Journal of Industrial and Management Optimization

https://www.aimspress.com/journal/jimo

JIMO, 22(1): 24–43.
DOI: 10.3934/jimo.2026002
Received: 11 August 2025
Revised: 12 October 2025
Accepted: 20 October 2025
Published: 13 November 2025

Research article

A proximal alternative directional method of multipliers for a class of low rank matrix recovery problems

Qihan Zhang and Xueting Cui*

School of Mathematics, Shanghai University of Finance and Economics, Shanghai 200433, China

* **Correspondence:** Email: cui.xueting@shufe.edu.cn.

Abstract: This paper considers the recovery problem of a low-rank matrix which arises in many areas, such as the Netflix ranking matrix recovery problem. We introduce the difference between the minimum nuclear norm and Ky Fan K-norm to approximate the rank of the target matrix and incorporate it into a linear least squares problem. As the resulting optimization problem is nonconvex and nonsmooth, albeit with some special structures, we propose a specially devised alternative directional method of multipliers (ADMM) for finding approximate or local optimal solutions. The efficiency of the proposed method is mainly due to the two subproblems at each iteration having explicit optimal solutions. A computational experiment is conducted to demonstrate the performance of the ADMM method by comparing with the existing methods. Computational results show that the ADMM method has better performance in terms of computing cost for most test instances.

Keywords: low rank matrix recovery; sparse optimization; ADMM; nuclear norm; Ky Fan K-norm
Mathematics Subject Classification: 90C26, 90C30

1. Introduction

Low-rank matrix recovery and completion have recently drawn growing interest from the research community in diverse areas of science and engineering, such as machine learning ([1, 2]), control [3] and recommendation systems [4], and biomedical signal processing [5]. One of the most important applications of rank minimization is to recover a low-rank matrix based on incomplete observations of its entries. Here are two real-world scenarios in which one would aim to reconstruct a low-rank matrix based on a subset of its entries.

Netflix problem:

Recommendation systems are designed to suggest items that users may find interesting or enjoyable, based on their preferences. They play a vital role in the success of many e-commerce platforms such as Amazon, MusicStrands, Pandora, Yahoo!, and Netflix. In the case of Netflix, a

movie rental and streaming service, its recommendation system, Cinematch, analyzes over 1.9 billion ratings from 11.7 million users to generate hundreds of millions of personalized suggestions daily. This helps improve user satisfaction and retention. However, because users typically rate only a small number of movies, the resulting user-rating matrix is often sparse. Despite this sparsity, it is commonly assumed that the matrix is approximately low-rank, as individual preferences are influenced by only a few underlying factors. Matrix completion techniques are therefore used to predict missing ratings and enhance recommendation accuracy (see [6–8]).

Structure-from-motion problem: Structure-from-motion (SfM) is the problem of recovering the 3D structure of a scene and the camera motion parameters from a sequence of images. Its core idea is to utilize the coordinates of feature points across multiple image frames, and apply matrix factorization techniques (e.g., singular value decomposition or SVD) to decompose the measurement matrix into shape and motion components, thereby reconstructing the object's 3D structure and the camera's motion. Ideally, the measurement matrix of feature points should be of low rank (for example, it is theoretically rank 4 under an affine camera model). However, due to occlusions, tracking failures, or other factors, the measurement matrix often contains a large amount of missing data in practical applications. Therefore, SfM can be viewed as a low-rank matrix completion problem, that is, accurately recovering the full low-rank matrix in the presence of noise and missing data, in order to improve the robustness and accuracy of structure and motion estimation (see [9–11]).

Let $\boldsymbol{M} \in \mathbb{R}^{m\times n}$ denote the matrix we aim to recover as accurately as possible. However, the only information available about $\boldsymbol{M}$ is a subset of its observed entries, namely M_{ij} for $\{i, j\} \in \Omega$, where Ω is a subset of the complete set of entries $[m] \times [n]$. (Here and in what follows, $[n]$ denotes the list $\{1, \cdots, n\}$). In this case, the matrix recovery problem can be formulated as follows:

$$\begin{aligned} \min \quad & \operatorname{rank}(\boldsymbol{X}), \\ \text{s.t.} \quad & X_{ij} = M_{ij}, (i, j) \in \Omega, \end{aligned} \tag{1.1}$$

where $\boldsymbol{X}$ is the decision variable and $\operatorname{rank}(\boldsymbol{X})$ is the rank of the matrix $\boldsymbol{X}$.

Unfortunately, the formulation (1.1) has limited practical applicability, as the optimization problem is NP-hard, and its exact solution methods have huge computational complexity that is doubly exponential in the matrix size (see [12]). If a matrix has rank r, then it has exactly r nonzero singular values, and the rank counts the number of nonzero singular values. Motivated by this observation, a popular alternative of low rank is to minimize the sum of the singular values, i.e., the nuclear norm (see [13]). This breakthrough result has immediately prompted many researchers to investigate efficient numerical algorithms for the nuclear-norm-based convex relaxation model. Fazel et al. [14] formulated the nuclear norm minimization problem as a tractable SDP problem which can be solved efficiently. We refer the reader to [15] for progress on interior-point methods concerning some special nuclear-norm-minimization problems. Candes and Recht [16] proved that most square matrices $\boldsymbol{X}$ of rank r can be perfectly recovered by solving the optimization problem of minimizing the nuclear norm with the constraints $X_{ij} = M_{ij}, (i, j) \in \Omega$, provided that $|\Omega|$, the number of observed samples, satisfies some rules. Chen et al. [17] showed that the alternating directional method of multipliers (ADMM) is a simple and effective approach for solving the nuclear norm minimization model.

By using a linear mapping $\mathcal{A}$ from $\mathbb{R}^{m\times n}$ to $\mathbb{R}^p$, defined by $\mathcal{A}(\boldsymbol{X}) = (\langle \boldsymbol{A}_1, \boldsymbol{X}\rangle, \langle \boldsymbol{A}_2, \boldsymbol{X}\rangle,$
$..., \langle \boldsymbol{A}_p, \boldsymbol{X}\rangle)^T$ with $\boldsymbol{A}_i \in \mathbb{R}^{m\times n}, i = 1, \cdots, p$, and $b \in \mathbb{R}^p$, we have a more general formulation for

low-rank matrix recovery:

$$\begin{aligned} \min_{\boldsymbol{X}} \quad & \operatorname{rank}(\boldsymbol{X}), \\ \text{s.t.} \quad & \mathcal{A}(\boldsymbol{X}) = b. \end{aligned}$$

Since the available observations are often polluted by noise, the following linear least squares problem with low-rank regularization are widely studied.

$$\min_{\boldsymbol{X}} \quad \frac{1}{2}\|\mathcal{A}(\boldsymbol{X}) - b\|_2^2 + \rho \operatorname{rank}(\boldsymbol{X}), \tag{1.2}$$

where $\|\mathcal{A}(\boldsymbol{X}) - b\|_2^2$ aims to measure the loss and $\rho > 0$ is a parameter.

This model presents two primary computational challenges. The first lies in the data fidelity term $\|\mathcal{A}(\boldsymbol{X}) - b\|_2^2$. While suitable for Gaussian noise, the squared L2-norm is highly sensitive to outliers or gross errors, which are data points that deviate significantly from the general pattern and are often caused by measurement failures or data corruption. This limitation has spurred the development of robust low-rank matrix completion. A major branch of this research focuses on replacing the L2-norm with more robust loss functions. For instance, Wang et al. [18] proposed using a non-convex truncated quadratic loss function within a matrix factorization framework to resist gross errors. This approach aims to design loss functions that are insensitive to large magnitude residuals. Wang and So [19] devised a general framework to generate a class of robust functions, such as the hybrid ordinary-Cauchy (HOC) and hybrid ordinary-ℓ_p (HOP) functions, which are designed to selectively down-weigh only the data points identified as outliers. Alternatively, some methods achieve robustness by explicitly identifying and removing outliers during recovery (See [20]).

The second challenge is the non-convex and combinatorial nature of the $\operatorname{rank}(\boldsymbol{X})$ term, which makes problem (1.2) NP-hard. In this paper, we will focus on this second challenge within the standard least squares framework. A seminal approach to make the problem tractable is to replace the rank function with its tightest convex surrogate, the nuclear norm $\|\boldsymbol{X}\|_*$. This leads to the well-known convex relaxation:

$$\min_{\boldsymbol{X}} \quad \frac{1}{2}\|\mathcal{A}(\boldsymbol{X}) - b\|_2^2 + \rho\|\boldsymbol{X}\|_*,$$

where $\|\boldsymbol{X}\|_*$ is the nuclear norm. In some sense, this is the tightest convex relaxation of the NP-hard-rank minimization problem (1.2) since the nuclear ball $\{\boldsymbol{X} : \|\boldsymbol{X}\|_* \leq 1\}$ is the convex hull of the set of rank-one matrices with spectral norm bounded by one. In the past two decades, the nuclear norm regularization method gained great progress in theory and numerical computation, but the produced solutions often deviate from the true low-rank matirx due to the remarkable difference between the nuclear norm and the rank function. This inspired some researchers to seek nonconvex surrogates of the rank function, such as the truncated nuclear norm ([21], [22]), the truncated frobenius norm [23], the weighted nuclear norm [24], and the schatten-p norm [25], and employ the associated nonconvex surrogate models to design efficient algorithms. Extensive numerical results by [26] demonstrate the superiority of nonconvex surrogates.

In addition to these optimization-based approaches with explicit regularizers, a distinctly different research direction has emerged in recent years: deep learning techniques. For instance, Li et al. [27] proposed a novel interpretable neural network model for matrix completion, called the bi-branch

neural network (BiBNN). In this model, the row and column indices of each entry are treated as inputs to the BiBNN, and the training objective is to minimize the fitting error between all observed entries and their predicted values. Unknown entries are then estimated by feeding the coordinates of missing positions into the trained network. Although such deep learning models have achieved impressive performance, they typically require substantial amounts of training data and computational resources, and their theoretical properties remain largely opaque. In contrast, the model-based optimization methods focused on in this work offer greater transparency and stronger convergence guarantees.

In this paper, we mainly study the following low rank approximation problem:

$$(LRP) \quad \min_X \quad \frac{1}{2}\|\mathcal{A}(\boldsymbol{X}) - b\|_2^2 + \rho(\|\boldsymbol{X}\|_* - |||\boldsymbol{X}|||_K),$$

where $|||\boldsymbol{X}|||_K$ is the Ky Fan K-norm of matrix $\boldsymbol{X}$, defined as the sum of the K largest singular values of $\boldsymbol{X}$. Notice that the rank of the matrix is equal to the number of nonzero singular values. Obviously, it is not necessary to minimize all the singular values, and we could just focus on some small singular values and expect them equal to zero. The term $\|\boldsymbol{X}\|_* - |||\boldsymbol{X}|||_K$ is exactly the sum of these small singular values. Gotoh et al. [28] proved that $\text{rank}(\boldsymbol{X}) \leq K$ is equivalent to the condition $\|\boldsymbol{X}\|_* - |||\boldsymbol{X}|||_K = 0$. Based on the equivalence, we know that when the difference $\|\boldsymbol{X}\|_* - |||\boldsymbol{X}|||_K$ approaches zero, it indicates that the matrix's rank does not exceed K. Therefore, $\|\boldsymbol{X}\|_* - |||\boldsymbol{X}|||_K$ can serve as an effective means to approximate the matrix rank. It is worth noting that this regularization term is mathematically equivalent to the well-known truncated nuclear norm (TNN), which has been proven effective for low-rank matrix recovery in [22]. While mathematically equivalent to the truncated nuclear norm (TNN), we specifically adopt this difference-of-convex (DC) representation, namely the difference between the convex nuclear norm and the convex Ky Fan K-norm, due to its significant algorithmic advantages. This structure is the cornerstone of our proposed method.

Problem (*LRP*) has computational difficulty due to the nonconvexity and nonsmoothness of the term $\|\boldsymbol{X}\|_* - |||\boldsymbol{X}|||_K$. Gotoh et al. [28] proposed iterative algorithms based on the DC (difference of convex functions) structure. At each iteration, they construct a separable approximation to the objective function at the current point, and furthermore drive its analytical solution to the separable problem, which reduces the computational cost greatly.

In this paper, we try to exploit the proximal operator of $\|\boldsymbol{X}\|_* - |||\boldsymbol{X}|||_K$ and the application and extension of ADMM methods in problem (*LRP*). The main contribution of this paper is twofold. First, we derive analytical solutions for the proximal mapping of $\|\boldsymbol{X}\|_* - |||\boldsymbol{X}|||_K$. Second, we propose a fast algorithm under the framework of ADMM where the two subproblems at each iteration can be solved in analytical manners, and prove that the proposed method converges to a first-order stationary point of the problem given a mild condition. Furthermore, the analytical solutions for the proximal mapping of $\|\boldsymbol{X}\|_* - |||\boldsymbol{X}|||_K$ can be applied to other possible fast algorithms.

The rest of the paper is organized as follows. We first derive in Section 2 analytical solutions for the proximal mapping of $\|\boldsymbol{X}\|_* - |||\boldsymbol{X}|||_K$. The main technique in finding the analytic solutions is the unitary invariance of $\|\boldsymbol{X}\|_* - |||\boldsymbol{X}|||_K$. In Section 3, the fast algorithm under the framework of ADMM and its convergence is discussed. In Section 4, we conduct computational experiments to illustrate the improved performance of the proposed ADMM method. Finally, some concluding remarks are given in Section 5.

Notations: The Frobenius norm of a matrix $\boldsymbol{X} \in \mathbb{R}^{m\times n}$ is denoted by $\|\boldsymbol{X}\|_F := \sqrt{\sum_{i=1}^m \sum_{j=1}^n X_{ij}^2}$. The inner product of two matrices $\boldsymbol{X}, \boldsymbol{Y} \in \mathbb{R}^{m\times n}$ is defined as $\langle \boldsymbol{X}, \boldsymbol{Y}\rangle := \text{Trace}(\boldsymbol{X}^T\boldsymbol{Y})$. For a proper function

$f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$, the subdifferential at $x \in \mathrm{dom}(f)$ is denoted by $\partial f(x)$. The positive part of a real number $a \in \mathbb{R}$ is denoted by $(a)_+ := \max\{a, 0\}$, and is applied entrywise when extended to vectors or matrices. For a matrix $\boldsymbol{X}$, $\mathrm{diag}(\boldsymbol{X})$ denotes the vector formed by the diagonal entries of $\boldsymbol{X}$. For a vector $x \in \mathbb{R}^n$, we denote by $\mathrm{diag}(x)$ the diagonal matirx whose diagonal entries are the elements of x.

2. Proximal operator

In this section, we present a closed-form solution of the proximal operator for $||\boldsymbol{X}||_* - |||\boldsymbol{X}|||_K$. For convenience, we denote $r_K(\boldsymbol{X}) = ||\boldsymbol{X}||_* - |||\boldsymbol{X}|||_K$, and the proximal operator is defined as follows,

$$Prox_{\alpha r_K}(\bar{\boldsymbol{X}}) = \arg\min_{X}\{||\boldsymbol{X}||_* - |||\boldsymbol{X}|||_K + \frac{1}{2\alpha}||\boldsymbol{X} - \bar{\boldsymbol{X}}||_F^2\},$$

where $\alpha > 0$ is a positive parameter and $\bar{\boldsymbol{X}}$ is a given matrix. Proximal operator is particularly useful in convex optimization [29] and the design of fast algorithms [30]. The unitary invariance of function $r_K(\boldsymbol{X})$ is given in Lemma 1, while the property of solutions to problem $Prox_{\alpha r_K(\boldsymbol{X})}(\bar{\boldsymbol{X}})$ is characterized in Theorem 1.

Lemma 1. *For any matrix $\boldsymbol{X} \in \mathbb{R}^{m\times n}$ and any K satisfying $1 \le K \le \min\{m, n\}$, the nuclear norm $||\boldsymbol{X}||_*$ and Ky Fan K-norm $|||\boldsymbol{X}|||_K$ are unitarily invariant. Thus, $r_K(\boldsymbol{X}) = ||\boldsymbol{X}||_* - |||\boldsymbol{X}|||_K$ is unitarily invariant.*

Proof. We perform singular value decomposition on $\boldsymbol{X}$: $\boldsymbol{X} = \boldsymbol{U}_X\boldsymbol{\Sigma}_X\boldsymbol{V}_X^T$, where $\boldsymbol{U}_X \in \mathbb{R}^{m\times m}$ and $\boldsymbol{V}_X \in \mathbb{R}^{n\times n}$ are unitary matrices and $\boldsymbol{\Sigma}_X \in \mathbb{R}^{m\times n}$ is diagonal. The diagonal elements of the matrix Σ_X are singular values of the matrix $\boldsymbol{X}$.

For any unitary matrices $\boldsymbol{U}$ and $\boldsymbol{V}$, i.e. $\boldsymbol{U}\boldsymbol{U}^T = \boldsymbol{I}$ and $\boldsymbol{V}\boldsymbol{V}^T = \boldsymbol{I}$, where $\boldsymbol{I}$ is the identity matrix, let $\boldsymbol{A} = \boldsymbol{U}\boldsymbol{X}\boldsymbol{V}^T$, and then we have

$$\boldsymbol{A} = \boldsymbol{U}(\boldsymbol{U}_X\Sigma_X\boldsymbol{V}_X^T)\boldsymbol{V}^T = (\boldsymbol{U}\boldsymbol{U}_X)\Sigma_X(\boldsymbol{V}\boldsymbol{V}_X)^T.$$

Since the product of two unitary matrices is still unitary, the singular values of $\boldsymbol{A}$ and $\boldsymbol{X}$ are equal.

Notice that the nuclear norm $||\boldsymbol{X}||_*$ and the Ky Fan K-norm $|||\boldsymbol{X}|||_K$ are defined as the sum of all the singular values and the sum of the K-largest singular values of the matrix, respectively. We conclude that $||\boldsymbol{X}||_* = ||\boldsymbol{U}\boldsymbol{X}\boldsymbol{V}^T||_*$ and $|||\boldsymbol{X}|||_K = |||\boldsymbol{U}\boldsymbol{X}\boldsymbol{V}^T|||_K$ for any unitary matrices $\boldsymbol{U}$ and $\boldsymbol{V}$. Furthermore, $r_K(\boldsymbol{X}) = ||\boldsymbol{X}||_* - |||\boldsymbol{X}|||_K$ is unitarily invariant. □

Now we turn to the problem of deriving the proximal operator

$$Prox_{\alpha r_K}(\bar{\boldsymbol{X}}) = \arg\min_{X}\{||\boldsymbol{X}||_* - |||\boldsymbol{X}|||_K + \frac{1}{2\alpha}||\boldsymbol{X} - \bar{\boldsymbol{X}}||_F^2\}.$$

Perform singular value decomposition on $\bar{\boldsymbol{X}}$, i.e., $\bar{\boldsymbol{X}} = \bar{\boldsymbol{U}}\bar{\boldsymbol{\Sigma}}\bar{\boldsymbol{V}}^T$, where $\bar{\boldsymbol{U}}$ and $\bar{\boldsymbol{V}}$ are unitary and $\bar{\boldsymbol{\Sigma}}$ is diagonal with its diagonal elements being the singular values arranged in descending order. Let $\tilde{\boldsymbol{X}} = \bar{\boldsymbol{U}}^T\boldsymbol{X}\bar{\boldsymbol{V}}$. Due to the unitary invariance of norms in Lemma 1, we have that $||\boldsymbol{X}||_* = ||\tilde{\boldsymbol{X}}||_*$, $|||\boldsymbol{X}|||_K = |||\tilde{\boldsymbol{X}}|||_K$, and

$$||\boldsymbol{X} - \bar{\boldsymbol{X}}||_F^2 = ||\bar{\boldsymbol{U}}(\tilde{\boldsymbol{X}} - \bar{\boldsymbol{\Sigma}})\bar{\boldsymbol{V}}^T||_F^2 = ||\tilde{\boldsymbol{X}} - \bar{\boldsymbol{\Sigma}}||_F^2.$$

Therefore, we can obtain a solution $\boldsymbol{X}^* = \bar{\boldsymbol{U}}\tilde{\boldsymbol{X}}^*\bar{\boldsymbol{V}}^T$ of the proximal operator $Prox_{\alpha r_K}(\bar{\boldsymbol{X}})$ if $\tilde{\boldsymbol{X}}^*$ is an optimal solution to the problem

$$\min_{\tilde{\boldsymbol{X}}} \{||\tilde{\boldsymbol{X}}||_* - |||\tilde{\boldsymbol{X}}|||_K + \frac{1}{2\alpha}||\tilde{\boldsymbol{X}} - \bar{\boldsymbol{\Sigma}}||_F^2\}. \tag{2.1}$$

Lemma 2 shows a property of solutions to problem (2.1). We assume that $m \leq n$ without loss of generality in the rest of the paper.

Lemma 2. *Consider the proximal problem (2.1) where $\bar{\boldsymbol{\Sigma}} \in \mathbb{R}^{m\times n}$ is a diagonal matrix with all diagonal elements $\sigma_1, \cdots, \sigma_m$ nonnegative and $\sigma_1 \geq \cdots \geq \sigma_m$. Assume that problem (2.1) has optimal solutions and $\boldsymbol{X}^*$ is the set of all optimal solutions to problem (2.1). Then,*

(i) any solution $\tilde{\boldsymbol{X}}^* \in \boldsymbol{X}^*$ is diagonal;

(ii) the diagonal elements $\tilde{X}_1^*, \cdots, \tilde{X}_m^*$ of optimal solution $\tilde{\boldsymbol{X}}^*$ satisfy

$$\tilde{X}_1^* \geq \tilde{X}_2^* \geq \cdots \geq \tilde{X}_m^* \geq 0.$$

Proof. (i) We prove the conclusion by contradiction. Assume that $\tilde{\boldsymbol{M}}^* \in \mathcal{X}^*$ is not diagonal. Perform singular value decomposition, $\tilde{\boldsymbol{M}}^* = \tilde{\boldsymbol{U}}^*\tilde{\boldsymbol{X}}^*(\tilde{\boldsymbol{V}}^*)^T$, where $\tilde{\boldsymbol{U}}^*$ and $\tilde{\boldsymbol{V}}^*$ are unitary but not diagonal, and $\tilde{\boldsymbol{X}}^*$ is diagonal with diagonal elements $\tilde{X}_1^*, \cdots, \tilde{X}_m^*$. We will show that the matrix $\tilde{\boldsymbol{X}}^*$ has a smaller objective value, which contradicts that $\tilde{\boldsymbol{M}}^*$ is optimal.

Due to the unitary invariance of the nuclear norm and the Ky Fan K-norm, we have $||\tilde{\boldsymbol{M}}^*||_F^2 = ||\tilde{\boldsymbol{X}}^*||_F^2$ and $|||\tilde{\boldsymbol{M}}^*|||_K = |||\tilde{\boldsymbol{X}}^*|||_K$. Note that the third part in problem (2.1) concerning the Frobenius norm is

$$||\tilde{\boldsymbol{M}}^* - \bar{\boldsymbol{\Sigma}}||_F^2 = ||\tilde{\boldsymbol{X}}^* - (\tilde{\boldsymbol{U}}^*)^T\bar{\boldsymbol{\Sigma}}\tilde{\boldsymbol{V}}^*||_F^2.$$

Denote $\tilde{\boldsymbol{N}}^* = (\tilde{\boldsymbol{U}}^*)^T\bar{\boldsymbol{\Sigma}}\tilde{\boldsymbol{V}}^*$ for convenience, and then $\tilde{\boldsymbol{N}}^*$ has the same singular values with $\bar{\boldsymbol{\Sigma}}$. According to the definition of the Frobenius norm, we have

$$\begin{aligned}
&||\tilde{\boldsymbol{M}}^* - \bar{\boldsymbol{\Sigma}}||_F^2 \\
=&||\tilde{\boldsymbol{X}}^* - \tilde{\boldsymbol{N}}^*||_F^2 \\
=&\text{Trace}([\tilde{\boldsymbol{X}}^* - \tilde{\boldsymbol{N}}^*]^T[\tilde{\boldsymbol{X}}^* - \tilde{\boldsymbol{N}}^*]) \\
=&\text{Trace}((\tilde{\boldsymbol{X}}^*)^T\tilde{\boldsymbol{X}}^*) + \text{Trace}((\tilde{\boldsymbol{N}}^*)^T\tilde{\boldsymbol{N}}^*) - 2\text{Trace}((\tilde{\boldsymbol{N}}^*)^T\tilde{\boldsymbol{X}}^*) \\
\geq&\text{Trace}((\tilde{\boldsymbol{X}}^*)^T\tilde{\boldsymbol{X}}^*) + \text{Trace}(\bar{\boldsymbol{\Sigma}}^T\bar{\boldsymbol{\Sigma}}) - 2\sum_{i=1}^{m}\sigma_i X_i^* \\
=&\text{Trace}((\tilde{\boldsymbol{X}}^*)^T\tilde{\boldsymbol{X}}^*) + \text{Trace}(\bar{\boldsymbol{\Sigma}}^T\bar{\boldsymbol{\Sigma}}) - 2\text{Trace}(\tilde{\boldsymbol{X}}^*\bar{\boldsymbol{\Sigma}}) \\
=&||\tilde{\boldsymbol{X}}^* - \bar{\boldsymbol{\Sigma}}||_F^2,
\end{aligned}$$

where the inequality above holds due to the Von Neumann trace inequality theorem introduced by [31].

As $\bar{\boldsymbol{X}}^*$ is a diagonal matrix while $\tilde{\boldsymbol{N}}^*$ is not diagonal, it is evident that the equality in Von Neumann trace inequality does not hold for $\bar{\boldsymbol{X}}^*$ and $\tilde{\boldsymbol{N}}^*$. Thus, we have

$$||\tilde{\boldsymbol{M}}^* - \bar{\boldsymbol{\Sigma}}||_F^2 > ||\tilde{\boldsymbol{X}}^* - \bar{\boldsymbol{\Sigma}}||_F^2.$$

Obviously, the diagonal matrix $\tilde{\boldsymbol{X}}^*$ has a smaller objective value in problem (2.1), which contradicts that $\tilde{\boldsymbol{M}}^*$ is optimal. Thus, any optimal solution to problem (2.1) is diagonal.

(ii) Now take an optimal solution $\tilde{\boldsymbol{X}}^* \in \mathcal{X}^*$ which is diagonal, and we will prove that its diagonal elements $\tilde{X}_1^*, \cdots, \tilde{X}_m^*$ satisfy

$$\tilde{X}_1^* \geq \tilde{X}_2^* \geq \cdots \geq \tilde{X}_m^* \geq 0.$$

First, we prove that all $\tilde{X}_i^*$ are nonnegative. Assume that $\tilde{X}_i^* < 0$ for some i. Let $\hat{\boldsymbol{X}}^* = \text{diag}(\tilde{X}_1^*, \cdots, \tilde{X}_{i-1}^*, 0, \tilde{X}_{i+1}^*, \cdots, \tilde{X}_m^*)$. Then,

$$\|\tilde{\boldsymbol{X}}^*\|_* - |||\tilde{\boldsymbol{X}}^*|||_K \geq \|\hat{\boldsymbol{X}}^*\|_* - |||\hat{\boldsymbol{X}}^*|||_K,$$

and

$$\|\tilde{\boldsymbol{X}}^* - \bar{\boldsymbol{\Sigma}}\|_F^2 - \|\hat{\boldsymbol{X}}^* - \bar{\boldsymbol{\Sigma}}\|_F^2 = (\tilde{X}_i^* - \sigma_i)^2 - \sigma_i^2 = \tilde{X}_i^*(\tilde{X}_i^* - 2\sigma_i) > 0,$$

which contradicts the fact that $\tilde{\boldsymbol{X}}^*$ is an optimal solution. Therefore, all diagonal elements of $\tilde{\boldsymbol{X}}^*$ are nonnegative.

Next, we proceed to prove that $\tilde{X}_1^* \geq \tilde{X}_2^* \geq \cdots \geq \tilde{X}_m^*$. Assume that the elements $\tilde{X}_1^*, \tilde{X}_2^*, \cdots, \tilde{X}_m^*$ are not in descending order. Let $\dot{\boldsymbol{X}}^* = \text{diag}(\tilde{X}_{[1]}^*, \tilde{X}_{[2]}^*, \cdots, \tilde{X}_{[m]}^*)$ where $\tilde{X}_{[i]}^*$ is the ith largest element among all elements $\tilde{X}_i^*$s. Since the nuclear norm and Ky Fan K-norm are independent of the ordering of the diagonal elements, it is obvious that, $\|\tilde{\boldsymbol{X}}^*\|_* - |||\tilde{\boldsymbol{X}}^*|||_K = \|\dot{\boldsymbol{X}}^*\|_* - |||\dot{\boldsymbol{X}}^*|||_K$.

As the rearrangement inequality states, for two sequences of real numbers, the sum of their element-wise products is maximized only when both sequences are sorted in the same order. Thus, we have

$$\sigma_1\tilde{X}_{[1]}^* + \sigma_2\tilde{X}_{[2]}^* + \cdots + \sigma_m\tilde{X}_{[m]}^* \geq \sigma_1\tilde{X}_1^* + \sigma_2\tilde{X}_2^* + \cdots + \sigma_m\tilde{X}_m^*.$$

Since $\tilde{X}_1^*, \tilde{X}_2^*, \cdots, \tilde{X}_m^*$ are not sorted in descending order, the above inequality holds strictly. Now we examine the difference between $\tilde{\boldsymbol{X}}^*$ and $\dot{\boldsymbol{X}}^*$ with respect to the Frobenius norm term:

$$\begin{aligned}
\|\tilde{\boldsymbol{X}}^* - \bar{\boldsymbol{\Sigma}}\|_F^2 - \|\dot{\boldsymbol{X}}^* - \bar{\boldsymbol{\Sigma}}\|_F^2 &= \sum_{i=1}^m (\tilde{X}_i^* - \sigma_i)^2 - \sum_{i=1}^m (\tilde{X}_{[i]}^* - \sigma_i)^2 \\
&= \sum_{i=1}^m [(\tilde{X}_i^*)^2 - (\tilde{X}_{[i]}^*)^2 + \sigma_i^2 - \sigma_i^2 - 2(\tilde{X}_i^*\sigma_i - \tilde{X}_{[i]}^*\sigma_i)] \\
&= -2\sum_{i=1}^m (\tilde{X}_i^*\sigma_i - \tilde{X}_{[i]}^*\sigma_i) > 0.
\end{aligned}$$

Then, $\dot{\boldsymbol{X}}^*$ has a smaller objective value, which contradicts the fact that $\tilde{\boldsymbol{X}}^*$ is optimal. Thus, $\tilde{X}_1^* \geq \tilde{X}_2^* \geq \cdots \geq \tilde{X}_m^*$. □

Based on the special properties of optimal solutions shown in Lemma 2, here we derive a closed-form solution to the proximal mapping problem (2.1).

Theorem 1. *The proximal mapping problem (2.1) where $\bar{\boldsymbol{\Sigma}} \in R^{m\times n}$ is diagonal with all diagonal elements $\sigma_1, \cdots, \sigma_m$ nonnegative and $\sigma_1 \geq \cdots \geq \sigma_m$, is equivalent to the constrained problem:*

$$\min \quad \sum_{i=1}^K \frac{1}{2\alpha}(\tilde{X}_i - \sigma_i)^2 + \sum_{i=K+1}^m \{\tilde{X}_i + \frac{1}{2\alpha}(\tilde{X}_i - \sigma_i)^2\} \tag{2.2}$$

$$\text{s.t.} \quad \tilde{X}_1 \geq \tilde{X}_2 \geq \cdots \geq \tilde{X}_m \geq 0.$$

Furthermore, the optimal solution $\tilde{\boldsymbol{X}}^*$ *to problem (2.1) is of the form* $\tilde{\boldsymbol{X}}^* = diag(\tilde{X}_1^*, \cdots, \tilde{X}_m^*)$*, where*

$$\tilde{X}_i^* = \begin{cases} \sigma_i, & \text{if } 1 \leq i \leq K; \\ (\sigma_i - \alpha)_+, & \text{if } K+1 \leq i \leq m. \end{cases} \tag{2.3}$$

Proof. According to Lemma 2, the optimal solution $\tilde{\boldsymbol{X}}^*$ to problem (2.1) is diagonal with $\tilde{X}_1^* \geq \tilde{X}_2^* \geq \cdots \geq \tilde{X}_m^* \geq 0$. It is evident that problem (2.1) is equivalent to problem (2.2). Solving problem (2.2) with only nonnegative constraints can give a solution of the form (2.3). We can easily check that $\tilde{X}_i^*$s in (2.3) satisfy $\tilde{X}_1^* \geq \tilde{X}_2^* \geq \cdots \geq \tilde{X}_m^* \geq 0$. Thus, problem (2.1) has an optimal solution $\tilde{\boldsymbol{X}}^* = diag(\tilde{X}_1^*, \cdots, \tilde{X}_m^*)$, where $\tilde{X}_i^*$ is defined as (2.3). □

Based on Theorem 1, we obtain the proximal operator of $r_K(X) = \|\boldsymbol{X}\|_* - |||\boldsymbol{X}|||_K$ as

$$Prox_{\alpha r_K}(\bar{\boldsymbol{X}}) = \bar{\boldsymbol{U}}\tilde{\boldsymbol{X}}^*\bar{\boldsymbol{V}}^T,$$

where $\bar{\boldsymbol{U}}$ and $\bar{\boldsymbol{V}}$ are the left and right unitary matrices, $\bar{\boldsymbol{\Sigma}} = diag(\sigma_1, \cdots, \sigma_m)$ is the singular value matrix in the singular value decomposition of $\bar{\boldsymbol{X}}$, and $\tilde{\boldsymbol{X}}^* = diag(\tilde{X}_1^*, \cdots, \tilde{X}_m^*)$ with

$$\tilde{X}_i^* = \begin{cases} \sigma_i, & \text{if } 1 \leq i \leq K; \\ (\sigma_i - \alpha)_+, & \text{if } K+1 \leq i \leq m. \end{cases}$$

3. Alternative directional methods of multipliers

In this section, we focus on the low-rank matrix recovery problem and restate it as follows:

$$(LRP) \quad \min_X \quad \frac{1}{2}\|\mathcal{A}(\boldsymbol{X}) - b\|_2^2 + \rho(\|\boldsymbol{X}\|_* - |||\boldsymbol{X}|||_K),$$

where $\mathcal{A}$ is a linear mapping from $\mathbb{R}^{m\times n}$ to $\mathbb{R}^p$, defined by $\mathcal{A}(\boldsymbol{X}) = (\langle \boldsymbol{A}_1, \boldsymbol{X}\rangle, \langle \boldsymbol{A}_2, \boldsymbol{X}\rangle, ..., \langle \boldsymbol{A}_p, \boldsymbol{X}\rangle)^T$ with $\boldsymbol{A}_i \in \mathbb{R}^{m\times n}, i = 1, \cdots, p$, and $b \in \mathbb{R}^p$. To the best of our knowledge, the nuclear norm is usually used to approximate the matrix rank in the low-rank matrix optimization problem. The following example illustrates that the matrix generated by $\|\boldsymbol{X}\|_* - |||\boldsymbol{X}|||_K$ minimization will have lower rank than the matrix from the nuclear norm minimization.

Example 1. *Consider a* 5×5 *matrix problem shown in Figure 1. Assume that only partial elements are observed. The matrices recovered by nuclear norm minimization and* (*LRP*) *with* $K = 2$ *are shown in the upper right and lower right matrices in Figure 1, respectively. The matrix by nuclear norm minimization is of rank 4 while the matrix by* (*LRP*) *has rank 2. For convenience of presentation, all numerical values in the matrices have been rounded to two decimal places.*

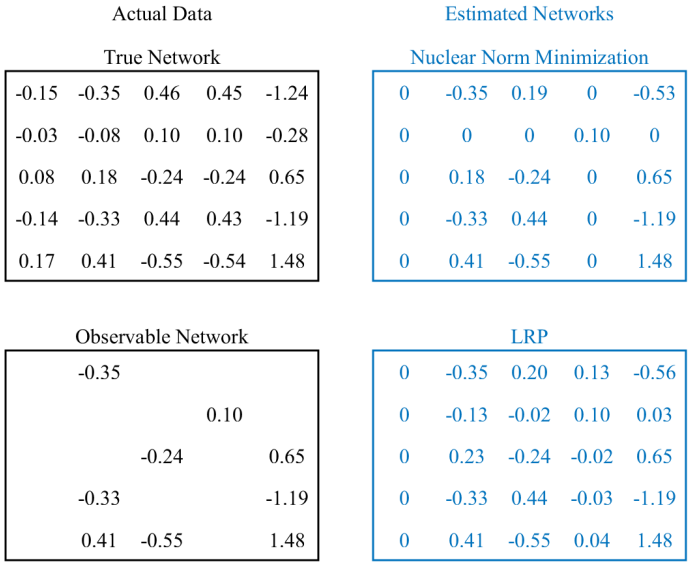

Figure 1. True matrix, observed entries, recovered matrices by nuclear norm minimization, and (*LRP*) in Example 1.

Now we proceed to develop the alternating directional method of multipliers to solve the problem (*LRP*). Alternative direction methods of multipliers have been successfully applied to convex programming and some nonconvex optimization problems arising from image processing and matrix optimization (see, e.g., [29, 32, 33]). The idea of the alternative direction methods of multipliers is to alternatively fix some variables in the augmented Lagrangian formulation of the constrained problem and solve the resulting more tractable subproblems at each iteration of the algorithm. It can also be viewed as a solution method that iterates along alternating directions and achieves global or local optimality eventually.

By observing the special structure and introducing coupling constraints $\boldsymbol{X} - \boldsymbol{Y} = 0$, problem (*LRP*) can then be rewritten as

$$\begin{aligned} \min \quad & \frac{1}{2}\|\mathcal{A}(\boldsymbol{Y}) - b\|_2^2 + \rho(\|\boldsymbol{X}\|_* - |||\boldsymbol{X}|||_K), \\ \text{s.t.} \quad & \boldsymbol{X} - \boldsymbol{Y} = \boldsymbol{0}, \\ & \boldsymbol{X}, \boldsymbol{Y} \in \mathbb{R}^{m\times n}. \end{aligned} \tag{3.1}$$

The augmented Lagrangian function for (3.1) is defined as

$$\mathcal{L}_\beta(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}) = \frac{1}{2}\|\mathcal{A}(\boldsymbol{Y}) - b\|_F^2 + \rho(\|\boldsymbol{X}\|_* - |||\boldsymbol{X}|||_K) + \langle \boldsymbol{Z}, \boldsymbol{X} - \boldsymbol{Y}\rangle + \frac{\beta}{2}\|\boldsymbol{X} - \boldsymbol{Y}\|_F^2$$

where $\beta > 0$ is the penalty parameter and $\boldsymbol{Z} \in \mathbb{R}^{m\times n}$ is the Lagrangian multiplier matrix for $\boldsymbol{X} - \boldsymbol{Y} = 0$. For given β and $\boldsymbol{Z}$, the augmented Lagrangian relaxation of (3.1) is

$$\min_{\boldsymbol{X},\boldsymbol{Y}} \mathcal{L}_\beta(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}), \tag{3.2}$$

which cannot be solved efficiently due to the nonconvexity.

A key observation is that if either $\boldsymbol{X}$ or $\boldsymbol{Y}$ is fixed in (3.2), then (3.2) can be reduced to a tractable subproblem with decision variable $\boldsymbol{Y}$ or $\boldsymbol{X}$. We can therefore apply the algorithmic framework of an alternative directional method of multilpliers to (3.1).

Suppose at the kth iteration, we have $\beta > 0$ and a tuple $(\boldsymbol{X}^k, \boldsymbol{Y}^k, \boldsymbol{Z}^k)$. We solve the following two subproblems alternatively:

$$\boldsymbol{X}^{k+1} = \arg\min \left\{ \mathcal{L}_\beta(\boldsymbol{X}, \boldsymbol{Y}^k, \boldsymbol{Z}^k) \right\}. \tag{3.3}$$

$$\boldsymbol{Y}^{k+1} = \arg\min \left\{ \mathcal{L}_\beta(\boldsymbol{X}^{k+1}, \boldsymbol{Y}, \boldsymbol{Z}^k) \right\}, \tag{3.4}$$

In the following, we show that the subproblem (3.3) can be solved efficiently. By expanding the inner product, ignoring constant terms and introducing an intermediate a variable $\bar{\boldsymbol{X}}^{k+1} = \boldsymbol{Y}^k - \frac{1}{\beta}\boldsymbol{Z}^k$, and we transform (3.3) equivalently into

$$\boldsymbol{X}^{k+1} = \arg\min_{\boldsymbol{X}} \|\boldsymbol{X}\|_* - |||\boldsymbol{X}|||_k + \frac{\beta}{2\rho}\|\boldsymbol{X} - \bar{\boldsymbol{X}}^{k+1}\|_F^2 \tag{3.5}$$

We perform a singular value decomposition on $\bar{\boldsymbol{X}}^{k+1}$,

$$\bar{\boldsymbol{X}}^{k+1} = \boldsymbol{U}^{k+1}\boldsymbol{\Sigma}^{k+1}(\boldsymbol{V}^{k+1})^T,$$

where $\boldsymbol{U}^{k+1} \in \mathbb{R}^{m\times m}$ and $\boldsymbol{V}^{k+1} \in \mathbb{R}^{n\times n}$ are unitary matrices and $\boldsymbol{\Sigma}^{k+1} = \mathrm{diag}(\sigma_1^{k+1}, \sigma_2^{k+1}, \cdots, \sigma_m^{k+1}) \in \mathbb{R}^{m\times n}$. According to Theorem 1, we have

$$\boldsymbol{X}^{k+1} = \boldsymbol{U}^{k+1}\tilde{\boldsymbol{X}}^{k+1}(\boldsymbol{V}^{k+1})^T,$$

where $\tilde{\boldsymbol{X}}^{k+1} = diag(\tilde{X}_1^{k+1}, \cdots, \tilde{X}_m^{k+1})$ with

$$\tilde{X}_i^{k+1} = \begin{cases} \sigma_i^{k+1}, & \text{if } 1 \le i \le K; \\ (\sigma_i^{k+1} - \frac{\rho}{\beta})_+, & \text{if } K+1 \le i \le m. \end{cases} \tag{3.6}$$

Note that $\mathcal{L}_\beta(\boldsymbol{X}^{k+1}, \boldsymbol{Y}, \boldsymbol{Z}^k)$ is a convex quadratic function of $\boldsymbol{Y}$ without any constraints. Then, problem (3.4) has a closed-form solution which is obtained by solving the proximal mapping problem

$$\boldsymbol{Y}^{k+1} = \arg\min_{\boldsymbol{Y}} \frac{1}{2}\|\mathcal{A}(\boldsymbol{Y}) - b\|_2^2 + \frac{\beta}{2}\|\boldsymbol{Y} - \bar{\boldsymbol{Y}}^{k+1}\|_F^2, \tag{3.7}$$

where $\bar{\boldsymbol{Y}}^{k+1} = \boldsymbol{X}^{k+1} + \frac{1}{\beta}\boldsymbol{Z}^k$.

The ADMM method for solving (*LRP*) is actually a procedure to solve subproblems (3.3) and (3.4) iteratively. Since subproblems (3.3) and (3.4) are proximal mapping problems and equivalent to (3.5) and (3.7), respectively, we name the proposed method in this paper a proximal alternating direction method of multipliers ($\mathrm{ADMM_P}$). We now formally describe the algorithm in the following.

The following Theorem 2 establishes the convergence property of the $\mathrm{ADMM_P}$ algorithm to a first-order stationary point of (*LRP*). For convenience, we denote $f(\boldsymbol{X}) = \frac{1}{2}\|\mathcal{A}(\boldsymbol{X}) - b\|_2^2$, which is a convex quadratic function of $\boldsymbol{X}$.

Algorithm 1: ADMM$_{\mathrm{P}}$ for sparse matrix optimization (*LRP*)

Step 0. Choose two accuracy tolerance parameters $\epsilon_1 > 0$ and $\epsilon_2 > 0$. Choose $\beta \geq 0$ and $\boldsymbol{Z}^0$. Choose $\boldsymbol{X}^0$ and $\boldsymbol{Y}^0$. Set $k := 0$.

Step 1. Update $\boldsymbol{X}^{k+1}$ by $\boldsymbol{X}^{k+1} = \boldsymbol{U}^{k+1}\tilde{\boldsymbol{X}}^{k+1}(\boldsymbol{V}^{k+1})^T$, where $\boldsymbol{U}^{k+1}, \boldsymbol{V}^{k+1}$ are the unitary matrices in the singular value decomposition of $\boldsymbol{Y}^k + \frac{1}{\beta}\boldsymbol{Z}^k$, and $\tilde{\boldsymbol{X}}^{k+1}$ is diagonal and defined as (3.6);

Step 2. Update $\boldsymbol{Y}^{k+1}$ by solving the problem (3.7).

Step 3. If $\|\boldsymbol{Y}^{k+1} - \boldsymbol{X}^{k+1}\|_F \leq \epsilon_1$ and $\|\boldsymbol{X}^{k+1} - \boldsymbol{X}^k\|_F / \|\boldsymbol{X}^k\|_F \leq \epsilon_2$, stop.

Step 4. Update the multiplier $\boldsymbol{Z}^{k+1}$ via:

$$\boldsymbol{Z}^{k+1} = \boldsymbol{Z}^k + \beta(\boldsymbol{X}^{k+1} - \boldsymbol{Y}^{k+1}).$$

Theorem 2. *Suppose that the objective function in* (*LRP*) *is coercive with respect to* $\boldsymbol{X}$*, that is,* $f(\boldsymbol{X}) + \rho(\|\boldsymbol{X}\|_* - \|\boldsymbol{X}\|_K) \to \infty$ *if* $\|\boldsymbol{X}\|_F^2 \to \infty$*. Let* $\bar{\lambda}$ *and* $\underline{\lambda}$ *be the maximum and minimum eigenvalues of the Hession matrix of function* $f(\boldsymbol{X}) = \frac{1}{2}\|\mathcal{A}(\boldsymbol{X}) - b\|_2^2$ *and* $\bar{\lambda} \geq \underline{\lambda} \geq 0$*. Assume that* $\beta > \max\{\bar{\lambda}, \frac{\bar{\lambda}}{\sqrt{\frac{\bar{\lambda}^2}{4} + \bar{\lambda}} + \frac{\bar{\lambda}}{2}}\}$*. Then,*

(i) the sequence $\{(\boldsymbol{X}^k, \boldsymbol{Y}^k, \boldsymbol{Z}^k)\}$ *generated by algorithm* (ADMM$_{\mathrm{P}}$) *is bounded and has at least one cluster point;*

(ii) for any cluster point $(\boldsymbol{X}^*, \boldsymbol{Y}^*, \boldsymbol{Z}^*)$ *of sequence* $\{(\boldsymbol{X}^k, \boldsymbol{Y}^k, \boldsymbol{Z}^k)\}$*, we have*

$$\lim_{k \leftarrow +\infty} \|\boldsymbol{X}^k - \boldsymbol{X}^*\|_F^2 + \|\boldsymbol{Y}^k - \boldsymbol{Y}^*\|_F^2 + \|\boldsymbol{Z}^k - \boldsymbol{Z}^*\|_F^2 = 0.$$

Furthermore, $\boldsymbol{X}^*$ *is a first-order stationary point such that*

$$0 \in \nabla f(\boldsymbol{X}^*) + \rho(\partial\|\boldsymbol{X}^*\|_* - \partial\|\boldsymbol{X}^*\|_K)$$

Proof. (i) Based on the optimality conditions of problems (3.3) and (3.4), the iterates $\{\boldsymbol{X}^k\}$ and $\{\boldsymbol{Y}^k\}$ satisfy

$$\begin{aligned} 0 &\in \rho\partial\|\boldsymbol{X}^{k+1}\|_* - \rho\partial\|\boldsymbol{X}^{k+1}\|_K + \boldsymbol{Z}^k + \beta(\boldsymbol{X}^{k+1} - \boldsymbol{Y}^k), \\ 0 &= \nabla f(\boldsymbol{Y}^{k+1}) - \boldsymbol{Z}^k - \beta(\boldsymbol{X}^{k+1} - \boldsymbol{Y}^{k+1}). \end{aligned}$$

From the definition of $\boldsymbol{Z}^{k+1}$, it holds that

$$\begin{aligned} 0 &\in \rho\partial\|\boldsymbol{X}^{k+1}\|_* - \rho\partial\|\boldsymbol{X}^{k+1}\|_K + \boldsymbol{Z}^{k+1} + \beta(\boldsymbol{Y}^{k+1} - \boldsymbol{Y}^k), && (3.8) \\ 0 &= \nabla f(\boldsymbol{Y}^{k+1}) - \boldsymbol{Z}^{k+1}, && (3.9) \\ \boldsymbol{X}^{k+1} &= \boldsymbol{Y}^{k+1} + \frac{1}{\beta}(\boldsymbol{Z}^{k+1} - \boldsymbol{Z}^k) \end{aligned}$$

Consequently, we have

$$\|\boldsymbol{X}^{k+1} - \boldsymbol{X}^k\|_F^2 = \|\boldsymbol{Y}^{k+1} + \frac{1}{\beta}(\boldsymbol{Z}^{k+1} - \boldsymbol{Z}^k) - \boldsymbol{Y}^k - \frac{1}{\beta}(\boldsymbol{Z}^k - \boldsymbol{Z}^{k-1})\|_F^2$$

$$\leq \|\boldsymbol{Y}^{k+1}-\boldsymbol{Y}^k\|_F^2+\frac{1}{\beta}\|\boldsymbol{Z}^{k+1}-\boldsymbol{Z}^k\|_F^2+\frac{1}{\beta}\|\boldsymbol{Z}^k-\boldsymbol{Z}^{k-1}\|_F^2, \tag{3.10}$$

and

$$\|\boldsymbol{Z}^{k+1}-\boldsymbol{Z}^k\|_F^2=\|\nabla f(\boldsymbol{Y}^{k+1})-\nabla f(\boldsymbol{Y}^k)\|_F^2\leq\bar{\lambda}\|\boldsymbol{Y}^{k+1}-\boldsymbol{Y}^k\|_F^2, \tag{3.11}$$

where the inequality holds due to the L-smoothness of the convex quadratic function.

According to the definition of $\boldsymbol{X}^{k+1}$ as the minimizer of $\mathcal{L}_\beta(\boldsymbol{X},\boldsymbol{Y}^k,\boldsymbol{Z}^k)$, we obtain

$$\mathcal{L}_\beta(\boldsymbol{X}^{k+1},\boldsymbol{Y}^k,\boldsymbol{Z}^k)-\mathcal{L}_\beta(\boldsymbol{X}^k,\boldsymbol{Y}^k,\boldsymbol{Z}^k)\leq 0. \tag{3.12}$$

Notice that $\mathcal{L}_\beta(\boldsymbol{X}^{k+1},\boldsymbol{Y},\boldsymbol{Z}^k)$ is quadratic with respect to $\boldsymbol{Y}$ with $\underline{\lambda}+\beta$ as its smallest eigenvalue, and $\boldsymbol{Y}^{k+1}$ is the minimizer of $\mathcal{L}_\beta(\boldsymbol{X}^{k+1},\boldsymbol{Y},\boldsymbol{Z}^k)$. We have

$$\mathcal{L}_\beta(\boldsymbol{X}^{k+1},\boldsymbol{Y}^k,\boldsymbol{Z}^k)-\mathcal{L}_\beta(\boldsymbol{X}^{k+1},\boldsymbol{Y}^{k+1},\boldsymbol{Z}^k) \geq (\underline{\lambda}+\beta)\|\boldsymbol{Y}^{k+1}-\boldsymbol{Y}^k\|_F^2. \tag{3.13}$$

Recall the definition of $\boldsymbol{Z}^{k+1}$, and we have

$$\begin{aligned}\mathcal{L}_\beta(\boldsymbol{X}^{k+1},\boldsymbol{Y}^{k+1},\boldsymbol{Z}^{k+1})-\mathcal{L}_\beta(\boldsymbol{X}^{k+1},\boldsymbol{Y}^{k+1},\boldsymbol{Z}^k) &= \langle\boldsymbol{Z}^{k+1}-\boldsymbol{Z}^k,\boldsymbol{X}^{k+1}-\boldsymbol{Y}^{k+1}\rangle\\ &= \frac{1}{\beta}\|\boldsymbol{Z}^{k+1}-\boldsymbol{Z}^k\|_F^2.\end{aligned} \tag{3.14}$$

Based on (3.11)-(3.14), we obtain that

$$\begin{aligned}\mathcal{L}_\beta(\boldsymbol{X}^{k+1},\boldsymbol{Y}^{k+1},\boldsymbol{Z}^{k+1})-\mathcal{L}_\beta(\boldsymbol{X}^k,\boldsymbol{Y}^k,\boldsymbol{Z}^k) &= \mathcal{L}_\beta(\boldsymbol{X}^{k+1},\boldsymbol{Y}^{k+1},\boldsymbol{Z}^{k+1})-\mathcal{L}_\beta(\boldsymbol{X}^{k+1},\boldsymbol{Y}^{k+1},\boldsymbol{Z}^k)\\ &+ \mathcal{L}_\beta(\boldsymbol{X}^{k+1},\boldsymbol{Y}^{k+1},\boldsymbol{Z}^k)-\mathcal{L}_\beta(\boldsymbol{X}^{k+1},\boldsymbol{Y}^k,\boldsymbol{Z}^k)\\ &+ \mathcal{L}_\beta(\boldsymbol{X}^{k+1},\boldsymbol{Y}^k,\boldsymbol{Z}^k)-\mathcal{L}_\beta(\boldsymbol{X}^k,\boldsymbol{Y}^k,\boldsymbol{Z}^k)\\ &\leq (\frac{\bar{\lambda}}{\beta}-\underline{\lambda}-\beta)\|\boldsymbol{Y}^{k+1}-\boldsymbol{Y}^k\|_F^2.\end{aligned} \tag{3.15}$$

Since $\beta>\frac{\bar{\lambda}}{\sqrt{\frac{\underline{\lambda}^2}{4}+\bar{\lambda}}+\frac{\underline{\lambda}}{2}}$, then $\frac{\bar{\lambda}}{\beta}-\underline{\lambda}-\beta<0$. Therefore, the sequence $\{\mathcal{L}_\beta(\boldsymbol{X}^k,\boldsymbol{Y}^k,\boldsymbol{Z}^k)\}$ is monotonically decreasing.

Note that $f(\boldsymbol{X})$ is L-smooth and $f(\boldsymbol{X})+\rho(\|\boldsymbol{X}\|_*-|||\boldsymbol{X}|||_K)$ is nonnegative. Together with (3.9) and $\beta>\bar{\lambda}$, we have

$$\begin{aligned}\mathcal{L}_\beta(\boldsymbol{X}^k,\boldsymbol{Y}^k,\boldsymbol{Z}^k) &= f(\boldsymbol{Y}^k)+\rho(\|\boldsymbol{X}^k\|_*-|||\boldsymbol{X}^k|||_K)+\langle\boldsymbol{Z}^k,\boldsymbol{X}^k-\boldsymbol{Y}^k\rangle+\frac{\beta}{2}\|\boldsymbol{X}^k-\boldsymbol{Y}^k\|_F^2\\ &= f(\boldsymbol{Y}^k)+\rho(\|\boldsymbol{X}^k\|_*-|||\boldsymbol{X}^k|||_K)+\langle\nabla f(\boldsymbol{Y}^k),\boldsymbol{X}^k-\boldsymbol{Y}^k\rangle+\frac{\beta}{2}\|\boldsymbol{X}^k-\boldsymbol{Y}^k\|_F^2\\ &\geq f(\boldsymbol{X}^k)+\rho(\|\boldsymbol{X}^k\|_*-|||\boldsymbol{X}^k|||_K)+\frac{\beta-\bar{\lambda}}{2}\|\boldsymbol{X}^k-\boldsymbol{Y}^k\|_F^2\\ &> -\infty.\end{aligned} \tag{3.16}$$

Thus, the sequence $\{\mathcal{L}_\beta(\boldsymbol{X}^k,\boldsymbol{Y}^k,\boldsymbol{Z}^k)\}$ is lower bounded.

Inequalities (3.15) and (3.16) show that $\{\mathcal{L}_\beta(\boldsymbol{X}^k,\boldsymbol{Y}^k,\boldsymbol{Z}^k)\}$, $\{f(\boldsymbol{X}^k)+\rho(\|\boldsymbol{X}^k\|_*-|||\boldsymbol{X}^k|||_K)\}$ and $\{\|\boldsymbol{X}^k-\boldsymbol{Y}^k\|_F^2\}$ are upper bounded by $\mathcal{L}_\beta(\boldsymbol{X}^0,\boldsymbol{Y}^0,\boldsymbol{Z}^0)$. Together with the coercivity of the function $f(\boldsymbol{X})+\rho(\|\boldsymbol{X}\|_*-$

$|||\boldsymbol{X}|||_K$), we have that the sequence $\{\boldsymbol{X}^k\}$ and then $\{\boldsymbol{Y}^k\}$ are bounded. The sequence $\{\boldsymbol{Z}^k\}$ is also bounded due to (3.11). By the Bolzano-Weierstrass theorem, the sequence $\{(\boldsymbol{X}^k, \boldsymbol{Y}^k, \boldsymbol{Z}^k)\}$ has at least one cluster point.

(ii) Summing (3.15) from $t = 1$ to $t = k$, we have

$$\mathcal{L}_\beta(\boldsymbol{X}^{k+1}, \boldsymbol{Y}^{k+1}, \boldsymbol{Z}^{k+1}) - \mathcal{L}_\beta(\boldsymbol{X}^1, \boldsymbol{Y}^1, \boldsymbol{Z}^1) \quad \leq \quad (\frac{\bar{\lambda}}{\beta} - \underline{\lambda} - \beta) \sum_{t=1}^{k} \|\boldsymbol{Y}^{t+1} - \boldsymbol{Y}^t\|_F^2.$$

Since $(\boldsymbol{X}^*, \boldsymbol{Y}^*, \boldsymbol{Z}^*)$ is a cluster point of sequence $\{(\boldsymbol{X}^k, \boldsymbol{Y}^k, \boldsymbol{Z}^k)\}$, there exists a convergent subsequence $\{(\boldsymbol{X}^{k_i}, \boldsymbol{Y}^{k_i}, \boldsymbol{Z}^{k_i})\}$ such that $\lim_{i\to\infty}(\boldsymbol{X}^{k_i}, \boldsymbol{Y}^{k_i}, \boldsymbol{Z}^{k_i}) = (\boldsymbol{X}^*, \boldsymbol{Y}^*, \boldsymbol{Z}^*)$. Note that $\mathcal{L}_\beta(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z})$ is continuous, and we have

$$\begin{aligned} & \mathcal{L}_\beta(\boldsymbol{X}^*, \boldsymbol{Y}^*, \boldsymbol{Z}^*) - \mathcal{L}_\beta(\boldsymbol{X}^1, \boldsymbol{Y}^1, \boldsymbol{Z}^1) \\ = & \lim_{i\to\infty} \mathcal{L}_\beta(\boldsymbol{X}^{k_i}, \boldsymbol{Y}^{k_i}, \boldsymbol{Z}^{k_i}) - \mathcal{L}_\beta(\boldsymbol{X}^1, \boldsymbol{Y}^1, \boldsymbol{Z}^1) \\ \leq & (\frac{\bar{\lambda}}{\beta} - \underline{\lambda} - \beta) \sum_{k=1}^{\infty} \|\boldsymbol{Y}^{k+1} - \boldsymbol{Y}^k\|_F^2. \end{aligned}$$

Thus, $\sum_{k=1}^{\infty} \|\boldsymbol{Y}^{k+1} - \boldsymbol{Y}^k\|_F^2 < +\infty$ and $\lim_{k\to\infty} \|\boldsymbol{Y}^{k+1} - \boldsymbol{Y}^k\|_F^2 = 0$. Furthermore, $\lim_{k\to\infty} \|\boldsymbol{X}^{k+1} - \boldsymbol{X}^k\|_F^2 = 0$ and $\lim_{k\to\infty} \|\boldsymbol{Z}^{k+1} - \boldsymbol{Z}^k\|_F^2 = 0$ hold implied by (3.10) and (3.11).

Taking the limit on the convergent subsequence $\{(\boldsymbol{X}^{k_i}, \boldsymbol{Y}^{k_i}, \boldsymbol{Z}^{k_i})\}$ in (3.8) and (3.9), we have

$$0 \quad \in \quad \nabla f(\boldsymbol{Y}^*) + \rho(\partial\|\boldsymbol{X}^*\|_* - \partial|||\boldsymbol{X}^*|||_K).$$

□

4. Numerical analysis

In this section, we present computational results of the proposed ADMM methods for solving problem (*LRP*). More specifically, we conduct the computational experiment on test instances of the following low rank matrix optimization problem:

$$(LRP_\Omega) \quad \min_{\boldsymbol{X}} \frac{1}{2}\|\boldsymbol{X}_\Omega - \boldsymbol{M}_\Omega\|_2^2 + \rho(\|\boldsymbol{X}\|_* - |||\boldsymbol{X}|||_K),$$

where Ω is a subset of the complete set of entries $[m] \times [n]$. In this section, we conduct several experiments on both synthetic and real data to demonstrate the effectiveness of our proposed matrix completion method. We compare the following two matrix completion approaches.

DCA: Consider that $\|\boldsymbol{X}\|_* - |||\boldsymbol{X}|||_K$ is DC-representable. We list a sequential DCA algorithm in the computational experiment. At the kth iteration, an approximate convex problem at the current point $\boldsymbol{X}^k$ will be solved to generate a new candidate solution $\boldsymbol{X}^{k+1}$. Here we use the following approximate subproblem (see [28]) at the kth iteration, which has closed-form solution,

$$\min_{\boldsymbol{X}} \frac{L}{2}\left\|\boldsymbol{X} - \left(\boldsymbol{X}^k - \frac{1}{L}(\boldsymbol{X}_\Omega - \boldsymbol{M}_\Omega - \rho\boldsymbol{B}^k)\right)\right\|_F^2 + \rho\|\boldsymbol{X}\|_*, \tag{4.1}$$

where $\boldsymbol{B}^k \in \partial|||\boldsymbol{X}^k|||_K$.

AIRNN: The accelerated iteratively reweighted nuclear norm (AIRNN) algorithm from [26] incorporates a Nesterov-style acceleration step by computing an extrapolated point $\boldsymbol{Y}^k$ and then solving a reweighted nuclear norm subproblem. In our experiment, the subproblem is,

$$\min_{X} \sum_{i=1}^{m} w_i \sigma_i(\boldsymbol{X}) + \frac{\mu}{2} \left\| \boldsymbol{X} - \left(\boldsymbol{Y}^k - \frac{1}{\mu} \nabla f(\boldsymbol{Y}^k) \right) \right\|_F^2,$$

where the weights w_i are zero for the largest K singular values and ρ otherwise.

4.1. Synthetic data

To build the test bed, we generate several groups of test instances for a variety of the matrix sizes m, n, ranks r, and numbers of observed entries p. We generate $\boldsymbol{M}$, an $m \times n$ matrix of rank r, by sampling two factors $\boldsymbol{M}_L \in \mathbb{R}^{m \times r}$ and $\boldsymbol{M}_R \in \mathbb{R}^{r \times n}$ with i.i.d. Gaussian entries and setting $\boldsymbol{M} = \boldsymbol{M}_L \boldsymbol{M}_R^T$. Note that an $m \times n$ matrix with rank r has $d_r := r(m + n - r)$ degrees of freedom. In this experiment, we change p/d_r, which determines the number of p, i.e., the number of the observed entries. The parameter ρ in (LRP_Ω) is set as $\|\boldsymbol{M}_\Omega\|_2/200$. The parameter L in the subproblem (4.1) of the DCA algorithm is set as 0.5.

In our implementation of ADMM algorithm, the initial matrices $\boldsymbol{X}^0$ and $\boldsymbol{Y}^0$ are both observed data matrices, filling unobserved entries with zeros. We also tested other initialization strategies, such as using a zero matrix or a random matrix. The numerical results indicated that the algorithm's performance is not sensitive to the choice of the initial point, as different strategies yielded comparable outcomes. Therefore, for the sake of logical coherence and intuitiveness, we opted to use the observed data matrix as the initial point for all experiments, as it better utilizes the available information. The initial matrices for all algorithms are set to the observed data matrix with unobserved entries filled with zeros. The value of β is initially chosen from the interval $[\frac{2}{\sqrt{mn}}, \frac{5}{\sqrt{mn}}]$, and is multiplied by 1.2 every five iterations. The ADMM algorithm is terminated if the stopping criteria $\|\boldsymbol{Y}^{k+1} - \boldsymbol{X}^{k+1}\|_F \leq 10^{-2}$ and $\frac{\|\boldsymbol{X}^{k+1} - \boldsymbol{X}^k\|_F}{\|\boldsymbol{X}^k\|_F} \leq 10^{-5}$ are satisfied. The termination tolerance of the DCA and AIRNN algorithms are set to 10^{-5}. The number of iterations of the four algorithms are limited to 500. The numerical tests were implemented in Matlab R2023a (64 bit) and run on a personal computer equipped with an AMD R7 7700 CPU (3.80GHz) and 32 GB RAM.

To evaluate the efficiency of the proposed method, we record the computing time and iterations when the test instances are solved by the ADMM_P, DCA, and AIRNN methods. We also record the relative deviation of the completed matrix from true matrix (Error) and sum of squared residuals (SSR) to measure the accuracy of the completed matrix, which are defined as

$$\text{Error} := \|\boldsymbol{X}^* - \boldsymbol{M}\|_F / \|\boldsymbol{M}\|_F, \ \text{SSR} := \|\boldsymbol{X}^*_\Omega - \boldsymbol{M}_\Omega\|_F. \tag{4.2}$$

Tables [1–3] present a detailed comparison of the performance of the ADMM_P, DCA, and AIRNN algorithms under different matrix sizes, where the first three columns list the parameters of the instances, "K" in the 4th column is the number in Ky Fan K-norm, "Time" is the CPU time (in seconds), "Iter." stands for the number of iterations, and "SSE" and "Error" measure the solution performance. All results are averaged for ten test instances. We mark the best-performing algorithm under different criteria in bold fonts. From these tables, it can be observed that for most of the test instances across varying scales, the ADMM_P algorithm demonstrates a superior and more balanced

performance, particularly in terms of computational efficiency. While DCA and AIRNN occasionally produce solutions with slightly lower SSR or Error, this marginal gain in accuracy comes at the cost of significantly more iterations and longer computing times. This efficiency gap becomes even more pronounced as the problem size increases. For the medium-scale problems in Table 2, ADMM_P consistently requires the fewest iterations and the shortest runtime, often by a large margin, while delivering highly competitive or superior accuracy. As for large-scale problems with $n = m = 1000$, ADMM_P can always generate matrices with smallest SSR and Error in the shortest time. These observations suggest that the ADMM_P algorithm often achieves more balanced performance, maintaining high solution accuracy while significantly reducing computational costs.

Table 1. Numerical results of DCA, AIRNN, and ADMM_P for instances with $n = m = 100$.

		DCA				AIRNN				ADMM_P			
Instance $(r, p, p/d_r)$	K	Iter	Time	SSR	Error	Iter	Time	SSR	Error	Iter	Time	SSR	Error
1, 1194, 6	60	6	**0.016**	3.166e-03	**9.418e-01**	12	0.029	3.456e-04	**9.418e-01**	41	0.044	**1.479e-05**	**9.418e-01**
	50	290	0.628	**4.086e-06**	**9.414e-01**	18	**0.039**	6.546e-04	**9.414e-01**	56	0.056	2.838e-05	**9.414e-01**
	40	265	0.554	**5.675e-06**	**9.399e-01**	22	**0.051**	1.245e-03	**9.399e-01**	76	0.075	8.530e-06	9.400e-01
	30	185	0.394	1.104e-05	9.363e-01	35	**0.078**	7.410e-04	**9.362e-01**	91	0.089	**9.407e-06**	9.364e-01
	20	72	0.157	1.741e-02	9.262e-01	50	**0.103**	3.270e-03	**9.238e-01**	111	0.114	**2.350e-05**	9.245e-01
	10	112	0.239	2.625e-02	8.928e-01	129	0.285	6.101e-03	**8.572e-01**	144	**0.147**	**1.412e-03**	8.617e-01
5, 3900, 4	65	501	2.056	**1.416e-04**	7.768e-01	30	0.126	1.453e-04	7.768e-01	51	**0.102**	4.119e-04	**7.767e-01**
	55	501	2.080	**1.551e-04**	7.683e-01	42	0.189	4.485e-04	7.682e-01	71	**0.142**	6.647e-04	**7.681e-01**
	45	435	1.863	**1.147e-04**	7.493e-01	56	0.228	2.007e-03	**7.490e-01**	104	**0.214**	1.856e-04	**7.490e-01**
	35	279	1.108	7.179e-02	7.032e-01	91	0.375	3.214e-03	**7.005e-01**	126	**0.245**	**2.693e-04**	7.029e-01
	25	230	0.949	4.087e-01	6.146e-01	225	1.008	**3.195e-02**	**5.457e-01**	151	**0.356**	4.040e-02	5.514e-01
	15	218	0.923	5.833e-01	4.667e-01	205	0.922	**6.445e-03**	3.465e-01	141	**0.319**	1.323e-02	**3.335e-01**
10, 5700, 3	70	498	3.548	**2.435e-04**	6.497e-01	28	0.177	3.104e-04	6.497e-01	51	**0.169**	6.150e-04	**6.484e-01**
	60	501	3.409	6.817e-04	6.350e-01	47	0.316	4.081e-04	6.349e-01	86	**0.279**	**3.677e-04**	**6.320e-01**
	50	501	3.362	7.331e-04	6.017e-01	69	0.482	7.881e-04	6.004e-01	110	**0.348**	**5.286e-04**	**5.919e-01**
	40	501	3.470	3.411e-01	5.095e-01	177	1.113	6.173e-03	4.979e-01	136	**0.422**	**5.785e-03**	**4.837e-01**
	30	366	2.590	1.234e+00	3.648e-01	179	1.161	**3.587e-02**	3.142e-01	156	**0.497**	7.151e-02	**2.909e-01**
	20	294	2.012	1.006e+00	2.113e-01	146	0.937	4.542e-03	1.837e-01	111	**0.364**	**1.127e-03**	**1.560e-01**

Table 2. Numerical results of DCA, AIRNN, and ADMM_P for instances with $n = m = 500$.

		DCA				AIRNN				ADMM_P			
Instance $(r, p, p/d_r)$	K	Iter	Time	SSR	Error	Iter	Time	SSR	Error	Iter	Time	SSR	Error
50, 142500, 3	110	250	49.769	1.699e+02	1.653e-01	412	88.462	3.180e-01	1.235e-01	103	**8.237**	**1.125e-02**	**9.937e-02**
	100	230	45.940	1.653e+02	1.412e-01	354	81.063	2.330e-01	1.079e-01	81	**7.018**	**1.921e-03**	**8.032e-02**
	90	210	42.046	1.529e+02	1.203e-01	302	68.543	1.649e-01	9.457e-02	66	**5.602**	**2.566e-03**	**5.935e-02**
	80	192	38.227	1.289e+02	9.792e-02	263	60.754	1.390e-01	7.532e-02	61	**5.190**	**2.323e-03**	**3.553e-02**
	70	174	35.059	1.020e+02	7.641e-02	161	36.666	4.085e-02	5.884e-02	66	**5.603**	**1.985e-03**	**1.000e-02**
	60	156	32.141	7.403e+01	5.258e-02	115	26.575	1.752e-02	3.743e-02	85	**7.337**	**2.501e-03**	**1.908e-05**
30, 116400, 4	90	202	39.446	1.057e+02	3.108e-01	376	83.831	4.174e-01	1.701e-01	153	**12.677**	**2.411e-01**	**1.392e-01**
	80	196	38.597	1.083e+02	2.719e-01	342	76.768	3.804e-01	1.452e-01	117	**9.606**	**1.528e-02**	**1.133e-01**
	70	190	38.017	1.098e+02	2.304e-01	305	67.583	2.944e-01	1.185e-01	96	**8.130**	**3.975e-03**	**8.819e-02**
	60	182	35.419	1.095e+02	1.909e-01	287	64.660	2.140e-01	9.490e-02	79	**6.610**	**3.491e-03**	**6.728e-02**
	50	172	34.477	1.066e+02	1.520e-01	221	49.961	1.334e-01	6.883e-02	81	**6.943**	**3.733e-03**	**4.215e-02**
	40	162	31.196	1.005e+02	1.079e-01	144	31.779	5.584e-02	4.057e-02	90	**7.167**	**2.858e-03**	**1.915e-02**
20, 117600, 6	80	160	27.773	9.958e+01	3.054e-01	303	68.620	3.716e-01	1.974e-01	156	**12.384**	**2.398e-01**	**1.687e-01**
	70	156	30.975	1.001e+02	2.676e-01	287	64.739	2.832e-01	1.724e-01	142	**11.968**	**7.740e-02**	**1.427e-01**
	60	150	29.408	9.875e+01	2.309e-01	262	59.084	2.291e-01	1.429e-01	105	**9.036**	**8.047e-03**	**1.182e-01**
	50	144	27.568	9.541e+01	1.921e-01	221	48.596	1.834e-01	1.154e-01	91	**7.130**	**2.936e-03**	**9.275e-02**
	40	136	26.404	8.896e+01	1.539e-01	176	38.844	1.118e-01	9.178e-02	86	**7.130**	**2.522e-03**	**7.062e-02**
	30	126	24.903	7.934e+01	1.152e-01	142	31.636	5.065e-02	6.066e-02	91	**7.553**	**2.300e-03**	**3.191e-02**

Table 3. Numerical results of DCA, AIRNN, and ADMM$_P$ for instances with $n = m = 1000$.

		DCA				AIRNN				ADMM$_P$			
Instance $(r, p, p/d_r)$	K	Iter	Time	SSR	Error	Iter	Time	SSR	Error	Iter	Time	SSR	Error
20, 237600, 6	80	120	57.352	1.726e+02	6.154e-01	501	252.668	8.463e-01	1.905e-01	161	**34.440**	**1.443e+00**	**1.500e-01**
	70	120	57.413	1.724e+02	6.003e-01	469	239.300	7.093e-01	1.624e-01	157	**33.948**	**8.409e-01**	**1.179e-01**
	60	120	57.238	1.715e+02	5.843e-01	425	216.339	6.165e-01	1.367e-01	148	**31.695**	**2.836e-01**	**9.067e-02**
	50	122	58.070	1.720e+02	5.632e-01	368	169.989	4.470e-01	1.074e-01	129	**27.685**	**1.472e-02**	**6.212e-02**
	40	122	51.475	1.694e+02	5.444e-01	310	137.145	2.866e-01	7.947e-02	122	**24.252**	**5.310e-03**	**3.844e-02**
	30	122	51.053	1.662e+02	5.241e-01	234	101.610	1.683e-01	4.277e-02	118	**22.839**	**5.234e-03**	**1.534e-02**
60, 465600, 4	120	172	89.743	9.205e+02	1.789e-01	501	277.187	5.350e-01	7.448e-02	78	**18.254**	**3.035e-03**	**3.535e-02**
	110	166	84.554	9.097e+02	1.632e-01	501	272.051	3.441e-01	6.379e-02	79	**17.972**	**2.877e-03**	**2.385e-02**
	100	162	83.031	8.964e+02	1.422e-01	485	278.161	1.718e-01	5.305e-02	81	**18.434**	**3.347e-03**	**1.513e-02**
	90	158	84.317	8.801e+02	1.206e-01	435	251.499	1.631e-01	4.058e-02	86	**20.456**	**3.972e-03**	**9.088e-03**
	80	152	75.401	8.490e+02	1.032e-01	395	201.474	1.105e-01	2.912e-02	96	**21.042**	**2.794e-03**	**1.151e-05**
	70	146	69.350	8.116e+02	8.607e-02	89	44.881	**1.747e-03**	2.086e-02	106	**22.711**	2.618e-03	**1.110e-05**
100, 570000, 3	160	184	61.030	1.200e+03	7.158e-02	501	170.421	4.050e-01	4.539e-02	59	**9.540**	**2.801e-03**	**5.026e-03**
	150	174	57.607	1.148e+03	6.359e-02	501	170.514	2.074e-01	3.962e-02	64	**10.319**	**2.753e-03**	**7.286e-06**
	140	166	55.011	1.041e+03	5.292e-02	501	171.035	1.742e-01	3.207e-02	71	**11.450**	**2.451e-03**	**7.296e-06**
	130	158	52.574	9.348e+02	4.190e-02	420	143.555	9.889e-02	2.298e-02	81	**12.954**	**2.370e-03**	**7.108e-06**
	120	162	53.923	1.500e+02	2.210e-02	368	122.304	7.885e-02	1.408e-02	95	**15.231**	**2.407e-03**	**6.631e-06**
	110	152	49.228	7.209e+01	1.198e-02	178	59.259	1.850e-02	6.390e-03	106	**16.650**	**2.542e-03**	**7.625e-06**

4.2. Real data

Futhermore, we implement ADMM$_P$, DCA, and AIRNN to solve the matrix completion problem arising in the well-known Jester joke data set, which contains 4.1 millions of ratings for 100 jokes from 73421 users. The data set is available on https://goldberg.berkeley.edu/jester-data/, and we refer to [34], [35], and [17], for the numerical study on this data set. In particular, the whole data set of the Jester joke problem consists of the following three categories:
(1) Jester-1: Data from 24,983 users who have rated 36 or more jokes;
(2) Jester-2: Data from 23,500 users who have rated 36 or more jokes;
(3) Jester-3: Data from 24,938 users who have rated between 15 and 35 jokes.

Since the number of users is much larger than the number of jokes, we randomly select m user ratings from the entire collection. In the numerical experiments, we set $m = 1000$, 2000, and 5000. For these experiments, all the parameters required in the algorithms are set the same as in the last subsection. Moreover, the computational environment is consistent with that used previously. The initial matrices for all algorithms are set to the observed data matrix with unobserved entries filled with zeros. The maximum number of iterations is limited to 500, and all reported results are the average of ten independent runs. Moreover, since some entries of $\boldsymbol{M}$ are not available, we are not able to compute the relative deviation of the completed matrix from true matrix (Error) defined by (4.2).

The results for the Jester dataset, presented in Table 4, further demonstrate the practical effectiveness of the proposed ADMM$_P$ algorithm. For this real-world matrix completion task, ADMM$_P$ consistently strikes a strong balance between computational efficiency and recovery accuracy. This positions ADMM$_p$ as a highly practical algorithm for real-world scenarios like recommendation systems, where achieving a strong level of accuracy rapidly is often more critical than attaining the ultimate precision at a higher computational expense.

Table 4. Comparison of experimental results of Jester-1, Jester-2, and Jester-3 across different methods.

		DCA			AIRNN			$ADMM_P$		
(m,n)	K	Iter	Time	SSR	Iter	Time	SSR	Iter	Time	SSR
Jester-1 Dataset										
(1000,100)	90	40	**1.205**	**4.068e-03**	501	8.191	4.057e-02	304	1.967	2.134e-02
(1000,100)	80	55	**1.833**	**6.449e-03**	497.6	8.491	5.998e-02	318	2.297	3.222e-02
(1000,100)	70	67	2.142	**8.767e-03**	501	7.582	8.206e-02	318	**1.943**	4.363e-02
(2000,100)	90	27	**13.274**	**3.466e-03**	501	15.578	4.072e-02	316	4.838	2.108e-02
(2000,100)	80	47	**22.209**	**5.803e-03**	501	15.205	6.124e-02	331	5.018	3.246e-02
(2000,100)	70	57	26.305	**8.435e-03**	501	16.178	8.428e-02	331	**4.999**	4.401e-02
(5000,100)	90	19	61.461	3.139e-03	458	17.332	3.932e-02	502	**0.786**	**2.242e-01**
(5000,100)	80	41	132.524	5.443e-03	482	18.173	5.962e-02	502	**0.358**	**5.860e-01**
(5000,100)	70	52	167.698	8.060e-03	501	19.153	8.249e-02	502	**0.025**	**8.564e-01**
Jester-2 Dataset										
(1000,100)	90	38	**1.179**	**4.258e-03**	481.6	8.639	4.314e-02	308	2.033	2.342e-02
(1000,100)	80	51	**1.793**	**6.892e-03**	481	8.858	6.361e-02	318	2.368	3.521e-02
(1000,100)	70	61	**2.032**	**9.482e-03**	501	8.830	8.641e-02	319	2.296	4.742e-02
(2000,100)	90	29	**14.665**	**3.556e-03**	501	15.965	4.042e-02	316	4.612	2.103e-02
(2000,100)	80	46	**24.840**	**5.966e-03**	501	17.792	6.037e-02	331	5.502	3.223e-02
(2000,100)	70	58	37.277	**8.331e-03**	501	17.829	8.382e-02	331	**5.906**	4.382e-02
(5000,100)	90	18	88.948	3.269e-03	467	20.635	4.027e-02	501.8	**0.742**	**2.290e-01**
(5000,100)	80	40	195.793	5.596e-03	484.6	21.299	6.023e-02	501.8	**0.373**	**6.050e-01**
(5000,100)	70	52	254.496	8.195e-03	501	22.056	8.371e-02	502	**0.031**	**8.367e-01**
Jester-3 Dataset										
(1000,100)	90	27	0.826	1.139e-04	30	0.398	3.243e-07	54.4	**0.347**	**4.966e-07**
(1000,100)	80	138	3.735	4.303e-04	46.4	**0.596**	7.538e-07	114.6	0.653	**5.745e-07**
(1000,100)	70	166	4.614	6.830e-04	48.2	0.664	1.454e-06	112.2	**0.661**	**1.097e-06**
(2000,100)	90	29	13.947	1.351e-04	30	**0.700**	1.591e-07	82	1.014	**1.209e-07**
(2000,100)	80	145	81.467	5.875e-04	44	**1.058**	**3.556e-07**	103	1.296	4.168e-07
(2000,100)	70	181	92.132	8.017e-04	50	**1.262**	**5.405e-07**	116	1.529	2.460e-07
(5000,100)	90	27	112.978	1.503e-04	37	**0.984**	**2.376e-07**	114.8	1.216	8.799e-07
(5000,100)	80	160	673.799	6.919e-04	45.2	**1.226**	**4.970e-07**	157.6	1.794	5.948e-07
(5000,100)	70	168	749.481	1.297e-03	50.2	**1.363**	**6.203e-07**	173.8	1.985	5.154e-07

Based on the numerical experiments we conducted on both simulated and real data, it can be observed that our proposed $ADMM_p$ algorithm has certain advantages. Especially for large-scale problems, our algorithm typically completes computation in a short time while achieving highly competitive SSR.

5. Conclusions

We have investigated in this paper a class of low-rank matrix recovery problems with $\|\boldsymbol{X}\|_* - \|\boldsymbol{X}\|_K$ regularization. Our main motivation is to develop an efficient solution methodology under the framework of ADMM for this nonconvex problem. By exploiting the special structure of the $\|\boldsymbol{X}\|_* - \|\boldsymbol{X}\|_K$ penalty, we have derived a closed-form proximal mapping of $\|\boldsymbol{X}\|_* - \|\boldsymbol{X}\|_K$ which promotes the solution efficiency of subproblems and the ADMM method. Numerical results reveal that the proposed ADMM methods are capable of finding good local solutions for large-scale problems and compare favorably with existing algorithms. Furthermore, the availability of a closed-form solution for the proximal mapping of $\|\boldsymbol{X}\|_* - \|\boldsymbol{X}\|_K$ enables its integration into a variety of efficient first-order optimization algorithms. For instance, it can be directly incorporated into the accelerated proximal gradient (APG) framework to accelerate convergence in nonconvex low-rank minimization. Moreover, due to the difference-of-convex (DC) structure of this regularizer, the proximal mapping serves as a key component in the difference of convex algorithms (DCA), where it is used iteratively to solve convex subproblems. These possibilities highlight the broader applicability of the derived proximal operator in designing scalable and effective algorithms for low-rank matrix recovery.

Author contributions

Qihan Zhang: Methodology; Writing – original draft; Software. Xueting Cui: Supervision; Writing – review & editing.

Use of Generative-AI tools declaration

The authors used Qwen (Tongyi Qianwen), a generative artificial intelligence tool developed by Alibaba Cloud, for language polishing to improve the fluency of the manuscript. All AI-assisted content was critically reviewed and revised by the authors, who take full responsibility for the final published version.

Acknowledgments

This research is supported partially by National Natural Science Foundation of China under Grant 72271250 and by the Joint NSFC/RGC grants under Grant 72461160315.

Conflict of interest

All authors declare no conflicts of interest in the paper.

References

1. J. Abernethy, F. Bach, T. Evgeniou, J. P. Vert, Low-rank matrix factorization with attributes, *arXiv preprint arXiv:cs/0611124*, 2006. https://doi.org/10.48550/arXiv.cs/0611124

2. Y. Amit, M. Fink, N. Srebro, S. Ullman, Uncovering shared structures in multiclass classification, *Proc. 24th Int. Conf. Mach. Learn.*, 2007. https://doi.org/10.1145/1273496.1273499
3. M. Mesbahi, G. P. Papavassilopoulos, On the rank minimization problem over a positive semidefinite linear matrix inequality, *IEEE Trans. Autom. Control*, **42** (1997), 239–243. https://doi.org/10.1109/9.554402
4. A. Ramlatchan, M. Y. Yang, Q. Liu, M. Li, J. X. Wang, Y. H. Li, A survey of matrix completion methods for recommendation systems, *Big Data Min. Anal.*, **1** (2018), 308–323. https://doi.org/10.26599/BDMA.2018.9020008
5. X. P. Li, Z. L. Shi, M. Dai, H. C. So, I. Frerichs, Z. Q. Zhao, Robust preprocessing of impulsive motion artifacts using low-rank matrix recovery for electrical impedance tomography, *IEEE Trans. Comput. Imaging*, **11** (2025), 942–954. https://doi.org/10.1109/TCI.2025.3587458
6. J. Bennett, S. Lanning, The netflix prize, *Proc. KDD Cup Workshop*, 2007.
7. J. D. M. Rennie, N. Srebro, Fast maximum margin matrix factorization for collaborative prediction, *Proc. 22nd Int. Conf. Mach. Learn.*, 2005, 713–719. https://doi.org/10.1145/1102351.1102441
8. N. Srebro, Learning with matrix factorizations, 2004.
9. P. Chen, D. Suter, Recovering the missing components in a large noisy low-rank matrix: Application to SFM, *IEEE Trans. Pattern Anal. Mach. Intell.*, **26** (2004), 1051–1063. https://doi.org/10.1109/TPAMI.2004.52
10. C. Tomasi, T. Kanade, Shape and motion from image streams under orthography: a factorization method, *Int. J. Comput. Vis.*, **9** (1992), 137–154. https://doi.org/10.1007/BF00129684
11. D. B. Zhang, Y. Hu, J. P. Ye, X. L. Li, X. F. He, Matrix completion by truncated nuclear norm regularization, *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, 2192–2199. https://doi.org/10.1109/cvpr.2012.6247927
12. A. L. Chistov, D. Y. Grigor'Ev, Complexity of quantifier elimination in the theory of algebraically closed fields, *LNCS*, **176** (1984), 17–31. https://doi.org/10.1007/bfb0030287
13. E. J. Candès, T. Tao, The power of convex relaxation: Near-optimal matrix completion, *IEEE Trans. Inf. Theory* , **56** (2010), 2053–2080. https://doi.org/10.1109/TIT.2010.2044061
14. M. Fazel, H. Hindi, S. P. Boyd, Log-det heuristic for matrix rank minimization with applications to hankel and Euclidean distance matrices, *Proc. Amer. Control Conf.*, 2003. https://doi.org/10.1109/acc.2003.1243393
15. Z. Liu, L. Vandenberghe, Interior-point method for nuclear norm approximation with application to system identification, *SIAM J. Matrix Anal. Appl.*, **31** (2010), 1235–1256. https://doi.org/10.1137/090755436
16. E. Candes, B. Recht, Exact matrix completion via convex optimization, *Commun. ACM*, **55** (2012), 111–119. https://doi.org/10.1145/2184319.2184343
17. C. H. Chen, B. S. He, X. M. Yuan, Matrix completion via an alternating direction method, *IMA J. Numer. Anal.*, **32** (2012), 227–245. https://doi.org/10.1093/imanum/drq039
18. Z. Y. Wang, X. P. Li, H. C. So, Robust matrix completion based on factorization and truncated-quadratic loss function, *IEEE Trans. Circuits Syst. Video Technol.*, **33** (2022), 1521–1534. https://doi.org/10.1109/TCSVT.2022.3214583
19. Z. Y. Wang, H. C. So, Robust matrix completion via novel M-estimator functions, *Proc. Int. Conf. Electr. Comput. Energy Technol.*, 2024, 1–5. https://doi.org/10.1109/icecet61485.2024.10698047

20. E. V. Laufer, B. Nadler, Rgnmr: A gauss-newton method for robust matrix completion with theoretical guarantees, *arXiv preprint arXiv: 2505.12919*, 2025.
21. T. Saeedi, M. Rezghi, A novel enriched version of truncated nuclear norm regularization for matrix completion of inexact observed data, *IEEE Trans. Knowl. Data Eng.*, **34** (2020), 519–530. https://doi.org/10.1109/tkde.2020.2983708
22. Y. Hu, D. B. Zhang, J. P. Ye, X. L. Li, X. F. He, Fast and accurate matrix completion via truncated nuclear norm regularization, *IEEE Trans. Pattern Anal. Mach. Intell.*, **35** (2012), 2117–2130. https://doi.org/10.1109/TPAMI.2012.271
23. H. L. Ye, H. Li, F. L. Cao, L. M. Zhang, A hybrid truncated norm regularization method for matrix completion, *IEEE Trans. Image Process.*, **28** (2019), 5171–5186. https://doi.org/10.1109/TIP.2019.2918733
24. C. Y. Lu, J. H. Tang, S. C. Yan, Z. C. Lin, Nonconvex nonsmooth low rank minimization via iteratively reweighted nuclear norm, *IEEE Trans. Image Process.*, **25** (2015), 829–839. https://doi.org/10.1109/TIP.2015.2511584
25. H. M. Zhang, J. J. Qian, B. Zhang, J. Yang, C. Gong, Y. Wei, Low-rank matrix recovery via modified schatten-p norm minimization with convergence guarantees, *IEEE Trans. Image Process.*, **29** (2019), 3132–3142. https://doi.org/10.1109/TIP.2019.2957925
26. D. N. Phan, T. N. Nguyen, An accelerated IRNN-Iteratively Reweighted Nuclear Norm algorithm for nonconvex nonsmooth low-rank minimization problems, *J. Comput. Appl. Math.*, **396** (2021), 113602. https://doi.org/10.1016/j.cam.2021.113602
27. X. P. Li, M. L. Wang, H. C. So, An interpretable bi-branch neural network for matrix completion, *Signal Process.*, **200** (2022), 108640. https://doi.org/10.1016/j.sigpro.2022.108640
28. J. Gotoh, A. Takeda, K. Tono, DC formulations and algorithms for sparse optimization problems, *Math. Program.*, **169** (2018), 141–176. https://doi.org/10.1007/s10107-017-1181-0
29. S. Boyd, L. Vandenberghe, *Convex optimization*, Cambridge University Press, 2004. https://doi.org/10.1017/CBO9780511804441
30. Y. F. Lou, M. Yan, Fast l1–l2 minimization via a proximal operator, *J. Sci. Comput.*, **74** (2018), 767–785. https://doi.org/10.1007/s10915-017-0463-2
31. J. V. Neumann, Some matrix-inequalities and metrization of matric space, 1937.
32. B. S. He, M. Tao, X. M. Yuan, Alternating direction method with Gaussian back substitution for separable convex programming, *Siam. J. Optimiz.*, **22** (2012), 313–340. https://doi.org/10.1137/110822347
33. X. T. Cui, X. L. Sun, S. S. Zhu, R. J. Jiang, D. Li, Portfolio optimization with nonparametric value-at-risk: A block coordinate descent method, *INFORMS J. Comput.*, **30** (2018), 454–471. https://doi.org/10.1287/ijoc.2017.0793
34. K. Goldberg, T. Roeder, D. Gupta, C. Perkins, Eigentaste: A constant time collaborative filtering algorithm, *Inf. Retr.*, **4** (2001), 133–151. https://doi.org/10.1023/A:1011419012209
35. S. Q. Ma, D. Goldfarb, L. F. Chen, Fixed point and bregman iterative methods for matrix rank minimization, *Math. Program.*, **128** (2011), 321–353. https://doi.org/10.1007/s10107-009-0306-5

AIMS Press