Mathematics

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

AIMS Mathematics, 10(8): 17602–17622.
DOI:10.3934/math.2025786
Received: 20 May 2025
Revised: 08 July 2025
Accepted: 22 July 2025
Published: 05 August 2025

Research article

Column $\ell_{2,0}$-norm regularized factorization model for inductive matrix completion

Ting Tao[1], **Lianghai Xiao**[2,*], **Zetian Chen**[1] **and Xijie Lin**[1]

[1] School of Mathematics, Foshan University, Foshan, China
[2] College of Information Science and Technology, Jinan University, Guangzhou, China

* **Correspondence:** Email: xiaolh@jnu.edu.cn.

Abstract: This paper is concerned with the model of column $\ell_{2,0}$-regularized factorization for inductive matrix completion. The column $\ell_{2,0}$-norm is an exact variational characterization of the rank function, promoting low-rank structure through column sparsity of the matrix. We established that the stationary points between the $\ell_{2,0}$-norm regularized factorization model and the rank regularized model are equivalent. Under a suitable assumption, an error bound to the true matrix was obtained for the stationary points with rank not more than that of the true matrix. For this nonconvex discontinuous optimization problem, we developed a proximal alternating minimization (PAM) algorithm with subspace correction and showed that the whole sequence globally converges to a stationary point of the rank regularized model. Numerical experiments conducted with examples of both synthetic and real data demonstrated the effectiveness of the proposed method.

Keywords: inductive matrix completion; column $\ell_{2,0}$-norm; low-rank factorization models; alternating minimization methods; subspace correction
Mathematics Subject Classification: 49J52, 49M27, 90C26

1. Introduction

The Low-rank matrix completion problem aims at recovering an unknown true matrix $M^* \in \mathbb{R}^{n\times m}$ with $\text{rank}(M^*) = r^*$ from observable entries in the index set $\Omega \subset \{1, \ldots, n\} \times \{1, \ldots, m\}$. In contrast, inductive matrix completion focuses on seeking a smaller matrix $X^* \in \mathbb{R}^{d_1\times d_2}$ by leveraging the additional side information that is summarized in the matrices $A \in \mathbb{R}^{n\times d_1}$ and $B \in \mathbb{R}^{m\times d_2}$ associated with a bilinear model $M^* = AX^*B^\top$. For convenience, in the sequel, we assume that $d_1 \leq d_2$ and $n \leq m$. If $\max\{d_1, d_2\} \ll \max\{n, m\}$, then the dimension of the recovery matrix is significantly reduced, improving the computational and storage efficiency. The side information matrices A and B can be viewed as feature representations, such as viewer preferences and movie properties in movie recommender systems [6,8], and also have wide applications in multi-label learning [20,22], disease prediction from gene/miRNA/lncRNA data [4,10,12], and link prediction in networks [5,11]. Generally, the inductive matrix completion problem can be casted into the following optimization convex problem:

$$\min_{X\in\mathbb{R}^{d_1\times d_2}} \|X\|_* \quad \text{s.t. } P_\Omega(AXB^\top) = P_\Omega(M^*), \tag{1.1}$$

where $P_\Omega(M^*)$ denotes the projection of M^* onto the set Ω, i.e., $[P_\Omega(M^*)]_{ij} = M^*_{ij}$ if $(i,j)\in\Omega$, and otherwise, $[P_\Omega(M^*)]_{ij} = 0$. The above model (1.1) follows the standard matrix completion, which has been shown to reduce the sample complexity to $O(\ln(n))$ [20]. When the observation has noise, i.e., $M = P_\Omega(M^* + \Xi)$ where Ξ is the noise matrix, one may consider the regularized model

$$\min_{X\in\mathbb{R}^{d_1\times d_2}} f(X) + \lambda\|X\|_*, \tag{1.2}$$

where $f(X) := \frac{1}{2}\|P_\Omega(AXB^\top - M)\|_F^2$ for any $X\in\mathbb{R}^{d_1\times d_2}$ and $\lambda > 0$ is the regularization parameter. While nuclear norm regularization provides strong recovery guarantees [5,20], most algorithms for (1.2), at each iteration, process full-matrix economic singular value decomposition (ESVD), which is the major computational difficulty and limits the solver's scalability. Inspired by the Burer–Monteiro factorization method [2], we replace X with $UV^\top$ for $(U,V)\in\mathbb{R}^{d_1\times\kappa}\times\mathbb{R}^{d_2\times\kappa}$, and obtain the factorization form of (1.2):

$$\min_{\substack{U\in\mathbb{R}^{d_1\times\kappa}\\ V\in\mathbb{R}^{d_2\times\kappa}}} \Big\{F_\lambda(U,V) := F(U,V) + \frac{\lambda}{2}(\|U\|_F^2 + \|V\|_F^2)\Big\}, \tag{1.3}$$

where $F(U,V) := \frac{1}{2}\big\|P_\Omega(AUV^\top B^\top - M)\|_F^2$. For $\lambda = 0$ and $\kappa = r^*$, [24] proposed the Gauss-Newton inductive matrix completion (GNIMC) method and established the error bound of iterates to the true matrix, which requires assumptions on A, B, and the initial point. Moreover, [22] considered the factorization model with balanced regularization in which they presented a gradient-based algorithm that converges to the true underlying matrix at a linear rate with incoherence assumptions and sample size conditions. Most existing theoretical guarantees for the factorized model are obtained under the implicit assumption that $\kappa = r^*$. However, in many scenarios, only a rough upper estimate for the true rank r^* is available. Thus, to ensure that these theoretical results fully work in practice, it is necessary to find a factorized model that involves a regularized term to reduce κ to r^*. To obtain a low-rank result from (1.3), a suitably large λ is necessary, which inevitably leads to a worse error bound condition to the recovered matrix X^*, see [18, Proposition A.1]. In fact, [13] employed $\ell_{2,1}$-norm regularization to promote low rank for inductive matrix completion. However, the $\ell_{2,1}$-norm is merely an approximation of the rank function. Furthermore, the theoretical results in [13] established risk estimation at the optimal solution under the assumptions of $k = r^*$ (true rank) and noiseless observations. Generally, the non-convex optimization algorithms can only guarantee convergence to stationary points rather than the globally optimal solution. To address these challenges, we consider the rank regularized problem:

$$\min_{X\in\mathbb{R}^{d_1\times d_2}} \Big\{\Psi(X) := f(X) + \mu\|X\|_* + \lambda\text{rank}(X)\Big\}, \tag{1.4}$$

which, in terms of the factorized form of $X = UV^\top$, can be equivalently written as (see [18])

$$\min_{\substack{U\in\mathbb{R}^{d_1\times\kappa}\\ V\in\mathbb{R}^{d_2\times\kappa}}} \Big\{\Phi(U,V) := F(U,V) + \frac{\mu}{2}(\|U\|_F^2 + \|V\|_F^2) + \frac{\lambda}{2}(\|U\|_{2,0} + \|V\|_{2,0})\Big\}, \tag{1.5}$$

where $\mu > 0$ is a small constant, and the term $\frac{1}{2}(\|U\|_{2,0} + \|V\|_{2,0})$ is an exact variational characterization of the rank function. The term $\frac{\mu}{2}(\|U\|_F^2 + \|V\|_F^2)$ is added to ensure that (1.5) has a nonempty and balanced stationary point set (see Proposition 3.1).

The main contribution of this work is to resolve the inductive matrix completion based on the column $\ell_{2,0}$-regularized factorization model (1.5). First, we establish the equivalence between the stationary points of (1.4) and (1.5). This means that we can achieve a stationary point of (1.4) by solving factorized model (1.5), enabling more efficient computation and less memory. Moreover, for the stationary points, we derive an error bound result to M^* with a rank less than or equal to $\mathrm{rank}(M^*)$, under an assumption that the condition number of the Hessian matrix $\nabla^2 F_\mu$ is restricted. Second, we propose a proximal alternating minimization algorithm with subspace correction to solve (1.5). Among others, the subspace correction strategy is added to make the corrected subproblems have a closed-form solution. By leveraging the KL property of the objective function of (1.5), we obtain the global convergence of the proposed method. Finally, we compare the performance of the PAM algorithm with subspace correction for (1.5) with that of the alternating method for (1.3) proposed in [4, 12]. The numerical results demonstrate the effectiveness of our proposed method.

The rest of the paper is structured as follows. Section 2 provides the preliminaries. Section 3 establishes the equivalence between stationary points of (1.5) and (1.4) and the error bound result to the true matrix for stationary points of (1.5). In Section 4, we present the PAM algorithm with subspace correction solving problem (1.5) and establish the global convergence result. In Section 5, we numerically evaluate the performance of the PAM algorithm with subspace correction and compare its performance with a proximal alternating method solving problem (1.3) on synthetic and real-world datasets.

2. Preliminaries

We first describe the notations and revisit the definitions used in this paper.

2.1. Notation

Let $\mathbb{R}^{n_1\times n_2}$ be the matrix space of all $n_1 \times n_2$ real matrices. The inner product is defined by the trace product $\langle X, Y\rangle = \mathrm{trace}(X^\top Y)$ for $X, Y \in \mathbb{R}^{n_1\times n_2}$. Let $\mathbb{O}^{n\times\kappa}$ be the set of column orthonormal matrices, and denote $\mathbb{O}^n$ for $\mathbb{O}^{n\times n}$. Let $\sigma(X) := (\sigma_1(X), \ldots, \sigma_n(X))^\top$ with $\sigma_1(X) \geq \cdots \geq \sigma_n(X)$ for $X \in \mathbb{R}^{n\times m}$. Write $\mathbb{O}^{n,m}(X) := \{(U, V) \in \mathbb{O}^n \times \mathbb{O}^m \mid X = U\mathrm{Diag}(\sigma(X))V^\top\}$, where $\mathrm{Diag}(z)$ represents a rectangular diagonal matrix with z as its diagonal elements and its dimensions are specified by context. For any $0 < \kappa \leq n$, denote $\sigma^\kappa(X) = \mathrm{Diag}(\sigma_1(X), \cdots, \sigma_\kappa(X))$. Let $\|X\|$, $\|X\|_*$, $\|X\|_F$, and $\|X\|_{2,0}$ denote the spectral norm, nuclear norm, Frobenius norm, and the number of nonzero columns of X, respectively, and X_i denotes the i-th column of X. For any $X \in \mathbb{R}^{n\times m}$, $\sqrt{X}$ denotes the matrix obtained by taking the element-wise square root of X.

2.2. Stationary points of (1.4) *and* (1.5)

In this part, we provide the relevant concepts and theories that are necessary for comprehending our research. We first recall from [16] the generalized subdifferentials for an extended real-valued function $h : \mathbb{R}^{n\times m} \to \overline{\mathbb{R}} := (-\infty, \infty]$ at a point with a finite value.

Definition 2.1. *Consider an extended real-value function $h\colon \mathbb{R}^{n\times m} \to \overline{\mathbb{R}}$ and a point x with $h(x)$ finite. Denoted by $\widehat{\partial}h(x)$ the regular subdifferential of h at x defined as*

$$\widehat{\partial}h(x) := \Big\{v \in \mathbb{R}^{n\times m} \;\Big|\; \liminf_{x\neq x'\to x} \frac{h(x') - h(x) - \langle v, x' - x\rangle}{\|x' - x\|_F} \geq 0\Big\},$$

and the basic subdifferential (also known as the limiting or Morduhovich subdifferential) of h at x defined as

$$\partial h(x) := \Big\{v \in \mathbb{R}^{n\times m} \mid \exists\, x^k \to x \text{ with } h(x^k) \to h(x) \text{ and } v^k \to v \text{ with } v^k \in \widehat{\partial}h(x^k)\Big\}.$$

Using the definition of the subdifferential and [16], the following lemmas characterize the subdifferential of the functions in problems (1.4) and (1.5).

Lemma 2.1. *Fix any $\mu, \lambda > 0$. For any $\overline{X} \in \mathbb{R}^{d_1\times d_2}$, it holds that $\widehat{\partial}\Psi(\overline{X}) = \partial\Psi(\overline{X})$ and $\partial\Psi(\overline{X}) = \nabla f(\overline{X}) + \mu\partial\|\cdot\|_*(\overline{X}) + \lambda\partial\mathrm{rank}(\overline{X})$ with $\nabla f(\overline{X}) = A^\top P_\Omega(A\overline{X}B^\top - M)B$ and*

$$\partial\Psi(\overline{X}) = \nabla f(\overline{X}) + \Big\{\mu\overline{P}\mathrm{Diag}(\zeta)\overline{Q}^\top \Big|\, \zeta \in \mathbb{R}^{d_1},\ \zeta_i = 1 \text{ for } \sigma_i(\overline{X}) > 0, (\overline{P}, \overline{Q}) \in \mathbb{O}^{d_1,d_2}(\overline{X})\Big\}.$$

Proof. For any $X \in \mathbb{R}^{d_1\times d_2}$, let $\vartheta_1(X) := \mu\|X\|_* = \mu\|\sigma(X)\|_1$ and $\vartheta_2(X) := \lambda\mathrm{rank}(X) = \lambda\|\sigma(X)\|_0$. Then, for given $\overline{X}$, by the regularity of the ℓ_1-norm and ℓ_0-norm, [9, Theorem 2.1], it immediately follows that

$$\begin{aligned}
\widehat{\partial}\vartheta_1(\overline{X}) = \partial\vartheta_1(\overline{X}) &= \Big\{\overline{P}\mathrm{Diag}(v)\overline{Q}^\top \;\Big|\; v \in \mu\partial\|\cdot\|_1(\sigma(\overline{X})),\ (\overline{P}, \overline{Q}) \in \mathbb{O}^{d_1,d_2}(\overline{X})\Big\},\\
\widehat{\partial}\vartheta_2(\overline{X}) = \partial\vartheta_2(\overline{X}) &= \Big\{\overline{P}\mathrm{Diag}(v)\overline{Q}^\top \;\Big|\; v \in \lambda\partial\|\cdot\|_0(\sigma(\overline{X})),\ (\overline{P}, \overline{Q}) \in \mathbb{O}^{d_1,d_2}(\overline{X})\Big\}.
\end{aligned}$$

From [16, Exercise 10.9 & Exercise 10.10] and the strict continuity of ϑ_1, we get $\partial(\vartheta_1+\vartheta_2) = \partial\vartheta_1+\partial\vartheta_2$. This combined with [16, Exercise 8.8(c)] and the smoothness of f imply

$$\widehat{\partial}\Psi(\overline{X}) = \partial\Psi(\overline{X}) = \nabla f(\overline{X}) + \partial\vartheta_1(\overline{X}) + \partial\vartheta_2(\overline{X}).$$

Then by the expressions of $\partial\vartheta_1(X)$ and $\partial\vartheta_2(X)$, we get the desired result. □

Lemma 2.2. *Fix an arbitrary $\mu, \lambda > 0$. For any $(U, V) \in \mathbb{R}^{d_1\times\kappa} \times \mathbb{R}^{d_2\times\kappa}$, write $J_U := \{j \mid U_j \neq 0\}$ and $J_V := \{j \mid V_j \neq 0\}$. Then, we obtain that $\widehat{\partial}\Phi(U, V) = \partial\Phi(U, V) = \partial_U\Phi(U, V) \times \partial_U\Phi(U, V)$ with*

$$\begin{aligned}
\partial_U\Phi(U, V) =& \Big\{G \in \mathbb{R}^{d_1\times\kappa} \mid G_j = \nabla f(UV^\top)V_j + \mu U_j,\ j \in J_U\Big\},\\
\partial_V\Phi(U, V) =& \Big\{H \in \mathbb{R}^{d_2\times\kappa} \mid H_j = [\nabla f(UV^\top)]^\top U_j + \mu V_j,\ j \in J_V\Big\}.
\end{aligned}$$

Proof. Let $g(Z) := \|Z\|_{2,0}$ for $Z \in \mathbb{R}^{l\times\kappa}$ with $l = d_1$ or d_2. By the smoothness of F and [16, Exercise 8.8(c) & Proposition 10.5], it follows that

$$\begin{aligned}
\widehat{\partial}\Phi(U,V) &= \widehat{\partial}_U\Phi(U,V) \times \widehat{\partial}_V\Phi(U,V)\\
&= \{\nabla_U F(U,V) + \mu U + \widehat{\partial}g(U)\} \times \{\nabla_V F(U,V) + \mu V + \widehat{\partial}g(V)\}, \qquad (2.2)\\
\partial\Phi(U,V) &= \partial_U\Phi(U, V) \times \partial_V\Phi(U,V)\\
&= \{\nabla_U F(U,V) + \mu U + \partial g(U)\} \times \{\nabla_V F(U,V) + \mu V + \partial g(V)\}. \qquad (2.3)
\end{aligned}$$

By [16, Proposition 10.5], we obtain that

$$\widehat{\partial}g(U) = \partial g(U) = S_1 \times \cdots \times S_\kappa \ \text{ with } S_j = \begin{cases} \{0\}^{d_1} & \text{if } j \in J_U; \\ \mathbb{R}^{d_1} & \text{if } j \notin J_U. \end{cases}$$

This combined with Eqs (2.2) and (2.3) implies that the desired results hold. □

The above discussions motivate us to introduce the following definitions of stationary points for problems (1.4) and (1.5).

Definition 2.2. *A matrix $X \in \mathbb{R}^{d_1\times d_2}$ with $0 \in \partial\Psi(X)$ is called a stationary point of problem* (1.4)*, and for any $(U, V) \in \mathbb{R}^{d_1\times\kappa} \times \mathbb{R}^{d_2\times\kappa}$ such that $0 \in \partial\Phi(U, V)$, it is called a stationary point of problem* (1.5).

2.3. The Kurdyka-Łojasiewicz (KL) property

We recall from [1] the KL property of an extended real-valued function.

Definition 2.3. *Let $h : \mathbb{X} \to \overline{\mathbb{R}}$ be a proper lower semicontinuous (lsc) function. The function h is said to have the Kurdyka-Łojasiewicz (KL) property at $\overline{x} \in \operatorname{dom}\partial h$ if there exists, for $\eta \in (0, \infty]$, a continuous concave function $\varphi\colon [0, \eta) \to \mathbb{R}_+$ satisfying*

(i) *$\varphi(0) = 0$ and φ is continuously differentiable on $(0, \eta)$,*

(ii) *for all $s \in (0, \eta)$, $\varphi'(s) > 0$;*

and a neighborhood $\mathcal{U}$ of $\overline{x}$ such that for all $x \in \mathcal{U} \cap [h(\overline{x}) < h(x) < h(\overline{x}) + \eta]$,

$$\varphi'(h(x) - h(\overline{x}))\text{dist}(0, \partial h(x)) \geq 1.$$

If h satisfies the KL property at each point of $\operatorname{dom}\partial h$, then it is called a KL function.

Remark 2.1. *According to Lemma 2.1 in [1], any proper lower semicontinuous (lsc) function possesses the KL property at all noncritical points. Therefore, to establish that a proper lsc function $h\colon \mathbb{X} \to \overline{\mathbb{R}}$ is a KL function, it is sufficient to verify the KL property at its critical points.*

3. Properties of stationary points of model (1.5)

Since the factorized problem (1.5) is nonconvex, we can generally only obtain stationary points. To achieve the recovery guarantees of the proposed model, it is crucial to discuss the relationship between stationary points of problems (1.4) and (1.5) and to establish the error bound to the true matrix for stationary points of (1.5).

3.1. Equivalence between stationary points of (1.4) *and* (1.5)

For a proper $h\colon \mathbb{X} \to \mathbb{R}$, the set of all stationary points of h is denoted by $\operatorname{crit} h$. The next proposition establishes the balance property of the stationary points of (1.5).

Proposition 3.1. *Given any stationary point $(U, V) \in \mathbb{R}^{d_1\times\kappa} \times \mathbb{R}^{d_2\times\kappa}$ of* (1.5)*, write $J_U := \{j \mid U_j \neq 0\}$ and $J_V := \{j \mid V_j \neq 0\}$. Then $J := J_U = J_V$ and (U, V) belongs to*

$$\mathcal{E}_\lambda := \Big\{(U, V) \in \mathbb{R}^{d_1\times\kappa} \times \mathbb{R}^{d_2\times\kappa} \mid U^\top U = V^\top V\Big\}. \tag{3.1}$$

Consequently, $\|U_j\| = \|V_j\|$ for $j \in [\kappa]$, $\operatorname{rank}(U) = \operatorname{rank}(V)$, and $\|U\|_{2,0} = \|V\|_{2,0}$.

Proof. First, we show that $J_U = J_V$. Pick any $(U, V) \in \mathrm{crit}\Phi$, and we have $\mathbf{0} \in \partial_U\Phi(U, V) \times \partial_V\Phi(U, V)$. If either J_U or J_V is empty, we get that both sets are empty. Indeed, if $J_U = \emptyset$, then $U = 0$. Then, by the expression of $\partial_V\Phi$ in Lemma 2.2, it immediately follows that $V = 0$ and $J_V = \emptyset$. Similarly, if $J_V = \emptyset$, then we get $J_U = \emptyset$. When $J_U \neq \emptyset$ and $J_V \neq \emptyset$, by Lemma 2.2, for each $j \in J_U$, since

$$U_j \neq 0 \text{ and } A^\top[P_\Omega(AUV^\top B^\top - M)]BV_j + \mu U_j = 0,$$

then we get $V_j \neq 0$. Hence $J_U \subset J_V$. Similarly, for each $j \in J_V$, we obtain $U_j \neq 0$ and $J_V \subset J_U$. Consequently, we have $J_U = J_V = J$. By Lemma 2.2, we have

$$\begin{cases} A^\top[P_\Omega(AUV^\top B^\top - M)]BV_J + \mu U_J = 0, \\ (A^\top[P_\Omega(AUV^\top B^\top - M)]B)^\top U_J + \mu V_J = 0. \end{cases}$$

By simple calculation, we get

$$\begin{cases} U_J^\top A^\top[P_\Omega(AUV^\top B^\top - M)])BV_J + \mu U_J^\top U_J = 0, \\ V_J^\top(A^\top[P_\Omega(AUV^\top B^\top - M)]B)^\top U_J + \mu V_J^\top V_J = 0. \end{cases}$$

Then, it is not hard to obtain that $U_J^\top U_J = V_J^\top V_J$. Thus, we get $U^\top U = V^\top V$, which implies that $\mathrm{rank}(U) = \mathrm{rank}(V)$ and $\|U\|_{2,0} = \|V\|_{2,0}$. □

The next proposition describes the equivalence between the stationary points of problem (1.4) and those of its factorized form (1.5).

Proposition 3.2. *Suppose* $(U, V) \in \mathbb{R}^{d_1\times\kappa} \times \mathbb{R}^{d_2\times\kappa}$ *is any arbitrary stationary point of* (1.5). *Write* $X = UV^\top$. *Then* X *is a stationary point of* (1.4). *Conversely, if* X *is an arbitrary stationary point of* (1.4) *with* $\mathrm{rank}(X) \leq \kappa$ *and the SVD* $X = \widetilde{P}\mathrm{Diag}(\sigma(X))\widetilde{Q}^\top$, *where* $\widetilde{P} \in \mathbb{O}^{d_1\times\kappa}$ *and* $\widetilde{Q} \in \mathbb{O}^{d_2\times\kappa}$, *then* $(\widetilde{P}\sqrt{\mathrm{Diag}(\sigma(X))}, \widetilde{Q}\sqrt{\mathrm{Diag}(\sigma(X))})$ *is the stationary point of* (1.5).

Proof. Fix any $(U, V) \in \mathrm{crit}\Phi$. By Proposition 3.1 and Lemma 2.2, we have $J := J_U = J_V$ and

$$\begin{cases} \nabla f(X)V_J + \mu U_J = 0, & (3.2a) \\ (\nabla f(X))^\top U_J + \mu V_J = 0. & (3.2b) \end{cases}$$

From the last two equalities, we get $U_J = -\frac{\nabla f(X)V_J}{\mu}$ and $V_J = -\frac{(\nabla f(X))^\top U_J}{\mu}$. Substituting U_J and V_J into (3.2b) and (3.2a), respectively, yields

$$(\nabla f(X))^\top\nabla f(X)V_J = \mu^2 V_J \text{ and } \nabla f(X)(\nabla f(X))^\top U_J = \mu^2 U_J. \tag{3.3}$$

From the above equalities, we get that U_J and V_J are the two matrices with columns corresponding to the eigenvectors of $\nabla f(X)(\nabla f(X))^\top$ and $(\nabla f(X))^\top\nabla f(X)$ with an eigenvalue equal to μ^2. Thus, there exists $\kappa_1 \in \mathbb{N}$ such that μ is the κ_1-multiple singular value root of $\nabla f(X)$. Let the SVD of $\nabla f(X)$ be

$$\nabla f(X) = \mu P_1 Q_1^\top + P_2\Sigma_2 Q_2^\top, \tag{3.4}$$

where $[P_1\ P_2] \in \mathbb{O}^{d_1\times\kappa}$, $[Q_1\ Q_2] \in \mathbb{O}^{d_2\times\kappa}$ and $P_1 \in \mathbb{O}^{d_1\times\kappa_1}$, $Q_1 \in \mathbb{O}^{d_2\times\kappa_1}$. Then

$$\nabla f(X)(\nabla f(X))^\top = \mu^2 P_1 P_1^\top + P_2\Sigma_2^2 P_2^\top,$$

$$(\nabla f(X))^\top \nabla f(X) = \mu^2 Q_1 Q_1^\top + Q_2 \Sigma_2^2 Q_2^\top.$$

Combining with Eq (3.3), there exist matrices $C_1 \in \mathbb{R}^{\kappa_1 \times |J|}$ and $C_2 \in \mathbb{R}^{\kappa_1 \times |J|}$ such that $U_J = P_1 C_1$ and $V_J = Q_1 C_2$. Then we shall proof

$$C_1 = -C_2.$$

Submitting $U_J = P_1 C_1$, $V_J = Q_1 C_2$, and Eq (3.4) into (3.2a), we get

$$(\mu P_1 Q_1^\top + P_2 \Sigma_2 Q_2^\top) Q_1 C_2 + \mu P_1 C_1 = 0.$$

Multiply the last equation by $P_1^\top$, and it immediately holds that $C_1 = -C_2$. Let $C := C_1 = -C_2$ and $r_1 := \operatorname{rank}(C)$. We can rewrite

$$U_J = P_1 C \text{ and } V_J = -Q_1 C.$$

Let the SVD of C be $C = \widetilde{P}\widetilde{\Sigma}\widetilde{Q}^\top$ with $\widetilde{P} \in \mathbb{O}^{\kappa_1 \times \kappa_1}$, $\widetilde{Q} \in \mathbb{O}^{|J| \times r_1}$, and $\widetilde{\Sigma} = \operatorname{Diag}(\sigma(C)) \in \mathbb{R}^{\kappa_1 \times r_1}$, and then

$$X = UV^\top = U_J V_J^\top = -\widetilde{P}_1 \widetilde{\Sigma}_1 \widetilde{Q}_1^\top \text{ with } \widetilde{P}_1 = P_1 \widetilde{P},\ \widetilde{Q}_1 = Q_1 \widetilde{P},\ \widetilde{\Sigma}_1 = \widetilde{\Sigma}\widetilde{\Sigma}^\top.$$

Note that$\widetilde{P}_1 \widetilde{Q}_1^\top = P_1 Q_1^\top$, and together with (3.4), we have

$$\nabla f(X) = \mu \widetilde{P}_1 \widetilde{Q}_1^\top + P_2 \Sigma_2 Q_2^\top.$$

This combined with $X = -\widetilde{P}_1 \widetilde{\Sigma}_1 \widetilde{Q}_1^\top$ and the expression of $\partial\Psi$ in Lemma 2.1 implies $0 \in \operatorname{crit}\Psi$. Consequently, X is the stationary point of (1.4).

Conversely, let X be a stationary point of (1.4) and the SVD of X be $X = P\operatorname{Diag}(\sigma(X))Q^\top$ with $(P, Q) \in \mathbb{O}^{d_1, d_2}$ and $U = P^\kappa \sqrt{\operatorname{Diag}(\sigma^\kappa(X))}$, $V = Q^\kappa \sqrt{\operatorname{Diag}(\sigma^\kappa(X))}$, where P^κ, Q^κ are the matrices obtained by deleting the last $d_1 - \kappa$ and $d_2 - \kappa$ columns of P and Q, respectively. By Lemma 2.1, we have $0 \in \operatorname{crit}\Psi$ and

$$-\frac{1}{\mu}\nabla f(X) \in \left\{ P\operatorname{Diag}(v)Q^\top \,\middle|\, v \in \mathbb{R}^{d_1}, v_j = 1 \text{ for } \sigma_i(X) > 0 \right\}.$$

Let $r := \operatorname{rank}(X)$. Then there exist $\overline{v} \in \mathbb{R}^{d_1}$ with $\overline{v}_j = 1$, $j = 1, \cdots, r$, such that $-\frac{1}{\mu}\nabla f(X) = P\operatorname{Diag}(\overline{v})Q^\top$. By the expressions of U and V and $r \le \kappa$, we have $UV^\top = X$,

$$-\frac{1}{\mu}\nabla f(X) V = P^\kappa \sqrt{\operatorname{Diag}(\sigma^\kappa(X))}, \text{ and } -\frac{1}{\mu} U^\top \nabla f(X) = \sqrt{\operatorname{Diag}(\sigma^\kappa(X))}(Q^\kappa)^\top,$$

which means that (U, V) is the stationary point of (1.5). □

3.2. *Error bound for the stationary point of* (1.5)

Given any stationary point (U, V) of (1.5) satisfying $\operatorname{rank}(UV^\top) \le r^*$, the following theorem gives an upper bound for the true solution M^* under certain mild assumptions. Recall that $\nabla f(X) = A^\top P_\Omega(AXB^\top - M)B$. For any $H \in \mathbb{R}^{d_1 \times d_2}$, it is not hard to calculate

$$\langle H, \nabla^2 f(X) H \rangle = \langle H, A^\top P_\Omega(AHB^\top)B \rangle = \|P_\Omega(AHB^\top)\|_F^2. \tag{3.5}$$

Assumption 3.1. *Given any* $X, H \in \mathbb{R}^{d_1\times d_2}$ *with* $\mathrm{rank}(X) \le 2r^*$ *and* $\mathrm{rank}(H) \le 4r^*$, *there exist* $0 < \alpha < \beta$ *with* $\alpha\|H\|_F^2 \le \|P_\Omega(AHB^\top)\|_F^2 \le \beta\|H\|_F^2$.

When the information matrices A and B are nonsingular, the above assumption simplifies to the restricted strong convexity (RSC) condition, which is a standard requirement for loss functions in low-rank matrix recovery problems (see [14, 23]). In this case, the assumption will hold with high probability.

Theorem 3.1. *Suppose Assumption 3.1 holds with* $0 < \beta/\alpha \le 1.38$. *Fix any* $\mu, \lambda > 0$. *Given any stationary point* (U, V) *of* (1.5) *satisfying* $\mathrm{rank}(UV^\top) \le r^*$ *and with* $\nabla^2 F_\mu(U, V)$ *being positive semidefinite, there exists a constant* $\gamma_0 > 0$ *(depending only on α and β) such that the following holds:*

$$\|UV^\top - M^*\|_F \le \gamma_0 \sqrt{r^*}(\mu + \|\nabla f(M^*)\|). \tag{3.6}$$

Proof. Take any $(U, V) \in \mathrm{crit}\Phi$ with $\mathrm{rank}(UV^\top) \le r^*$. By Proposition 3.1, for any $(U, V) \in \mathrm{crit}\Phi$, it holds that $U^\top U = V^\top V$. Let $X = UV^\top$. Then $\mathrm{rank}(X) \le r^*$. From the expression of $\nabla^2 f$ in (3.5) and Assumption 3.1, for any $Y, H \in \mathbb{R}^{d_1\times d_2}$ with $\mathrm{rank}(Y) \le 2r^*$ and $\mathrm{rank}(H) \le 4r^*$, there exist constants $0 < \alpha < \beta$ such that

$$\alpha\|H\|_F^2 \le \langle H, \nabla^2 f(X)H\rangle \le \beta\|H\|_F^2.$$

This means that the Hession of f satisfies both $(2r^*, 4r^*)$-RSC and $(2r^*, 4r^*)$-RSS in [17]. Combining with the condition $\nabla^2 F_\mu(U, V) \succeq 0$ and [17, Theorem 3.1], we get the desired result. □

Remark 3.1. *By the equivalence of the stationary points of problems* (1.4) *and* (1.5) *in Proposition 3.2, any stationary point X of problem* (1.4) *with* $\mathrm{rank}(X) \le r^*$ *has the error bound* (3.6) *under the conditions of Theorem 3.1. Furthermore, we can get a low-rank stationary point of* (1.4) *that satisfies the error bound by solving the factorized problem* (1.5).

4. A PAM algorithm with subspace correction for solving problem (1.5)

Recall that $f(X) = \frac{1}{2}\|P_\Omega(AXB^\top - M)\|_F^2$ is a smooth function which has a Lipschitz continuous gradient $\nabla f(X)$ with module $c_f = \|A\|^2\|B\|^2$. Then by the descent lemma [15, Proposition A.24], for any $\gamma \ge c_f$, fix any $X^k \in \mathbb{R}^{d_1\times d_2}$, and it holds that

$$f(X) \le f(X^k) + \langle\nabla f(X^k), X - X^k\rangle + \frac{\gamma}{2}\|X - X^k\|_F^2, \ \ \forall \ X \in \mathbb{R}^{d_1\times d_2}.$$

Then, for any fixed $(U^k, V^k) \in \mathbb{R}^{d_1\times\kappa} \times \mathbb{R}^{d_2\times\kappa}$, write $X^k = U^k V^{k^\top}$, and it holds that

$$\begin{aligned} f(UV^\top) &\le f(U^k V^{k^\top}) + \langle\nabla f(U^k V^{k^\top}), UV^\top - U^k V^{k^\top}\rangle + \frac{\gamma}{2}\|UV^\top - U^k V^{k^\top}\|_F^2 \\ &:= \widehat{F}(U, V; U^k, V^k) \quad \text{for any } (U, V) \in \mathbb{R}^{d_1\times\kappa} \times \mathbb{R}^{d_2\times\kappa}, \end{aligned}$$

which together with the expression of $\Phi_{\lambda,\mu}$ immediately implies that

$$\Phi(U, V) \le \widehat{\Phi}(U, V, U^k, V^k) := \widehat{F}(U, V; U^k, V^k) + \frac{\lambda}{2}(\|U\|_{2,0} + \|V\|_{2,0}) + \frac{\mu}{2}(\|U\|_F^2 + \|V\|_F^2).$$

Since $\widehat{\Phi}(U^k, V^k, U^k, V^k) = \Phi(U^k, V^k)$, $\widehat{\Phi}(\cdot, \cdot, U^k, V^k)$ is a majorization of Φ at (U^k, V^k). Let (U^k, V^k) be the current iterate. Then, we use a majorized proximal alternative minimizing algorithm with subspace correction steps proposed in [18, 19] to solve problem (1.5) by minimizing the following subproblem alternately:

$$\begin{cases} U^{k+1} \in \underset{U\in\mathbb{R}^{d_1\times\kappa}}{\arg\min}\Big\{\langle \nabla f(U^k(V^k)^\top)V^k, U\rangle + \frac{\lambda}{2}\|U\|_{2,0} + \frac{\mu}{2}\|U\|_F^2 + \frac{L_f}{2}\|(U - U^k)(V^k)^\top\|_F^2\Big\}, \\ V^{k+1} \in \underset{V\in\mathbb{R}^{d_2\times\kappa}}{\arg\min}\Big\{\langle [\nabla f(U^{k+1}(V^k)^\top)]^\top U^{k+1}, V\rangle + \frac{\lambda}{2}\|V\|_{2,0} + \frac{\mu}{2}\|V\|_F^2 + \frac{L_f}{2}\|U^{k+1}(V - V^k)^\top\|_F^2\Big\}. \end{cases}$$

Its iteration steps are described below.

Algorithm 1 (A majorized PAM with subspace correction for solving (1.5)**)**

Initialization: Input parameters $\varrho \in (0,1), \underline{\gamma_1} > 0, \underline{\gamma_2} > 0, \gamma_{1,0} > 0$, and $\gamma_{2,0} > 0$. Choose $\overline{P}^0 \in \mathbb{O}^{d_2\times\kappa}$, $\widetilde{P}^0 \in \mathbb{O}^{d_1\times\kappa}$, $\overline{Q}^0 = \overline{D}^0 = I_\kappa$, and let $(\overline{U}^0, \overline{V}^0) = (\widetilde{P}^0, \overline{P}^0)$, $\overline{X}^0 = \overline{U}^0(\overline{V}^0)^\top$.

For $k = 0, 1, 2, \ldots$ **do**

1. Compute the following problem:

$$\begin{aligned} U^{k+1} \in \underset{U\in\mathbb{R}^{d_1\times\kappa}}{\arg\min}\Big\{&\langle \nabla f(\overline{X}^k), U(\overline{V}^k)^\top\rangle + \frac{\lambda}{2}\|U\|_{2,0} + \frac{\mu}{2}\|U\|_F^2 \\ &+ \frac{c_f}{2}\|\overline{X}^k - U(\overline{V}^k)^\top\|_F^2 + \frac{\gamma_{1,k}}{2}\|U - \overline{U}^k\|_F^2\Big\}. \end{aligned} \tag{4.2}$$

2. Find a thin SVD of $U^{k+1}\overline{D}^k$ as $U^{k+1}\overline{D}^k = \widetilde{P}^{k+1}(\widetilde{D}^{k+1})^2(\widetilde{Q}^{k+1})^\top$. Set

$$\widetilde{U}^{k+1} = \widetilde{P}^{k+1}\widetilde{D}^{k+1},\ \widetilde{V}^{k+1} = \overline{P}^k\widetilde{Q}^{k+1}\widetilde{D}^{k+1},\ \widetilde{X}^{k+1} = \widetilde{U}^{k+1}(\widetilde{V}^{k+1})^\top,\ \kappa = \mathrm{rank}(\widetilde{D}^{k+1}).$$

3. Compute the following problem:

$$\begin{aligned} V^{k+1} \in \underset{V\in\mathbb{R}^{d_2\times\kappa}}{\arg\min}\Big\{&\langle \nabla f(\widetilde{X}^{k+1}), \widetilde{U}^{k+1}V^\top\rangle + \frac{\lambda}{2}\|V\|_{2,0} + \frac{\mu}{2}\|V\|_F^2 \\ &+ \frac{c_f}{2}\|\widetilde{X}^{k+1} - \widetilde{U}^{k+1}V^\top\|_F^2 + \frac{\gamma_{2,k}}{2}\|V - \widetilde{V}^{k+1}\|_F^2\Big\}. \end{aligned} \tag{4.3}$$

4. Find a thin SVD of $V^{k+1}\widetilde{D}^{k+1}$ as $V^{k+1}\widetilde{D}^{k+1} = \overline{P}^{k+1}(\overline{D}^{k+1})^2(\overline{Q}^{k+1})^\top$. Set

$$\overline{U}^{k+1} = \widetilde{P}^{k+1}\overline{Q}^{k+1}\overline{D}^{k+1},\ \overline{V}^{k+1} = \overline{P}^{k+1}\overline{D}^{k+1},\ \overline{X}^{k+1} = \overline{U}^{k+1}(\overline{V}^{k+1})^\top,\ \kappa = \mathrm{rank}(\overline{D}^{k+1}).$$

5. Set $\gamma_{1,k+1} = \max(\underline{\gamma_1}, \varrho\gamma_{1,k})$ and $\gamma_{2,k+1} = \max(\underline{\gamma_2}, \varrho\gamma_{2,k})$.

end (For)

Remark 4.1. **(i)** *Let* $M_U^k = \overline{X}^k - c_f^{-1}\nabla f(\overline{X}^k)$. *Since* $\overline{V}^k = \overline{Q}^k\overline{D}^k$, *the subproblem* (4.2) *is equivalent to the following:*

$$U^{k+1} \in \min_{U\in\mathbb{R}^{n_1\times\kappa}} \Big\{\frac{c_f}{2}\|M_U^k\overline{Q}^k - U\overline{D}^k\|_F^2 + \frac{\lambda}{2}\|U\|_{2,0} + \frac{\mu}{2}\|U\|_F^2 + \frac{\gamma_{1,k}}{2}\|U - \overline{U}^k\|_F^2\Big\}.$$

Let $G^k = (c_f M_U^k\overline{P}^k + \gamma_{1,k}\overline{U}^k)(S^k)^{-1}$ *with* $S^k = (c_f(\overline{D}^k)^2 + (\mu + \gamma_{1,k})I)^{1/2}$. *After elementary calculation, the columns of* U^{k+1} *yield that*

$$U_i^{k+1} = \begin{cases} G_i^k/S_{ii}^k & \text{if } \|G_i^k\|^2 > \lambda; \\ 0 & \text{otherwise.} \end{cases}$$

Similarly, let $M_V^{k+1} = \widetilde{X}^{k+1} - c_f^{-1}\nabla f(\widetilde{X}^{k+1})$, $\widetilde{G}^{k+1} = (c_f(M_V^{k+1})^\top\widetilde{P}^{k+1} + \gamma_{2,k}\widetilde{V}^{k+1})(\widetilde{S}^{k+1})^{-1}$ *with* $\widetilde{S}^{k+1} = (c_f(\widetilde{D}^{k+1})^2 + (\mu + \gamma_{2,k})I)^{1/2}$. *After elementary calculation, the columns of* V^{k+1} *yield that*

$$V_i^{k+1} = \begin{cases} \widetilde{G}_i^{k+1}/\widetilde{S}_{ii}^{k+1} & \text{if } \|\widetilde{G}_i^{k+1}\|^2 > \lambda; \\ 0 & \text{otherwise.} \end{cases}$$

By the expressions of $\overline{U}^k$, $\overline{V}^k$ *and* $\widetilde{U}^k$, $\widetilde{V}^k$, *we deduce that each step of Algorithm 1 involves about* $4(nm + nd_1 + md_2)\kappa$ *flops. From [19, Proposition 3], in each iteration of Algorithm 1,* κ *may decrease, which leads to lower computational complexity.*
(ii) *The first-order optimality conditions of* (4.2) *and* (4.3) *have the following form:*

$$0\in\nabla f(\overline{X}^k)\overline{V}^k + c_f(\overline{X}^k - \widetilde{X}^k)\overline{V}^k + \mu U^{k+1} + \gamma_{1,k}(U^{k+1} - \overline{U}^k) + \partial\|\cdot\|_{2,0}(U^{k+1}),$$
$$0\in\nabla f(\widetilde{X}^{k+1})^\top\widetilde{U}^{k+1} + c_f(\widetilde{X}^{k+1} - \overline{X}^{k+1})^\top\widetilde{U}^{k+1} + \mu V^{k+1} + \gamma_{2,k}(V^{k+1} - \widetilde{V}^{k+1}) + \partial\|\cdot\|_{2,0}(V^{k+1}).$$

(iii) *In Steps 1 and 3 of Algorithm 1, the proximal parameters* $\gamma_{1,k} > 0$ *and* $\gamma_{2,k} > 0$ *enforce strict monotonic decrease of the objective function values sequence* $\{\Phi(\overline{U}^k, \overline{V}^k)\}_{k\in\mathbb{N}}$, *thereby guaranteeing convergence of* $\{\overline{X}^k\}_{k\in\mathbb{N}}$ *and* $\{\widetilde{X}^k\}_{k\in\mathbb{N}}$. *Smaller values of* $\gamma_{1,k} > 0$ *and* $\gamma_{2,k} > 0$ *yield better approximations of the primal problem in subproblems* (4.2) *and* (4.3), *while simultaneously increasing computational difficulty. We therefore employ a diminishing scheme to adaptively decrease their values in Algorithm 1 for all numerical experiments.*

Since the zero-norm and the function $\theta(Z) := (\|Z_1\|, \ldots, \|Z_\kappa\|)$ for $Z \in \mathbb{R}^{l\times\kappa}$ are semialgebraic, the column $\ell_{2,0}$-norm, as a composition of θ and the zero-norm, is semialgebraic. This means that Φ is a KL function (see [1, Section 4]). By the lower boundedness of f, Proposition 3.2, and [19, Theorem 4.3], the global convergence of Algorithm 1 follows.

Theorem 4.1. *The sequences* $\{\widetilde{X}^k\}_{k\in\mathbb{N}}$ *and* $\{\overline{X}^k\}_{k\in\mathbb{N}}$ *converge to the same stationary point of* (1.4), *say,* X^*, *and* $\left(P_1^*\sqrt{\mathrm{Diag}(\sigma^\kappa(X^*))}, Q_1^*\sqrt{\mathrm{Diag}(\sigma^\kappa(X^*))}\right)$ *is a stationary point of problem* (1.5), *where* P_1^* *and* Q_1^* *are the matrices obtained by deleting the last* $d_1 - \kappa$ *and* $d_2 - \kappa$ *columns of* P^* *and* Q^*, *respectively, with* $(P^*, Q^*) \in \mathbb{O}^{n,m}(X^*)$.

5. Numerical experiments

In this section, we conduct a numerical test for Algorithm 1 and compare its performance with several candidate methods. All experiments were conducted using MATLAB on a workstation running a 64-bit Windows operating system, equipped with an Intel(R) Core(TM) i9-12900 CPU operating at 2.40 GHz and 64 GB of RAM.

5.1. Model comparison and parameter sensitivity analysis

We compare the proposed Algorithm 1 with the baseline Algorithm 2, both of which are proximal alternating minimization methods. The main distinction is that Algorithm 2 is designed to solve the relaxed problem (1.3), which omits the $\ell_{2,0}$-norm regularization term present in our formulation. Such relaxed models are commonly used in the literature (e.g., [20,22,24]). All experiments are conducted with fixed parameters for Algorithm 1:

$$\mu = 10^{-8},\ \varrho = 0.8,\ \gamma_{1,0} = \gamma_{2,0} = 10^{-5},\ \underline{\gamma_1} = \underline{\gamma_2} = 10^{-8}.$$

We focus on analyzing the sensitivity with respect to the regularization parameter λ, which plays a key role in performance. Other parameters showed limited impact and are thus omitted from the sensitivity analysis.

The test instances are generated using normal distributions. Matrices $A \in \mathbb{R}^{n\times d_1}$, $B \in \mathbb{R}^{m\times d_2}$, and the ground-truth low-rank matrix $X^* \in \mathbb{R}^{d_1\times d_2}$ are drawn from standard Gaussian distributions and normalized to maintain scale. The true data matrix is constructed as $M^* = AX^*B^\top$.

To simulate partial observations, we define a random index set $\Omega = \{(i_t, j_t)\}_{t=1}^{p}$ with uniformly sampled entries, and define the sampling operator $\mathcal{A} : \mathbb{R}^{n\times m} \to \mathbb{R}^p$ as $\mathcal{A}(Y) := (Y_{i_1,j_1}, Y_{i_2,j_2}, \ldots, Y_{i_p,j_p})^\top$. The observation vector is defined as $b = \mathcal{A}(M_\Omega)$, where $M_\Omega \in \mathbb{R}^{n\times m}$. Noisy observations are generated via

$$[M_\Omega]_{i_t,j_t} = \begin{cases} 0, & \text{if } (i_t, j_t) \notin \Omega, \\ M^*_{i_t,j_t} + \varpi_t, & \text{if } (i_t, j_t) \in \Omega, \end{cases} \quad \text{where} \quad \varpi = \frac{0.1\|b\|}{\|w\|} w, \quad w \sim \mathcal{N}(0, I_p). \tag{5.1}$$

Algorithm 2 (A proximal alternating method for solving problem (1.3))

```
Initialization: Choose $U^0 \in \mathbb{O}^{d_1\times\kappa}$, $U^0 \in \mathbb{O}^{d_2\times\kappa}$.
For $k = 0, 1, 2, \ldots$ do
    1. Choose $\widetilde{\gamma}_{1,k} = c_f\|V^k\|^2$. Compute the following problem:
       $U^{k+1} \in \arg\min_{U\in\mathbb{R}^{d_1\times\kappa}} \Big\{\langle \nabla_U F(U^k, V^k), U - U^k\rangle + \frac{\lambda}{2}\|U\|_F^2 + \frac{\widetilde{\gamma}_{1,k}}{2}\|U - U^k\|_F^2\Big\}.$
    2. Choose $\widetilde{\gamma}_{2,k} = c_f\|U^{k+1}\|^2$. Compute the following problem:
       $V^{k+1} \in \arg\min_{V\in\mathbb{R}^{d_2\times\kappa}} \Big\{\langle \nabla_V F(U^{k+1}, V^k), V - V^k\rangle + \frac{\lambda}{2}\|V\|_F^2 + \frac{\widetilde{\gamma}_{2,k}}{2}\|V - V^k\|_F^2\Big\}.$
end (For)
```

Remark 5.1. *(i) From Steps 1 and 2 of Algorithm 2, we observe that each iteration requires approximately* $4(nm + nd_1 + md_2)\kappa$ *flops. Although this per-iteration complexity matches that of Algorithm 1, a key distinction lies in the treatment of the parameter* κ*: Algorithm 1 adaptively reduces* κ *as the iterations proceed, whereas* κ *remains fixed throughout the execution of Algorithm 2. As a result, the per-iteration computational cost of Algorithm 1 decreases monotonically over time.*

(ii) The iteration behavior of Algorithm 2 is influenced by the Lipschitz constants of the partial gradients $\nabla_U F(\cdot, V^k)$ *and* $\nabla_V F(U^k, \cdot)$*, given, respectively, by* $L_1(V^k) = c_f\|V^k\|^2$ *and* $L_1(U^k) = c_f\|U^k\|^2$.

These quantities are usually much larger than c_f, the global Lipschitz constant of ∇f used in Algorithm 1. For instance, if the spectral norm $\|V^k\|$ exceeds 10^3 and $c_f \geq 1$, then $L_1(V^k) > 10^6$; a similar situation arises for $L_1(U^k)$. Such large Lipschitz moduli significantly deteriorate the performance of first-order methods, as their iteration complexity and local convergence rates worsen with increasing values of $L_1(U^k)$ and $L_1(V^k)$.

For initialization, both algorithms use random Gaussian matrices U^0, V^0 orthonormalized to form initial subspaces, and set the diagonal matrix D_0 as a vector of ones. The subspace dimension is fixed to $\kappa = \min\{\min(d_1, d_2), 100\}$.

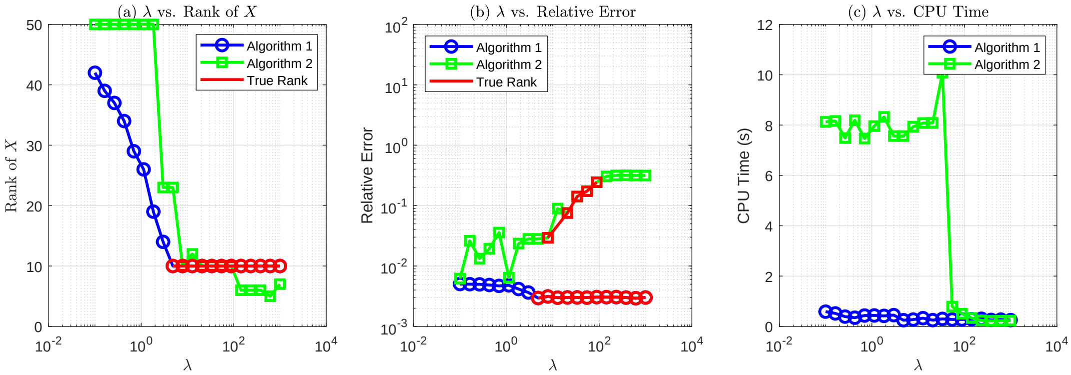
Each algorithm terminates when the maximum number of iterations (1000) is reached, or one of the following conditions is met:

$$\frac{\max_{j\in\{1,\ldots,9\}} |\Phi(U^k, V^k) - \Phi(U^{k-j}, V^{k-j})|}{\max\{1, \Phi(U^k, V^k)\}} \leq 10^{-5}, \quad \frac{\|X^k - X^{k-1}\|_F}{\max\{1, \|X^k\|_F\}} \leq 10^{-5}.$$

5.2. *Parameter sensitivity analysis*

We conduct a sensitivity analysis on the regularization parameter λ for both algorithms. Figure 1 illustrates the effect of λ on rank recovery, relative error, and computational time for a fixed problem with dimensions $(n, m, d_1, d_2, r^*) = (1000, 1000, 50, 50, 10)$. In the figure, red markers indicate successful identification of the true rank.

As shown in Figure 1(a), Algorithm 1 correctly identifies the true rank over a wide range of λ values (approximately $10^{0.3}$ to $10^{3.2}$), which demonstrates its robustness and insensitivity to λ. This wide tolerance range reduces the effort required for parameter tuning. In contrast, Algorithm 2 is more sensitive to λ, achieving correct rank recovery only in a narrow interval, which complicates calibration and compromises reliability. Figure 1(b) shows that Algorithm 1 consistently achieves lower relative errors when it recovers the correct rank, and Figure 1(c) confirms that its runtime is also significantly shorter.


(a) λ vs. Rank of X

(b) λ vs. Relative Error

(c) λ vs. CPU Time

Figure 1. λ versus Relative Error, Rank of X, and CPU Time. Here, $(n, m, d_1, d_2, r^*) = (1000, 1000, 50, 50, 10)$.

Figure 2 further compares the average computational time across various problem sizes. Algorithm 1 maintains superior efficiency, especially as the dimension of the problem increases.

The experimental results highlight the advantages of Algorithm 1 in both accuracy and efficiency. It can reliably find low-rank solutions with small relative error over a broad range of λ, and requires minimal parameter tuning to achieve good performance with a shorter runtime.

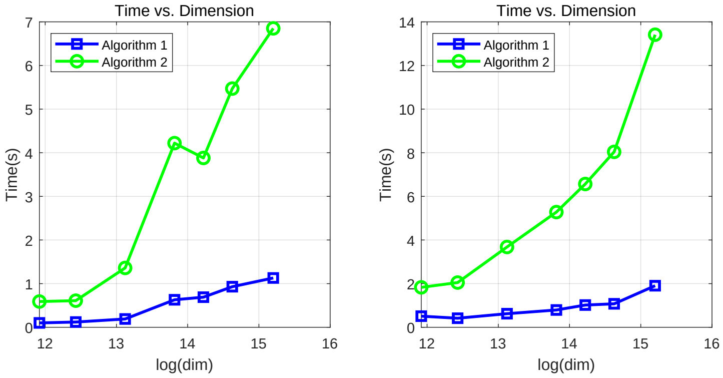


Figure 2. CPU Time versus Dimensions. Left: $r^* = 10$, right: $r^* = 20$.

5.3. *Synthetic data experiments*

In this subsection, we compare the proposed Algorithm 1 against Algorithm 2, the GNIMC algorithm [24], and the SGIMC algorithm. To thoroughly evaluate their performance, we generate synthetic matrix completion problems of varying dimensions with $n \in \{1000, 2000, 3000\}$, $m \in \{1000, 2000, 3000, 5000\}$, and true rank $r^* \in \{10, 20\}$. All instances are randomly generated using the same procedure described in Subsection 5.1. Results are reported across different sampling ratios in $\{0.1, 0.15, 0.2\}$, with detailed outcomes presented in Tables 1–3.

For Algorithms 1 and 2, we report the optimal relative error, the selected λ values, the recovered rank, and the computational time. For GNIMC, which lacks rank adaptation ability, we report the relative error and runtime under the assumption that the true rank is known (otherwise the algorithm fails to converge). The initial points for our algorithms are generated according to the method in Subsection 5.1, while GNIMC's and SGIMC's initializations follow their default procedures.

The experimental results demonstrate that Algorithm 1 can consistently deliver superior performance. It achieves the lowest relative error and shortest computational time across all tested scenarios. Notably, Algorithm 1 is robust to the choice of the regularization parameter λ; it requires no manual tuning or prior knowledge of the true rank. In most cases, high-quality solutions with correct rank recovery are achieved with $\lambda \leq 10^3$. Therefore, we report in Tables 1–3 the smallest value of λ that successfully recovers the true rank.

In comparison, Algorithm 2 is sensitive to the choice of λ. Poorly chosen parameters lead to large relative errors and longer runtimes. The GNIMC algorithm demonstrates good performance only when the true rank is explicitly provided. Without this information, it either fails to converge or yields suboptimal solutions. Moreover, GNIMC's computational and memory requirements grow rapidly with problem size. In many high-dimensional cases, GNIMC encounters out-of-memory errors (as indicated by "-" in Tables 1–3). The SGIMC algorithm adopts an ADMM-based optimization framework, which is inherently more expensive. As a result, SGIMC incurs substantially higher computational cost. In our experiments, since SGIMC was consistently slower and less scalable, we only report the results with runtimes less than 5000 seconds.

Table 1. Random examples with SR = 0.10.

nc	nr	r^*	Algorithm 1				Algorithm 2				GNIMC		SGIMC			
			λ	rank(X)	relerr	time	λ	rank(X)	relerr	time	relerr	time	λ	rank(X)	relerr	time
1000	1000	10	13.895	10	0.013	0.59	0.189	50	0.015	18.73	0.013	5.27	0.100	50	0.014	683.22
1000	1000	20	71.969	20	0.027	1.20	0.672	100	0.025	3.81	0.027	22.66	0.100	100	0.029	1411.44
1000	2000	10	13.895	10	0.009	1.04	0.137	50	0.012	53.25	0.171	32.18	0.100	50	0.010	1237.43
1000	2000	20	51.795	20	0.019	1.65	0.489	100	0.017	16.57	0.019	179.93	0.100	100	0.020	1969.05
1000	3000	10	13.895	10	0.008	1.61	0.100	50	0.012	123.73	0.008	16.75	0.100	50	0.008	1884.57
1000	3000	20	51.795	20	0.016	2.44	0.924	100	0.015	31.50	0.015	665.01	0.100	100	0.016	2813.01
2000	2000	10	13.895	10	0.006	1.87	0.924	50	0.037	80.46	-	-	0.100	50	0.007	2213.77
2000	2000	20	26.827	20	0.013	2.79	0.259	100	0.019	97.77	-	-	0.100	100	0.014	3283.95
2000	3000	10	7.197	10	0.005	2.84	6.210	10	0.087	103.29	-	-	0.100	50	0.006	3250.41
2000	3000	20	26.827	20	0.011	4.03	0.100	100	0.015	118.25	-	-	0.100	100	0.011	4650.86
3000	3000	10	78.805	10	0.003	11.49	41.753	10	0.190	148.89	-	-	-	-	-	-
3000	3000	20	204.336	20	0.007	14.57	0.137	100	0.020	172.19	-	-	-		-	-
3000	5000	10	78.805	10	0.003	18.37	78.805	10	0.263	251.65	-	-	-		-	-
3000	5000	20	204.336	20	0.005	24.02	0.672	100	0.027	297.92	-	-	-		-	-
5000	5000	10	78.805	10	0.002	30.31	108.264	10	0.285	635.92	-	-	-		-	-
5000	5000	20	126.896	20	0.004	39.51	11.721	20	0.095	719.70	-	-	-		-	-

Note:

- λ: The regularization parameter that yields the lowest relative error for each algorithm.
- rankX: The estimated rank of matrix X corresponding to the optimal λ.
- time: The computational time (in seconds) required by the algorithm.
- relerr: The lowest relative error between the estimated result and the ground truth.

The results shown in the table represent the average of 5 independent runs with different initial points. The dimensions of matrix X are set as $(d_1, d_2) = (5r^*, 5r^*)$. To evaluate sensitivity to the regularization parameter λ, we use the `logspace` command to generate 50 logarithmically spaced values from 10^{-1} to 10^4.

Table 2. Random examples with SR = 0.15.

nc	nr	r^*	Algorithm 1				Algorithm 2				GNIMC		SGIMC			
			λ	rank(X)	relerr	time	λ	rank(X)	relerr	time	relerr	time	λ	rank(X)	relerr	time
1000	1000	10	78.805	10	0.008	2.06	57.362	10	0.482	19.81	0.014	29.30	0.100	50	2.569	1604.93
1000	1000	20	529.832	20	0.017	1.91	22.122	20	0.341	27.39	0.029	114.07	0.100	100	2.505	5651.60
1000	2000	10	78.805	10	0.006	2.41	78.805	10	0.484	51.36	0.010	21.14	0.100	50	3.145	3845.07
1000	2000	20	329.034	20	0.011	3.48	30.392	20	0.322	57.97	0.020	808.15	0.100	100	4.483	6917.13
1000	3000	10	48.939	10	0.004	3.62	108.264	10	0.546	72.00	0.008	18.08	0.100	50	4.549	5539.41
1000	3000	20	204.336	20	0.009	4.48	41.753	20	0.387	77.42	-	-	0.100	100	5.754	9671.27
2000	2000	10	48.939	10	0.004	4.33	108.264	10	0.454	92.38	-	-	0.100	50	5.441	5481.21
2000	2000	20	126.896	20	0.008	5.88	78.805	19	0.481	103.17	-	-	-		-	-
2000	3000	10	48.939	10	0.003	6.79	148.735	9	0.513	144.72	-	-	-		-	-
2000	3000	20	126.896	20	0.006	8.25	78.805	20	0.426	155.44	-	-	-		-	-
3000	3000	10	30.392	10	0.003	9.70	204.336	10	0.560	118.30	-	-	-		-	-
3000	3000	20	126.896	20	0.005	11.78	108.264	20	0.456	247.31	-	-	-		-	-
3000	5000	10	30.392	10	0.002	15.75	280.722	10	0.578	15.51	-	-	-		-	-
3000	5000	20	78.805	20	0.004	18.71	148.735	20	0.470	404.37	-	-	-		-	-
5000	5000	10	30.392	10	0.002	25.25	280.722	10	0.500	636.28	-	-	-		-	-
5000	5000	20	78.805	20	0.003	29.55	204.336	19.5	0.491	669.52	-	-	-		-	-

Note:

- λ: The regularization parameter that yields the lowest relative error for each algorithm.
- rankX: The estimated rank of matrix X corresponding to the optimal λ.
- time: The computational time (in seconds) required by the algorithm.
- relerr: The lowest relative error between the estimated result and the ground truth.

The results shown in the table represent the average of 5 independent runs with different initial points. The dimensions of matrix X are set as $(d_1, d_2) = (5r^*, 5r^*)$. To evaluate sensitivity to the regularization parameter λ, we use the `logspace` command to generate 50 logarithmically spaced values from 10^{-1} to 10^4.

Table 3. Random examples with SR = 0.20.

nc	nr	r^*	Algorithm 1				Algorithm 2				GNIMC		SGIMC			
			λ	rank(X)	relerr	time	λ	rank(X)	relerr	time	relerr	time	λ	rank(X)	relerr	time
1000	1000	10	4.520	10	0.004	1.26	78.805	9	0.575	34.90	0.008	13.68	0.100	50	1.479	1978.92
1000	1000	20	18.874	20	0.008	1.92	57.362	18	0.594	32.69	0.017	602.86	0.100	100	2.377	2859.85
1000	2000	10	4.520	10	0.003	2.85	148.735	9	0.696	5.01	0.006	30.00	0.100	50	3.326	3560.12
1000	2000	20	18.874	20	0.006	3.90	78.805	18	0.568	113.28	-	-	-		-	-
1000	3000	10	4.520	10	0.002	2.73	148.735	8	0.601	156.54	-	-	-		-	-
1000	3000	20	11.721	20	0.005	3.49	108.264	16	0.614	179.08			-		-	-
2000	2000	10	4.520	10	0.002	5.90	148.735	9	0.543	226.34	-	-	-		-	-
2000	2000	20	11.721	20	0.004	4.71	148.735	15	0.692	138.26	-	-	-		-	-
2000	3000	10	2.807	10	0.002	5.54	280.722	9	0.742	12.41	-	-	-		-	-
2000	3000	20	11.721	20	0.003	7.26	148.735	17	0.600	411.19	-	-	-		-	-
3000	3000	10	30.392	10	0.002	10.29	280.722	9	0.644	305.75	-	-	-		-	-
3000	3000	20	78.805	20	0.004	13.30	204.336	17	0.655	334.70	-	-	-		-	-
3000	5000	10	18.874	10	0.002	17.53	385.662	10	0.663	40.20	-	-	-		-	-
3000	5000	20	78.805	20	0.004	22.37	280.722	19	0.686	46.57	-	-	-		-	-
5000	5000	10	18.874	10	0.001	17.72	529.832	8	0.697	31.73	-	-	-		-	-
5000	5000	20	48.939	20	0.003	22.05	280.722	17	0.567	957.68	-	-	-		-	-

Note:

- λ: The regularization parameter that yields the lowest relative error for each algorithm.
- rankX: The estimated rank of matrix X corresponding to the optimal λ.
- time: The computational time (in seconds) required by the algorithm.
- relerr: The lowest relative error between the estimated result and the ground truth.

The results shown in the table represent the average of 5 independent runs with different initial points. The dimensions of matrix X are set as $(d_1, d_2) = (5r^*, 5r^*)$. To evaluate sensitivity to the regularization parameter λ, we use the `logspace` command to generate 50 logarithmically spaced values from 10^{-1} to 10^4.

5.4. MR image recovery experiments

Recovering images from incomplete and noisy measurements is a fundamental application of low-rank matrix completion. To assess the practical performance of our proposed method, we conducted experiments using magnetic resonance (MR) brain images from the AANLIB dataset (Harvard Whole Brain Atlas) [7]. Each MR image is represented as a 256×256 grayscale matrix M, and the objective is to recover the original image from randomly undersampled observations. The sampling rates range from 10% to 25%. To incorporate side information, we follow the procedure described in [21]. Specifically, we apply truncated singular value decomposition (SVD) to the fully observed image matrix M using the `lansvd` function:

$$[A, s, B] = \texttt{lansvd}(M, r),$$

where $r = 50$ is the assumed underlying rank. The matrices $A \in \mathbb{R}^{256\times r}$ and $B \in \mathbb{R}^{256\times r}$ are treated as approximate row and column subspaces of M. To simulate noise in the side information, we perturb the matrix B by adding Gaussian noise. The noisy matrix is generated as

$$\widetilde{B} = B + \eta \cdot G,$$

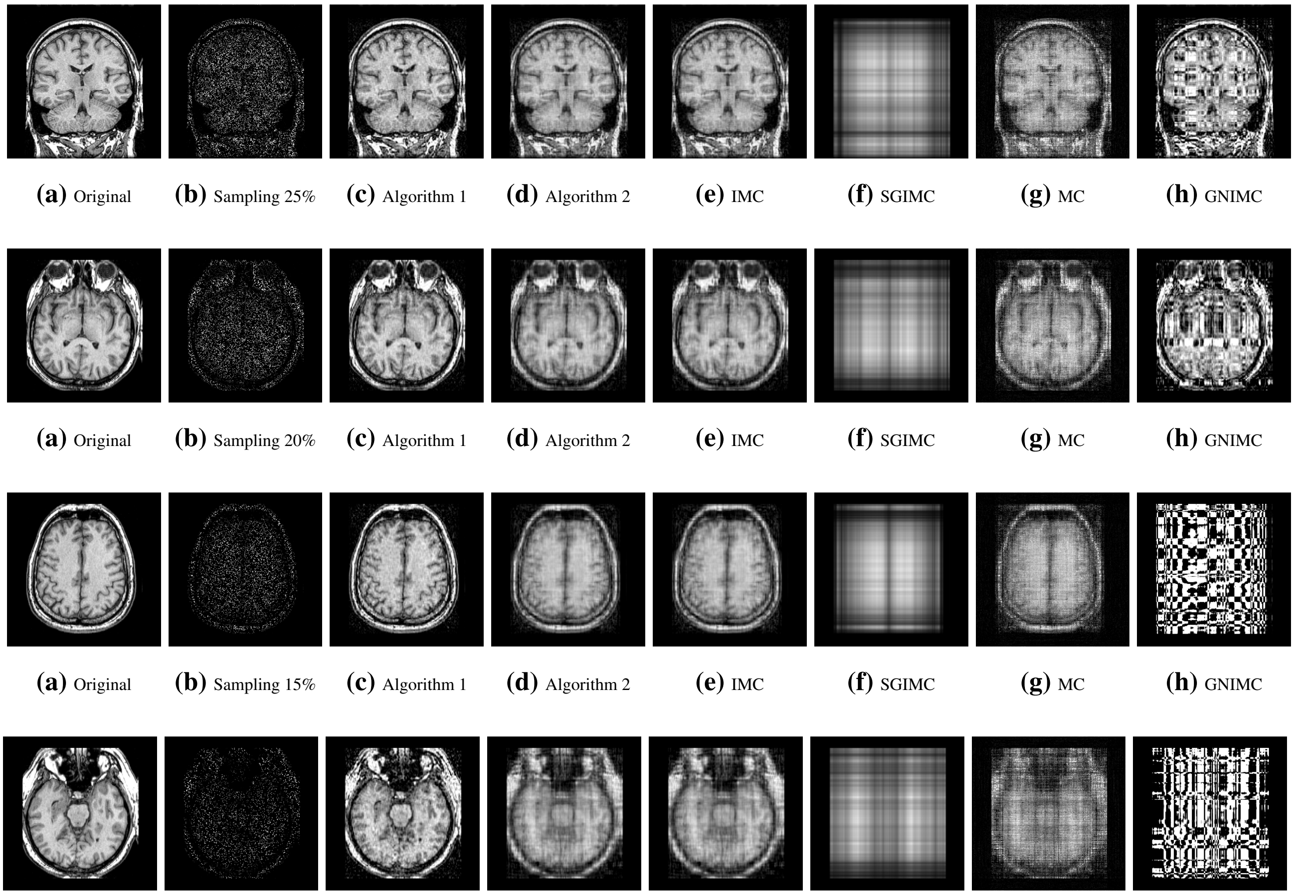
where $G \in \mathbb{R}^{256\times r}$ is a noise matrix that follows the generation procedure described in Section 5.1, and $\eta = 0.1$ is a scaling parameter that controls the noise level. In this experiment, we take $\lambda = 0.5$ for all methods.

Table 4. Experimental results of MR image recovery.

Algorithm	Sampling 10% Relative Error	 PSNR	 Execution Time (s)	Sampling 15% Relative Error	 PSNR	 Execution Time (s)
Algorithm 1	0.190	21.250	0.178	0.125	25.513	0.466
Algorithm 2	0.280	17.890	0.726	0.220	20.633	2.706
IMC	0.280	17.883	0.125	0.221	20.612	0.465
SGIMC	0.490	13.037	62.958	0.390	15.657	106.449
MC	0.457	13.632	17.487	0.346	16.702	19.158
GNIMC	8.217	-11.462	30.882	12.042	-14.130	68.242
	Sampling 20%			Sampling 25%		
Algorithm	Relative Error	PSNR	Execution Time (s)	Relative Error	PSNR	Execution Time (s)
Algorithm 1	0.106	26.653	0.059	0.128	24.566	0.045
Algorithm 2	0.189	21.641	0.529	0.186	21.363	0.574
IMC	0.189	21.623	0.049	0.185	21.401	0.048
SGIMC	0.430	14.479	58.913	0.442	13.834	79.299
MC	0.322	16.995	6.135	0.327	16.440	5.938
GNIMC	0.498	13.213	38.304	0.393	14.861	49.793

The reconstruction results under different sampling rates are displayed in Figure 3. Each row corresponds to a sampling rate of 25%, 20%, 15%, and 10%. For each setting, Figure 3(a) shows the original MR image, Figure 3(b) shows the undersampled observations, and Figures 3(c)–3(h) display the reconstructed outputs from Algorithm 1, Algorithm 2, IMC [20], SGIMC [13], MC [3], and GNIMC [24], respectively.

As shown in Table 4 and Figure 3, our method consistently achieves the lowest relative error, the highest PSNR, and the shortest runtime among all algorithms tested. While IMC requires less computational time, its reconstruction quality degrades significantly in the presence of noise. The SGIMC method, which relies on ADMM, is computationally expensive and offers no clear advantage in accuracy. GNIMC performs poorly when noise is present. This is primarily because GNIMC assumes access to the true rank, which is unavailable or unreliable in noisy scenarios.



(a) Original **(b)** Sampling 25% **(c)** Algorithm 1 **(d)** Algorithm 2 **(e)** IMC **(f)** SGIMC **(g)** MC **(h)** GNIMC

(a) Original **(b)** Sampling 20% **(c)** Algorithm 1 **(d)** Algorithm 2 **(e)** IMC **(f)** SGIMC **(g)** MC **(h)** GNIMC

(a) Original **(b)** Sampling 15% **(c)** Algorithm 1 **(d)** Algorithm 2 **(e)** IMC **(f)** SGIMC **(g)** MC **(h)** GNIMC

(a) Original **(b)** Sampling 10% **(c)** Algorithm 1 **(d)** Algorithm 2 **(e)** IMC **(f)** SGIMC **(g)** MC **(h)** GNIMC

Figure 3. Experimental results of MR image recovery.

6. Conclusions

In this paper, we concerned with the large-scale inductive matrix completion problem with observation noise. We proposed a $\ell_{2,0}$-norm regularization matrix factorization model to solve this problem and established the error bound to the true matrix for the stationary points. According to the results of the numerical experiments, we concluded that $\ell_{2,0}$-norm regularization has better performance and less computing time. Moreover, Algorithm 1 can efficiently handle large-scale problems. Future work will address scenarios with simultaneous noise corruption in both observations and the side-information matrix.

Author contributions

Ting Tao: Conceptualization, Methodology, Writing–original draft; Lianghai Xiao: Software, Investigation, Formal analysis, Visualization; Zetian Chen and Xijie Lin: Data curation, Validation, Writing–review & editing.

Use of Generative-AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

Ting Tao was supported by the Guangdong Basic and Applied Basic Research Foundation No. 2023A1515111167 and Lianghai Xiao was supported by the Guangdong Basic and Applied Basic Research Foundation No. 2022A1515110959.

Conflict of interest

The authors declare that they have no conflict of interest.

References

1. H. Attouch, J. Bolte, P. Redont, A. Soubeyran, Proximal alternating minimization and projection methods for nonconvex problems: an approach based on the Kerdyka-Łojasiewicz inequality, *Math. Oper. Res.*, **35** (2010), 438–457. https://doi.org/10.1287/moor.1100.0449
2. S. Burer, R. D. C. Monteiro, A nonlinear programming algorithm for solving semidefinite programs with low-rank factorization, *Math. Program.*, **95** (2003), 329–357. https://doi.org/10.1007/s10107-002-0352-8
3. E. J. Candès, B. Recht, Exact matrix completion via convex optimization, *Found. Comput. Math.*, **9** (2009), 717–772. https://doi.org/10.1007/s10208-009-9045-5
4. X. Chen, L. Wang, J. Qu, N.-N. Guan, J.-Q. Li, Predicting miRNA-disease association based on inductive matrix completion, *Bioinformatics*, **34** (2018), 4256–4265. https://doi.org/10.1093/bioinformatics/bty503
5. K.-Y. Chiang, I. S. Dhillon, C.-J. Hsieh, Using side information to reliably learn low-rank matrices from missing and corrupted observations, *J. Mach. Learn. Res.*, **19** (2018), 1–35.
6. J. Hannon, M. Bennett, B. Smyth, Recommending twitter users to follow using content and collaborative filtering approaches, In: *RecSys '10: Proceedings of the fourth ACM conference on Recommender systems*, 2010, 199–206. https://doi.org/10.1145/1864708.1864746
7. K. A. Johnson, J. A. Becker, The whole brain atlas, In: neural information processing systems, **26** 2013. Accessed on: July, 2019. `https://www.med.harvard.edu/aanlib/home.html`.

8. Y. Koren, R. Bell, C. Volinsky, Matrix factorization techniques for recommender systems, *Computer*, **42** (2009), 30–37. https://doi.org/10.1109/MC.2009.263

9. A. S. Lewis, H. S. Sendov, Nonsmooth analysis of singular values. Part II: applications, *Set-Valued Anal.*, **13** (2005), 243–264. https://doi.org/10.1007/s11228-004-7198-6

10. C. Q. Lu, M. Y. Yang, F. Luo, F. X. Wu, M. Li, Y. Pan, et al., Prediction of lncRNA-disease associations based on inductive matrix completion, *Bioinformatics*, **34** (2018), 3357–3364. https://doi.org/10.1093/bioinformatics/bty327

11. A. K. Menon, C. Elkan, Link prediction via matrix factorization, In: *Machine learning and knowledge discovery in databases*, Berlin, Heidelberg: Springer, 2011, 437–452. https://doi.org/10.1007/978-3-642-23783-6_28

12. N. Natarajan, I. S. Dhillon, Inductive matrix completion for predicting gene-disease associations, *Bioinformatics*, **30** (2014), i60–i68. https://doi.org/10.1093/bioinformatics/btu269

13. I. Nazaro, B. Shirokik, M. Burkin, G. Fedoni, M. Pano, Sparse group inductive matrix completion, arXiv:1804.10653, 2018. https://doi.org/10.48550/arXiv.1804.10653

14. S. Negahban, M. J. Wainwright, Restricted strong convexity and weighted matrix completion: Optimal bounds with noise, *J. Mach. Learn. Res.*, **13** (2012), 1665–1697.

15. J. Nocedal, S. J. Wright, *Numerical optimization*, 2 Eds., New York: Springer, 2006. https://doi.org/10.1007/978-0-387-40065-5

16. R. T. Rockafellar, R. J.-B. Wets, *Variational analysis*, 1 Eds., Berlin: Springer, 1998. http://doi.org/10.1007/978-3-642-02431-3

17. T. Tao, S. H. Pan, S. J. Bi, Error bound of critical points and KL property of exponent 1/2 for squared F-norm regularized factorization, *J. Glob. Optim.*, **81** (2021), 991–1017. https://doi.org/10.1007/s10898-021-01077-0

18. T. Tao, Y. T. Qian, S. H. Pan, Column $\ell_{2,0}$-norm regularized factorization model of low-rank matrix recovery and its computation, *SIAM J. Optim.*, **32** (2022), 959–988. https://doi.org/10.1137/20M136205X

19. T. Tao, Y. T. Qian, S. H. Pan, Convergence of the majorized PAM method with subspace correction for low-rank composite factorization model, *Comput. Optim. Appl.*, **91** (2025), 1–37. https://doi.org/10.1007/s10589-025-00708-6

20. M. Xu, R. Jin, Z. H. Zhou, Speedup matrix completion with side information: application to multi-label learning, In: *NIPS'13: Proceedings of the 27th international conference on neural information processing systems*, **2**(2013), 2301–2309.

21. K. F. Yi, H. W. Hu, Y. Yu, W. Hao, Regularized matrix completion with partial side information, *Neurocomputing*, **383** (2020), 151–164. https://doi.org/10.1016/j.neucom.2019.12.021

22. X. Zhang, S. S. Du, Q. Q. Gu, Fast and sample efficient inductive matrix completion via multi-phase procrustes flow, In: *Proceedings of the 35th international conference on machine learning*, 10–15 July 2018, Stockholmsmässan, Stockholm, Sweden, 5756–5765.

23. Z. H. Zhu, Q. W. Li, G. G. Tang, M. B. Wakin, The global optimization geometry in low-rank matrix optimization, *IEEE Trans. Inform. Theory*, **67** (2021), 1308–1331. https://doi.org/10.1109/TIT.2021.3049171

24. P. Zilber, B. Nadler, Inductive matrix completion: no bad local minima and a fast algorithm, In: *Proceedings of the 39th international conference on machine learning*, 17–23 July 2022, Baltimore, Maryland, USA, 27671–27692.