gamma_1\lambda_1+\lambda_2\alpha(\rho)\beta(\rho)+\epsilon_\infty\right),\\
&\|\boldsymbol{D}_{\mathcal{T}}\|_{*} \leqslant r\left(\frac{2\alpha(\rho)}{1-\alpha(\rho)\beta(\rho)}(\lambda_2+\gamma_1\lambda_1+\epsilon_\infty)+2\epsilon_{2\to 2}+\lambda_1\gamma_2\right),\\
&\|\boldsymbol{D}_{\mathcal{S}}\|_{vec(1)} \leqslant \frac{s}{1-\alpha(\rho)\beta(\rho)}(\lambda_2+\lambda_1\gamma_1+\epsilon_\infty),\\
&\|\boldsymbol{D}_{\mathcal{T}}+\boldsymbol{D}_{\mathcal{S}}\|_2^2 \leqslant (\lambda_2+\epsilon_\infty)\|\boldsymbol{D}_{\mathcal{S}}\|_{vec(1)}+(\lambda_1\gamma_2+\epsilon_{2\to 2})\|\boldsymbol{D}_{\mathcal{T}}\|_{*}.
\end{aligned} \tag{37}$$

5. The Noiseless Case

Although this paper focuses on noisy imaging recovery problems, our study could also be extended to the noiseless case. When the noise term $\boldsymbol{E}=\boldsymbol{0}$, model (1) reduces to

$$\boldsymbol{Z}=\boldsymbol{P}\boldsymbol{X}_0\boldsymbol{Q}^\top+\boldsymbol{Y}_0. \tag{38}$$

We consider the following optimization problem to solve (38)

$$\begin{aligned}
\min_{\boldsymbol{X}\in\mathbb{R}^{n\times m},\boldsymbol{Y}\in\mathbb{R}^{N\times M}} \quad & \|\boldsymbol{X}\|_*+\mu\|\boldsymbol{Y}\|_{\mathrm{vec}(1)}\\
\text{s.t.} \quad & \boldsymbol{P}\boldsymbol{X}\boldsymbol{Q}^\top+\boldsymbol{Y}=\boldsymbol{Z}.
\end{aligned} \tag{39}$$

Here the parameter μ could be viewed as the ratio of regularization parameters in problem (2), i.e., $\mu=\lambda_2/\lambda_1$. The problem (39) could be solved by a similar proximal gradient as in Section 3. We omit the details of computation but rather focus on the theoretical analysis.

Let γ_1 and γ_2 be defined as in (23) and error quantities $\epsilon_{2\to 2}$ and $\epsilon_\infty=0$ be as in (24). Under such noiseless case, we have $\epsilon_{2\to 2}=\epsilon_\infty=0$. As a consequence, the regularization condition (25) to (27) reduce to the following form:

$$\inf_{\rho>0}\alpha(\rho)\beta(\rho)<1, \tag{40}$$

$$\frac{c\gamma_1}{1-(1+c)\alpha(\rho)\beta(\rho)} \leqslant \mu \leqslant \frac{\sigma_{\max}^{-1}(\boldsymbol{P})\sigma_{\max}^{-1}(\boldsymbol{Q})}{c\cdot\alpha(\rho)}-\gamma_1, \tag{41}$$

where as before, $c>1$ and $\rho>0$ are certain constants. Building on this regularization requirement, we have the following exact recovery performance of PRPCA.

Theorem 9 *Suppose the noiseless model (38) holds for certain $\boldsymbol{X}_0$, $\boldsymbol{Y}_0$ and full column rank matrices $\boldsymbol{P}$ and $\boldsymbol{Q}$. Assume the condition (40) and (41) hold for certain $c>1$ and $\rho>0$. Then, the solution $(\widehat{\boldsymbol{X}},\widehat{\boldsymbol{Y}})$ to the problem (39) exactly recovers $(\boldsymbol{X}_0,\boldsymbol{Y}_0)$:*

$$\widehat{\boldsymbol{X}}=\boldsymbol{X}_0, \quad \widehat{\boldsymbol{Y}}=\boldsymbol{Y}_0.$$

The exact recovery performance of standard RPCA has been showed in many existing literature, e.g., Candès et al. (2011); Hsu et al. (2011). We provide here in Theorem

9 a generalization of such results by allowing general full-column rank matrices $\boldsymbol{P}$ and $\boldsymbol{Q}$. As discussed earlier, when $\sigma_{\min}(\boldsymbol{P}) \asymp \sigma_{\max}(\boldsymbol{P}) \asymp \sigma_{\min}(\boldsymbol{Q}) \asymp \sigma_{\max}(\boldsymbol{Q}) \asymp \mathcal{O}(1)$ as of interpolation matrices and $\gamma_1\alpha(\rho) \asymp \gamma_2 \asymp \mathcal{O}(1)$, we have

$$\mu = \lambda_2/\lambda_1 = \mathcal{O}(1/\alpha(\rho)). \tag{42}$$

Theorem 9 suggests that when $\inf_{\rho>0} \alpha(\rho)\beta(\rho) < 1$ and $\mu = \mathcal{O}(1/\alpha(\rho))$, $\boldsymbol{X}_0$ and $\boldsymbol{Y}_0$ could be exactly recovered. To further understand the condition $\inf_{\rho>0} \alpha(\rho)\beta(\rho) < 1$, we consider the case of recovering a square matrix, i.e., $M = N$. Under such case, $\alpha(1)$ measures the maximum column-wise and row-wise sparsity of $\boldsymbol{Y}_0$, i.e., the maximum number of nonzero entries in any row or column. We denote such sparsity as s_0. While for $\beta(\rho)$, if the entries of $\widetilde{\boldsymbol{U}}_0$ and $\widetilde{\boldsymbol{V}}_0$ are bounded by $\sqrt{c/M}$ for certain $c \geqslant 1$, then $\beta(1) \leqslant 3cr/M$. Therefore, to guarantee the condition $\inf_{\rho>0} \alpha(\rho)\beta(\rho) < 1$, a sufficient requirement is $\alpha(1)\beta(1) \leqslant 3crs_0/M < 1$. In other words, the maximum row/column-wise sparsity s_0 is allowed to be upper bounded by $\mathcal{O}(M/r)$. Note that when s_0 is of order $\mathcal{O}(M/r)$, the total number of nonzero entries s of the sparse component could be as large as $\mathcal{O}(MN/r)$. That is to say, PRPCA works even for the case that the sparse and smooth component has $\mathcal{O}(MN/r)$ nonzero entries.

6. Simulation Studies

In this section, we conduct a comprehensive simulation study to demonstrate the performance of PRPCA. Without loss of generality, all the simulations are for square matrix recovery, i.e., $M = N$.

We consider the model

$$\boldsymbol{Z} = \boldsymbol{P}_0\boldsymbol{X}_0\boldsymbol{Q}_0^\top + \boldsymbol{Y}_0 + \boldsymbol{E} \tag{43}$$

under two cases

(Case 1) $\boldsymbol{P}_0$ and $\boldsymbol{Q}_0$ are the interpolation matrices, i.e., $\boldsymbol{P}_0 = \boldsymbol{Q}_0 = \boldsymbol{J}_N$

(Case 2) $\boldsymbol{P}_0$ and $\boldsymbol{Q}_0$ are the generalized interpolation matrices in (6) with all the i.i.d weights w_{ur}^i, w_{lr}^i generated from $\mathcal{N}(0.5, 0.2^2)$.

We study the the following optimization problem

$$(\widehat{\boldsymbol{X}}, \widehat{\boldsymbol{Y}}) \in \underset{\substack{\boldsymbol{X}\in\mathbb{R}^{n(\boldsymbol{P})\times m(\boldsymbol{Q})} \\ \boldsymbol{Y}\in\mathbb{R}^{N\times M}}}{\arg\min} \frac{1}{2}\left\|\boldsymbol{Z} - \boldsymbol{P}\boldsymbol{X}\boldsymbol{Q}^\top - \boldsymbol{Y}\right\|_F^2 + \lambda_1\|\boldsymbol{X}\|_* + \lambda_2\|\boldsymbol{Y}\|_{\mathrm{vec}(1)}, \tag{44}$$

with four sets of $(\boldsymbol{P}, \boldsymbol{Q})$, where $n(\boldsymbol{P})$ and $m(\boldsymbol{Q})$ refers to the number of columns of $\boldsymbol{P}$ and $\boldsymbol{Q}$, respectively.

- The standard RPCA, with both $\boldsymbol{P}$ and $\boldsymbol{Q}$ being identity matrices, is denoted as "no interpolation". With such $\boldsymbol{P}$ and $\boldsymbol{Q}$, apparently there exists $\boldsymbol{X}$ such that $\boldsymbol{P}\boldsymbol{X}\boldsymbol{Q}^\top = \boldsymbol{P}_0\boldsymbol{X}_0\boldsymbol{Q}_0^\top$ under both cases. In other words, the optimization problem (44) is correctly specified under both cases.

- Both $\boldsymbol{P}$ and $\boldsymbol{Q}$ are interpolation matrices, i.e., $\boldsymbol{P} = \boldsymbol{Q} = \boldsymbol{J}_N$, denoted as "single interpolation". With such $\boldsymbol{P}$ and $\boldsymbol{Q}$, the optimization problem (44) is correctly specified under Case 1, while mis-specified under Case 2.

- Both $\boldsymbol{P}$ and $\boldsymbol{Q}$ are estimated based on robust linear regression (LR) with Huber Loss described in Section 2.2, denoted as "LR interpolation". With such $\boldsymbol{P}$ and $\boldsymbol{Q}$, the optimization problem (44) is mis-specified under both cases as the estimated $\widehat{\boldsymbol{P}}$ and $\widehat{\boldsymbol{Q}}$ may not recover $\boldsymbol{P}_0$ and $\boldsymbol{Q}_0$ exactly with probability goes to 1.

- Both $\boldsymbol{P}$ and $\boldsymbol{Q}$ are double interpolation matrices, i.e., $\boldsymbol{P} = \boldsymbol{Q} = \boldsymbol{J}_N \times \boldsymbol{J}_{N/2}$, denoted as "double interpolation". With such $\boldsymbol{P}$ and $\boldsymbol{Q}$, the optimization problem (44) is mis-specified under both cases.

Under both cases, we generate each entry of the noise term $\boldsymbol{E}$ from an i.i.d $\mathcal{N}(0, \sigma^2)$ distribution. The low-rank matrix $\boldsymbol{X}_0$ is generated as $\boldsymbol{X}_0 = \boldsymbol{U}_0 \boldsymbol{V}_0^\top$, where both $\boldsymbol{U}_0$ and $\boldsymbol{V}_0$ are $N \times r$ matrices with i.i.d. $\mathcal{N}(0, \sigma_0^2)$ entries. Each entry of the sparse component $\boldsymbol{Y}_0$ is i.i.d. generated, being 0 with probability $1 - \rho_s$, and uniformly distributed in $[-5, 5]$ with probability $1 - \rho_s$. The simulation is run over a grid of values for the matrix dimension N, noise level σ and sparsity level ρ_s.

- $\sigma = 0.2, 0.4, 0.6, 0.8, 1$

- $\rho_s = 0.05, 0.1, 0.15, 0.2, 0.25$

- $N = 60, 100, 200, 300, 400$

The other parameters are fixed at $r = 10$ and $\sigma_0 = 0.6$ if not otherwise specified. For all four sets of $(\boldsymbol{P}, \boldsymbol{Q})$, we use the same penalty level with $\lambda_1 = \sqrt{2N}\sigma$ and $\lambda_1 = \sqrt{2}\sigma$. This penalty level is commonly used in RPCA with noise, for example, in Zhou et al. (2010). When $\boldsymbol{P}$ and $\boldsymbol{Q}$ are single or double interpolation matrices, other carefully tuned penalty levels may further increase the estimation accuracy. In other words, this penalty level setup may not favor the PRPCA with interpolation matrices. But it allows us to better tell the effects of $(\boldsymbol{P}, \boldsymbol{Q})$ on the matrix recovery accuracy.

We report the root mean square errors (RMSE) of recovering $\boldsymbol{P}_0 \boldsymbol{X}_0 \boldsymbol{Q}_0^\top$, $\boldsymbol{Y}_0$ and $\boldsymbol{\Theta}$ with different choices of $(\boldsymbol{P}, \boldsymbol{Q})$:

$$\begin{aligned} \text{RMSE}(\boldsymbol{P}\boldsymbol{X}\boldsymbol{Q}^\top) &= \frac{\left\| \boldsymbol{P}\widehat{\boldsymbol{X}}\boldsymbol{Q}^\top - \boldsymbol{P}_0\boldsymbol{X}_0\boldsymbol{Q}_0^\top \right\|_F}{\sqrt{N * M}}, \\ \text{RMSE}(\boldsymbol{Y}) &= \frac{\|\widehat{\boldsymbol{Y}} - \boldsymbol{Y}_0\|_F}{\sqrt{N * M}}, \\ \text{RMSE}(\boldsymbol{\Theta}) &= \frac{\|\widehat{\boldsymbol{\Theta}} - \boldsymbol{\Theta}\|_F}{\sqrt{N * M}}. \end{aligned}$$

Finally, we report the required computation time (in seconds; all calculations were performed on a 2018 MacBook Pro laptop with 2.3 GHz Quad-Core Processor and 16GB Memory).

6.1 Case 1 study: effect of noise level σ

Figure 1 below reports the performance of PRPCA and RPCA over different noise levels under Case 1, with $\sigma_0 = \sigma$ and fixed $\rho_s = 0.1$, $N = M = 200$ and $r = 10$.

We first observe that the PRPCA with both single interpolation and LR interpolation demonstrate clear advantages in recovering all three targets: $\boldsymbol{P}_0\boldsymbol{X}_0\boldsymbol{Q}_0^\top$, $\boldsymbol{Y}_0$ and $\boldsymbol{\Theta}$. Particularly, PRPCA with single interpolation performs the best across the whole range of σ. While for the LR interpolation based PRPCA, we need to estimate the weights in the interpolation matrix first and then recover $\widehat{\boldsymbol{X}}$ and $\widehat{\boldsymbol{Y}}$. It still outperforms RPCA in recovering $\boldsymbol{\Theta}$ across the whole range of σ, and in recovering $\boldsymbol{P}\boldsymbol{X}\boldsymbol{Q}^\top$ and $\boldsymbol{Y}$ with relatively large σ ($\sigma \geqslant 0.6$). In terms of PRPCA with the mis-specified double interpolation matrix, its overall performance is not as good as the RPCA. But note that when the noise level is small, it achieves similar recovery accuracy in recovering $\boldsymbol{Y}_0$ and Θ compared to RPCA.

Regarding the computation time, it is clear that imposing interpolation matrices expedites the computation and such improvement is significant.

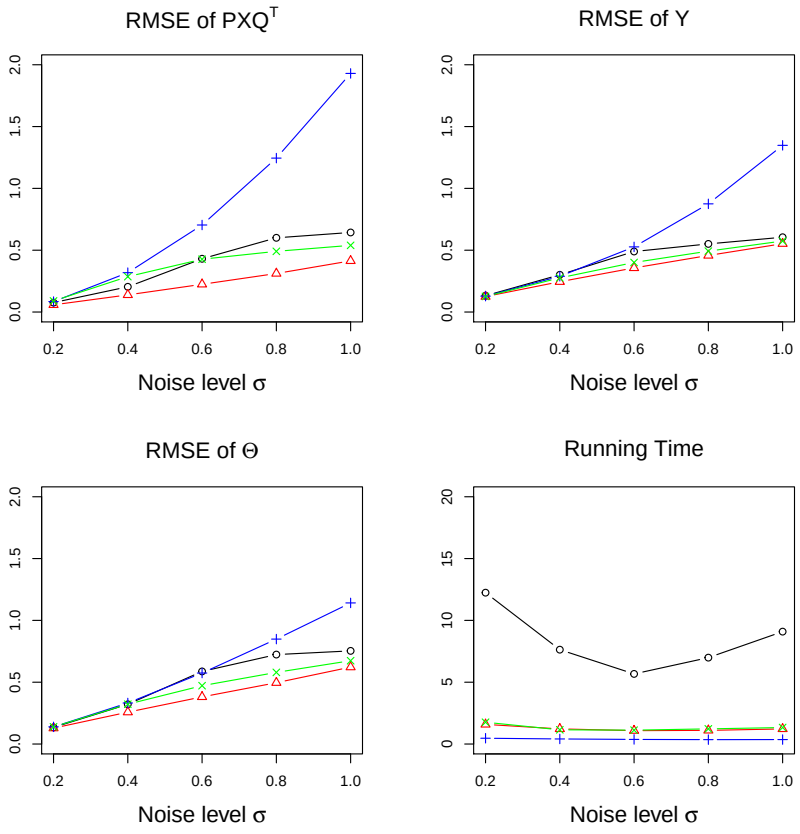

Figure 1: RMSE and running time with different $(\boldsymbol{P}, \boldsymbol{Q})$ ranges over different σ under Case 1. $\rho_s = 0.1$, $N = M = 200$, $r = 10$. Here: $\circ$ refers to no interpolation, $-\triangle-$ refers to single interpolation, $-\times-$ refers to LR interpolation, $-+-$ refers to double interpolation. The running times are in seconds.

6.2 Case 1 study: effect of sparsity level ρ_s

Figure 2 reports the performance of PRPCA and RPCA over different ρ_s, the sparsity level of $\boldsymbol{Y}_0$, under Case 1, with fixed $\sigma = 0.6$, $N = M = 200$ and $r = 10$.

It is clear that both the single interpolation and LR interpolation based PRPCA demonstrate clear advantages in recovering $\boldsymbol{P}_0\boldsymbol{X}_0\boldsymbol{Q}_0^\top$, $\boldsymbol{Y}_0$ and $\boldsymbol{\Theta}$. Indeed, both of them outperforms RPCA across the whole range of ρ_s in estimating all three targets. Moreover, the PRPCA with mis-specified double interpolation also achieves smaller RMSE compared to RPCA in recovering $\boldsymbol{\Theta}$ when ρ_s is relatively large, e.g., $\rho_s \geqslant 0.15$. One possible explanation for such phenomena is that when ρ_s goes up, the mis-modeled entries in $\boldsymbol{PXQ}^\top$ are more likely to be modeled by the sparse component $\boldsymbol{Y}$, thus further leveling up the performance of PRPCA. For the computation time, we also see a significant speed-up when interpolation matrices are imposed.

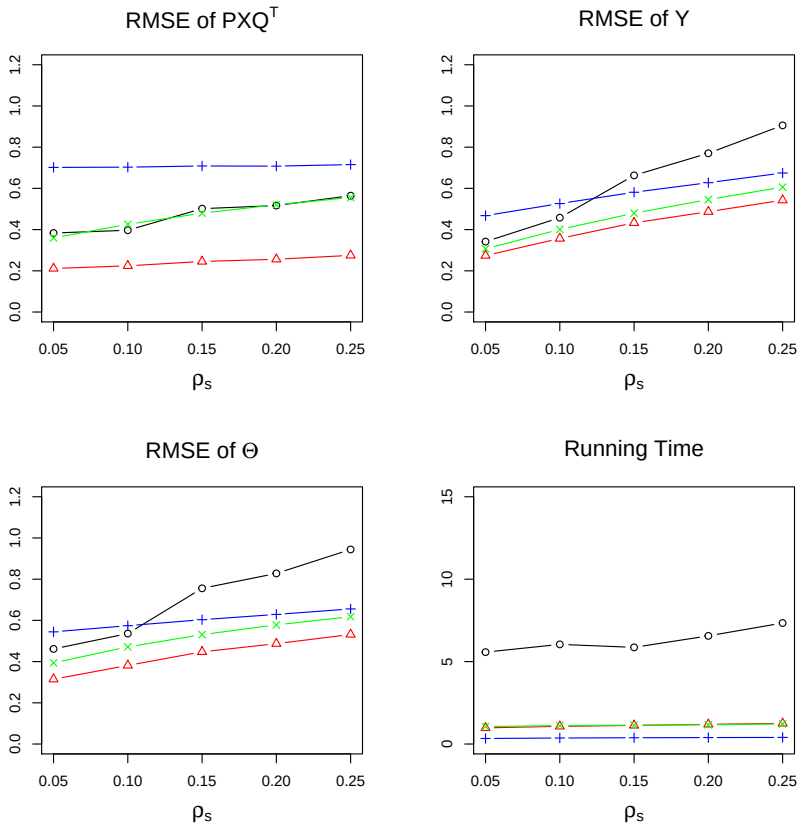

Figure 2: RMSE and running time with different $(\boldsymbol{P}, \boldsymbol{Q})$ ranges over different ρ_s under Case 1. The sparsity of $\boldsymbol{Y}_0$. $\sigma = 0.6$, $N = M = 200$, $r = 10$. Here: $-\circ-$ refers to no interpolation, $-\triangle-$ refers to single interpolation, $-+-$ refers to double interpolation, $-\times-$ refers to LR interpolation. The running times are in seconds.

6.3 Case 1 study: effect of matrix dimension N

Figure 3 reports the performance of PRPCA and RPCA over different matrix dimension N under Case 1. The noise level and sparsity level of $\boldsymbol{Y}_0$ are fixed at $\sigma = 0.6$ and $\rho_s = 0.05$.

We again see that the advantage of PRPCA with single and LR interpolation in recovering $\boldsymbol{P}_0\boldsymbol{X}_0\boldsymbol{Q}_0^\top$, $\boldsymbol{Y}_0$ and $\boldsymbol{\Theta}$ across all the values of N. In addition, the PRPCA with double interpolation also outperforms RPCA in recovering $\boldsymbol{Y}_0$ and $\boldsymbol{\Theta}$ when the matrix is of high dimension, e.g., $N \geqslant 200$. In terms of computation, the running time of RPCA increases almost exponentially as N increase. The computational benefits of applying PRPCA are even more significant for high-dimensional matrix problems.

Under Case 1, the PRPCA with single interpolation is expected to outperform LR interpolation as the $\boldsymbol{P}$ and $\boldsymbol{Q}$ are correctly specified. We now consider Case 2, under which the $\boldsymbol{P}_0$ and $\boldsymbol{Q}_0$ are generated with noise. This allows us to test the performance of PRPCA when $\boldsymbol{P}$ and $\boldsymbol{Q}$ are not perfectly specified.

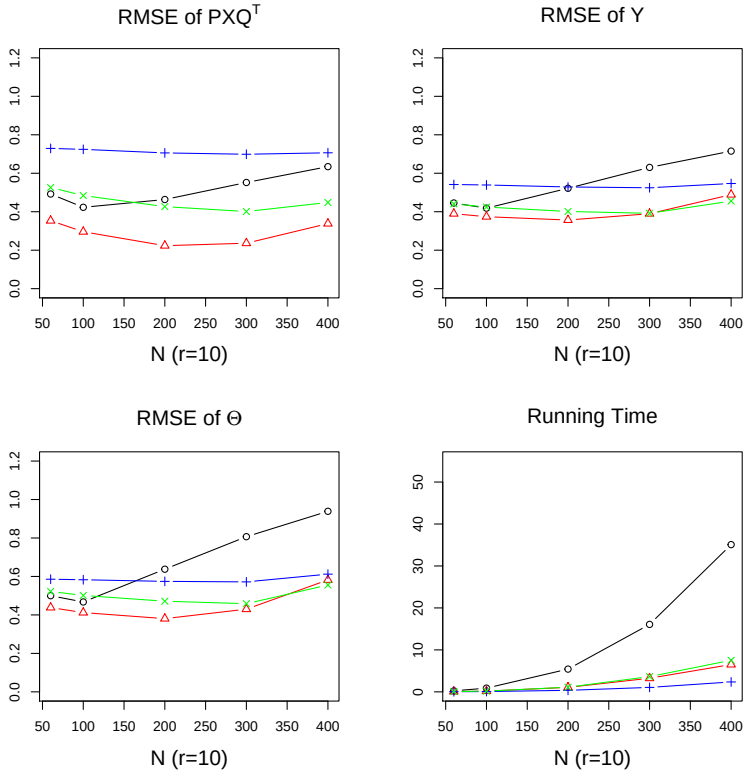

Figure 3: RMSE and running time with different $(\boldsymbol{P}, \boldsymbol{Q})$ ranges over different N and r. $\sigma = 0.6$, $\rho_s = 0.1$, $M = N$. In the first four plots, r is fixed at 10, while in the second four plots, $r = 0.05N$. Here: $\circ$ refers to no interpolation, $-\triangle-$ refers to single interpolation, $- \times -$ refers to LR interpolation, $- + -$ refers to double interpolation. The running times are in seconds.

6.4 Case 2 study: effect of mis-specified $\boldsymbol{P}_0$ and $\boldsymbol{Q}_0$ across N

Figure 4 reports the performance of PRPCA and RPCA when $\boldsymbol{P}_0$ and $\boldsymbol{Q}_0$ are generated with noise across N. The noise level and sparsity level of $\boldsymbol{Y}_0$ are fixed at $\sigma = 0.2$ and $\rho_s = 0.05$.

Under Case 2, we observe that the performance of PRPCA with LR interpolation demonstrate clear advantages over other approaches. Such advantage is resulted from the fact that the LR interpolation is data-dependent and able to achieve a better estimation of $(\boldsymbol{P}_0, \boldsymbol{Q}_0)$. While for the PRPCA with single and double interpolation matrices, they use a mis-specified $(\boldsymbol{P}, \boldsymbol{Q})$ but still demonstrate robust performances, especially for recovering Θ. Indeed, although the PRPCA with mis-specified smoothing mechanism may not perform as good as RPCA in recovering $\boldsymbol{PXQ}^\top$ and $\boldsymbol{Y}$, but they in general achieve an better accuracy in recovering $\boldsymbol{\Theta}$. An explanation for such phenomena is that the PRPCA could decompose $\boldsymbol{\Theta}$ differently with the sparse component accounting for more signals. As a consequence, the target $\boldsymbol{\Theta}$ can be recovered well even with mis-specified smoothing matrices. In terms of computation, we see a similar pattern as before: imposing the interpolation matrices is able to expedite computation significantly.

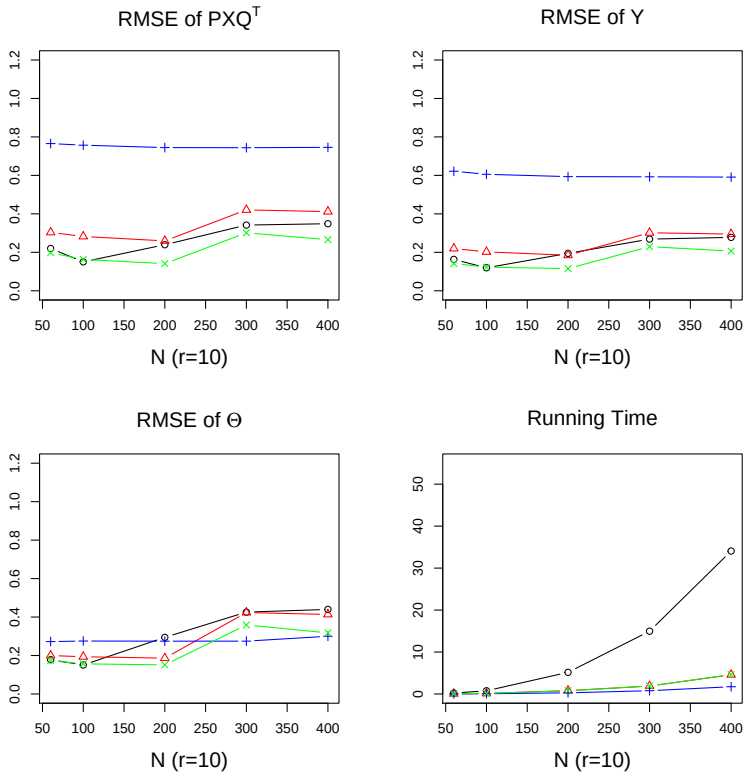

Figure 4: RMSE and running time with different $(\boldsymbol{P}, \boldsymbol{Q})$ ranges over different N and r. $\sigma = 0.6$, $\rho_s = 0.05$, $M = N$, $r = 10$. Here: $\circ$ refers to no interpolation, $-\triangle-$ refers to single interpolation, $-\times-$ refers to LR interpolation, $-+-$ refers to double interpolation. The running times are in seconds.

After all, we conclude that the PRPCA with interpolation matrices performs consistently well across a large range of noise levels, matrix dimensions, sparsities of $\boldsymbol{Y}_0$. Moreover, even with mis-specified smoothing matrices, the PRPCA is still able to produce robust estimation, especially for recovering the mean matrix $\boldsymbol{\Theta}$.

7. Real Data Analysis

In this section, we conduct two real data analyses: 1) the image of Lenna, a benchmark in image data analysis, and 2) a surveillance video, to demonstrate the advantage of PRPCA with interpolation matrices over the RPCA.

7.1 The Lenna image analysis

We consider the gradyscale Lenna image, which is of dimension 512×512 and can be found at https://www.ece.rice.edu/ wakin/images/. The image is displayed in Figure 5 (left).

Figure 5: The Lenna image with different noise levels, $\sigma = 0.05$ (left), $\sigma = 0.15$ (middle), $\sigma = 0.25$ (right).

We re-scale the Lenna image such that each pixel of the image ranges from 0 to 1, with 0 representing pure black and 1 representing pure white. Our target is to recover the Lenna image from its noisy version with different noise levels. That is, we observe

$$\boldsymbol{Z} = \boldsymbol{\Theta} + \boldsymbol{E},$$

where $\boldsymbol{\Theta}$ is the true Lenna image and $\boldsymbol{E}$ is the noise term with i.i.d entries generated from $\mathcal{N}(0, \sigma^2)$. We consider the noise levels ranging from 0.05 to 0.25. Specifically, we let $\sigma = 0.05, 0.1, 0.15, 0.2, 0.25$. Figure 5 below plots the Lenna image with different noise levels.

As in the simulation study, we recover the image with three sets of $(\boldsymbol{P}, \boldsymbol{Q})$: 1) identity matrices 2) single interpolation matrices 3) LR interpolation and 4) double interpolation matrices.

The penalty levels are still fixed at $\lambda_1 = \sqrt{2N}\sigma$ and $\lambda_1 = \sqrt{2}\sigma$ for all three sets of $(\boldsymbol{P}, \boldsymbol{Q})$. As the true low-rank component $\boldsymbol{P}\boldsymbol{X}_0\boldsymbol{Q}^\top$ and sparse component $\boldsymbol{Y}_0$ are not available in the real image analysis, we only measure the RMSE of $\boldsymbol{\Theta}$ and the computation time. We generate 100 independent noise terms $\boldsymbol{E}$ and report the mean running time and RMSE in Figure 7. In addition, we plot the recovered Lenna image with different noise levels in one implementation in Figure 6.

Figure 6: Recovered Lenna image with no interpolation (first column), single interpolation (second column), double interpolation (third column), LR interpolation (last column) when, $\sigma = 0.05$ (the first row), $\sigma = 0.15$ (the middle row), $\sigma = 0.25$ (the last row).

From Figure 7 and Figure 6, it is clear that the three PRPCA approaches outperform the RPCA significantly in terms of both image recovery accuracy and computation time across the whole range of σ. Recall that the Lenna image is of dimension 512×512. Under such dimension, the computational benefits of PRPCA are even more significant. The PRPCA with single or LR interpolation is on average 10 times faster than RPCA, while PRPCA

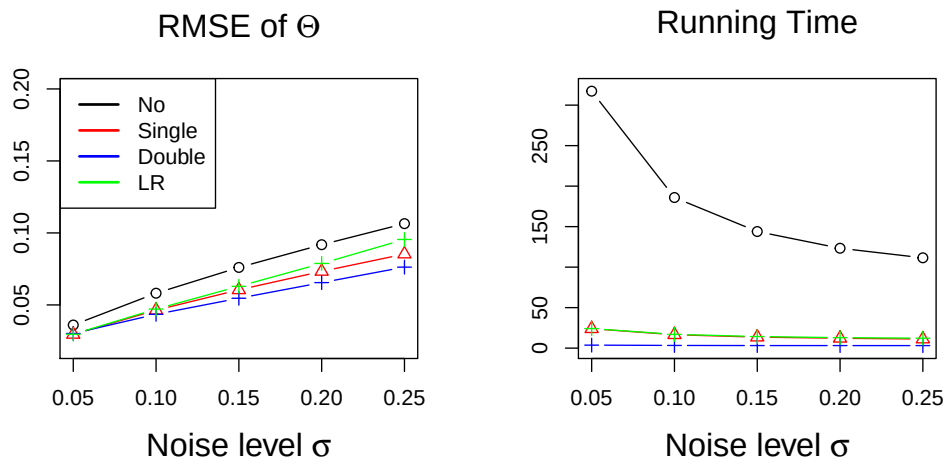

Figure 7: The RMSE of $\boldsymbol{\Theta}$ and running time of the Lenna image analysis with different $(\boldsymbol{P}, \boldsymbol{Q})$ and different σ. Here: $\circ$ refers to no interpolation, $-\triangle-$ refers to single interpolation, $-\times-$ refers to LR interpolation, $-+-$ refers to double interpolation. The running times are in seconds.

with double interpolation is at least 30 times faster than RPCA. In extreme cases, when the noise level is low, e.g., $\sigma = 0.05$, the average running time of PRPCA with double interpolation is 3.6 seconds. While RPCA requires 311.7 second, more than 86 times of that of PRPCA.

In terms of recovery accuracy, we see that the PRPCA with double interpolation even outperforms PRPCA with single or LR interpolation across the whole range of σ. In the simulation study, we conclude that when the target matrix is of large dimension and the sparsity of $\boldsymbol{Y}_0$, ρ_s, is high, the PRPCA with double or even more interpolation matrices would work well in terms of mean matrix $\boldsymbol{\Theta}$ recovery. The Lenna image can be viewed as such kind, with resolution 512×512 and although unknown, but potentially large ρ_s. Thus it is not surprised to see the outstanding performance of double interpolated PRPCA for the Lenna image analysis.

On the other hand, we note that for the Lenna image analysis, the single interpolated PRPCA demonstrates clear advantages not only compared to the classical RPCA, but also compared to the more adaptive PRPCA with LR interpolation. In other words, although the single interpolation matrix is not data-dependent, a simple equal-weight average smoothing mechanism could benefit many image problems tremendously.

7.2 The surveillance video analysis

In this subsection, we consider the problem of recovering a surveillance video from the CAVIAR project: https://homepages.inf.ed.ac.uk/rbf/CAVIARDATA1/. In this project, a number of video clips were recorded for different scenarios. In our study, we consider the scenario with one person walking on straight line.

The video is half-resolution PAL standard (288*384 pixels, 25 frames per second), with a total of 611 frames. From computational perspectives, we first resize each frame of video into 56×80 $(= 4480)$ pixels and then keep the last 411 frames. We stack each frame into a matrix column and obtain a large matrix of dimension 4480×411. Denote the large matrix as $\boldsymbol{\Theta}$.

As in the Lenna image analysis, we consider recovering $\boldsymbol{\Theta}$ from noisy observations $\boldsymbol{Z} = \boldsymbol{\Theta} + \boldsymbol{E}$, with $\boldsymbol{E} \sim \mathcal{N}(0, \sigma^2)$ and the same set of σ: $\sigma = 0.05, 0.1, 0.15, 0.2, 0.25$. Such setting refers to the case that each frame is compromised with independent white noise. We implement PRPCA with the same sets of $(\boldsymbol{P}, \boldsymbol{Q})$ as in the previous subsection. The RMSE and running time for the surveillance video analysis are plotted in Figure 8 below. Note that we omit the running time for "no interpolation" as its running time is of difference scale compared to the others (5 times more than that of single interpolation).

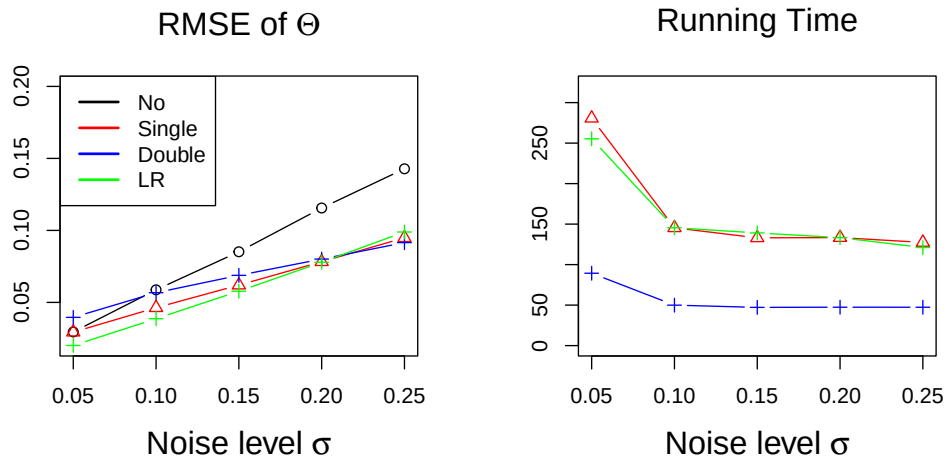

Figure 8: The RMSE of $\boldsymbol{\Theta}$ and running time of the surveillance video analysis with different $(\boldsymbol{P}, \boldsymbol{Q})$ and different σ. Here: $\circ$ refers to no interpolation, $-\triangle-$ refers to single interpolation, $- \times -$ refers to LR interpolation, $- + -$ refers to double interpolation. We omit the running time for "no interpolation" as its running time is of difference scale compared with others (5 times more than that of single interpolation). The running times are in seconds.

In Figure 8, we see a similar pattern as in the Lenna image analysis: all three PRPCA approaches outperform the RPCA in terms of both image recovery accuracy and computation time across different noise levels. Within three PRPCA approaches, the LR interpolation shows more advantages when the noise is small. It is because under such cases we could produce a more accurate estimates for the LR weights. While on the other hand, the double interpolation performs better when the noise is large. It is more beneficial to impose the smoothness structure and allow the neighborhood pixels to learn from each other when the noise is large.

After all, the surveillance video and Lenna image analysis further validate the advantages of PRPCA over RPCA for smooth image and video recovery. Especially when image/video is of high resolution and complex (large latent ρ_s or noise level σ), the advantages of PRPCA could be more significant. Such advantages are reflected not only in computation, but also in recovery accuracy.

8. Conclusions and Future Work

In this paper, we develop a novel framework of projected RPCA motivated by smooth image recovery. This framework is general in the sense that it includes not only the classical RPCA as a special case, but also works for multivariate reduced rank regression with outliers. Theoretically, we derive explicit error bounds on the estimation of $\boldsymbol{P}\boldsymbol{X}_0\boldsymbol{Q}^\top$ and $\boldsymbol{Y}_0$. Our bounds match the optimum bounds in RPCA. In addition, by bringing the interpolation matrices into PRPCA model, we could not only significantly speed up the computation of the RPCA, but also improve matrix accuracy, which was demonstrated by a comprehensive simulation study and a real image data analysis. Due to the prevalence of low-rank and smooth images and (stacked) videos, this paper would greatly advance future research on many computer vision problems and demonstrate the potential of statistical methods on computer vision study.

We conclude with the discussion of future works. The first direction is to explore the performance of PRPCA in a missing entry scenario. That is, when the entries of $\boldsymbol{Z}$ are observed with both missingness and noise, how would the PRPCA perform in terms of matrix recovery accuracy compared to RPCA? Consider an image inpainting problem, where images are observed with missing pixels. Intuitively, it would be more beneficial if we could borrow information from the observed pixels for its neighbor missing entries. In other words, the interpolation matrices could play an even more significant role in image inpainting problems. Empirically, it is interesting to discover how different missing patterns and missing rates would affect the performance of PRPCA. Theoretically, it would also be significant to derive the error bounds under missing entry scenarios. Such derivation may be more challenging as the dual certificate we construct in Theorem 8 may not be generalized directly.

Second, note that our study is restricted to the matrix version. Indeed, considering the fact the tensor data would usually be of even higher dimension, it would be interesting to generalize PRPCA to the tensor case. However, a tensor version RPCA or PRPCA is far more than simple generalizations of their matrix version. In particular, the low-rankness of a tensor is a much more complicated property compared to its matrix counterpart. Indeed, the tensor rank is defined differently under different tensor decompositions, such as Canonical Polyadic (CP) decomposition or Tucker decomposition. In a rather recent work, Cai et al. (2021) considered a tensor recovery problem when the input tensor can be decomposed as a sparse component plus a (Tucker) low-rank component, without considering the potential smoothness in image recovery. Moreover, they proposed a Riemannian gradient descent algorithm to solve such a tensor decomposition problem. For tensor PRPCA, how to appropriately incorporate the smooth property into tensors would be our great interest in the future work.

References

Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1):183–202, 2009.

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.

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

Jian-Feng Cai, Jingyang Li, and Dong Xia. Generalized low-rank plus sparse tensor estimation by fast riemannian optimization. *arXiv preprint arXiv:2103.08895*, 2021.

Emmanuel J Candès, Xiaodong Li, Yi Ma, and John Wright. Robust principal component analysis? *Journal of the ACM (JACM)*, 58(3):11, 2011.

Venkat Chandrasekaran, Sujay Sanghavi, Pablo A Parrilo, and Alan S Willsky. Rank-sparsity incoherence for matrix decomposition. *SIAM Journal on Optimization*, 21(2): 572–596, 2011.

Yudong Chen. Incoherence-optimal matrix completion. *IEEE Transactions on Information Theory*, 61(5):2909–2923, 2015.

Yuxin Chen, Jianqing Fan, Cong Ma, and Yuling Yan. Bridging convex and nonconvex optimization in robust pca: Noise, outliers, and missing data. *arXiv preprint arXiv:2001.05484*, 2020.

Kenneth R Davidson and Stanislaw J Szarek. Local operator theory, random matrices and banach spaces. *Handbook of the geometry of Banach spaces*, 1(317-366):131, 2001.

Trevor Hastie, Rahul Mazumder, Jason D Lee, and Reza Zadeh. Matrix completion and low-rank svd via fast alternating least squares. *The Journal of Machine Learning Research*, 16(1):3367–3402, 2015.

Vahan Hovhannisyan, Yannis Panagakis, Panos Parpas, and Stefanos Zafeiriou. Fast multilevel algorithms for compressive principal component pursuit. *SIAM Journal on Imaging Sciences*, 12(1):624–649, 2019.

Daniel Hsu, Sham M Kakade, and Tong Zhang. Robust matrix decomposition with sparse corruptions. *IEEE Transactions on Information Theory*, 57(11):7221–7234, 2011.

Peter J Huber. Robust estimation of a location parameter. In *Breakthroughs in statistics*, pages 492–518. Springer, 1992.

Prateek Jain and Inderjit S Dhillon. Provable inductive matrix completion. *arXiv preprint arXiv:1306.0626*, 2013.

Olga Klopp, Karim Lounici, and Alexandre B Tsybakov. Robust matrix completion. *Probability Theory and Related Fields*, 169(1-2):523–564, 2017.

Joseph A Maldjian, Paul J Laurienti, Robert A Kraft, and Jonathan H Burdette. An automated method for neuroanatomic and cytoarchitectonic atlas-based interrogation of fmri data sets. *Neuroimage*, 19(3):1233–1239, 2003.

Yu Nesterov. Gradient methods for minimizing composite functions. *Mathematical Programming*, 140(1):125–161, 2013.

Omkar M Parkhi, Andrea Vedaldi, and Andrew Zisserman. Deep face recognition. 2015.

Leonid I Rudin and Stanley Osher. Total variation based image restoration with free local constraints. *Proceedings of 1st International Conference on Image Processing*, 1:31–35, 1994.

Leonid I Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1-4):259–268, 1992.

Yiyuan She and Kun Chen. Robust reduced-rank regression. *Biometrika*, 104(3):633–647, 2017.

Robert Tibshirani, Michael Saunders, Saharon Rosset, Ji Zhu, and Keith Knight. Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(1):91–108, 2005.

Namrata Vaswani, Thierry Bouwmans, Sajid Javed, and Praneeth Narayanamurthy. Robust subspace learning: Robust pca, robust subspace tracking, and robust subspace recovery. *IEEE signal processing magazine*, 35(4):32–55, 2018.

Xiao Wang, Hongtu Zhu, and Alzheimer's Disease Neuroimaging Initiative. Generalized scalar-on-image regression models via total variation. *Journal of the American Statistical Association*, 112(519):1156–1168, 2017.

John Wright, Arvind Ganesh, Kerui Min, and Yi Ma. Compressive principal component pursuit. *Information and Inference: A Journal of the IMA*, 2(1):32–68, 2013.

Zihan Zhou, Xiaodong Li, John Wright, Emmanuel Candes, and Yi Ma. Stable principal component pursuit. In *2010 IEEE international symposium on information theory*, pages 1518–1522. IEEE, 2010.

Appendix A.

In the appendix, we provide proofs in the following order: Proposition 3, Proposition 4, Theorem 7, Theorem 8, Theorem 5.

Proof of Proposition 3. We first show that the smallest singular value of interpolation matrices $\sigma_{\min}(\boldsymbol{P})$ is greater than 1. This is because for any $\boldsymbol{u} \in \mathbb{R}^n$,

$$\|\boldsymbol{P}\boldsymbol{u}\|_2^2 = \sum_{j=1}^{N} (\boldsymbol{P}_{j,\cdot}\boldsymbol{u})^2 > \sum_{1 \leqslant j \leqslant N \ \&\ j \text{ is even}} (\boldsymbol{P}_{j,\cdot}\boldsymbol{u})^2 = \sum_{j=1}^{n} \boldsymbol{u}_j^2 = \|\boldsymbol{u}\|_2^2.$$

Similarly we have $\sigma_{\min}(\boldsymbol{Q}) > 1$. Then it follows that

$$\|\boldsymbol{PX}\|_* \geqslant \sum_{i=1}^{r} \sigma_i(\boldsymbol{X})\sigma_{\min}(\boldsymbol{P}) > \|\boldsymbol{X}\|_*$$

where r is the rank of $\boldsymbol{X}$. Furthermore,

$$\|\boldsymbol{PXQ}^\top\|_* \geqslant \sum_{i=1}^{r} \sigma_i(\boldsymbol{PX})\sigma_{\min}(\boldsymbol{Q}) > \|\boldsymbol{PX}\|_* > \|\boldsymbol{X}\|_*.$$

This completes the proof.

Proof of Proposition 4. When $\|\mathcal{P}_{\mathcal{T}_0^\perp}(\boldsymbol{X})\|_* = 0$, we have $\boldsymbol{X} = \mathcal{P}_{\mathcal{T}_0}(\boldsymbol{X})$, in other words, $\boldsymbol{X} \in \mathcal{T}_0$. Thus $\boldsymbol{X}$ can be written as $\boldsymbol{X} = \boldsymbol{U}_0\boldsymbol{X}_1^\top + \boldsymbol{X}_2\boldsymbol{V}_0^\top$ for certain matrices $\boldsymbol{X}_1 \in \mathbb{R}^{m\times r}$ and $\boldsymbol{X}_2 \in \mathbb{R}^{n\times r}$. It then follows that

$$\begin{aligned}
&\mathcal{P}_{\mathcal{T}}(\boldsymbol{PXQ}^\top)\\
=&\mathcal{P}_{\mathcal{T}}(\boldsymbol{PU}_0\boldsymbol{X}_1^\top\boldsymbol{Q}^\top + \boldsymbol{PX}_2\boldsymbol{V}_0^\top\boldsymbol{Q}^\top)\\
=&\tilde{\boldsymbol{U}}_0\tilde{\boldsymbol{U}}_0^T\boldsymbol{PU}_0\boldsymbol{X}_1^\top\boldsymbol{Q}^\top + \tilde{\boldsymbol{U}}_0\tilde{\boldsymbol{U}}_0^T\boldsymbol{PX}_2\boldsymbol{V}_0^\top\boldsymbol{Q}^\top\\
&+ \boldsymbol{PU}_0\boldsymbol{X}_1^\top\boldsymbol{Q}^\top\tilde{\boldsymbol{V}}_0\tilde{\boldsymbol{V}}_0^\top + \boldsymbol{PX}_2\boldsymbol{V}_0^\top\boldsymbol{Q}^\top\tilde{\boldsymbol{V}}_0\tilde{\boldsymbol{V}}_0^\top\\
&- \tilde{\boldsymbol{U}}_0\tilde{\boldsymbol{U}}_0^T\boldsymbol{PU}_0\boldsymbol{X}_1^\top\boldsymbol{Q}^\top\tilde{\boldsymbol{V}}_0\tilde{\boldsymbol{V}}_0^\top + \tilde{\boldsymbol{U}}_0\tilde{\boldsymbol{U}}_0^T\boldsymbol{PX}_2\boldsymbol{V}_0^\top\boldsymbol{Q}^\top\tilde{\boldsymbol{V}}_0\tilde{\boldsymbol{V}}_0^\top\\
=&\boldsymbol{PU}_0\boldsymbol{X}_1^\top\boldsymbol{Q}^\top + \tilde{\boldsymbol{U}}_0\tilde{\boldsymbol{U}}_0^T\boldsymbol{PX}_2\boldsymbol{V}_0^\top\boldsymbol{Q}^\top\\
&+ \boldsymbol{PU}_0\boldsymbol{X}_1^\top\boldsymbol{Q}^\top\tilde{\boldsymbol{V}}_0\tilde{\boldsymbol{V}}_0^\top + \boldsymbol{PX}_2\boldsymbol{V}_0^\top\boldsymbol{Q}^\top\\
&- \boldsymbol{PU}_0\boldsymbol{X}_1^\top\boldsymbol{Q}^\top\tilde{\boldsymbol{V}}_0\tilde{\boldsymbol{V}}_0^\top + \tilde{\boldsymbol{U}}_0\tilde{\boldsymbol{U}}_0^T\boldsymbol{PX}_2\boldsymbol{V}_0^\top\boldsymbol{Q}^\top\\
=&\boldsymbol{PU}_0\boldsymbol{X}_1^\top\boldsymbol{Q}^\top + \boldsymbol{PX}_2\boldsymbol{V}_0^\top\boldsymbol{Q}^\top\\
=&\boldsymbol{PXQ}^\top
\end{aligned} \tag{45}$$

Therefore, $\mathcal{P}_{\mathcal{T}^\perp}(\boldsymbol{PXQ}^\top) = \boldsymbol{0}$. This completes the proof.

Proof of Theorem 7. The condition

$$\mathcal{P}_{\mathcal{T}}(\boldsymbol{D}) = \boldsymbol{\Gamma}$$

suggests that

$$\tilde{\boldsymbol{U}}_0\tilde{\boldsymbol{U}}_0^T\boldsymbol{D} + \boldsymbol{D}\tilde{\boldsymbol{V}}_0\tilde{\boldsymbol{V}}_0^\top - \tilde{\boldsymbol{U}}_0\tilde{\boldsymbol{U}}_0^\top\boldsymbol{D}\tilde{\boldsymbol{V}}_0\tilde{\boldsymbol{V}}_0^\top = \boldsymbol{\Gamma}. \tag{46}$$

Left-multiply $\boldsymbol{U}_0\boldsymbol{U}_0^\top\boldsymbol{P}^\top$ and right-multiply $\boldsymbol{Q}$ on both sides of (46), we get

$$\boldsymbol{U}_0\boldsymbol{U}_0^\top\boldsymbol{P}^\top\left(\tilde{\boldsymbol{U}}_0\tilde{\boldsymbol{U}}_0^T\boldsymbol{D} + \boldsymbol{D}\tilde{\boldsymbol{V}}_0\tilde{\boldsymbol{V}}_0^\top - \tilde{\boldsymbol{U}}_0\tilde{\boldsymbol{U}}_0^\top\boldsymbol{D}\tilde{\boldsymbol{V}}_0\tilde{\boldsymbol{V}}_0^\top\right)\boldsymbol{Q} = \boldsymbol{U}_0\boldsymbol{U}_0^\top\boldsymbol{P}^\top\boldsymbol{\Gamma}\boldsymbol{Q}. \tag{47}$$

For the LHS of (47),

$$\begin{aligned}
&\boldsymbol{U}_0\boldsymbol{U}_0^\top\boldsymbol{P}^\top\left(\tilde{\boldsymbol{U}}_0\tilde{\boldsymbol{U}}_0^T\boldsymbol{D}+\boldsymbol{D}\tilde{\boldsymbol{V}}_0\tilde{\boldsymbol{V}}_0^\top-\tilde{\boldsymbol{U}}_0\tilde{\boldsymbol{U}}_0^\top\boldsymbol{D}\tilde{\boldsymbol{V}}_0\tilde{\boldsymbol{V}}_0^\top\right)\boldsymbol{Q}\\
=&\left(\boldsymbol{U}_0\boldsymbol{U}_0^\top\boldsymbol{P}^\top\boldsymbol{D}+\boldsymbol{U}_0\boldsymbol{U}_0\boldsymbol{P}^\top\boldsymbol{D}\tilde{\boldsymbol{V}}_0\tilde{\boldsymbol{V}}_0^\top-\boldsymbol{U}_0\boldsymbol{U}_0\boldsymbol{P}^\top\boldsymbol{D}\tilde{\boldsymbol{V}}_0\tilde{\boldsymbol{V}}_0^\top\right)\boldsymbol{Q}\\
=&\boldsymbol{U}_0\boldsymbol{U}_0^\top\boldsymbol{P}^\top\boldsymbol{D}\boldsymbol{Q},
\end{aligned}\tag{48}$$

For the RHS of (47),

$$\begin{aligned}
&\boldsymbol{U}_0\boldsymbol{U}_0^\top\boldsymbol{P}^\top\boldsymbol{\Gamma}\boldsymbol{Q}\\
=&\boldsymbol{U}_0\boldsymbol{U}_0^\top\boldsymbol{P}^\top\left((\boldsymbol{P}\boldsymbol{U}_0)^+\right)^\top\boldsymbol{V}_0^\top\boldsymbol{Q}^+\boldsymbol{Q}+\boldsymbol{U}_0\boldsymbol{U}_0^\top\boldsymbol{P}^\top(\boldsymbol{P}^+)^\top\boldsymbol{U}_0(\boldsymbol{Q}\boldsymbol{V}_0)^+\boldsymbol{Q}\\
&-\boldsymbol{U}_0\boldsymbol{U}_0^\top\boldsymbol{P}^\top\left((\boldsymbol{P}\boldsymbol{U}_0)^+\right)^\top(\boldsymbol{Q}\boldsymbol{V}_0)^+\boldsymbol{Q}\\
=&\boldsymbol{U}_0\boldsymbol{V}_0^\top+\boldsymbol{U}_0(\boldsymbol{Q}\boldsymbol{V}_0)^+\boldsymbol{Q}-\boldsymbol{U}_0(\boldsymbol{Q}\boldsymbol{V}_0)^+\boldsymbol{Q}\\
=&\boldsymbol{U}_0\boldsymbol{V}_0^\top.
\end{aligned}\tag{49}$$

Plug (48) and (49) into (47), we have

$$\boldsymbol{U}_0\boldsymbol{U}_0^\top\boldsymbol{P}^\top\boldsymbol{D}\boldsymbol{Q}=\boldsymbol{U}_0\boldsymbol{V}_0^\top.\tag{50}$$

Similarly, left-multiply $\boldsymbol{P}^\top$ and right-multiply $\boldsymbol{Q}\boldsymbol{V}_0\boldsymbol{V}_0^\top$ on both sides of (46), we get

$$\boldsymbol{P}^\top\boldsymbol{D}\boldsymbol{Q}\boldsymbol{V}_0\boldsymbol{V}_0^\top=\boldsymbol{U}_0\boldsymbol{V}_0^\top.\tag{51}$$

Finally, left-multiply $\boldsymbol{U}_0\boldsymbol{U}_0^\top\boldsymbol{P}^\top$ and right-multiply $\boldsymbol{Q}\boldsymbol{V}_0\boldsymbol{V}_0^\top$ on both sides of (46), we get

$$\boldsymbol{U}_0\boldsymbol{U}_0^\top\boldsymbol{P}^\top\boldsymbol{D}\boldsymbol{Q}\boldsymbol{V}_0\boldsymbol{V}_0^\top=\boldsymbol{U}_0\boldsymbol{V}_0^\top.\tag{52}$$

Combine (50), (51) and (52) together, we have

$$\boldsymbol{U}_0\boldsymbol{U}_0^\top\boldsymbol{P}^\top\boldsymbol{D}\boldsymbol{Q}+\boldsymbol{P}^\top\boldsymbol{D}\boldsymbol{Q}\boldsymbol{V}_0\boldsymbol{V}_0^\top-\boldsymbol{U}_0\boldsymbol{U}_0^\top\boldsymbol{P}^\top\boldsymbol{D}\boldsymbol{Q}\boldsymbol{V}_0\boldsymbol{V}_0^\top=\boldsymbol{U}_0\boldsymbol{V}_0^\top.$$

In other words,

$$\mathcal{P}_{\mathcal{T}_0}(\boldsymbol{P}^\top\boldsymbol{D}\boldsymbol{Q})=\boldsymbol{U}_0\boldsymbol{V}_0^\top.$$

This completes the proof.

Before proving Theorem 8, we first provide the following Definitions and Lemmas from Hsu et al. (2011).

Definition 10 *The matrix norm $\|\cdot\|_{\#'}$ is said to be the dual norm of $\|\cdot\|_{\#}$ if for all $\boldsymbol{M}$, $\|\boldsymbol{M}\|_{\#'}=\sup_{\|N\|_{\#}\leqslant 1}\langle\boldsymbol{M},\boldsymbol{N}\rangle$.*

Lemma 11 *For any linear matrix operator $\mathcal{T}:\mathbb{R}^{n\times m}\to\mathbb{R}^{n\times m}$, and any pair of matrix norms $\|\cdot\|_{\#}$ and $\|\cdot\|_{*}$, we have*

$$\|\mathcal{T}\|_{\#\to *}=\|\mathcal{T}^*\|_{*'\to\#'}$$

where $\|\cdot\|_{\#'}$ is the dual norm of $\|\cdot\|_{\#}$ and $\|\cdot\|_{*'}$ is the dual norm of $\|\cdot\|_{*}$.

Lemma 12 *For any matrix $\boldsymbol{M} \in \mathbb{R}^{n\times m}$ and $p \in \{1, \infty\}$, we have*

$$\|\mathcal{P}_{\mathcal{S}}\|_{vec(\infty)\to\star(\rho)} \leqslant \alpha(\rho).$$

where the norm $\|\cdot\|_{\star(\rho)}$ is defined as

$$\|\boldsymbol{M}\|_{\star(\rho)} = \max\{\rho\|\boldsymbol{M}\|_{1\to1}, \rho^{-1}\|\boldsymbol{M}\|_{\infty\to\infty}\}.$$

Lemma 13 *For any matrix $\boldsymbol{M} \in \mathbb{R}^{n\times m}$, we have for all $\rho > 0$,*

$$\|\boldsymbol{M}\|_{2\to2} \leqslant \|\boldsymbol{M}\|_{\star(\rho)}.$$

Lemma 14 *For any matrix $\boldsymbol{M} \in \mathbb{R}^{N\times M}$ and $p \in \{1, \infty\}$, we have*

$$\begin{aligned}
&\|\mathcal{P}_{\mathcal{S}}(\boldsymbol{M})\|_{p\to p} \leqslant \|\mathrm{sgn}(\boldsymbol{X}_0)\|_{p\to p}\|\boldsymbol{M}\|_{vec(\infty)},\\
&\|\mathcal{P}_{\mathcal{S}}\|_{vec(\infty)\to\star(\rho)} \leqslant \alpha(\rho),\\
&\|\mathcal{P}_{\mathcal{T}}(\boldsymbol{M})\|_{*} \leqslant 2r\|\boldsymbol{M}\|_{2\to2},\\
&\|\mathcal{P}_{\mathcal{T}}(\boldsymbol{M})\|_{vec(2)} \leqslant 2\sqrt{r}\|\boldsymbol{M}\|_{2\to2}.
\end{aligned}$$

Lemma 15 *For any linear matrix operator $\mathcal{T}_1 : \mathbb{R}^{n\times m} \to \mathbb{R}^{n\times m}$ and $\mathcal{T}_2 : \mathbb{R}^{n\times m} \to \mathbb{R}^{n\times m}$, and any matrix norm $\|\cdot\|_{\#}$, if $\|\mathcal{T}_1 \circ \mathcal{T}_2\|_{\#} < 1$, then $\mathcal{I} - \mathcal{T}_1 \circ \mathcal{T}_2$ is convertible and satisfies*

$$\|(\mathcal{I} - \mathcal{T}_1 \circ \mathcal{T}_2)^{-1}\|_{\#\to\#} \leqslant \frac{1}{1 - \|\mathcal{T}_1 \circ \mathcal{T}_2\|_{\#\to\#}},$$

where $\mathcal{I}$ is the identity operator.

Now we are ready to prove Theorem 8.

Proof of Theorem 8. First, it is not hard to verify that $\boldsymbol{D}_{\mathcal{S}} \in \mathcal{S}$, $\boldsymbol{D}_{\mathcal{T}} \in \mathcal{T}$ and the first two equality of (35). The third equality of (35) followed by Theorem 7. We now prove (37).

$$\begin{aligned}
& \|\boldsymbol{D}_{\mathcal{S}}\|_{2\to2}\\
\leqslant\ & \|\boldsymbol{D}_{\mathcal{S}}\|_{\star(\rho)}\\
=\ & \left\|(\mathcal{I} - \mathcal{P}_{\mathcal{S}} \circ \mathcal{P}_{\mathcal{T}})^{-1}\left(\lambda_2\mathrm{sgn}(\boldsymbol{Y}_0) - \lambda_1\mathcal{P}_{\mathcal{S}}(\boldsymbol{\Gamma}) - (\mathcal{P}_{\mathcal{S}} \circ \mathcal{P}_{\mathcal{T}^\perp})(\boldsymbol{E})\right)\right\|_{\star(\rho)}\\
\leqslant\ & \frac{1}{1 - \alpha(\rho)\beta(\rho)}\left\|\lambda_2\mathrm{sgn}(\boldsymbol{Y}_0) - \lambda_1\mathcal{P}_{\mathcal{S}}(\boldsymbol{\Gamma}) - (\mathcal{P}_{\mathcal{S}} \circ \mathcal{P}_{\mathcal{T}^\perp})(\boldsymbol{E})\right\|_{\star(\rho)}\\
=\ & \frac{1}{1 - \alpha(\rho)\beta(\rho)}\Big(\lambda_2\|\mathrm{sgn}(\boldsymbol{Y}_0)\|_{\star(\rho)} + \lambda_1\left\|\mathcal{P}_{\mathcal{S}}(\boldsymbol{\Gamma})\right\|_{\star(\rho)} + \left\|(\mathcal{P}_{\mathcal{S}} \circ \mathcal{P}_{\mathcal{T}^\perp})(\boldsymbol{E})\right\|_{\star(\rho)}\Big)\\
\leqslant\ & \frac{\alpha(\rho)}{1 - \alpha(\rho)\beta(\rho)}(\lambda_2 + \gamma_1\lambda_1 + \epsilon_\infty),
\end{aligned}$$

where the first inequality holds by Lemma 13, the second inequality holds by Lemma 15, the last inequality hods by Lemma 14 and

$$\|(\mathcal{P}_{\mathcal{S}} \circ \mathcal{P}_{\mathcal{T}^\perp})(\boldsymbol{E})\|_{\star(\rho)} \leqslant \alpha(\rho)\|\mathcal{P}_{\mathcal{T}^\perp}(\boldsymbol{E})\|_{\infty} \leqslant \alpha(\rho)(\|\boldsymbol{E}\|_{\infty} + \|\mathcal{P}_{\mathcal{T}}(\boldsymbol{E})\|_{\infty}) \leqslant \alpha(\rho)\epsilon_\infty.$$

Similarly, for $\|\boldsymbol{D}_{\mathcal{T}}\|_\infty$,

$$\begin{aligned}
& \|\boldsymbol{D}_{\mathcal{T}}\|_\infty \\
= & \left\|(\mathcal{I}-\mathcal{P}_{\mathcal{T}}\circ\mathcal{P}_{\mathcal{S}})^{-1}\Big(\lambda_1\boldsymbol{\Gamma}-\lambda_2\mathcal{P}_{\mathcal{T}}\big(\mathrm{sgn}(\boldsymbol{Y}_0)\big)-(\mathcal{P}_{\mathcal{T}}\circ\mathcal{P}_{\mathcal{S}^\perp})(\boldsymbol{E})\Big)\right\|_\infty \\
\leqslant & \frac{1}{1-\alpha(\rho)\beta(\rho)}\left(\lambda_1\|\boldsymbol{\Gamma}\|_\infty+\lambda_2\big\|\mathcal{P}_{\mathcal{T}}\big(\mathrm{sgn}(\boldsymbol{Y}_0)\big)\big\|_\infty+\big\|(\mathcal{P}_{\mathcal{T}}\circ\mathcal{P}_{\mathcal{S}^\perp})(\boldsymbol{E})\big\|_\infty\right) \\
\leqslant & \frac{1}{1-\alpha(\rho)\beta(\rho)}\left(\gamma_1\lambda_1+\lambda_2\alpha(\rho)\beta(\rho)+\epsilon_\infty\right),
\end{aligned}$$

where for the last inequality we used the bound

$$\big\|(\mathcal{P}_{\mathcal{T}}\circ\mathcal{P}_{\mathcal{S}^\perp})(\boldsymbol{E})\big\|_\infty\leqslant\|\mathcal{P}_{\mathcal{T}}(\boldsymbol{E})-(\mathcal{P}_{\mathcal{T}}\circ\mathcal{P}_{\mathcal{S}})(\boldsymbol{E})\|_\infty\leqslant\|\mathcal{P}_{\mathcal{T}}(\boldsymbol{E})\|+\alpha(\rho)\beta(\rho)\|(\boldsymbol{E})\|_\infty\leqslant\epsilon_\infty.$$

For $\|\boldsymbol{D}_{\mathcal{T}}\|_*$,

$$\begin{aligned}
& \|\boldsymbol{D}_{\mathcal{T}}\|_* \\
\leqslant & r\|\boldsymbol{D}_{\mathcal{T}}\|_{2\to2} \\
= & r\|\mathcal{P}_{\mathcal{T}}(\boldsymbol{D}_{\mathcal{S}}+\boldsymbol{E})-\lambda_1\boldsymbol{\Gamma}\|_{2\to2} \\
= & r\left(\|\boldsymbol{D}_{\mathcal{S}}\|_{2\to2}+\|\boldsymbol{E}\|_{2\to2}+\lambda_1\gamma_2\right) \\
\leqslant & r\left(\frac{2\alpha(\rho)}{1-\alpha(\rho)\beta(\rho)}(\lambda_2+\gamma_1\lambda_1+\epsilon_{\mathrm{vec}(\infty)})+2\epsilon_{2\to2}+\lambda_1\gamma_2\right).
\end{aligned}$$

For $\|\boldsymbol{D}_{\mathcal{S}}\|_{\mathrm{vec}(1)}$, we have

$$\begin{aligned}
& \|\boldsymbol{D}_{\mathcal{S}}\|_{\mathrm{vec}(1)} \\
\leqslant & s\|\boldsymbol{D}_{\mathcal{S}}\|_{\mathrm{vec}(\infty)} \\
\leqslant & \frac{s}{1-\alpha(\rho)\beta(\rho)}\Big(\lambda_2\|\mathrm{sgn}(\boldsymbol{Y}_0)\|_{\mathrm{vec}(\infty)}+\lambda_1\big\|\mathcal{P}_{\mathcal{S}}(\boldsymbol{\Gamma})\big\|_{\mathrm{vec}(\infty)}+\|(\mathcal{P}_{\mathcal{S}}\circ\mathcal{P}_{\mathcal{T}^\perp})(\boldsymbol{E})\|_{\mathrm{vec}(\infty)}\Big) \\
\leqslant & \frac{s}{1-\alpha(\rho)\beta(\rho)}(\lambda_2+\lambda_1\gamma_1+\epsilon_\infty).
\end{aligned}$$

Finally,

$$\begin{aligned}
\|\boldsymbol{D}_{\mathcal{T}}+\boldsymbol{D}_{\mathcal{S}}\|_2^2 &= \langle\boldsymbol{D}_{\mathcal{S}},\mathcal{P}_{\mathcal{S}}(\boldsymbol{D}_{\mathcal{S}}+\boldsymbol{D}_{\mathcal{T}})\rangle+\langle\boldsymbol{D}_{\mathcal{T}},\mathcal{P}_{\mathcal{T}}(\boldsymbol{D}_{\mathcal{S}}+\boldsymbol{D}_{\mathcal{T}})\rangle \\
&= \langle\boldsymbol{D}_{\mathcal{S}},\lambda_2\mathcal{P}_{\mathcal{S}}(\mathrm{sgn}(\boldsymbol{Y}_0))-\mathcal{P}_{\mathcal{S}}(\boldsymbol{E})\rangle \\
&\quad +\langle\boldsymbol{D}_{\mathcal{T}},\lambda_1\mathcal{P}_{\mathcal{T}}(\boldsymbol{\Gamma})-\mathcal{P}_{\mathcal{T}}(\boldsymbol{E})\rangle \\
&\leqslant \|\boldsymbol{D}_{\mathcal{S}}\|_{\mathrm{vec}(1)}(\lambda_2+\epsilon_\infty)+\|\boldsymbol{D}_{\mathcal{T}}\|_*(\lambda_1\gamma_2+\epsilon_{2\to2}).
\end{aligned}$$

This finish the proof for (37). To prove (36), let $\boldsymbol{D}=\boldsymbol{D}_{\mathcal{S}}+\boldsymbol{D}_{\mathcal{T}}+\boldsymbol{E}$,

$$\begin{aligned}
\|\mathcal{P}_{\mathcal{T}_0^\perp}(\boldsymbol{P}^\top\boldsymbol{D}\boldsymbol{Q})\|_{2\to2} &= \left\|\mathcal{P}_{\mathcal{T}_0^\perp}\left(\boldsymbol{P}^\top(\boldsymbol{D}_{\mathcal{S}}+\boldsymbol{E})\boldsymbol{Q}\right)\right\|_{2\to2} \\
&\leqslant \sigma_{\max}(\boldsymbol{P})\sigma_{\max}(\boldsymbol{Q})\left(\|\boldsymbol{D}_{\mathcal{S}}\|_{2\to2}+\|\boldsymbol{E}\|_{2\to2}\right) \\
&\leqslant \sigma_{\max}(\boldsymbol{P})\sigma_{\max}(\boldsymbol{Q})\left(\frac{\alpha(\rho)(\lambda_2+\gamma_1\lambda_1+\epsilon_\infty)}{1-\alpha(\rho)\beta(\rho)}+\epsilon_{2\to2}\right)\leqslant\frac{\lambda_1}{c},
\end{aligned}$$

where the last inequality holds by penalty condition (i). Similarly, we can bound $\big\|\mathcal{P}_{\mathcal{S}^\perp}(\boldsymbol{D})\big\|_{\mathrm{vec}(\infty)}$ as below,

$$\big\|\mathcal{P}_{\mathcal{S}^\perp}(\boldsymbol{D})\big\|_{\mathrm{vec}(\infty)} = \left\|\mathcal{P}_{\mathcal{S}^\perp}\left(\boldsymbol{D}_{\mathcal{T}}+\boldsymbol{E}\right)\right\|_{\mathrm{vec}(\infty)}$$

$$\begin{aligned} &\leqslant \|\boldsymbol{D}_{\mathcal{T}}\|_{\infty} + \epsilon_{\infty} \\ &\leqslant \frac{1}{1-\alpha(\rho)\beta(\rho)}\left(\gamma_1\lambda_1 + \lambda_2\alpha(\rho)\beta(\rho) + \epsilon_{\infty}\right) + \epsilon_{\infty} \leqslant \frac{\lambda_2}{c}, \end{aligned}$$

where the last inequality holds by penalty condition (ii).

Proof of Theorem 5. To prove Theorem 5, we need the following Propositions.

Proposition 16 *For any $\lambda_1 > 0$, $\lambda_2 > 0$, define the penalty function $Pen(\boldsymbol{X}, \boldsymbol{Y}) = \lambda_1\|\boldsymbol{X}\|_* + \lambda_2\|\boldsymbol{Y}\|_{vec(1)}$ with domain $\boldsymbol{X} \in \mathbb{R}^{n\times m}$ and $\boldsymbol{Y} \in \mathbb{R}^{N\times M}$. Then, if there exists $\boldsymbol{D}$ satisfies*

$$\begin{aligned} \mathcal{P}_{\mathcal{T}}(\boldsymbol{D}) &= \lambda_1\boldsymbol{\Gamma} \\ \mathcal{P}_{\mathcal{S}}(\boldsymbol{D}) &= \lambda_2\ \mathrm{sgn}(\boldsymbol{Y}_0), \end{aligned}$$

and $\|\mathcal{P}_{\mathcal{T}_0^{\perp}}(\boldsymbol{P}^{\top}\boldsymbol{D}\boldsymbol{Q})\|_{2\to 2} \leqslant \lambda_1/c$, $\|\mathcal{P}_{\mathcal{S}^{\perp}}(\boldsymbol{D})\|_{vec(\infty)} \leqslant \lambda_2/c$, we have

$$\lambda_1\left\|\boldsymbol{X}_0 + \boldsymbol{\Delta}_{\boldsymbol{X}}\right\|_* - \lambda_1\|\boldsymbol{X}_0\|_* - \langle \boldsymbol{P}^{\top}\boldsymbol{D}\boldsymbol{Q}, \boldsymbol{\Delta}_{\boldsymbol{X}}\rangle \geqslant \lambda_1(1-1/c)\|\mathcal{P}_{\mathcal{T}_0^{\perp}}(\boldsymbol{\Delta}_{\boldsymbol{X}})\|_*,$$

and

$$\lambda_2\left\|\boldsymbol{Y}_0 + \boldsymbol{\Delta}_{\boldsymbol{Y}}\right\|_* - \lambda_2\left\|\boldsymbol{Y}_0\right\|_* - \langle \boldsymbol{D}, \boldsymbol{\Delta}_{\boldsymbol{Y}}\rangle \geqslant \lambda_2(1-1/c)\left\|\mathcal{P}_{\mathcal{S}^{\perp}}(\boldsymbol{\Delta}_{\boldsymbol{Y}})\right\|_{vec(1)}.$$

Proof of Proposition 16. First, by the construction of $\boldsymbol{D}$ and Theorem 7, we have $\mathcal{P}_{\mathcal{T}_0}(\boldsymbol{P}^T\boldsymbol{D}\boldsymbol{Q}) = \boldsymbol{U}_0\boldsymbol{V}_0^{\top}$. On the other hand, for any other sub-gradient $\boldsymbol{G} \in \partial_{\boldsymbol{X}}(\lambda_1\|\boldsymbol{X}_0\|_*)$, we have

$$\mathcal{P}_{\mathcal{T}_0}(\boldsymbol{G}) = \boldsymbol{U}_0\boldsymbol{V}_0^{\top}.$$

It then follows that

$$\begin{aligned} \boldsymbol{G} - \boldsymbol{P}^{\top}\boldsymbol{D}\boldsymbol{Q} &= \mathcal{P}_{\mathcal{T}_0}(\boldsymbol{G}) + \mathcal{P}_{\mathcal{T}_0^{\perp}}(\boldsymbol{G}) - \mathcal{P}_{\mathcal{T}_0}(\boldsymbol{P}^{\top}\boldsymbol{D}\boldsymbol{Q}) - \mathcal{P}_{\mathcal{T}_0^{\perp}}(\boldsymbol{P}^{\top}\boldsymbol{D}\boldsymbol{Q}) \\ &= \mathcal{P}_{\mathcal{T}_0^{\perp}}(\boldsymbol{G}) - \mathcal{P}_{\mathcal{T}_0^{\perp}}(\boldsymbol{P}^{\top}\boldsymbol{D}\boldsymbol{Q}). \end{aligned}$$

As a consequence,

$$\begin{aligned} &\lambda_1\left\|\boldsymbol{X}_0 + \boldsymbol{\Delta}_{\boldsymbol{X}}\right\|_* - \lambda_1\|\boldsymbol{X}_0\|_* - \langle \boldsymbol{P}^{\top}\boldsymbol{D}\boldsymbol{Q}, \boldsymbol{\Delta}_{\boldsymbol{X}}\rangle \\ \geqslant &\sup\left\{\langle \boldsymbol{G}, \boldsymbol{\Delta}_{\boldsymbol{X}}\rangle - \langle \boldsymbol{P}^{\top}\boldsymbol{D}\boldsymbol{Q}, \boldsymbol{\Delta}_{\boldsymbol{X}}\rangle : \quad \boldsymbol{G} \in \partial_{\boldsymbol{X}}(\lambda_1\|\boldsymbol{X}_0\|_*)\right\} \\ = &\sup\left\{\langle \boldsymbol{G} - \boldsymbol{P}^{\top}\boldsymbol{D}\boldsymbol{Q}, \boldsymbol{\Delta}_{\boldsymbol{X}}\rangle : \quad \boldsymbol{G} \in \partial_{\boldsymbol{X}}(\lambda_1\|\boldsymbol{X}_0\|_*)\right\} \\ = &\sup\left\{\langle \mathcal{P}_{\mathcal{T}_0^{\perp}}(\boldsymbol{G} - \boldsymbol{P}^{\top}\boldsymbol{D}\boldsymbol{Q}), \boldsymbol{\Delta}_{\boldsymbol{X}}\rangle : \quad \boldsymbol{G} \in \partial_{\boldsymbol{X}}(\lambda_1\|\boldsymbol{X}_0\|_*)\right\} \\ = &\sup\left\{\langle \mathcal{P}_{\mathcal{T}_0^{\perp}}(\boldsymbol{G}), \mathcal{P}_{\mathcal{T}_0^{\perp}}(\boldsymbol{\Delta}_{\boldsymbol{X}})\rangle - \langle \mathcal{P}_{\mathcal{T}_0^{\perp}}(\boldsymbol{P}^{\top}\boldsymbol{D}\boldsymbol{Q}), \boldsymbol{\Delta}_{\boldsymbol{X}}\rangle : \quad \boldsymbol{G} \in \partial_{\boldsymbol{X}}(\lambda_1\|\boldsymbol{X}_0\|_*)\right\} \\ \geqslant &\lambda_1\|\mathcal{P}_{\mathcal{T}_0^{\perp}}(\boldsymbol{\Delta}_{\boldsymbol{X}})\|_* - \|\mathcal{P}_{\mathcal{T}_0^{\perp}}(\boldsymbol{P}^{\top}\boldsymbol{D}\boldsymbol{Q})\|_{2\to 2}\|\mathcal{P}_{\mathcal{T}_0^{\perp}}(\boldsymbol{\Delta}_{\boldsymbol{X}})\|_* \\ \geqslant &\lambda_1(1-1/c)\|\mathcal{P}_{\mathcal{T}_0^{\perp}}(\boldsymbol{\Delta}_{\boldsymbol{X}})\|_*. \end{aligned} \tag{53}$$

Similarly,

$$\begin{aligned}
&\lambda_2 \|\boldsymbol{Y}_0 + \boldsymbol{\Delta}_{\boldsymbol{Y}}\|_* - \lambda_2 \|\boldsymbol{Y}_0\|_* - \langle \boldsymbol{D}, \boldsymbol{\Delta}_{\boldsymbol{Y}} \rangle \\
\geqslant & \sup\left\{ \langle \boldsymbol{G}, \boldsymbol{\Delta}_{\boldsymbol{Y}} \rangle - \langle \boldsymbol{D}, \boldsymbol{\Delta}_{\boldsymbol{Y}} \rangle : \quad \boldsymbol{G} \in \partial_{\boldsymbol{Y}} (\lambda_2 \|\boldsymbol{Y}_0\|_{\mathrm{vec}(1)}) \right\} \\
= & \sup\left\{ \left\langle \left\{ \mathcal{P}_{\mathcal{S}^\perp}(\boldsymbol{G}) - \mathcal{P}_{\mathcal{S}^\perp}(\boldsymbol{D}) \right\}, \ \boldsymbol{\Delta}_{\boldsymbol{Y}} \right\rangle : \quad \boldsymbol{G} \in \partial_{\boldsymbol{Y}} (\lambda_1 \|\boldsymbol{Y}_0\|_{\mathrm{vec}(1)}) \right\} \\
\geqslant & \lambda_2 \left\| \mathcal{P}_{\mathcal{S}^\perp}(\boldsymbol{\Delta}_{\boldsymbol{Y}}) \right\|_{\mathrm{vec}(1)} - \left\langle \mathcal{P}_{\mathcal{S}^\perp}(\boldsymbol{D}), \ \mathcal{P}_{\mathcal{S}^\perp}(\boldsymbol{\Delta}_{\boldsymbol{Y}}) \right\rangle \\
\geqslant & \lambda_2 \left\| \mathcal{P}_{\mathcal{S}^\perp}(\boldsymbol{\Delta}_{\boldsymbol{Y}}) \right\|_{\mathrm{vec}(1)} - \left\| \mathcal{P}_{\mathcal{S}^\perp}(\boldsymbol{D}) \right\|_{\mathrm{vec}(\infty)} \left\| \mathcal{P}_{\mathcal{S}^\perp}(\boldsymbol{\Delta}_{\boldsymbol{X}}) \right\|_{\mathrm{vec}(1)} \\
\geqslant & \lambda_2 (1 - 1/c) \left\| \mathcal{P}_{\mathcal{S}^\perp}(\boldsymbol{\Delta}_{\boldsymbol{Y}}) \right\|_{\mathrm{vec}(1)}.
\end{aligned} \tag{54}$$

This completes the proof of Proposition 16.

Proposition 17 *Suppose the conditions in Theorem 8 holds and let $\boldsymbol{D}_{\mathcal{S}}$ and $\boldsymbol{D}_{\mathcal{T}}$ be as in Theorem 8, then*

$$\lambda_1 \|\mathcal{P}_{\mathcal{T}_0^\perp}(\boldsymbol{\Delta}_{\boldsymbol{X}})\|_* + \lambda_2 \left\| \mathcal{P}_{\mathcal{S}^\perp}(\boldsymbol{\Delta}_{\boldsymbol{Y}}) \right\|_{vec(1)} \leqslant (1 - 1/c)^{-1} (1/2) \|\boldsymbol{D}_{\mathcal{T}} + \boldsymbol{D}_{\mathcal{S}}\|_2^2$$

Proof of Proposition 17. By the optimality of $(\widehat{\boldsymbol{X}}, \widehat{\boldsymbol{Y}})$, we have

$$\begin{aligned}
& \lambda_1 (\|\widehat{\boldsymbol{X}}\|_* - \|\boldsymbol{X}_0\|_*) + \lambda_2 (\|\widehat{\boldsymbol{Y}}\|_1 - \|\boldsymbol{Y}_0\|_1) \\
\leqslant \quad & -\frac{1}{2} \left\| \widehat{\boldsymbol{\Theta}} - \boldsymbol{\Theta} \right\|^2 + \langle \boldsymbol{E}, \boldsymbol{P} \boldsymbol{\Delta}_X \boldsymbol{Q}^\top \rangle + \langle \boldsymbol{E}, \boldsymbol{\Delta}_Y \rangle
\end{aligned} \tag{55}$$

Combining (53), (54) and (55), we have

$$\begin{aligned}
& \lambda_1 (1 - 1/c) \|\mathcal{P}_{\mathcal{T}_0^\perp}(\boldsymbol{\Delta}_{\boldsymbol{X}})\|_* + \lambda_2 (1 - 1/c) \left\| \mathcal{P}_{\mathcal{S}^\perp}(\boldsymbol{\Delta}_{\boldsymbol{Y}}) \right\|_{\mathrm{vec}(1)} \\
\leqslant \quad & -\langle \boldsymbol{P}^\top (\boldsymbol{D}_{\mathcal{T}} + \boldsymbol{D}_{\mathcal{S}}) \boldsymbol{Q}, \boldsymbol{\Delta}_{\boldsymbol{X}} \rangle - \langle \boldsymbol{D}_{\mathcal{T}} + \boldsymbol{D}_{\mathcal{S}}, \boldsymbol{\Delta}_{\boldsymbol{Y}} \rangle - \frac{1}{2} \|\widehat{\Theta} - \Theta\|_F^2 \\
= \quad & -\langle \boldsymbol{D}_{\mathcal{T}} + \boldsymbol{D}_{\mathcal{S}}, \boldsymbol{P} \boldsymbol{\Delta}_{\boldsymbol{X}} \boldsymbol{Q}^\top \rangle - \langle \boldsymbol{D}_{\mathcal{T}} + \boldsymbol{D}_{\mathcal{S}}, \boldsymbol{\Delta}_{\boldsymbol{Y}} \rangle - \frac{1}{2} \|\boldsymbol{P} \boldsymbol{\Delta}_{\boldsymbol{X}} \boldsymbol{Q}^\top + \boldsymbol{\Delta}_{\boldsymbol{Y}}\|_F^2 \\
\leqslant \quad & \frac{1}{2} \|\boldsymbol{D}_{\mathcal{T}} + \boldsymbol{D}_{\mathcal{S}}\|_2^2.
\end{aligned}$$

This completes the proof of Proposition 17.

Now we are ready to prove Theorem 5. By KKT condition,

$$\begin{aligned}
\lambda_1 \boldsymbol{G}_X &= \boldsymbol{P}^\top (\boldsymbol{Z} - \boldsymbol{P} \widehat{\boldsymbol{X}} \boldsymbol{Q}^\top - \widehat{\boldsymbol{Y}}) \boldsymbol{Q} \\
\lambda_2 \boldsymbol{G}_Y &= \boldsymbol{Z} - \boldsymbol{P} \widehat{\boldsymbol{X}} \boldsymbol{Q}^\top - \widehat{\boldsymbol{Y}}
\end{aligned} \tag{56}$$

By algebra,

$$\lambda_2 \boldsymbol{G}_Y = \boldsymbol{E} - \boldsymbol{P} \boldsymbol{\Delta}_{\boldsymbol{X}} \boldsymbol{Q}^\top - \boldsymbol{\Delta}_{\boldsymbol{Y}} \tag{57}$$

Left-multiply $\boldsymbol{P}^*$ and right-multiply $\boldsymbol{Q}^*$ on both sides,

$$\lambda_2 \boldsymbol{P}^* \boldsymbol{G}_Y \boldsymbol{Q}^* = \boldsymbol{P}^* \boldsymbol{E} \boldsymbol{Q}^* - \boldsymbol{P} \boldsymbol{\Delta}_{\boldsymbol{X}} \boldsymbol{Q}^\top - \boldsymbol{P}^* \boldsymbol{\Delta}_{\boldsymbol{Y}} \boldsymbol{Q}^* \tag{58}$$

Project both sides of (58) onto $\mathcal{S}$, we have

$$\lambda_2 \mathcal{P}_{\mathcal{S}}(\boldsymbol{P}^* \boldsymbol{G}_Y \boldsymbol{Q}^*) = \mathcal{P}_{\mathcal{S}}(\boldsymbol{P}^* \boldsymbol{E} \boldsymbol{Q}^*) - \mathcal{P}_{\mathcal{S}}(\boldsymbol{P} \boldsymbol{\Delta}_{\boldsymbol{X}} \boldsymbol{Q}^\top) - \mathcal{P}_{\mathcal{S}}(\boldsymbol{P}^* \boldsymbol{\Delta}_{\boldsymbol{Y}} \boldsymbol{Q}^*), \tag{59}$$

Similarly,

$$\lambda_1 \boldsymbol{G}_X \quad = \quad \boldsymbol{P}^\top \boldsymbol{E} \boldsymbol{Q} - \boldsymbol{P}^\top \boldsymbol{P} \boldsymbol{\Delta}_{\boldsymbol{X}} \boldsymbol{Q}^\top \boldsymbol{Q} - \boldsymbol{P}^\top \boldsymbol{\Delta}_{\boldsymbol{Y}} \boldsymbol{Q}. \tag{60}$$

Multiply above by $(\boldsymbol{P}^+)^\top$ and $\boldsymbol{Q}^+$ on left and right respectively,

$$\lambda_1 (\boldsymbol{P}^+)^\top \boldsymbol{G}_X \boldsymbol{Q}^+ = \boldsymbol{P}^* \boldsymbol{E} \boldsymbol{Q}^* - \boldsymbol{P} \boldsymbol{\Delta}_{\boldsymbol{X}} \boldsymbol{Q}^\top - \boldsymbol{P}^* \boldsymbol{\Delta}_{\boldsymbol{Y}} \boldsymbol{Q}^*. \tag{61}$$

Project onto $\mathcal{T}$ on both sides, we obtain

$$\begin{aligned} &\lambda_1 \mathcal{P}_{\mathcal{T}} \left((\boldsymbol{P}^+)^\top \boldsymbol{G}_X \boldsymbol{Q}^+ \right) \\ =&\mathcal{P}_{\mathcal{T}}(\boldsymbol{P}^* \boldsymbol{E} \boldsymbol{Q}^*) - \mathcal{P}_{\mathcal{T}}(\boldsymbol{P} \boldsymbol{\Delta}_{\boldsymbol{X}} \boldsymbol{Q}^\top) - \mathcal{P}_{\mathcal{T}}(\boldsymbol{P}^* \boldsymbol{\Delta}_{\boldsymbol{Y}} \boldsymbol{Q}^*) \end{aligned} \tag{62}$$

Further project onto $\mathcal{S}$ on both sides of (62),

$$\begin{aligned} &\lambda_1 (\mathcal{P}_{\mathcal{S}} \circ \mathcal{P}_{\mathcal{T}}) \left((\boldsymbol{P}^+)^\top \boldsymbol{G}_X \boldsymbol{Q}^+ \right) \\ =&(\mathcal{P}_{\mathcal{S}} \circ \mathcal{P}_{\mathcal{T}})(\boldsymbol{P}^* \boldsymbol{E} \boldsymbol{Q}^*) - (\mathcal{P}_{\mathcal{S}} \circ \mathcal{P}_{\mathcal{T}})(\boldsymbol{P} \boldsymbol{\Delta}_{\boldsymbol{X}} \boldsymbol{Q}^\top) \\ &- (\mathcal{P}_{\mathcal{S}} \circ \mathcal{P}_{\mathcal{T}})(\boldsymbol{P}^* \boldsymbol{\Delta}_{\boldsymbol{Y}} \boldsymbol{Q}^*) \end{aligned} \tag{63}$$

Subtracting (63) from (59) we have

$$\begin{aligned} &\mathcal{P}_{\mathcal{S}}(\boldsymbol{P}^* \boldsymbol{\Delta}_{\boldsymbol{Y}} \boldsymbol{Q}^*) - (\mathcal{P}_{\mathcal{S}} \circ \mathcal{P}_{\mathcal{T}})(\boldsymbol{P}^* \boldsymbol{\Delta}_{\boldsymbol{Y}} \boldsymbol{Q}^*) + (\mathcal{P}_{\mathcal{S}} \circ \mathcal{P}_{\mathcal{T}^\perp})(\boldsymbol{P} \boldsymbol{\Delta}_{\boldsymbol{X}} \boldsymbol{Q}^\top) \\ =&(\mathcal{P}_{\mathcal{S}} \circ \mathcal{P}_{\mathcal{T}^\perp})(\boldsymbol{P}^* \boldsymbol{E} \boldsymbol{Q}^*) - \lambda_2 \mathcal{P}_{\mathcal{S}}(\boldsymbol{P}^* \boldsymbol{G}_Y \boldsymbol{Q}^*) + \lambda_1 (\mathcal{P}_{\mathcal{S}} \circ \mathcal{P}_{\mathcal{T}}) \left((\boldsymbol{P}^+)^\top \boldsymbol{G}_X \boldsymbol{Q}^+ \right). \end{aligned} \tag{64}$$

As $\langle \operatorname{sgn}\left((\boldsymbol{P}^* \boldsymbol{\Delta}_Y \boldsymbol{Q}^*)_{\mathcal{S}} \right), \mathcal{P}_{\mathcal{S}}(\boldsymbol{P}^* \boldsymbol{\Delta}_Y \boldsymbol{Q}^*) \rangle = \| \mathcal{P}_{\mathcal{S}}(\boldsymbol{P}^* \boldsymbol{\Delta}_Y \boldsymbol{Q}^*) \|_{\mathrm{vec}(1)}$, take inner product on both sides of (64), we have

$$\begin{aligned} &\| \mathcal{P}_{\mathcal{S}}(\boldsymbol{P}^* \boldsymbol{\Delta}_{\boldsymbol{Y}} \boldsymbol{Q}^*) \|_{\mathrm{vec}(1)} \\ \leqslant&\| (\mathcal{P}_{\mathcal{S}} \circ \mathcal{P}_{\mathcal{T}})(\boldsymbol{P}^* \boldsymbol{\Delta}_{\boldsymbol{Y}} \boldsymbol{Q}^*) \|_{\mathrm{vec}(1)} + \| (\mathcal{P}_{\mathcal{S}} \circ \mathcal{P}_{\mathcal{T}^\perp})(\boldsymbol{P} \boldsymbol{\Delta}_{\boldsymbol{X}} \boldsymbol{Q}^\top) \|_{\mathrm{vec}(1)} \\ &+ \| (\mathcal{P}_{\mathcal{S}} \circ \mathcal{P}_{\mathcal{T}^\perp})(\boldsymbol{P}^* \boldsymbol{E} \boldsymbol{Q}^*) \|_{\mathrm{vec}(1)} + \lambda_2 \| \mathcal{P}_{\mathcal{S}}(\boldsymbol{P}^* \boldsymbol{G}_Y \boldsymbol{Q}^*) \|_{\mathrm{vec}(1)} \\ &+ \lambda_1 \| (\mathcal{P}_{\mathcal{S}} \circ \mathcal{P}_{\mathcal{T}}) \left((\boldsymbol{P}^+)^\top \boldsymbol{G}_X \boldsymbol{Q}^+ \right) \|_{\mathrm{vec}(1)} \\ \leqslant&\alpha(\rho) \beta(\rho) \| \boldsymbol{P}^* \boldsymbol{\Delta}_{\boldsymbol{Y}} \boldsymbol{Q}^* \|_{\mathrm{vec}(1)} \\ &+ \sqrt{s} \| \mathcal{P}_{\mathcal{T}^\perp}(\boldsymbol{P} \boldsymbol{\Delta}_{\boldsymbol{X}} \boldsymbol{Q}^\top) \|_{\mathrm{vec}(2)} + \| (\mathcal{P}_{\mathcal{S}} \circ \mathcal{P}_{\mathcal{T}^\perp})(\boldsymbol{P}^* \boldsymbol{E} \boldsymbol{Q}^*) \|_{\mathrm{vec}(1)} \\ &+ \lambda_2 s + \lambda_1 \sqrt{s} \| \mathcal{P}_{\mathcal{T}} \left((\boldsymbol{P}^+)^\top \boldsymbol{G}_X \boldsymbol{Q}^+ \right) \|_{\mathrm{vec}(2)} \\ \leqslant&\alpha(\rho) \beta(\rho) \| \mathcal{P}_{\mathcal{S}}(\boldsymbol{P}^* \boldsymbol{\Delta}_{\boldsymbol{Y}} \boldsymbol{Q}^*) \|_{\mathrm{vec}(1)} + \alpha(\rho) \beta(\rho) \| \mathcal{P}_{\mathcal{S}^\perp}(\boldsymbol{P}^* \boldsymbol{\Delta}_{\boldsymbol{Y}} \boldsymbol{Q}^*) \|_{\mathrm{vec}(1)} \\ &+ \sqrt{s} \| \mathcal{P}_{\mathcal{T}^\perp}(\boldsymbol{P} \boldsymbol{\Delta}_{\boldsymbol{X}} \boldsymbol{Q}^\top) \|_* + s \| \mathcal{P}_{\mathcal{T}^\perp}(\boldsymbol{P}^* \boldsymbol{E} \boldsymbol{Q}^*) \|_{\mathrm{vec}(\infty)} \\ &+ \lambda_2 s + 2 \sigma_{\min}^{-1}(\boldsymbol{P}) \sigma_{\min}^{-1}(\boldsymbol{Q}) \lambda_1 \sqrt{sr}, \end{aligned}$$

where in the last inequality, we used the Lemma 14 and the inequalities $\|\boldsymbol{P}^+\|_{2\to 2} \leqslant \sigma_{\min}^{-1}(\boldsymbol{P})$, $\|\boldsymbol{Q}^+\|_{2\to 2} \leqslant \sigma_{\min}^{-1}(\boldsymbol{Q})$. Rearrange above inequality, we obtain

$$\begin{aligned}
&(1-\alpha(\rho)\beta(\rho))\,\|\mathcal{P}_{\mathcal{S}}(\boldsymbol{P}^*\boldsymbol{\Delta}_{\boldsymbol{Y}}\boldsymbol{Q}^*)\|_{\mathrm{vec}(1)}\\
\leqslant&\alpha(\rho)\beta(\rho)\|\mathcal{P}_{\mathcal{S}^\perp}(\boldsymbol{P}^*\boldsymbol{\Delta}_{\boldsymbol{Y}}\boldsymbol{Q}^*)\|_{\mathrm{vec}(1)}+\sqrt{s}\|\mathcal{P}_{\mathcal{T}^\perp}(\boldsymbol{P}\boldsymbol{\Delta}_{\boldsymbol{X}}\boldsymbol{Q}^\top)\|_*\\
&+s\|\mathcal{P}_{\mathcal{T}^\perp}(\boldsymbol{P}^*\boldsymbol{E}\boldsymbol{Q}^*)\|_{\mathrm{vec}(\infty)}+\lambda_2 s+2\sigma_{\min}^{-1}(\boldsymbol{P})\sigma_{\min}^{-1}(\boldsymbol{Q})\lambda_1\sqrt{sr}. \qquad (65)
\end{aligned}$$

To bound $\alpha(\rho)\beta(\rho)\|\mathcal{P}_{\mathcal{S}^\perp}(\boldsymbol{P}^*\boldsymbol{\Delta}_{\boldsymbol{Y}}\boldsymbol{Q}^*)\|_{\mathrm{vec}(1)}+\sqrt{s}\|\mathcal{P}_{\mathcal{T}^\perp}(\boldsymbol{P}\boldsymbol{\Delta}_{\boldsymbol{X}}\boldsymbol{Q}^\top)\|_*$, we subtract (57) from (58), we have

$$\boldsymbol{\Delta}_{\boldsymbol{Y}}=\lambda_2\boldsymbol{P}^*\boldsymbol{G}_Y\boldsymbol{Q}^*-\lambda_2\boldsymbol{G}_Y+\boldsymbol{E}-\boldsymbol{P}^*\boldsymbol{E}\boldsymbol{Q}^*+\boldsymbol{P}^*\boldsymbol{\Delta}_{\boldsymbol{Y}}\boldsymbol{Q}^* \qquad (66)$$

Project both sides of (66) onto $\mathcal{S}$, we have

$$\begin{aligned}
\mathcal{P}_{\mathcal{S}}(\boldsymbol{\Delta}_{\boldsymbol{Y}})=&\lambda_2\mathcal{P}_{\mathcal{S}}\left(\boldsymbol{P}^*\boldsymbol{G}_Y\boldsymbol{Q}^*\right)-\lambda_2\mathcal{P}_{\mathcal{S}}\left(\boldsymbol{G}_Y\right)\\
&+\mathcal{P}_{\mathcal{S}}(\boldsymbol{E})-\mathcal{P}_{\mathcal{S}}(\boldsymbol{P}^*\boldsymbol{E}\boldsymbol{Q}^*)+\mathcal{P}_{\mathcal{S}}(\boldsymbol{P}^*\boldsymbol{\Delta}_{\boldsymbol{Y}}\boldsymbol{Q}^*). \qquad (67)
\end{aligned}$$

It then follows that

$$\begin{aligned}
&\|\mathcal{P}_{\mathcal{S}}(\boldsymbol{\Delta}_{\boldsymbol{Y}})\|_{\mathrm{vec}(1)}\\
\leqslant&\lambda_2\left\|\mathcal{P}_{\mathcal{S}}(\boldsymbol{P}^*\boldsymbol{G}_Y\boldsymbol{Q}^*)\right\|_{\mathrm{vec}(1)}+\lambda_2\left\|\mathcal{P}_{\mathcal{S}}(\boldsymbol{G}_Y)\right\|_{\mathrm{vec}(1)}+\left\|\mathcal{P}_{\mathcal{S}}(\boldsymbol{E})\right\|_{\mathrm{vec}(1)}\\
&+\left\|\mathcal{P}_{\mathcal{S}}(\boldsymbol{P}^*\boldsymbol{E}\boldsymbol{Q}^*)\right\|_{\mathrm{vec}(1)}+\left\|\mathcal{P}_{\mathcal{S}}(\boldsymbol{P}^*\boldsymbol{\Delta}_{\boldsymbol{Y}}\boldsymbol{Q}^*)\right\|_{\mathrm{vec}(1)}\\
\leqslant&2\lambda_2 s+s\left\|\boldsymbol{E}\right\|_{\mathrm{vec}(\infty)}+s\left\|\boldsymbol{P}^*\boldsymbol{E}\boldsymbol{Q}^*\right\|_{\mathrm{vec}(\infty)}+\left\|\mathcal{P}_{\mathcal{S}}(\boldsymbol{P}^*\boldsymbol{\Delta}_{\boldsymbol{Y}}\boldsymbol{Q}^*)\right\|_{\mathrm{vec}(1)}. \qquad (68)
\end{aligned}$$

On the other hand, by the definition of η_2, we have

$$\|\boldsymbol{P}^*\boldsymbol{\Delta}_{\boldsymbol{Y}}\boldsymbol{Q}^*\|_{\mathrm{vec}(1)}\leqslant\|\mathcal{P}_{\mathcal{S}}(\boldsymbol{\Delta}_{\boldsymbol{Y}})\|_{\mathrm{vec}(1)}+\eta_2^{-1}\|\mathcal{P}_{\mathcal{S}^\perp}(\boldsymbol{\Delta}_{\boldsymbol{Y}})\|_{\mathrm{vec}(1)}. \qquad (69)$$

Combine (68) and (69), it follows that

$$\begin{aligned}
\|\mathcal{P}_{\mathcal{S}^\perp}(\boldsymbol{P}^*\boldsymbol{\Delta}_{\boldsymbol{Y}}\boldsymbol{Q}^*)\|_{\mathrm{vec}(1)}\quad\leqslant\quad&\eta_2^{-1}\|\mathcal{P}_{\mathcal{S}^\perp}(\boldsymbol{\Delta}_{\boldsymbol{Y}})\|_{\mathrm{vec}(1)}+2\lambda_2 s\\
&+s\left\|\boldsymbol{E}\right\|_{\mathrm{vec}(\infty)}+s\left\|\boldsymbol{P}^*\boldsymbol{E}\boldsymbol{Q}^*\right\|_{\mathrm{vec}(\infty)}. \qquad (70)
\end{aligned}$$

Similarly, by the definition of η_1,

$$\|\mathcal{P}_{\mathcal{T}^\perp}(\boldsymbol{P}\boldsymbol{\Delta}_{\boldsymbol{X}}\boldsymbol{Q}^\top)\|_*\leqslant\eta_1^{-1}\|\mathcal{P}_{\mathcal{T}_0^\perp}(\boldsymbol{\Delta}_{\boldsymbol{X}})\|_*. \qquad (71)$$

Combine (70) and (71) and Proposition 17, we have

$$\begin{aligned}
&\alpha(\rho)\beta(\rho)\|\mathcal{P}_{\mathcal{T}^\perp}(\boldsymbol{P}\boldsymbol{\Delta}_{\boldsymbol{X}}\boldsymbol{Q}^\top)\|_*+\sqrt{s}\left\|\mathcal{P}_{\mathcal{S}^\perp}(\boldsymbol{P}^*\boldsymbol{\Delta}_{\boldsymbol{Y}}\boldsymbol{Q}^*)\right\|_{\mathrm{vec}(1)}\\
\leqslant&\left(\frac{\alpha(\rho)\beta(\rho)}{2\lambda_2\eta_2}\vee\frac{\sqrt{s}}{2\lambda_1\eta_1}\right)(1-1/c)^{-1}\|\boldsymbol{D}_{\mathcal{T}}+\boldsymbol{D}_{\mathcal{S}}\|_2^2\\
&+2\lambda_2 s+s\left\|\boldsymbol{E}\right\|_{\mathrm{vec}(\infty)}+s\left\|\boldsymbol{P}^*\boldsymbol{E}\boldsymbol{Q}^*\right\|_{\mathrm{vec}(\infty)}\\
\leqslant&[2(1-1/c)\eta_0\lambda_2]^{-1}\|\boldsymbol{D}_{\mathcal{T}}+\boldsymbol{D}_{\mathcal{S}}\|_2^2\\
&+2\lambda_2 s+s\left\|\boldsymbol{E}\right\|_{\mathrm{vec}(\infty)}+s\left\|\boldsymbol{P}^*\boldsymbol{E}\boldsymbol{Q}^*\right\|_{\mathrm{vec}(\infty)}. \qquad (72)
\end{aligned}$$

where we used the fact $\alpha(\rho)\beta(\rho) < 1$, $\alpha(\rho) \geqslant \sqrt{s}$ and $\lambda_1 \geqslant \lambda_2 \alpha(\rho)\sigma_{\max}(\boldsymbol{P})\sigma_{\max}(\boldsymbol{Q})$ by (26). Further combine (65) and (72), we have

$$\begin{aligned}
&(1-\alpha(\rho)\beta(\rho))\,\|\mathcal{P}_{\mathcal{S}}(\boldsymbol{P}^*\boldsymbol{\Delta}_{\boldsymbol{Y}}\boldsymbol{Q}^*)\|_{\mathrm{vec}(1)}\\
\leqslant&[2(1-1/c)\eta_0\lambda_2]^{-1}\|\boldsymbol{D}_{\mathcal{T}}+\boldsymbol{D}_{\mathcal{S}}\|_2^2+3\lambda_2 s+s\,\|\boldsymbol{E}\|_{\mathrm{vec}(\infty)}\\
&+s\,\|\boldsymbol{P}^*\boldsymbol{E}\boldsymbol{Q}^*\|_{\mathrm{vec}(\infty)}+s\|\mathcal{P}_{\mathcal{T}^\perp}(\boldsymbol{P}^*\boldsymbol{E}\boldsymbol{Q}^*)\|_{\mathrm{vec}(\infty)}\\
&+2\sigma_{\min}^{-1}(\boldsymbol{P})\sigma_{\min}^{-1}(\boldsymbol{Q})\lambda_1\sqrt{sr}.
\end{aligned}\tag{73}$$

It further follows from (73), (70) and and Proposition 17 that

$$\begin{aligned}
&(1-\alpha(\rho)\beta(\rho))\,\|\boldsymbol{P}^*\boldsymbol{\Delta}_{\boldsymbol{Y}}\boldsymbol{Q}^*\|_{\mathrm{vec}(1)}\\
\leqslant\;&(1-\alpha(\rho)\beta(\rho))\left(\|\mathcal{P}_{\mathcal{S}}(\boldsymbol{P}^*\boldsymbol{\Delta}_{\boldsymbol{Y}}\boldsymbol{Q}^*)\|_{\mathrm{vec}(1)}+\|\mathcal{P}_{\mathcal{S}^\perp}(\boldsymbol{P}^*\boldsymbol{\Delta}_{\boldsymbol{Y}}\boldsymbol{Q}^*)\|_{\mathrm{vec}(1)}\right)\\
\leqslant\;&[2\lambda_2(1-1/c)]^{-1}(\eta_0^{-1}+\eta_2^{-1})\|\boldsymbol{D}_{\mathcal{T}}+\boldsymbol{D}_{\mathcal{S}}\|_2^2\\
&+5\lambda_2 s+2s\,\|\boldsymbol{E}\|_{\mathrm{vec}(\infty)}+2s\,\|\boldsymbol{P}^*\boldsymbol{E}\boldsymbol{Q}^*\|_{\mathrm{vec}(\infty)}\\
&+s\|\mathcal{P}_{\mathcal{T}^\perp}(\boldsymbol{P}^*\boldsymbol{E}\boldsymbol{Q}^*)\|_{\mathrm{vec}(\infty)}+2\sigma_{\min}^{-1}(\boldsymbol{P})\sigma_{\min}^{-1}(\boldsymbol{Q})\lambda_1\sqrt{sr}.\\
\leqslant\;&[\lambda_2(1-1/c)\eta_0]^{-1}\|\boldsymbol{D}_{\mathcal{T}}+\boldsymbol{D}_{\mathcal{S}}\|_2^2\\
&+5\lambda_2 s+2s\epsilon_\infty+3s\epsilon'_\infty+2\sigma_{\min}^{-1}(\boldsymbol{P})\sigma_{\min}^{-1}(\boldsymbol{Q})\lambda_1\sqrt{sr}.
\end{aligned}\tag{74}$$

This proves (29). To prove (30), we note that by (68), (73) and Proposition 17,

$$\begin{aligned}
&(1-\alpha(\rho)\beta(\rho))\|\boldsymbol{\Delta}_{\boldsymbol{Y}}\|_{\mathrm{vec}(1)}\\
\leqslant&(1-\alpha(\rho)\beta(\rho))\|\mathcal{P}_{\mathcal{S}}(\boldsymbol{\Delta}_{\boldsymbol{Y}})\|_{\mathrm{vec}(1)}+\|\mathcal{P}_{\mathcal{S}^\perp}(\boldsymbol{\Delta}_{\boldsymbol{Y}})\|_{\mathrm{vec}(1)}\\
\leqslant&2\lambda_2 s+s\,\|\boldsymbol{E}\|_{\mathrm{vec}(\infty)}+s\,\|\boldsymbol{P}^*\boldsymbol{E}\boldsymbol{Q}^*\|_{\mathrm{vec}(\infty)}\\
&+(1-\alpha(\rho)\beta(\rho))\,\|\mathcal{P}_{\mathcal{S}}(\boldsymbol{P}^*\boldsymbol{\Delta}_{\boldsymbol{Y}}\boldsymbol{Q}^*)\|_{\mathrm{vec}(1)}+[2(1-1/c)\lambda_2]^{-1}\|\boldsymbol{D}_{\mathcal{T}}+\boldsymbol{D}_{\mathcal{S}}\|_2^2\\
\leqslant&5\lambda_2 s+2s\,\|\boldsymbol{E}\|_{\mathrm{vec}(\infty)}+2s\,\|\boldsymbol{P}^*\boldsymbol{E}\boldsymbol{Q}^*\|_{\mathrm{vec}(\infty)}+s\|\mathcal{P}_{\mathcal{T}^\perp}(\boldsymbol{P}^*\boldsymbol{E}\boldsymbol{Q}^*)\|_{\mathrm{vec}(\infty)}\\
&+2\sigma_{\min}^{-1}(\boldsymbol{P})\sigma_{\min}^{-1}(\boldsymbol{Q})\lambda_1\sqrt{sr}+[2\lambda_2(1-1/c)]^{-1}(\eta_0^{-1}+1)\|\boldsymbol{D}_{\mathcal{T}}+\boldsymbol{D}_{\mathcal{S}}\|_2^2\\
\leqslant&5\lambda_2 s+2s\,\|\boldsymbol{E}\|_{\mathrm{vec}(\infty)}+3s\epsilon'_\infty+2\sigma_{\min}^{-1}(\boldsymbol{P})\sigma_{\min}^{-1}(\boldsymbol{Q})\lambda_1\sqrt{sr}\\
&+[2\lambda_2(1-1/c)]^{-1}(\eta_0^{-1}+1)\|\boldsymbol{D}_{\mathcal{T}}+\boldsymbol{D}_{\mathcal{S}}\|_2^2.
\end{aligned}\tag{75}$$

Finally, to prove (31), note that by (62),

$$\begin{aligned}
&\|\mathcal{P}_{\mathcal{T}}(\boldsymbol{P}\boldsymbol{\Delta}_{\boldsymbol{X}}\boldsymbol{Q}^\top)\|_*\\
\leqslant&\lambda_1\left\|\mathcal{P}_{\mathcal{T}}\left((\boldsymbol{P}^+)^\top\boldsymbol{G}_X\boldsymbol{Q}^+\right)\right\|_*+\|\mathcal{P}_{\mathcal{T}}(\boldsymbol{P}^*\boldsymbol{E}\boldsymbol{Q}^*)\|_*+\|\mathcal{P}_{\mathcal{T}}(\boldsymbol{P}^*\boldsymbol{\Delta}_{\boldsymbol{Y}}\boldsymbol{Q}^*)\|_*\\
\leqslant&2\sigma_{\min}^{-1}(\boldsymbol{P})\sigma_{\min}^{-1}(\boldsymbol{Q})\lambda_1 r+\epsilon_*+\sqrt{2r}\|\boldsymbol{P}^*\boldsymbol{\Delta}_{\boldsymbol{Y}}\boldsymbol{Q}^*\|_{\mathrm{vec}(2)}.
\end{aligned}\tag{76}$$

Then (31) follows from above.

Proof of Theorem 9.
By the feasibility of $\widehat{\boldsymbol{X}}$ and $\widehat{\boldsymbol{Y}}$, we have

$$\boldsymbol{P}\boldsymbol{\Delta}_X\boldsymbol{Q}^\top+\boldsymbol{\Delta}_Y=\boldsymbol{P}(\widehat{\boldsymbol{X}}-\boldsymbol{X}_0)\boldsymbol{Q}^\top+(\widehat{\boldsymbol{Y}}-\boldsymbol{Y}_0)=\widetilde{\boldsymbol{\Theta}}-\widetilde{\boldsymbol{\Theta}}=0\tag{77}$$

By Proposition 16 we have

$$\lambda_1 \|\boldsymbol{X}_0 + \boldsymbol{\Delta}_X\|_* - \lambda_1 \|\boldsymbol{X}_0\|_* - \langle \boldsymbol{D}, \boldsymbol{P}\boldsymbol{\Delta}_X\boldsymbol{Q}^\top \rangle \geqslant \lambda_1(1-1/c)\|\mathcal{P}_{\mathcal{T}_0^\perp}(\boldsymbol{\Delta}_X)\|_*,$$

and

$$\lambda_2 \|\boldsymbol{Y}_0 + \boldsymbol{\Delta}_Y\|_1 - \lambda_2 \|\boldsymbol{Y}_0\|_1 - \langle \boldsymbol{D}, \boldsymbol{\Delta}_Y \rangle \geqslant \lambda_2(1-1/c)\big\|\mathcal{P}_{\mathcal{S}^\perp}(\boldsymbol{\Delta}_Y)\big\|_{\mathrm{vec}(1)}.$$

Combine above two inequalities with (77) we have

$$\begin{array}{rl} & \lambda_1 \|\boldsymbol{X}_0 + \boldsymbol{\Delta}_X\|_* + \lambda_2 \|\boldsymbol{Y}_0 + \boldsymbol{\Delta}_Y\|_{\mathrm{vec}(1)} - \lambda_1 \|\boldsymbol{X}_0\|_* - \lambda_2 \|\boldsymbol{Y}_0\|_{\mathrm{vec}(1)} \\ \geqslant & \lambda_1(1-1/c)\|\mathcal{P}_{\mathcal{T}_0^\perp}(\boldsymbol{\Delta}_X)\|_* + \lambda_2(1-1/c)\big\|\mathcal{P}_{\mathcal{S}^\perp}(\boldsymbol{\Delta}_Y)\big\|_{\mathrm{vec}(1)} \end{array} \tag{78}$$

On the other hand, by the optimality of $\widehat{\boldsymbol{X}}$ and $\widehat{\boldsymbol{Y}}$, we have

$$\lambda_1 \|\boldsymbol{X}_0 + \boldsymbol{\Delta}_X\|_* + \lambda_2 \|\boldsymbol{Y}_0 - \boldsymbol{\Delta}_Y\|_* - \lambda_1 \|\boldsymbol{X}_0\|_* - \lambda_2 \|\boldsymbol{Y}_0\|_* \leqslant 0 \tag{79}$$

Combine (78) and (79) we have

$$\lambda_1 \|\mathcal{P}_{\mathcal{T}_0^\perp}(\boldsymbol{\Delta}_X)\|_* + \lambda_2 \big\|\mathcal{P}_{\mathcal{S}^\perp}(\boldsymbol{\Delta}_Y)\big\|_{\mathrm{vec}(1)} = 0.$$

As a consequence,

$$\|\mathcal{P}_{\mathcal{T}_0^\perp}(\boldsymbol{\Delta}_X)\|_* = 0, \tag{80}$$

and

$$\big\|\mathcal{P}_{\mathcal{S}^\perp}(\boldsymbol{\Delta}_Y)\big\|_{\mathrm{vec}(1)} = 0. \tag{81}$$

It further follows from (80) and Proposition 4 that

$$\|\mathcal{P}_{\mathcal{T}^\perp}(\boldsymbol{P}\boldsymbol{\Delta}_X\boldsymbol{Q}^\top)\|_* = 0 \tag{82}$$

On the other hand, due to (77),

$$\mathcal{P}_{\mathcal{T}}(\boldsymbol{P}\boldsymbol{\Delta}_X\boldsymbol{Q}^\top) + \mathcal{P}_{\mathcal{T}^\perp}(\boldsymbol{P}\boldsymbol{\Delta}_X\boldsymbol{Q}^\top) + \mathcal{P}_{\mathcal{S}}(\boldsymbol{\Delta}_Y) + \mathcal{P}_{\mathcal{S}^\perp}(\boldsymbol{\Delta}_Y) = 0$$

It is equivalently to

$$\mathcal{P}_{\mathcal{T}}(\boldsymbol{P}\boldsymbol{\Delta}_X\boldsymbol{Q}^\top) + \mathcal{P}_{\mathcal{S}}(\boldsymbol{\Delta}_Y) = -\mathcal{P}_{\mathcal{T}^\perp}(\boldsymbol{P}\boldsymbol{\Delta}_X\boldsymbol{Q}^\top) - \mathcal{P}_{\mathcal{S}^\perp}(\boldsymbol{\Delta}_Y)$$

Separately applying $\mathcal{P}_{\mathcal{T}}$ and $\mathcal{P}_{\mathcal{S}}$ on both sides gives

$$\begin{pmatrix} \mathcal{I} & \mathcal{P}_{\mathcal{T}} \\ \mathcal{P}_{\mathcal{S}} & \mathcal{I} \end{pmatrix} \begin{pmatrix} \mathcal{P}_{\mathcal{T}}(\boldsymbol{P}\boldsymbol{\Delta}_X\boldsymbol{Q}^\top) \\ \mathcal{P}_{\mathcal{S}}(\boldsymbol{\Delta}_Y) \end{pmatrix} = \begin{pmatrix} -(\mathcal{P}_{\mathcal{T}} \circ \mathcal{P}_{\mathcal{S}^\perp})(\boldsymbol{\Delta}_Y) \\ -(\mathcal{P}_{\mathcal{S}} \circ \mathcal{P}_{\mathcal{T}^\perp})(\boldsymbol{P}\boldsymbol{\Delta}_X\boldsymbol{Q}^\top) \end{pmatrix}$$

It further follows that

$$\mathcal{P}_{\mathcal{S}}(\boldsymbol{\Delta}_Y) = (\mathcal{I} - \mathcal{P}_{\mathcal{S}} \circ \mathcal{P}_{\mathcal{T}})^{-1} \left((\mathcal{P}_{\mathcal{S}} \circ \mathcal{P}_{\mathcal{T}} \circ \mathcal{P}_{\mathcal{S}^\perp})(\boldsymbol{\Delta}_Y) - (\mathcal{P}_{\mathcal{S}} \circ \mathcal{P}_{\mathcal{T}^\perp})(\boldsymbol{P}\boldsymbol{\Delta}_X\boldsymbol{Q}^\top) \right)$$

Then,

$$
\begin{aligned}
&\|\mathcal{P}_{\mathcal{S}}(\boldsymbol{\Delta}_Y)\|_{\mathrm{vec}(1)} \\
&\leqslant \frac{1}{1-\alpha(\rho)\beta(\rho)}\left(\|(\mathcal{P}_{\mathcal{S}}\circ\mathcal{P}_{\mathcal{T}}\circ\mathcal{P}_{\mathcal{S}^\perp})(\boldsymbol{\Delta}_Y)\|_{\mathrm{vec}(1)}+\left\|(\mathcal{P}_{\mathcal{S}}\circ\mathcal{P}_{\mathcal{T}^\perp})(\boldsymbol{P}\boldsymbol{\Delta}_X\boldsymbol{Q}^\top)\right\|_{\mathrm{vec}(1)}\right) \\
&\leqslant \frac{1}{1-\alpha(\rho)\beta(\rho)}\left(\|(\mathcal{P}_{\mathcal{S}}\circ\mathcal{P}_{\mathcal{T}}\circ\mathcal{P}_{\mathcal{S}^\perp})(\boldsymbol{\Delta}_Y)\|_{\mathrm{vec}(1)}+\sqrt{s}\left\|\mathcal{P}_{\mathcal{T}^\perp}(\boldsymbol{P}\boldsymbol{\Delta}_X\boldsymbol{Q}^\top)\right\|_{\mathrm{vec}(2)}\right) \\
&\leqslant \frac{1}{1-\alpha(\rho)\beta(\rho)}\left(\|(\mathcal{P}_{\mathcal{S}}\circ\mathcal{P}_{\mathcal{T}}\circ\mathcal{P}_{\mathcal{S}^\perp})(\boldsymbol{\Delta}_Y)\|_{\mathrm{vec}(1)}+\sqrt{s}\left\|\mathcal{P}_{\mathcal{T}^\perp}(\boldsymbol{P}\boldsymbol{\Delta}_X\boldsymbol{Q}^\top)\right\|_{*}\right) \\
&\leqslant \frac{1}{1-\alpha(\rho)\beta(\rho)}\left(\alpha(\rho)\beta(\rho)\|\mathcal{P}_{\mathcal{S}^\perp}(\boldsymbol{\Delta}_Y)\|_{\mathrm{vec}(1)}+\sqrt{s}\left\|\mathcal{P}_{\mathcal{T}^\perp}(\boldsymbol{P}\boldsymbol{\Delta}_X\boldsymbol{Q}^\top)\right\|_{*}\right) \\
&= 0, \qquad (83)
\end{aligned}
$$

where the last equality holds due to (81) and (82). Combining (81) and (83), we have

$$\|\boldsymbol{\Delta}_Y\|_{\mathrm{vec}(1)} = \|\mathcal{P}_{\mathcal{S}}(\boldsymbol{\Delta}_Y)\|_{\mathrm{vec}(1)} + \|\mathcal{P}_{\mathcal{S}^\perp}(\boldsymbol{\Delta}_Y)\|_{\mathrm{vec}(1)} = 0$$

As a consequence, we have

$$\boldsymbol{\Delta}_Y = \mathbf{0}.$$

Finally,

$$\boldsymbol{\Delta}_X = (\boldsymbol{P}^\top\boldsymbol{P})^{-1}\boldsymbol{P}^\top(\boldsymbol{P}\boldsymbol{X}\boldsymbol{Q}^\top)\boldsymbol{Q}(\boldsymbol{Q}^\top\boldsymbol{Q})^{-1} = \mathbf{0}.$$

This completes the proof.