Operations Research

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Recursive Importance Sketching for Rank Constrained Least Squares: Algorithms and High-Order Convergence

Yuetian Luo, Wen Huang, Xudong Li, Anru Zhang

To cite this article:
Yuetian Luo, Wen Huang, Xudong Li, Anru Zhang (2024) Recursive Importance Sketching for Rank Constrained Least Squares: Algorithms and High-Order Convergence. Operations Research 72(1):237-256. https://doi.org/10.1287/opre.2023.2445

https://pubsonline.informs.org/journal/opre

OPERATIONS RESEARCH
Vol. 72, No. 1, January–February 2024, pp. 237–256
ISSN 0030-364X (print), ISSN 1526-5463 (online)

Crosscutting Areas

Recursive Importance Sketching for Rank Constrained Least Squares: Algorithms and High-Order Convergence

Yuetian Luo,[a] Wen Huang,[b] Xudong Li,[c] Anru Zhang[d,*]

[a] Data Science Institute, University of Chicago, Chicago, Illinois 60637; [b] School of Mathematical Sciences, Xiamen University, Xiamen 361005, China; [c] School of Data Science, Fudan University, Shanghai 200433, China; [d] Department of Biostatistics & Bioinformatics, Duke University, Durham, North Carolina 27710

*Corresponding author

Contact: yuetian@uchicago.edu (YL); wen.huang@xmu.edu.cn, https://orcid.org/0000-0001-8324-2416 (WH); lixudong@fudan.edu.cn, https://orcid.org/0000-0002-9275-1965 (XL); anru.zhang@duke.edu, https://orcid.org/0000-0002-8721-5252 (AZ)

Received: March 11, 2021
Revised: July 25, 2022; November 20, 2022
Accepted: December 1, 2022
Published Online in Articles in Advance: May 2, 2023

Area of Review: Machine Learning and Data Science

https://doi.org/10.1287/opre.2023.2445

Abstract. In this paper, we propose a recursive importance sketching algorithm for rank-constrained least squares optimization (RISRO). The key step of RISRO is recursive importance sketching, a new sketching framework based on deterministically designed recursive projections, and it significantly differs from the randomized sketching in the literature. Several existing algorithms in the literature can be reinterpreted under this new sketching framework, and RISRO offers clear advantages over them. RISRO is easy to implement and computationally efficient, and the core procedure in each iteration is to solve a dimension-reduced least squares problem. We establish the local quadratic-linear and quadratic rate of convergence for RISRO under some mild conditions. We also discover a deep connection of RISRO to the Riemannian Gauss–Newton algorithm on fixed rank matrices. The effectiveness of RISRO is demonstrated in two applications in machine learning and statistics: low-rank matrix trace regression and phase retrieval. Simulation studies demonstrate the superior numerical performance of RISRO.

Funding: Y. Luo and A. Zhang were partially supported by the National Science Foundation [Grant CAREER-2203741]. W. Huang was partially supported by the Fundamental Research Funds for the Central Universities [Grant 20720190060] and the National Natural Science Foundation of China [Grant 12001455]. X. Li was partially supported by the National Natural Science Foundation of China [Grants 62141407 and 12271107], the Chenguang Program by the Shanghai Education Development Foundation and Shanghai Municipal Education Commission [Grant 19CG02], and the Shanghai Science and Technology Program [Grant 21JC1400600].

Supplemental Material: The online appendix is available at https://doi.org/10.1287/opre.2023.2445.

Keywords: rank-constrained least squares • sketching • quadratic convergence • Riemannian manifold optimization • low-rank matrix recovery • nonconvex optimization

1. Introduction

The focus of this paper is on the rank-constrained least squares:

$$\min_{\mathbf{X}\in\mathbb{R}^{p_1\times p_2}} f(\mathbf{X}) := \frac{1}{2}\|\mathbf{y}-\mathcal{A}(\mathbf{X})\|_2^2, \quad \text{subject to} \quad \operatorname{rank}(\mathbf{X}) = r. \tag{1}$$

Here, $\mathbf{y}\in\mathbb{R}^n$ is the given response and $\mathcal{A}\in\mathbb{R}^{p_1\times p_2}\to\mathbb{R}^n$ is a known linear map that can be explicitly represented as

$$\begin{aligned}\mathcal{A}(\mathbf{X}) &= [\langle\mathbf{A}_1,\mathbf{X}\rangle,\ldots,\langle\mathbf{A}_n,\mathbf{X}\rangle]^\top, \langle\mathbf{A}_i,\mathbf{X}\rangle \\ &= \sum_{1\leqslant j\leqslant p_1, 1\leqslant k\leqslant p_2} (\mathbf{A}_i)_{[j,k]}\mathbf{X}_{[j,k]}\end{aligned} \tag{2}$$

with given measurement matrices $\mathbf{A}_i\in\mathbb{R}^{p_1\times p_2}$, $i=1,\ldots,n$.

The rank-constrained least square (1) is motivated by the widely studied low-rank matrix recovery problem, in which the goal is to recover a low-rank matrix $\mathbf{X}^*$ from the observation $\mathbf{y}=\mathcal{A}(\mathbf{X}^*)+\epsilon$ (ϵ is the noise). This problem is of fundamental importance in a variety of fields, such as optimization, machine learning, signal processing, scientific computation, and statistics. With different realizations of $\mathcal{A}$, (1) covers many applications, such as matrix trace regression (Candès and Plan 2011, Davenport and Romberg 2016), matrix completion (Keshavan et al. 2009, Candès and Tao 2010, Koltchinskii et al. 2011, Miao et al. 2016), phase retrieval (Candès et al. 2013, Shechtman et al. 2015), blind deconvolution (Ahmed et al. 2013), and matrix recovery via rank-one projections (Cai and Zhang 2015, Chen et al. 2015). To overcome the nonconvexity and NP-hardness of directly solving (1) (Recht et al. 2010), various computational feasible schemes have been developed in the past decade, including the prominent convex relaxation (Recht et al. 2010,

Candès and Plan 2011)

$$\min_{\mathbf{X}\in\mathbb{R}^{p_1\times p_2}} \frac{1}{2}\|\mathbf{y}-\mathcal{A}(\mathbf{X})\|_2^2+\lambda\|\mathbf{X}\|_*, \tag{3}$$

where $\|\mathbf{X}\|_*=\sum_{i=1}^{\min(p_1,p_2)}\sigma_i(\mathbf{X})$ is the nuclear norm of $\mathbf{X}$ and $\lambda>0$ is a tuning parameter. Nevertheless, the convex relaxation technique has one well-documented limitation: the parameter space after relaxation is usually much larger than that of the target problem. Also, algorithms for solving the convex program often require the singular value decomposition as the stepping-stone and can be prohibitively time-consuming for large-scale instances.

In addition, nonconvex optimization, which directly enforces the rank-r constraint on the iterates, renders another important class of algorithms for solving (1). Because each iterate lies in a low-dimensional space, the computation cost of the nonconvex approach can be much smaller than the convex regularized approach. Over the last a few years, there is a flurry of research on nonconvex methods in solving (1) (Wen et al. 2012, Jain et al. 2013, Hardt 2014, Chen and Wainwright 2015, Zhao et al. 2015, Zheng and Lafferty 2015, Miao et al. 2016, Sun and Luo 2016, Tu et al. 2016, Tran-Dinh 2021), and many of the algorithms, such as gradient descent (GD) and alternating minimization (alter mini), are shown to have nice convergence results under proper model assumptions (Jain et al. 2013, Hardt 2014, Zhao et al. 2015, Sun and Luo 2016, Tu et al. 2016, Tong et al. 2021a). We refer readers to Section 1.2 for more review of recent works.

In the existing literature, many algorithms for solving (1) either require careful tuning of hyperparameters or have a convergence rate no faster than linear. Thus, we raise the following question: can we develop an easy-to-compute and efficient (we hope with comparable per-iteration computational complexity as the first order methods) algorithm with provable high-order convergence guarantees (possibly converging to a stationary point because of the nonconvexity) for solving (1)?

In this paper, we give an affirmative answer to this question by making contributions as outlined next.

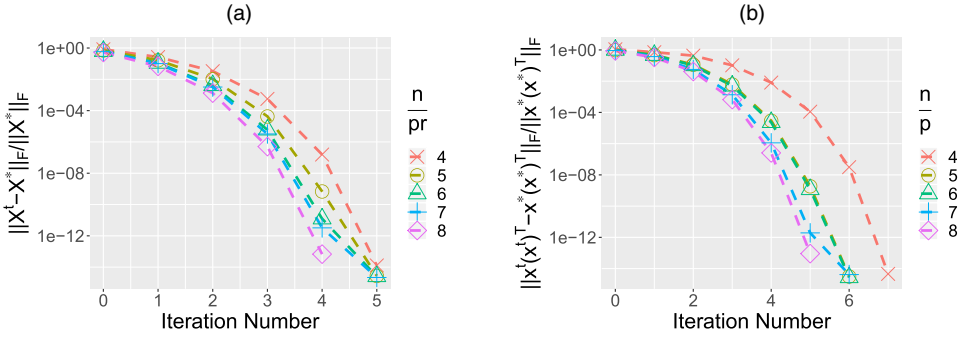
1.1. Our Contributions

We introduce an easy-to-implement and computationally efficient algorithm, recursive importance sketching for rank-constrained least squares optimization (RISRO), for solving (1) in this paper. The proposed algorithm is tuning-free and has the same per-iteration computational complexity as alternating minimization (Jain et al. 2013) as well as comparable complexity to many popular first order methods, such as iterative hard thresholding (Jain et al. 2010) and gradient descent (Tu et al. 2016) when $r\ll p_1,p_2,n$. We then illustrate the key idea of RISRO under a general framework of recursive importance sketching. This framework also renders a platform to compare RISRO and several existing algorithms for rank-constrained least squares.

Assuming $\mathcal{A}$ satisfies the restricted isometry property (RIP), we prove RISRO is local quadratic-linearly convergent in general and quadratically convergent to a stationary point under some extra conditions. Figure 1 provides a numerical example of the performance of RISRO in the noiseless low-rank matrix trace regression (left panel) and phase retrieval (right panel). In both problems, RISRO converges to the underlying parameter quadratically and reaches a highly accurate solution within five iterations.

In addition, we discover a deep connection between RISRO and the Riemannian Gauss–Newton optimization algorithm on fixed rank matrices manifold. The least squares step in RISRO implicitly solves a Fisher scoring or Riemannian Gauss–Newton equation on the Riemannian optimization of low-rank matrices, and the updating rule in RISRO can be seen as a retraction map. With this connection, our theory on RISRO also improves the existing

Figure 1. (Color online) RISRO Achieves a Quadratic Rate of Convergence



Notes. Spectral initialization is used in each setting and more details about the simulation setup are given in Section 7. (a) Noiseless low-rank matrix trace regression. Here, $\mathbf{y}_i=\langle\mathbf{A}_i,\mathbf{X}^*\rangle$ for $1\leqslant i\leqslant n$, $\mathbf{X}^*\in\mathbb{R}^{p\times p}$ with $p=100$, $\sigma_1(\mathbf{X}^*)=\cdots=\sigma_3(\mathbf{X}^*)=3$, $\sigma_k(\mathbf{X}^*)=0$ for $4\leqslant k\leqslant 100$ and $\mathbf{A}_i$ has independently and identically distributed (i.i.d.) standard Gaussian entries. (b) Phase retrieval. Here, $\mathbf{y}_i=\langle\mathbf{a}_i\mathbf{a}_i^\top,\mathbf{x}^*\mathbf{x}^{*\top}\rangle$ for $1\leqslant i\leqslant n$, $\mathbf{x}^*\in\mathbb{R}^p$ with $p=1,200$, $\mathbf{a}_i\overset{i.i.d.}{\sim}N(0,\mathbf{I}_p)$.

convergence results on the Riemannian Gauss–Newton method for the rank-constrained least squares problem.

Next, we further apply RISRO to two prominent problems in machine learning and statistics: low-rank matrix trace regression and phase retrieval. In the noisy low-rank matrix trace regression, we prove the estimation error rate of RISRO converges quadratically to the information-theoretical limit with only a double-logarithmic number of iterations under the Gaussian ensemble design. To the best of our knowledge, RISRO is the first algorithm that provably achieves the minimax rate-optimal estimation error in matrix trace regression with only a double-logarithmic number of iterations, which offers an exponential improvement over the existing results of first order methods (Jain et al. 2010, 2013; Chen and Wainwright 2015). We also discover a new "quadratic + one-iteration optimality" phenomenon for RISRO on low-rank matrix recovery (Remark 12). In phase retrieval, in which $\mathcal{A}$ does not satisfy the RIP condition, we can still establish the local convergence of RISRO given a proper initialization. We also develop RISRO in the matrix completion and robust principal component analysis (PCA) applications, in which the restricted isometry property completely fails. We find RISRO still has similar empirical performance as in the setting in which the RIP condition holds.

Finally, we conduct simulation studies to support our theoretical findings and compare RISRO with many existing algorithms. The numerical results show RISRO not only offers faster and more robust convergence, but also requires a smaller sample size for low-rank matrix recovery compared with existing approaches.

1.2. Related Literature

This work is related to a range of literature on low-rank matrix recovery, convex/nonconvex optimization, and sketching arising from several communities, including optimization, machine learning, statistics, and applied mathematics. We make an attempt to review the related literature without claiming the survey is exhaustive.

One class of the most popular approaches to solve (1) is the nuclear norm minimization (NNM) (3). Many algorithms are proposed to solve NNM, such as proximal gradient descent (Toh and Yun 2010), fixed-point continuation (Goldfarb and Ma 2011), and proximal point methods (Jiang et al. 2014). It is shown that the solution of NNM has desirable properties under proper models, such as matrix trace regression and matrix completion (Recht et al. 2010; Candès and Plan 2011; Cai and Zhang 2013, 2014, 2015). In addition to NNM, the max norm minimization is another widely considered convex realization for the rank-constrained optimization (Lee et al. 2010, Cai and Zhou 2013). However, these convex programs are usually computationally intensive to solve, which motivates a line of work on using nonconvex approaches. Since Burer and Monteiro (2003), one of the most popular nonconvex methods for solving (1) is to first factor the low-rank matrix $\mathbf{X}$ to $\mathbf{RL}^\top$ with two factor matrices $\mathbf{R} \in \mathbb{R}^{p_1 \times r}$, $\mathbf{L} \in \mathbb{R}^{p_2 \times r}$ and then run either gradient descent or alternating minimization on $\mathbf{R}$ and $\mathbf{L}$ (Candès et al. 2015, Zhao et al. 2015, Zheng and Lafferty 2015, Sun and Luo 2016, Tu et al. 2016, Sanghavi et al. 2017, Wang et al. 2017c, Park et al. 2018, Li et al. 2019b, Ma et al. 2019, Tong et al. 2021b). Other methods, such as singular value projection or iterative hard thresholding (Jain et al. 2010, Goldfarb and Ma 2011, Tanner and Wei 2013), Grassmann manifold optimization (Keshavan et al. 2009, Boumal and Absil 2011), and Riemannian manifold optimization (Meyer et al. 2011, Vandereycken 2013, Mishra et al. 2014, Wei et al. 2016, Huang and Hand 2018), have also been proposed and studied. We refer readers to the recent survey paper Chi et al. (2019) for a comprehensive overview of existing literature on convex and nonconvex approaches on solving (1). Most of the convergence analyses in the literature were conducted under certain statistical models (e.g., noisy/noiseless matrix trace regression, matrix completion, and phase retrieval), and the goal was to recover the underlying parameter matrix. Here, we study (1) from both an optimization perspective (how the algorithm converges to a stationary point) and a statistical perspective (how the iterates estimate the underlying true parameter). These two perspectives overlap in the noiseless settings as the parameter becomes a stationary point and is disjoint in the more general noisy settings.

There are a few recent attempts in connecting the geometric structures of different approaches (Li et al. 2019a, Ha et al. 2020), and the landscape of Problem (1) is also studied in various settings (Bhojanapalli et al. 2016, Ge et al. 2017, Zhu et al. 2018, Zhang et al. 2019, Uschmajew and Vandereycken 2020).

Our work is also related to the idea of sketching in numerical linear algebra. Performing sketching to speed up the computation via dimension reduction has been explored extensively in recent years (Mahoney 2011, Woodruff 2014). Sketching methods are applied to solve a number of problems, including but not limited to matrix approximation (Drineas et al. 2012, Zheng et al. 2012, Song et al. 2017), linear regression (Pilanci and Wainwright 2016, Raskutti and Mahoney 2016, Clarkson and Woodruff 2017, Dobriban and Liu 2019), ridge regression (Wang et al. 2017b), etc. In most of the sketching literature, the sketching matrices are randomly constructed (Mahoney 2011, Woodruff 2014). Randomized sketching matrices are easy to generate and require little storage for sparse sketching. However, randomized sketching can be suboptimal in statistical settings (Raskutti and Mahoney 2016). To overcome this, Zhang et al. (2020) introduce an idea of importance sketching in the context of low-rank tensor regression. In contrast to the randomized sketching, importance sketching matrices are constructed deterministically with the supervision of the data and are shown capable of achieving better

statistical efficiency. However, the method developed in Zhang et al. (2020) is essentially a "one-time" importance sketching, which yields a suboptimal outcome when the noise level is small or moderate. This paper proposes a more powerful recursive importance sketching algorithm that iteratively refines the sketching matrices. We also provide a comprehensive convergence analysis for the proposed algorithm without the sample-splitting assumption used in Zhang et al. (2020); our theory demonstrates the optimality of the proposed algorithm at all different noise levels and advantages over other algorithms for the rank-constrained least squares problem.

1.3. Organization of the Paper

The rest of this article is organized as follows. After a brief introduction of notation in Section 1.4, we present our main algorithm RISRO with an interpretation from the recursive importance sketching perspective in Section 2. The theoretical results of RISRO are given in Section 3. In Section 4, we present another interpretation for RISRO from Riemannian manifold optimization. The computational complexity of RISRO and its applications to low-rank matrix trace regression and phase retrieval are discussed in Sections 5 and 6, respectively. Numerical studies of RISRO and the comparison with existing algorithms in the literature are presented in Section 7. Conclusion and future work are given in Section 8.

1.4. Notation

The following notation is used throughout this article. Uppercase and lowercase letters (e.g., A, B, a, b), lowercase boldface letters (e.g., $\mathbf{u}, \mathbf{v}$), and uppercase boldface letters (e.g., $\mathbf{U}, \mathbf{V}$) are used to denote scalars, vectors, and matrices, respectively. For any two series of numbers, say $\{a_n\}$ and $\{b_n\}$, denote $a = O(b)$ if there exists uniform constants $C > 0$ such that $a_n \leqslant Cb_n$, $\forall n$. For any $a, b \in \mathbb{R}$, let $a \wedge b := \min\{a, b\}, a \vee b = \max\{a, b\}$. For any matrix $\mathbf{X} \in \mathbb{R}^{p_1 \times p_2}$ with singular value decomposition $\sum_{i=1}^{p_1 \wedge p_2} \sigma_i(\mathbf{X}) \mathbf{u}_i \mathbf{v}_i^\top$, where $\sigma_1(\mathbf{X}) \geqslant \sigma_2(\mathbf{X}) \geqslant \cdots \geqslant \sigma_{p_1 \wedge p_2}(\mathbf{X})$, let $\mathbf{X}_{\max(r)} = \sum_{i=1}^{r} \sigma_i(\mathbf{X}) \mathbf{u}_i \mathbf{v}_i^\top$ be the best rank-r approximation of $\mathbf{X}$ and denote $\|\mathbf{X}\|_{\mathrm{F}} = \sqrt{\sum_i \sigma_i^2(\mathbf{X})}$ and $\|\mathbf{X}\| = \sigma_1(\mathbf{X})$ as the Frobenius and spectral norm, respectively. Let $\mathrm{QR}(\mathbf{X})$ be the Q part of the QR decomposition outcome of $\mathbf{X}$. We denote $\mathrm{vec}(\mathbf{X}) \in \mathbb{R}^{p_1 p_2}$ as the vectorization of $\mathbf{X}$ by its columns. In addition, $\mathbf{I}_r$ is the r-by-r identity matrix. Let $\mathbb{O}_{p,r} = \{\mathbf{U} : \mathbf{U}^\top \mathbf{U} = \mathbf{I}_r\}$ be the set of all p-by-r matrices with orthonormal columns. For any $\mathbf{U} \in \mathbb{O}_{p,r}$, $P_{\mathbf{U}} = \mathbf{U}\mathbf{U}^\top$ represents the orthogonal projector onto the column space of $\mathbf{U}$; we also note $\mathbf{U}_\perp \in \mathbb{O}_{p,p-r}$ as the orthonormal complement of $\mathbf{U}$. We use bracket subscripts to denote submatrices. For example, $\mathbf{X}_{[i_1,i_2]}$ is the entry of $\mathbf{X}$ on the i_1th row and i_2th column; $\mathbf{X}_{[(r+1):p_1,:]}$ contains the $(r+1)$th to the p_1th rows of $\mathbf{X}$. For any matrix $\mathbf{X}$, we use $\mathbf{X}^\dagger$ to denote its Moore–Penrose inverse. For matrices $\mathbf{U} \in \mathbb{R}^{p_1 \times p_2}, \mathbf{V} \in \mathbb{R}^{m_1 \times m_2}$, let

$$\mathbf{U} \otimes \mathbf{V} = \begin{bmatrix} \mathbf{U}_{[1,1]} \cdot \mathbf{V} & \cdots & \mathbf{U}_{[1,p_2]} \cdot \mathbf{V} \\ \vdots & & \vdots \\ \mathbf{U}_{[p_1,1]} \cdot \mathbf{V} & \cdots & \mathbf{U}_{[p_1,p_2]} \cdot \mathbf{V} \end{bmatrix} \in \mathbb{R}^{(p_1 m_1) \times (p_2 m_2)}$$

be their Kronecker product. Finally, for any given linear operator $\mathcal{L}$, we use $\mathcal{L}^*$ to denote its adjoint and use $\mathrm{Ran}(\mathcal{L})$ to denote its range space.

2. Recursive Importance Sketching for Rank-Constrained Least Squares

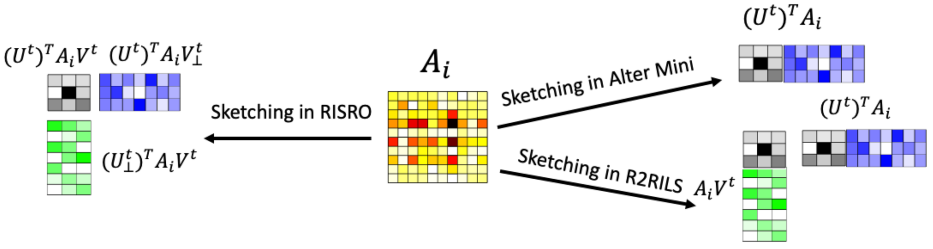
In this section, we discuss the procedure and interpretations of RISRO and then compare it with existing algorithms from a sketching perspective. The pseudocode of RISRO is summarized in Algorithm 1.

2.1. RISRO Procedure and Recursive Importance Sketching

In each iteration $t = 1, 2, \ldots$, RISRO includes three steps:

1. We sketch each $\mathbf{A}_i$ $(i = 1, \ldots, n)$ onto the subspace spanned by $[\mathbf{U}^t \otimes \mathbf{V}^t, \mathbf{U}_\perp^t \otimes \mathbf{V}^t, \mathbf{U}^t \otimes \mathbf{V}_\perp^t]$, where $\mathbf{U}^t$ and $\mathbf{V}^t$ span the column and row subspaces of $\mathbf{X}^t$, respectively. This yields the sketched importance covariates $\mathbf{U}^{t\top} \mathbf{A}_i \mathbf{V}^t, \mathbf{U}_\perp^{t\top} \mathbf{A}_i \mathbf{V}^t, \mathbf{U}^{t\top} \mathbf{A}_i \mathbf{V}_\perp^t$. See Figure 2, left panel,

Figure 2. (Color online) Illustration of Sketching Strategies of RISRO (this Work), Alter Mini (Jain et al. 2013, Hardt 2014), and R2RILS (Bauch et al. 2021)



Notes. Here, $\mathbf{A}_i$ denotes the covariate matrix of the ith observation; $\mathbf{U}^t$ and $\mathbf{V}^t$ span the column and row subspaces of $\mathbf{X}^t$, respectively. The 3*3 core covariate matrices (colored in gray) represent the sketching of $\mathbf{A}_i$ onto the column and row subspaces of $\mathbf{X}^t$, the 7*3 thin covariate matrices (colored in green) represent the sketching of $\mathbf{A}_i$ onto the perpendicular column subspace and row subspace of $\mathbf{X}^t$, and the 3*7 fact covariate matrices (colored in blue) represent the sketching of $\mathbf{A}_i$ onto the column subspace and perpendicular row subspace of $\mathbf{X}^t$. In alter mini and R2RILS, the sketched covariates are combined to represent the actual algorithmic implementation.

for an illustration of the sketching scheme of RISRO. Then, we construct the covariate maps $\mathcal{A}_B : \mathbb{R}^{r\times r} \to \mathbb{R}^n$, $\mathcal{A}_{D_1} : \mathbb{R}^{(p_1-r)\times r} \to \mathbb{R}^n$, and $\mathcal{A}_{D_2} : \mathbb{R}^{r\times(p_2-r)} \to \mathbb{R}^n$: for matrix "$\cdot$," let

$$[\mathcal{A}_B(\cdot)]_i = \langle \cdot, \mathbf{U}^{t\top}\mathbf{A}_i\mathbf{V}^t\rangle, \quad [\mathcal{A}_{D_1}(\cdot)]_i = \langle \cdot, \mathbf{U}_\perp^{t\top}\mathbf{A}_i\mathbf{V}^t\rangle,$$
$$[\mathcal{A}_{D_2}(\cdot)]_i = \langle \cdot, \mathbf{U}^{t\top}\mathbf{A}_i\mathbf{V}_\perp^t\rangle, \quad i = 1,\dots,n. \tag{4}$$

2. We solve a dimension-reduced least squares problem (5) (provided in the box of Algorithm 1) in which the number of parameters is reduced to $(p_1 + p_2 - r)r$, whereas the sample size remains n.

3. We update the sketching matrices $\mathbf{U}^{t+1}, \mathbf{V}^{t+1}$, and $\mathbf{X}^{t+1}$ in steps 6 and 7. By construction, $\mathbf{U}^{t+1}, \mathbf{V}^{t+1}$ contain both the column and row spans of $\mathbf{X}^{t+1}$.

Algorithm 1 (RISRO)

1: Input: $\mathcal{A}(\cdot) : \mathbb{R}^{p_1\times p_2} \to \mathbb{R}^n$, $\mathbf{y} \in \mathbb{R}^n$, rank r, and initialization $\mathbf{X}^0$, which admits singular value decomposition $\mathbf{U}^0\mathbf{\Sigma}^0\mathbf{V}^{0\top}$, where $\mathbf{U}^0 \in \mathbb{O}_{p_1,r}, \mathbf{V}^0 \in \mathbb{O}_{p_2,r}, \mathbf{\Sigma}^0 \in \mathbb{R}^{r\times r}$.

2: **for** $t = 0, 1, \dots$ **do**

3: Perform importance sketching on $\mathcal{A}$ and construct the covariate maps $\mathcal{A}_B : \mathbb{R}^{r\times r} \to \mathbb{R}^n$, $\mathcal{A}_{D_1} : \mathbb{R}^{(p_1-r)\times r} \to \mathbb{R}^n$, and $\mathcal{A}_{D_2} : \mathbb{R}^{r\times(p_2-r)} \to \mathbb{R}^n$: for matrix "$\cdot$," let

$$[\mathcal{A}_B(\cdot)]_i = \langle \cdot, \mathbf{U}^{t\top}\mathbf{A}_i\mathbf{V}^t\rangle,$$
$$[\mathcal{A}_{D_1}(\cdot)]_i = \langle \cdot, \mathbf{U}_\perp^{t\top}\mathbf{A}_i\mathbf{V}^t\rangle,$$
$$[\mathcal{A}_{D_2}(\cdot)]_i = \langle \cdot, \mathbf{U}^{t\top}\mathbf{A}_i\mathbf{V}_\perp^t\rangle, \quad i = 1,\dots,n.$$

4: Solve the unconstrained least squares problem

$$(\mathbf{B}^{t+1}, \mathbf{D}_1^{t+1}, \mathbf{D}_2^{t+1}) = \underset{\mathbf{B}\in\mathbb{R}^{r\times r},\, \mathbf{D}_i\in\mathbb{R}^{(p_i-r)\times r},\, i=1,2}{\arg\min} \|\mathbf{y} - \mathcal{A}_B(\mathbf{B}) - \mathcal{A}_{D_1}(\mathbf{D}_1) - \mathcal{A}_{D_2}(\mathbf{D}_2^\top)\|_2^2. \tag{5}$$

5: Compute $\mathbf{X}_U^{t+1} = (\mathbf{U}^t\mathbf{B}^{t+1} + \mathbf{U}_\perp^t\mathbf{D}_1^{t+1})$ and $\mathbf{X}_V^{t+1} = (\mathbf{V}^t\mathbf{B}^{t+1\top} + \mathbf{V}_\perp^t\mathbf{D}_2^{t+1})$.

6: Perform QR orthogonalization: $\mathbf{U}^{t+1} = \mathrm{QR}(\mathbf{X}_U^{t+1})$, $\mathbf{V}^{t+1} = \mathrm{QR}(\mathbf{X}_V^{t+1})$.

7: Update $\mathbf{X}^{t+1} = \mathbf{X}_U^{t+1}(\mathbf{B}^{t+1})^\dagger\mathbf{X}_V^{t+1\top}$.

8: **end for**

We give a high-level explanation of RISRO through a decomposition of $\mathbf{y}_i$. Suppose $\mathbf{y}_i = \langle \mathbf{A}_i, \bar{\mathbf{X}}\rangle + \bar{\boldsymbol{\epsilon}}_i$, where $\bar{\mathbf{X}}$ is a rank-r target matrix with singular value decomposition $\bar{\mathbf{U}}\bar{\mathbf{\Sigma}}\bar{\mathbf{V}}^\top$ with $\bar{\mathbf{U}} \in \mathbb{O}_{p_1,r}$, $\bar{\mathbf{\Sigma}} \in \mathbb{R}^{r\times r}$, and $\bar{\mathbf{V}} \in \mathbb{O}_{p_2,r}$. Then,

$$\begin{aligned}\mathbf{y}_i &= \langle \mathbf{U}^{t\top}\mathbf{A}_i\mathbf{V}^t, \mathbf{U}^{t\top}\bar{\mathbf{X}}\mathbf{V}^t\rangle + \langle \mathbf{U}_\perp^{t\top}\mathbf{A}_i\mathbf{V}^t, \mathbf{U}_\perp^{t\top}\bar{\mathbf{X}}\mathbf{V}^t\rangle \\ &\quad + \langle \mathbf{U}^{t\top}\mathbf{A}_i\mathbf{V}_\perp^t, \mathbf{U}^{t\top}\bar{\mathbf{X}}\mathbf{V}_\perp^t\rangle + \langle \mathbf{U}_\perp^{t\top}\mathbf{A}_i\mathbf{V}_\perp^t, \mathbf{U}_\perp^{t\top}\bar{\mathbf{X}}\mathbf{V}_\perp^t\rangle + \bar{\boldsymbol{\epsilon}}_i \\ &:= \langle \mathbf{U}^{t\top}\mathbf{A}_i\mathbf{V}^t, \mathbf{U}^{t\top}\bar{\mathbf{X}}\mathbf{V}^t\rangle + \langle \mathbf{U}_\perp^{t\top}\mathbf{A}_i\mathbf{V}^t, \mathbf{U}_\perp^{t\top}\bar{\mathbf{X}}\mathbf{V}^t\rangle \\ &\quad + \langle \mathbf{U}^{t\top}\mathbf{A}_i\mathbf{V}_\perp^t, \mathbf{U}^{t\top}\bar{\mathbf{X}}\mathbf{V}_\perp^t\rangle + \boldsymbol{\epsilon}_i^t. \end{aligned} \tag{6}$$

Here, $\boldsymbol{\epsilon}^t := \mathcal{A}(P_{\mathbf{U}_\perp^t}\bar{\mathbf{X}}P_{\mathbf{V}_\perp^t}) + \bar{\boldsymbol{\epsilon}} \in \mathbb{R}^n$ can be seen as the residual of the new regression model (6), and $\mathbf{U}^{t\top}\mathbf{A}_i\mathbf{V}^t, \mathbf{U}_\perp^{t\top}\mathbf{A}_i\mathbf{V}^t, \mathbf{U}^{t\top}\mathbf{A}_i\mathbf{V}_\perp^t$ are exactly the importance covariates constructed in (4). Let

$$\tilde{\mathbf{B}}^t := \mathbf{U}^{t\top}\bar{\mathbf{X}}\mathbf{V}^t, \tilde{\mathbf{D}}_1^t := \mathbf{U}_\perp^{t\top}\bar{\mathbf{X}}\mathbf{V}^t, \tilde{\mathbf{D}}_2^{t\top} := \mathbf{U}^{t\top}\bar{\mathbf{X}}\mathbf{V}_\perp^t. \tag{7}$$

If $\boldsymbol{\epsilon}^t = 0$, we have that $(\tilde{\mathbf{B}}^t, \tilde{\mathbf{D}}_1^t, \tilde{\mathbf{D}}_2^t)$ is a solution of the least squares in (5). Hence, we could set $\mathbf{B}^{t+1} = \tilde{\mathbf{B}}^t, \mathbf{D}_1^{t+1} = \tilde{\mathbf{D}}_1^t, \mathbf{D}_2^{t+1} = \tilde{\mathbf{D}}_2^t$, and thus, $\mathbf{X}_U^{t+1} = \bar{\mathbf{X}}\mathbf{V}^t, \mathbf{X}_V^{t+1} = \bar{\mathbf{X}}^\top\mathbf{U}^t$. Furthermore, if $\mathbf{B}^{t+1}$ is invertible, then it holds that

$$\mathbf{X}_U^{t+1}(\mathbf{B}^{t+1})^{-1}\mathbf{X}_V^{t+1\top} = \bar{\mathbf{X}}\mathbf{V}^t(\mathbf{U}^{t\top}\bar{\mathbf{X}}\mathbf{V}^t)^{-1}(\bar{\mathbf{X}}^\top\mathbf{U}^t)^\top = \bar{\mathbf{X}}, \tag{8}$$

which means $\bar{\mathbf{X}}$ can be exactly recovered by one iteration of RISRO.

In general, $\boldsymbol{\epsilon}^t \neq 0$. When the column spans of $\mathbf{U}^t, \mathbf{V}^t$ well-approximate the ones of $\bar{\mathbf{U}}, \bar{\mathbf{V}}$, that is, the column and row subspaces on which the target parameter $\bar{\mathbf{X}}$ lies, we expect $\mathbf{U}_\perp^{t\top}\bar{\mathbf{X}}\mathbf{V}_\perp^t$ and $\boldsymbol{\epsilon}_i^t = \langle \mathbf{U}_\perp^{t\top}\mathbf{A}_i\mathbf{V}_\perp^t, \mathbf{U}_\perp^{t\top}\bar{\mathbf{X}}\mathbf{V}_\perp^t\rangle + \bar{\boldsymbol{\epsilon}}_i$ to have a small amplitude; then, $\mathbf{B}^{t+1}, \mathbf{D}_1^{t+1}, \mathbf{D}_2^{t+1}$, the outcome of the least squares problem (5), can well-approximate $\tilde{\mathbf{B}}^t, \tilde{\mathbf{D}}_1^t, \tilde{\mathbf{D}}_2^t$. In Lemma 1, we give a precise characterization for this approximation. Before that, let us introduce a convenient notation so that (5) can be written in a more compact way.

Define the linear operator $\mathcal{L}_t$ as

$$\mathcal{L}_t : \mathbf{W} = \begin{bmatrix} \mathbf{W}_0 \in \mathbb{R}^{r\times r} & \mathbf{W}_2 \in \mathbb{R}^{r\times(p_2-r)} \\ \mathbf{W}_1 \in \mathbb{R}^{(p_1-r)\times r} & \mathbf{0}_{(p_1-r)\times(p_2-r)} \end{bmatrix} \to [\mathbf{U}^t \quad \mathbf{U}_\perp^t]\begin{bmatrix} \mathbf{W}_0 & \mathbf{W}_2 \\ \mathbf{W}_1 & \mathbf{0} \end{bmatrix}[\mathbf{V}^t \quad \mathbf{V}_\perp^t]^\top, \tag{9}$$

and it is easy to compute its adjoint $\mathcal{L}_t^* : \mathbf{M} \in \mathbb{R}^{p_1\times p_2} \to \begin{bmatrix} \mathbf{U}^{t\top}\mathbf{M}\mathbf{V}^t & \mathbf{U}^{t\top}\mathbf{M}\mathbf{V}_\perp^t \\ (\mathbf{U}_\perp^t)^\top\mathbf{M}\mathbf{V}^t & \mathbf{0} \end{bmatrix}$. Then, the least squares problem in (5) can be written as

$$(\mathbf{B}^{t+1}, \mathbf{D}_1^{t+1}, \mathbf{D}_2^{t+1}) = \underset{\mathbf{B}\in\mathbb{R}^{r\times r}, \mathbf{D}_i\in\mathbb{R}^{(p_i-r)\times r}, i=1,2}{\arg\min} \left\| \mathbf{y} - \mathcal{A}\mathcal{L}_t\left(\begin{bmatrix} \mathbf{B} & \mathbf{D}_2^\top \\ \mathbf{D}_1 & \mathbf{0} \end{bmatrix}\right)\right\|_2^2. \tag{10}$$

Lemma 1 (Iteration Error Analysis for RISRO). *Let $\bar{\mathbf{X}}$ be any given target matrix. Recall the definition of $\boldsymbol{\epsilon}^t = \bar{\boldsymbol{\epsilon}} + \mathcal{A}(P_{\mathbf{U}_\perp^t}\bar{\mathbf{X}}P_{\mathbf{V}_\perp^t})$ from (6). If the operator $\mathcal{L}_t^*\mathcal{A}^*\mathcal{A}\mathcal{L}_t$ is invertible over $\mathrm{Ran}(\mathcal{L}_t^*)$, then $\mathbf{B}^{t+1}, \mathbf{D}_1^{t+1}, \mathbf{D}_2^{t+1}$ in (5) satisfy*

$$\begin{bmatrix} \mathbf{B}^{t+1} - \tilde{\mathbf{B}}^t & \mathbf{D}_2^{t+1\top} - \tilde{\mathbf{D}}_2^{t\top} \\ \mathbf{D}_1^{t+1} - \tilde{\mathbf{D}}_1^t & \mathbf{0} \end{bmatrix} = (\mathcal{L}_t^*\mathcal{A}^*\mathcal{A}\mathcal{L}_t)^{-1}\mathcal{L}_t^*\mathcal{A}^*\boldsymbol{\epsilon}^t, \tag{11}$$

and

$$\|\mathbf{B}^{t+1} - \tilde{\mathbf{B}}^t\|_{\mathrm{F}}^2 + \sum_{k=1}^2 \|\mathbf{D}_k^{t+1} - \tilde{\mathbf{D}}_k^t\|_{\mathrm{F}}^2 = \|(\mathcal{L}_t^*\mathcal{A}^*\mathcal{A}\mathcal{L}_t)^{-1}\mathcal{L}_t^*\mathcal{A}^*\boldsymbol{\epsilon}^t\|_{\mathrm{F}}^2. \tag{12}$$

In view of Lemma 1, the approximation errors of $\mathbf{B}^{t+1}$, $\mathbf{D}_1^{t+1}, \mathbf{D}_2^{t+1}$ to $\tilde{\mathbf{B}}^t, \tilde{\mathbf{D}}_1^t, \tilde{\mathbf{D}}_2^t$ are driven by the least squares residual $\|(\mathcal{L}_t^*\mathcal{A}^*\mathcal{A}\mathcal{L}_t)^{-1}\mathcal{L}_t^*\mathcal{A}^*\boldsymbol{\epsilon}^t\|_{\mathrm{F}}^2$. This fact plays a key

role in the proof for the high-order convergence theory of RISRO; see later in Remark 7.

Remark 1 (Comparison with Randomized Sketching). The importance sketching in RISRO is significantly different from the randomized sketching in the literature (see surveys Mahoney 2011, Woodruff 2014, and the references therein). The randomized sketching matrices are often randomly generated and reduce the sample size (n), the importance sketching matrices are deterministically constructed under the supervision of $\mathbf{y}$ and reduce the dimension of parameter space (p_1p_2). See Zhang et al. (2020, sections 1.3 and 2) for more comparison of randomized and importance sketchings.

2.2. Comparison with More Algorithms in the View of Sketching

In addition to RISRO, several classic algorithms for rank-constrained least squares can be interpreted from the recursive importance sketching perspective. Through the lens of the sketching, RISRO exhibits advantages over these existing algorithms.

We first focus on alter mini proposed and studied in Hardt (2014), Jain et al. (2013), and Zhao et al. (2015). Suppose $\mathbf{U}^t$ is the left singular vectors of $\mathbf{X}^t$, the outcome of the tth iteration; alter mini solves the following least squares problems to update $\mathbf{U}$ and $\mathbf{V}$:

$$\begin{aligned}
\check{\mathbf{V}}^{t+1} &= \arg\min_{\mathbf{V}\in\mathbb{R}^{p_2\times r}} \sum_{i=1}^{n} (\mathbf{y}_i - \langle \mathbf{A}_i, \mathbf{U}^t\mathbf{V}^\top\rangle)^2 \\
&= \arg\min_{\mathbf{V}\in\mathbb{R}^{p_2\times r}} \sum_{i=1}^{n} (\mathbf{y}_i - \langle \mathbf{U}^{t\top}\mathbf{A}_i, \mathbf{V}^\top\rangle)^2, \\
\check{\mathbf{U}}^{t+1} &= \arg\min_{\mathbf{U}\in\mathbb{R}^{p_1\times r}} \sum_{i=1}^{n} (\mathbf{y} - \langle \mathbf{A}_i, \mathbf{U}(\mathbf{V}^{t+1})^\top\rangle)^2 \\
&= \arg\min_{\mathbf{U}\in\mathbb{R}^{p_1\times r}} \sum_{i=1}^{n} (\mathbf{y} - \langle \mathbf{A}_i\mathbf{V}^{t+1}, \mathbf{U}\rangle)^2, \\
\mathbf{V}^{t+1} &= \mathrm{QR}(\check{\mathbf{V}}^{t+1}), \quad \mathbf{U}^{t+1} = \mathrm{QR}(\check{\mathbf{U}}^{t+1}).
\end{aligned} \tag{13}$$

Then, alter mini essentially solves least squares problems with sketched covariates $\mathbf{U}^{t\top}\mathbf{A}_i, \mathbf{A}_i\mathbf{V}^{t+1}$ to update $\check{\mathbf{V}}^{t+1}$, $\check{\mathbf{U}}^{t+1}$ alternatively and iteratively. The number of parameters of the least squares in (13) are rp_2 and rp_1 as opposed to p_1p_2, the number of parameters in the original least squares problem. See Figure 2, upper right panel, for an illustration of the sketching scheme in alter mini. Consider the following decomposition of $\mathbf{y}_i$:

$$\begin{aligned}
\mathbf{y}_i &= \langle \mathbf{A}_i, P_{\mathbf{U}^t}\bar{\mathbf{X}}\rangle + \langle \mathbf{A}_i, P_{\mathbf{U}_\perp^t}\bar{\mathbf{X}}\rangle + \bar{\boldsymbol{\epsilon}}_i \\
&= \langle \mathbf{U}^{t\top}\mathbf{A}_i, \mathbf{U}^{t\top}\bar{\mathbf{X}}\rangle + \langle \mathbf{A}_i, P_{\mathbf{U}_\perp^t}\bar{\mathbf{X}}\rangle + \bar{\boldsymbol{\epsilon}}_i \\
&:= \langle \mathbf{U}^{t\top}\mathbf{A}_i, \mathbf{U}^{t\top}\bar{\mathbf{X}}\rangle + \check{\boldsymbol{\epsilon}}_i^t,
\end{aligned} \tag{14}$$

where $\check{\boldsymbol{\epsilon}}^t := \mathcal{A}(P_{\mathbf{U}_\perp^t}\bar{\mathbf{X}}) + \bar{\boldsymbol{\epsilon}} \in \mathbb{R}^n$. Define $\check{\mathbf{A}}^t \in \mathbb{R}^{n\times p_2 r}$ with $\check{\mathbf{A}}_{[i,:]} = \mathrm{vec}(\mathbf{U}^{t\top}\mathbf{A}_i)$. Similar to how Lemma 1 is proved, we can show $\|\check{\mathbf{V}}^{t+1\top} - \mathbf{U}^{t\top}\bar{\mathbf{X}}\|_{\mathrm{F}}^2 = \|(\check{\mathbf{A}}^{t\top}\check{\mathbf{A}}^t)^{-1}\check{\mathbf{A}}^{t\top}\check{\boldsymbol{\epsilon}}^t\|_2^2$, which implies the approximation error of $\mathbf{V}^{t+1} = \mathrm{QR}(\check{\mathbf{V}}^{t+1})$ (i.e., the outcome of one iteration of alter mini) to $\bar{\mathbf{V}}$ (i.e., true row span of the target matrix $\bar{\mathbf{X}}$) is driven by $\check{\boldsymbol{\epsilon}}^t = \mathcal{A}(P_{\mathbf{U}_\perp^t}\bar{\mathbf{X}}) + \bar{\boldsymbol{\epsilon}}$, that is, the residual of the least squares problem (14). Recall that, for RISRO, Lemma 1 shows the approximation error of $\mathbf{V}^{t+1}$ is driven by $\boldsymbol{\epsilon}^t = \mathcal{A}(P_{\mathbf{U}_\perp^t}\bar{\mathbf{X}}P_{\mathbf{V}_\perp^t}) + \bar{\boldsymbol{\epsilon}}$. Because $\|P_{\mathbf{U}_\perp^t}\bar{\mathbf{X}}P_{\mathbf{V}_\perp^t}\|_{\mathrm{F}} \leqslant \|P_{\mathbf{U}_\perp^t}\bar{\mathbf{X}}\|_{\mathrm{F}}$, the approximation error in per iteration of RISRO can be smaller than the one of alter mini. Such a difference between RISRO and alter mini is due to the following fact: in alter mini, the sketching captures that the importance covariates correspond to only the row (or column) span of $\mathbf{X}^t$ in updating $\mathbf{V}^{t+1}$ (or $\mathbf{U}^{t+1}$), whereas the importance sketching of RISRO in (4) catches the importance covariates from both the row and column spans of $\mathbf{X}^t$. As a consequence, alter mini iterations yield first order convergence, whereas RISRO iterations render high-order convergence as is established in Section 3.

Remark 2. Recently, Kümmerle and Sigl (2018) propose a harmonic mean iterative reweighted least squares (HM-IRLS) method for low-rank matrix recovery: they specifically solve $\min_{\mathbf{X}\in\mathbb{R}^{p_1\times p_2}}\|\mathbf{X}\|_q^q$ subject to $\mathbf{y} = \mathcal{A}(\mathbf{X})$, where $\|\mathbf{X}\|_q = (\sum_i \sigma_i^q(\mathbf{X}))^{1/q}$ is the Schatten-q norm of the matrix $\mathbf{X}$. Compared with the original IRLS (Fornasier et al. 2011, Mohan and Fazel 2012), which only involves either the column span or the row span of $\mathbf{X}^t$ in constructing the reweighting matrix, HM-IRLS leverages both the column and row spans of $\mathbf{X}^t$ in constructing the reweighting matrix per iteration and performs better. Such a comparison of HM-IRLS versus IRLS shares the same spirit as RISRO versus alter mini: the importance sketching of RISRO simultaneously captures the information of both column and row spans of $\mathbf{X}^t$ per iteration and achieves a better performance. Utilizing both row and column spans of $\mathbf{X}^t$ simultaneously is the key to achieve high-order convergence performance by RISRO.

Another example is the rank-$2r$ iterative least squares (R2RILS) proposed in Bauch et al. (2021) for solving ill-conditioned matrix completion problems. In particular, at the tth iteration, step 1 of R2RILS solves the following least squares problem:

$$\min_{\mathbf{M}\in\mathbb{R}^{p_1\times r},\mathbf{N}\in\mathbb{R}^{p_2\times r}} \sum_{(i,j)\in\Omega} \{(\mathbf{U}^t\mathbf{N}^\top + \mathbf{M}\mathbf{V}^{t\top} - \mathbf{X})_{[i,j]}\}^2, \tag{15}$$

where Ω is the set of index pairs of the observed entries. In the matrix completion setting, it turns out the following equivalence holds (proof given in the online appendix):

$$\begin{aligned}
&\arg\min_{\mathbf{M}\in\mathbb{R}^{p_1\times r},\mathbf{N}\in\mathbb{R}^{p_2\times r}} \sum_{(i,j)\in\Omega} \{(\mathbf{U}^t\mathbf{N}^\top + \mathbf{M}\mathbf{V}^{t\top} - \mathbf{X})_{[i,j]}\}^2 \\
= &\arg\min_{\mathbf{M}\in\mathbb{R}^{p_1\times r},\mathbf{N}\in\mathbb{R}^{p_2\times r}} \sum_{(i,j)\in\Omega} (\langle \mathbf{U}^{t\top}\mathbf{A}^{ij}, \mathbf{N}^\top\rangle + \langle \mathbf{M}, \mathbf{A}^{ij}\mathbf{V}^t\rangle - \mathbf{X}_{[i,j]})^2,
\end{aligned} \tag{16}$$

where $\mathbf{A}^{ij} \in \mathbb{R}^{p_1 \times p_2}$ is the special covariate in matrix completion satisfying $(\mathbf{A}^{ij})_{[k,l]} = 1$ if $(i,j) = (k,l)$ and $(\mathbf{A}^{ij})_{[k,l]} = 0$ otherwise. This equivalence reveals that the least squares step (15) in R2RILS can be seen as an implicit sketched least squares problem similar to (5) and (13) with covariates $\mathbf{U}^{t\top}\mathbf{A}^{ij}$ and $\mathbf{A}^{ij}\mathbf{V}^t$ for $(i,j) \in \Omega$.

We give a pictorial illustration for the sketching interpretation of R2RILS on the bottom right part of Figure 2. Different from the sketching in RISRO, R2RILS incorporates the core sketch $\mathbf{U}^{t\top}\mathbf{A}_i\mathbf{V}^t$ twice, which results in the rank deficiency in the least squares problem (15) and brings difficulties in both implementation and theoretical analysis. RISRO overcomes this issue by performing a better designed sketching and covers more general low-rank matrix recovery settings than R2RILS. With the new sketching scheme, we are able to give a new and solid theory for RISRO with high-order convergence.

3. Theoretical Analysis

In this section, we provide convergence analysis for the proposed algorithm. For technical convenience, we assume $\mathcal{A}$ satisfies the RIP (Candès 2008). The RIP condition, first introduced in compressed sensing, is widely used as one of the most standard assumptions in the low-rank matrix recovery literature (Jain et al. 2010; Recht et al. 2010; Candès and Plan 2011; Cai and Zhang 2013, 2014; Chen and Wainwright 2015; Zhao et al. 2015; Tu et al. 2016). It also plays a critical role in analyzing the landscape of the rank-constrained optimization problem (1) (Bhojanapalli et al. 2016, Ge et al. 2017, Zhu et al. 2018, Zhang et al. 2019, Uschmajew and Vandereycken 2020). On the other hand, RIP is only a sufficient but not necessary condition for the convergence of RISRO. We illustrate later in several examples that RISRO converges quadratically, whereas RIP completely fails.

Definition 1 (RIP). Let $\mathcal{A}: \mathbb{R}^{p_1 \times p_2} \to \mathbb{R}^n$ be a linear map. For every integer r with $1 \leqslant r \leqslant \min(p_1, p_2)$, define the r-restricted isometry constant to be the smallest number R_r such that $(1 - R_r)\|\mathbf{Z}\|_{\mathrm{F}}^2 \leqslant \|\mathcal{A}(\mathbf{Z})\|_2^2 \leqslant (1 + R_r)\|\mathbf{Z}\|_{\mathrm{F}}^2$ holds for all $\mathbf{Z}$ of rank at most r. And $\mathcal{A}$ is said to satisfy the r-restricted isometry property ($r-$RIP) if $0 \leqslant R_r < 1$.

The RIP condition provably holds when $\mathcal{A}$ has independent random sub-Gaussian design or $\mathcal{A}$ is a random projection (Recht et al. 2010, Candès and Plan 2011). In addition, this definition of RIP can be equivalently stated in a matrix format: define $\tilde{\mathbf{A}} = [\mathrm{vec}(\mathbf{A}_1), \ldots, \mathrm{vec}(\mathbf{A}_n)]^\top$ and $\mathcal{A}(\mathbf{Z}) = \tilde{\mathbf{A}}\mathrm{vec}(\mathbf{Z})$. Then, that $\mathcal{A}$ satisfies the RIP condition is equivalent to $(1 - R_r)\|\mathrm{vec}(\mathbf{Z})\|_2^2 \leqslant \|\tilde{\mathbf{A}}(\mathrm{vec}(\mathbf{Z}))\|_2^2 \leqslant (1 + R_r)\|\mathrm{vec}(\mathbf{Z})\|_2^2$ for all matrices $\mathbf{Z}$ of rank at most r. By definition, $R_r \leqslant R_{r'}$ for any $r \leqslant r'$.

By assuming RIP for $\mathcal{A}$, we can show that the linear operator $\mathcal{L}_t^*\mathcal{A}^*\mathcal{A}\mathcal{L}_t$ mentioned in Lemma 1 is always invertible over $\mathrm{Ran}(\mathcal{L}_t^*)$ (i.e., the least squares (5) has a unique solution). The following lemma gives explicit lower and upper bounds for the spectrum of this operator.

Lemma 2 (Bounds for the Spectrum of $\mathcal{L}_t^*\mathcal{A}^*\mathcal{A}\mathcal{L}_t$). *Recall the definition of $\mathcal{L}_t$ in (9). It holds that*

$$\|\mathcal{L}_t(\mathbf{M})\|_{\mathrm{F}} = \|\mathbf{M}\|_{\mathrm{F}}, \quad \forall\, \mathbf{M} \in \mathrm{Ran}(\mathcal{L}_t^*). \tag{17}$$

Suppose the linear map $\mathcal{A}$ satisfies the 2r-RIP. Then, it holds that, for any matrix $\mathbf{M} \in \mathrm{Ran}(\mathcal{L}_t^*)$,

$$(1 - R_{2r})\|\mathbf{M}\|_{\mathrm{F}} \leqslant \|\mathcal{L}_t^*\mathcal{A}^*\mathcal{A}\mathcal{L}_t(\mathbf{M})\|_{\mathrm{F}} \leqslant (1 + R_{2r})\|\mathbf{M}\|_{\mathrm{F}}.$$

Remark 3 (Bounds for the Spectrum of $(\mathcal{L}_t^*\mathcal{A}^*\mathcal{A}\mathcal{L}_t)^{-1}$). By the relationship of the spectrum of an operator and its inverse, from Lemma 2, we also have that the spectrum of $(\mathcal{L}_t^*\mathcal{A}^*\mathcal{A}\mathcal{L}_t)^{-1}$ is lower and upper bounded by $\frac{1}{(1+R_{2r})}$ and $\frac{1}{(1-R_{2r})}$, respectively.

In the following Proposition 1, we bound the iteration approximation error given in Lemma 1.

Proposition 1 (Upper Bound for Iteration Approximation Error). *Let $\bar{\mathbf{X}}$ be a given target rank-r matrix and $\bar{\boldsymbol{\epsilon}} = \mathbf{y} - \mathcal{A}(\bar{\mathbf{X}})$. Suppose that $\mathcal{A}$ satisfies the 2r-RIP. Then, at the tth iteration of RISRO, the approximation error (12) has the following upper bound:*

$$\begin{aligned}
&\|(\mathcal{L}_t^*\mathcal{A}^*\mathcal{A}\mathcal{L}_t)^{-1}\mathcal{L}_t^*\mathcal{A}^*\boldsymbol{\epsilon}^t\|_{\mathrm{F}}^2 \\
&\leqslant \frac{R_{3r}^2\|\mathbf{X}^t - \bar{\mathbf{X}}\|^2\|\mathbf{X}^t - \bar{\mathbf{X}}\|_{\mathrm{F}}^2}{(1 - R_{2r})^2\sigma_r^2(\bar{\mathbf{X}})} + \frac{\|\mathcal{L}_t^*\mathcal{A}^*(\bar{\boldsymbol{\epsilon}})\|_{\mathrm{F}}^2}{(1 - R_{2r})^2} \\
&\quad + \|\mathcal{L}_t^*\mathcal{A}^*(\bar{\boldsymbol{\epsilon}})\|_{\mathrm{F}}\frac{2R_{3r}\|\mathbf{X}^t - \bar{\mathbf{X}}\|\|\mathbf{X}^t - \bar{\mathbf{X}}\|_{\mathrm{F}}}{\sigma_r(\bar{\mathbf{X}})(1 - R_{2r})^2}.
\end{aligned} \tag{18}$$

Note that Proposition 1 is rather general in the sense that it applies to any $\bar{\mathbf{X}}$ of rank r, and we pick different choices of $\bar{\mathbf{X}}$ depending on our purposes. For example, in studying the convergence of RISRO, for example, the upcoming Theorem 1, we treat $\bar{\mathbf{X}}$ as a stationary point, and in the setting of estimating the model parameter in matrix trace regression, we take $\bar{\mathbf{X}}$ to be the ground truth (see Theorem 3).

Now, we are ready to establish the deterministic convergence theory for RISRO. For Problem (1), we use the following definition of stationary points: a rank-r matrix $\bar{\mathbf{X}}$ is said to be a stationary point of (1) if $\nabla f(\bar{\mathbf{X}})^\top\bar{\mathbf{U}} = 0$ and $\nabla f(\bar{\mathbf{X}})\bar{\mathbf{V}} = 0$, where $\nabla f(\bar{\mathbf{X}}) = \mathcal{A}^*(\mathcal{A}(\bar{\mathbf{X}}) - \mathbf{y})$, and $\bar{\mathbf{U}}, \bar{\mathbf{V}}$ are the left and right singular vectors of $\bar{\mathbf{X}}$. See also Ha et al. (2020). In Theorem 1, we show that, given any target stationary point $\bar{\mathbf{X}}$ and proper initialization, RISRO has a local quadratic-linear convergence rate in general and quadratic convergence rate if $\mathbf{y} = \mathcal{A}(\bar{\mathbf{X}})$.

Theorem 1 (Local Quadratic-Linear and Quadratic Convergence of RISRO). *Let $\bar{\mathbf{X}}$ be a stationary point to Problem (1) and $\bar{\boldsymbol{\epsilon}} = \mathbf{y} - \mathcal{A}(\bar{\mathbf{X}})$. Suppose that $\mathcal{A}$ satisfies the 2r-RIP, and the initialization $\mathbf{X}^0$ satisfies*

$$\|\mathbf{X}^0 - \bar{\mathbf{X}}\|_{\mathrm{F}} \leqslant \left(\frac{1}{4} \wedge \frac{1 - R_{2r}}{4\sqrt{5}R_{3r}}\right)\sigma_r(\bar{\mathbf{X}}), \tag{19}$$

and $\|\mathcal{A}^*(\bar{\boldsymbol{\epsilon}})\|_{\mathrm{F}} \leqslant \frac{1-R_{2r}}{4\sqrt{5}}\sigma_r(\bar{\mathbf{X}})$. *Then, we have* $\{\mathbf{X}^t\}$, *the sequence generated by RISRO (Algorithm 1), converges linearly to* $\bar{\mathbf{X}}$: $\|\mathbf{X}^{t+1}-\bar{\mathbf{X}}\|_{\mathrm{F}} \leqslant \frac{3}{4}\|\mathbf{X}^t-\bar{\mathbf{X}}\|_{\mathrm{F}}, \quad \forall\, t \geqslant 0.$

More precisely, it holds that $\forall\, t \geqslant 0$:

$$\|\mathbf{X}^{t+1}-\bar{\mathbf{X}}\|_{\mathrm{F}}^2 \leqslant \frac{5\|\mathbf{X}^t-\bar{\mathbf{X}}\|^2}{(1-R_{2r})^2\sigma_r^2(\bar{\mathbf{X}})} \cdot (R_{3r}^2\|\mathbf{X}^t-\bar{\mathbf{X}}\|_{\mathrm{F}}^2 + 4R_{3r}\|\mathcal{A}^*(\bar{\boldsymbol{\epsilon}})\|_{\mathrm{F}}\|\mathbf{X}^t-\bar{\mathbf{X}}\|_{\mathrm{F}} + 4\|\mathcal{A}^*(\bar{\boldsymbol{\epsilon}})\|_{\mathrm{F}}^2). \tag{20}$$

In particular, if $\bar{\boldsymbol{\epsilon}} = 0$, *then* $\{\mathbf{X}^t\}$ *converges quadratically to* $\bar{\mathbf{X}}$ *as*

$$\|\mathbf{X}^{t+1}-\bar{\mathbf{X}}\|_{\mathrm{F}} \leqslant \frac{\sqrt{5}R_{3r}}{(1-R_{2r})\sigma_r(\bar{\mathbf{X}})}\|\mathbf{X}^t-\bar{\mathbf{X}}\|_{\mathrm{F}}^2, \quad \forall\, t \geqslant 0.$$

Remark 4 (Quadratic-Linear and Quadratic Convergence of RISRO). We call the convergence in (20) quadratic-linear because the sequence $\{\mathbf{X}^t\}$ generated by RISRO exhibits a phase transition from quadratic to linear convergence: when $\|\mathbf{X}^t-\bar{\mathbf{X}}\|_{\mathrm{F}} \gg \|\mathcal{A}^*(\bar{\boldsymbol{\epsilon}})\|_{\mathrm{F}}$, the algorithm has a quadratic convergence rate; when $\mathbf{X}^t$ becomes close to $\bar{\mathbf{X}}$ such that $\|\mathbf{X}^t-\bar{\mathbf{X}}\|_{\mathrm{F}} \leqslant c\|\mathcal{A}^*(\bar{\boldsymbol{\epsilon}})\|_{\mathrm{F}}$ for some $c > 0$, the convergence rate becomes linear. Even though the ultimate convergence of RISRO is linear to a stationary point in the noisy setting, we show later in Section 6.1 that RISRO achieves quadratic convergence in estimating the underlying parameter matrix in statistical applications. Moreover, as $\bar{\boldsymbol{\epsilon}}$ becomes smaller, the stage of quadratic convergence becomes longer (see Section 7.1 for a numerical illustration of this convergence pattern). In the extreme case $\bar{\boldsymbol{\epsilon}} = 0$, Theorem 1 covers the widely studied matrix sensing problem under the RIP framework (Jain et al. 2010, Recht et al. 2010, Chen and Wainwright 2015, Zhao et al. 2015, Zheng and Lafferty 2015, Tu et al. 2016, Park et al. 2018). It shows that, as long as the initialization error is within a constant factor of $\sigma_r(\bar{\mathbf{X}})$, RISRO enjoys quadratic convergence to the target matrix $\bar{\mathbf{X}}$. To the best of our knowledge, we are among the first to give quadratic-linear algorithmic convergence guarantees for general rank-constrained least squares and quadratic convergence for matrix sensing. Recently, Charisopoulos et al. (2021) formulated (1) as a nonconvex composite optimization problem based on $\mathbf{X} = \mathbf{R}\mathbf{L}^\top$ factorization and showed that the prox-linear algorithm (Burke 1985, Lewis and Wright 2016) achieves local quadratic convergence when $\bar{\boldsymbol{\epsilon}} = 0$. In each iteration therein, a carefully tuned convex program needs to be solved exactly, and the tuning parameter relies on the unknown weak convexity parameter of the composite objective function. In contrast, the proposed RISRO is tuning-free, only solves a dimension-reduced least squares in each step, and can be as cheaply as many first order methods. See Section 5 for a detailed discussion on the computational complexity of RISRO.

Moreover, a quadratic-linear convergence rate also appears in several other methods under different settings: Pilanci and Wainwright (2017) study the local convergence of the randomized Newton sketch for objectives with strong convexity and smooth properties; Erdogdu and Montanari (2015) consider the subsampled Newton method to optimize an objective function in the form of a sum of convex functions and establish their convergence theory with the well-conditioned subsampled Hessian. We consider the nonconvex matrix optimization problem (1) and use the recursive importance sketching method. Our quadratic-linear convergence result can be boosted to quadratic when $\bar{\boldsymbol{\epsilon}} = 0$.

Remark 5 (Initialization). The convergence theory in Theorem 1 requires a good initialization condition. Practically, the spectral method often provides a sufficiently good initialization that meets the requirement in (19) in many statistical applications. In Sections 6 and 7, we illustrate this point from two applications: matrix trace regression and phase retrieval.

Remark 6 (Small Residual Condition in Theorem 1). In addition to the initialization condition, the small residual condition $\|\mathcal{A}^*(\bar{\boldsymbol{\epsilon}})\|_{\mathrm{F}} \leqslant \frac{1-R_{2r}}{4\sqrt{5}}\sigma_r(\bar{\mathbf{X}})$ is also needed in Theorem 1. This condition essentially means that the signal strength at point $\bar{\mathbf{X}}$ needs to dominate the noise. If $\bar{\boldsymbol{\epsilon}} = \mathbf{y} - \mathcal{A}(\bar{\mathbf{X}}) = 0$, then the aforementioned small residual condition holds automatically.

Remark 7. We provide a proof sketch of Theorem 1 and discuss our technical contributions therein.

Step 1. We bound $\|\mathcal{L}_t^*\mathcal{A}^*(\bar{\boldsymbol{\epsilon}})\|_{\mathrm{F}} \leqslant \frac{4\|\mathbf{X}^t-\bar{\mathbf{X}}\|^2}{\sigma_r^2(\bar{\mathbf{X}})}\|\mathcal{A}^*(\bar{\boldsymbol{\epsilon}})\|_{\mathrm{F}}^2$ and then apply Proposition 1 to obtain an upper bound for the approximation error in (12):

$$\|(\mathcal{L}_t^*\mathcal{A}^*\mathcal{A}\mathcal{L}_t)^{-1}\mathcal{L}_t^*\mathcal{A}^*\boldsymbol{\epsilon}^t\|_{\mathrm{F}}^2 \leqslant \frac{\|\mathbf{X}^t-\bar{\mathbf{X}}\|^2}{(1-R_{2r})^2\sigma_r^2(\bar{\mathbf{X}})} \cdot (R_{3r}^2\|\mathbf{X}^t-\bar{\mathbf{X}}\|_{\mathrm{F}}^2 + 4\|\mathcal{A}^*(\bar{\boldsymbol{\epsilon}})\|_{\mathrm{F}}^2 + 4R_{3r}\|\mathcal{A}^*(\bar{\boldsymbol{\epsilon}})\|_{\mathrm{F}}\|\mathbf{X}^t-\bar{\mathbf{X}}\|_{\mathrm{F}}). \tag{21}$$

Step 2. We use induction to show the following three claims:

(*C*1) $\max\{\|\sin\Theta(\mathbf{U}^t,\bar{\mathbf{U}})\|, \|\sin\Theta(\mathbf{V}^t,\bar{\mathbf{V}})\|\} \leqslant \frac{1}{2}$;

(*C*2) $\mathbf{B}^{t+1}$ *in* (5) *is invertible*;

(*C*3) $\|\mathbf{X}^{t+1}-\bar{\mathbf{X}}\|_{\mathrm{F}}^2 \leqslant \frac{5\|\mathbf{X}^t-\bar{\mathbf{X}}\|^2}{(1-R_{2r}^2)\sigma_r^2(\bar{\mathbf{X}})} (R_{3r}^2\|\mathbf{X}^t-\bar{\mathbf{X}}\|_{\mathrm{F}}^2 + 4R_{3r}\|\mathcal{A}^*(\bar{\boldsymbol{\epsilon}})\|_{\mathrm{F}}\|\mathbf{X}^t-\bar{\mathbf{X}}\|_{\mathrm{F}} + 4\|\mathcal{A}^*(\bar{\boldsymbol{\epsilon}})\|_{\mathrm{F}}^2) \leqslant \frac{9}{16}\|\mathbf{X}^t-\bar{\mathbf{X}}\|_{\mathrm{F}}^2.$

Here, (C2) means the iterates $\mathbf{X}^t$ are always rank-r. This fact is useful in Section 4 in connecting RISRO to Riemannian optimization on fixed-rank matrix

manifolds. (C2) is proved by (C1) and Lemma 1. In proving (C3), we introduce an intermediate quantity $\rho_{t+1} = \max\{\|\mathbf{D}_1^{t+1}(\mathbf{B}^{t+1})^{-1}\|, \|(\mathbf{B}^{t+1})^{-1}\mathbf{D}_2^{t+1\top}\|\}$ and obtain

$$\begin{aligned}
&\|\mathbf{X}^{t+1} - \bar{\mathbf{X}}\|_{\mathrm{F}}^2 \\
&= \left\| \begin{matrix} \mathbf{B}^{t+1} - \tilde{\mathbf{B}}^t & \mathbf{D}_2^{t+1\top} - \tilde{\mathbf{D}}_2^{t\top} \\ \mathbf{D}_1^{t+1} - \tilde{\mathbf{D}}_1^t & \mathbf{D}_1^{t+1}(\mathbf{B}^{t+1})^{-1}\mathbf{D}_2^{t+1\top} - \tilde{\mathbf{D}}_1^t(\tilde{\mathbf{B}}^t)^{-1}\tilde{\mathbf{D}}_2^{t\top} \end{matrix} \right\|_{\mathrm{F}}^2 \\
&\overset{(a)}{\leqslant} 5\|(\mathcal{L}_t^*\mathcal{A}^*\mathcal{A}\mathcal{L}_t)^{-1}\mathcal{L}_t^*\mathcal{A}^*\boldsymbol{\epsilon}^t\|_{\mathrm{F}}^2, \qquad (22)
\end{aligned}$$

Here, (a) is by the induction assumptions, Lemma 1, and Lemma 7. Finally, (C3) follows by plugging (21) into (22) and the induction assumptions, and this proves the main result of Theorem 1.

4. A Riemannian Manifold Optimization Interpretation of RISRO

The superior performance of RISRO yields the following question: is there a connection of RISRO to any class of optimization algorithms in the literature?

In this section, we give an affirmative answer to this question. We show RISRO can be viewed as a Riemannian optimization algorithm on the manifold $\mathcal{M}_r := \{\mathbf{X} \in \mathbb{R}^{p_1\times p_2} | \mathrm{rank}(\mathbf{X}) = r\}$. We find the sketched least squares in (5) of RISRO actually solves the Fisher scoring or Riemannian Gauss–Newton equation, and step 7 in RISRO performs a type of retraction under the framework of Riemannian optimization.

Riemannian optimization concerns optimizing a real-valued function f defined on a Riemannian manifold $\mathcal{M}$. One commonly encountered manifold is a submanifold of $\mathbb{R}^n$. Under such circumstances, a manifold can be viewed as a smooth subset of $\mathbb{R}^n$. When a smooth-varying inner product is further defined on the subset, the subset together with the inner product is called a Riemannian manifold. We refer to Absil et al. (2008) for the rigorous definition of Riemannian manifolds. Optimization on a Riemannian manifold often relies on the notion of Riemannian gradient/Riemannian Hessian (which are used to find a search direction) and the notion of retraction (which is defined for the motion of iterates on the manifold). The remainder of this section describes the required Riemannian optimization tools and the connection of RISRO to Riemannian optimization.

It is shown in Lee (2013, example 8.14) that the set $\mathcal{M}_r$ is a smooth submanifold of $\mathbb{R}^{p_1\times p_2}$ and the tangent space is also given therein. The result is given in Proposition 2 for completeness.

Proposition 2 (Lee 2013, Example 8.14). *$\mathcal{M}_r = \{\mathbf{X} \in \mathbb{R}^{p_1\times p_2} : \mathrm{rank}(\mathbf{X}) = r\}$ is a smooth embedded submanifold of dimension $(p_1 + p_2 - r)r$. Its tangent space $T_{\mathbf{X}}\mathcal{M}_r$ at $\mathbf{X} \in \mathcal{M}_r$ with the singular value decomposition $\mathbf{X} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top$ ($\mathbf{U} \in \mathbb{O}_{p_1,r}$ and $\mathbf{V} \in \mathbb{O}_{p_2,r}$) is given by*

$$T_{\mathbf{X}}\mathcal{M}_r = \left\{ [\mathbf{U} \quad \mathbf{U}_\perp] \begin{bmatrix} \mathbb{R}^{r\times r} & \mathbb{R}^{r\times(p_2-r)} \\ \mathbb{R}^{(p_1-r)\times r} & \mathbf{0}_{(p_1-r)\times(p_2-r)} \end{bmatrix} [\mathbf{V} \quad \mathbf{V}_\perp]^\top \right\}. \qquad (23)$$

The Riemannian metric of $\mathcal{M}_r$ that we use throughout this paper is the Euclidean inner product, that is, $\langle \mathbf{U}, \mathbf{V}\rangle = \mathrm{trace}(\mathbf{U}^\top\mathbf{V})$.

In the Euclidean setting, the update formula in an iterative algorithm is $X^t + \alpha\eta^t$, where α is the step size and η^t is a descent direction. However, in the framework of Riemannian optimization, $\mathbf{X}^t + \alpha\eta^t$ is generally neither well-defined nor lying in the manifold. To overcome this difficulty, the notion of retraction is used; see, for example, Absil et al. (2008). Considering the manifold $\mathcal{M}_r$, we have the definition that a retraction R is a smooth map from $T\mathcal{M}_r$ to $\mathcal{M}_r$ satisfying (i) $R(\mathbf{X}, 0) = \mathbf{X}$ and (ii) $\frac{d}{dt}R(\mathbf{X}, t\eta)|_{t=0} = \eta$ for all $\mathbf{X} \in \mathcal{M}_r$ and $\eta \in T_{\mathbf{X}}\mathcal{M}_r$, where $T\mathcal{M}_r = \{(\mathbf{X}, T_{\mathbf{X}}\mathcal{M}_r) : \mathbf{X} \in \mathcal{M}_r\}$ is the tangent bundle of $\mathcal{M}_r$. The two conditions guarantee that $R(\mathbf{X}, t\eta)$ stays in $\mathcal{M}_r$ and $R(\mathbf{X}, t\eta)$ is a first order approximation of $\mathbf{X} + t\eta$ at $t = 0$.

Next, we show that step 7 in Algorithm 1 performs the orthographic retraction on the manifold of fixed-rank matrices given in Absil and Malick (2012). Suppose at iteration $t+1$ $\mathbf{B}^{t+1}$ is invertible (this is true under the RIP framework; see Remark 7 and step 2 in the proof of Theorem 1). We can show by some algebraic calculations that the update $\mathbf{X}^{t+1}$ in step 7 can be rewritten as

$$\begin{aligned}
\mathbf{X}^{t+1} &= \mathbf{X}_U^{t+1}(\mathbf{B}^{t+1})^{-1}\mathbf{X}_V^{t+1\top} \\
&= [\mathbf{U}^t \quad \mathbf{U}_\perp^t] \begin{bmatrix} \mathbf{B}^{t+1} & \mathbf{D}_2^{t+1\top} \\ \mathbf{D}_1^{t+1} & \mathbf{D}_1^{t+1}(\mathbf{B}^{t+1})^{-1}\mathbf{D}_2^{t+1\top} \end{bmatrix} [\mathbf{V}^t \quad \mathbf{V}_\perp^t]^\top. \qquad (24)
\end{aligned}$$

Let $\eta^t \in T_{\mathbf{X}^t}\mathcal{M}_r$ be the update direction and $\mathbf{X}^t + \eta^t$ has the following representation:

$$\mathbf{X}^t + \eta^t = [\mathbf{U}^t \quad \mathbf{U}_\perp^t] \begin{bmatrix} \mathbf{B}^{t+1} & \mathbf{D}_2^{t+1\top} \\ \mathbf{D}_1^{t+1} & \mathbf{0} \end{bmatrix} [\mathbf{V}^t \quad \mathbf{V}_\perp^t]^\top. \qquad (25)$$

Comparing (24) and (25), we can view the update of $\mathbf{X}^{t+1}$ from $\mathbf{X}^t + \eta^t$ as simply completing the $\mathbf{0}$ matrix in $\begin{bmatrix} \mathbf{B}^{t+1} & \mathbf{D}_2^{t+1\top} \\ \mathbf{D}_1^{t+1} & \mathbf{0} \end{bmatrix}$ by $\mathbf{D}_1^{t+1}(\mathbf{B}^{t+1})^{-1}\mathbf{D}_2^{t+1\top}$. This operation maps the tangent vector on $T_{\mathbf{X}^t}\mathcal{M}_r$ back to the manifold $\mathcal{M}_r$, and it turns out that it coincides with the orthographic retraction

$$R(\mathbf{X}^t, \eta^t) = [\mathbf{U}^t \quad \mathbf{U}_\perp^t] \begin{bmatrix} \mathbf{B}^{t+1} & \mathbf{D}_2^{t+1\top} \\ \mathbf{D}_1^{t+1} & \mathbf{D}_1^{t+1}(\mathbf{B}^{t+1})^{-1}\mathbf{D}_2^{t+1\top} \end{bmatrix} [\mathbf{V}^t \quad \mathbf{V}_\perp^t]^\top \qquad (26)$$

on the set of fixed-rank matrices (Absil and Malick 2012). Therefore, we have $\mathbf{X}^{t+1} = R(\mathbf{X}^t, \eta^t)$.

Remark 8. Although the orthographic retraction defined in Absil and Malick (2012) requires that $\mathbf{U}^t$ and $\mathbf{V}^t$ are left and right singular vectors of $\mathbf{X}^t$, one can verify that, even if the $\mathbf{U}^t$ and $\mathbf{V}^t$ are not exactly the left and right singular vectors but satisfy $\mathbf{U}^t = \tilde{\mathbf{U}}^t\mathbf{O}$, $\mathbf{V}^t = \tilde{\mathbf{V}}^t\mathbf{Q}$, then the mapping (26) is equivalent to the orthographic retraction in Absil and Malick (2012). Here, $\mathbf{O}, \mathbf{Q} \in \mathbb{O}_{r,r}$, and $\tilde{\mathbf{U}}^t$ and $\tilde{\mathbf{V}}^t$ are left and right singular vectors of $\mathbf{X}^t$.

The Riemannian gradient of a smooth function $f : \mathcal{M}_r \to \mathbb{R}$ at $\mathbf{X} \in \mathcal{M}_r$ is defined as the unique tangent vector $\operatorname{grad} f(\mathbf{X}) \in T_{\mathbf{X}}\mathcal{M}_r$ such that $\langle \operatorname{grad} f(\mathbf{X}), \mathbf{Z}\rangle = \mathrm{D} f(\mathbf{X})[\mathbf{Z}]$, $\forall\, \mathbf{Z} \in T_{\mathbf{X}}\mathcal{M}_r$, where $\mathrm{D}f(\mathbf{X})[\mathbf{Z}]$ denotes the directional derivative of f at point $\mathbf{X}$ along the direction $\mathbf{Z}$. Because $\mathcal{M}_r$ is an embedded submanifold of $\mathbb{R}^{p_1\times p_2}$ and the Euclidean metric is used, from Absil et al. (2008, (3.37)), we know, in our problem,

$$\operatorname{grad}\ f(\mathbf{X}) = P_{T_{\mathbf{X}}}(\mathcal{A}^*(\mathcal{A}(\mathbf{X}) - \mathbf{y})), \tag{27}$$

and here, $P_{T_{\mathbf{X}}}$ is the orthogonal projector onto the tangent space at $\mathbf{X}$ defined as follows:

$$P_{T_{\mathbf{X}}}(\mathbf{Z}) = P_{\mathbf{U}}\mathbf{Z}P_{\mathbf{V}} + P_{\mathbf{U}_\perp}\mathbf{Z}P_{\mathbf{V}} + P_{\mathbf{U}}\mathbf{Z}P_{\mathbf{V}_\perp}, \quad \forall\, \mathbf{Z} \in \mathbb{R}^{p_1\times p_2}, \tag{28}$$

where $\mathbf{U} \in \mathbb{O}_{p_1,r}, \mathbf{V} \in \mathbb{O}_{p_2,r}$ are the left and right singular vectors of $\mathbf{X}$.

Next, we introduce the Riemannian Hessian. The Riemannian Hessian of f at $\mathbf{X} \in \mathcal{M}_r$ is the linear map $\operatorname{Hess} f(\mathbf{X})$ of $T_{\mathbf{X}}\mathcal{M}_r$ onto itself defined as $\operatorname{Hess} f(\mathbf{X})[\mathbf{Z}] = \overline{\nabla}_{\mathbf{Z}}\operatorname{grad} f$, $\forall \mathbf{Z} \in T_{\mathbf{X}}\mathcal{M}_r$, where $\overline{\nabla}$ is the Riemannian connection on $\mathcal{M}_r$ (Absil et al. 2008, section 5.3). Lemma 3 gives an explicit formula for Riemannian Hessian in our problem.

Lemma 3 (Riemannian Hessian). *Consider $f(\mathbf{X})$ in (1). If $\mathbf{X} \in \mathcal{M}_r$ has singular value decomposition $\mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$ and $\mathbf{Z} \in T_{\mathbf{X}}\mathcal{M}_r$ has representation*

$$\mathbf{Z} = [\mathbf{U} \quad \mathbf{U}_\perp]\begin{bmatrix}\mathbf{Z}_B & \mathbf{Z}_{D_2}^\top \\ \mathbf{Z}_{D_1} & 0\end{bmatrix}[\mathbf{V} \quad \mathbf{V}_\perp]^\top,$$

then the Hessian operator in this setting satisfies

$$\begin{aligned}\operatorname{Hess} f(\mathbf{X})[\mathbf{Z}] = {} & P_{T_{\mathbf{X}}}(\mathcal{A}^*(\mathcal{A}(\mathbf{Z}))) + P_{\mathbf{U}_\perp}\mathcal{A}^*(\mathcal{A}(\mathbf{X}) - \mathbf{y})\\ & \mathbf{V}_p\mathbf{\Sigma}^{-1}\mathbf{V}^\top P_{\mathbf{V}} + P_{\mathbf{U}}\mathbf{U}\mathbf{\Sigma}^{-1}\mathbf{U}_p^\top\mathcal{A}^*(\mathcal{A}(\mathbf{X}) - \mathbf{y})P_{\mathbf{V}_\perp},\end{aligned} \tag{29}$$

where $\mathbf{U}_p = \mathbf{U}_\perp\mathbf{Z}_{D_1}, \mathbf{V}_p = \mathbf{V}_\perp\mathbf{Z}_{D_2}$.

Next, we show that the update direction η^t, implicitly encoded in (25), finds the Riemannian Gauss–Newton direction in the manifold optimization of $\mathcal{M}_r$. Similar to the classic Newton's method, at the tth iteration, the Riemannian Newton method aims to find the Riemannian Newton direction η^t_{Newton} in $T_{\mathbf{X}^t}\mathcal{M}_r$ that solves the following Newton equation:

$$-\operatorname{grad} f(\mathbf{X}^t) = \operatorname{Hess} f(\mathbf{X}^t)[\eta^t_{\text{Newton}}]. \tag{30}$$

If the residual $(\mathbf{y} - \mathcal{A}(\mathbf{X}^t))$ is small, the last two terms in $\operatorname{Hess} f(\mathbf{X}^t)[\eta]$ of (29) are expected to be small, which means we can approximately solve the Riemannian Newton direction via

$$-\operatorname{grad} f(\mathbf{X}^t) = P_{T_{\mathbf{X}^t}}(\mathcal{A}^*(\mathcal{A}(\eta))), \quad \eta \in T_{\mathbf{X}^t}\mathcal{M}_r. \tag{31}$$

In fact, Equation (31) has an interpretation from the Fisher scoring algorithm. Consider the statistical setting $\mathbf{y} = \mathcal{A}(\mathbf{X}) + \boldsymbol{\epsilon}$, where $\mathbf{X}$ is a fixed low-rank matrix and $\boldsymbol{\epsilon}_i \overset{i.i.d.}{\sim} N(0, \sigma^2)$. Then, for any η,

$$\{\mathbb{E}(\operatorname{Hess} f(\mathbf{X})[\eta])\}|_{\mathbf{X}=\mathbf{X}^t} = P_{T_{\mathbf{X}^t}}(\mathcal{A}^*(\mathcal{A}(\eta))),$$

where, on the left-hand side, the expression is evaluated at $\mathbf{X}^t$ after taking expectation. In the literature, the Fisher scoring algorithm computes the update direction via solving the modified Newton equation, which replaces the Hessian with its expected value (Lange 2010), that is,

$$\{\mathbb{E}(\operatorname{Hess} f(\mathbf{X})[\eta])\}|_{\mathbf{X}=\mathbf{X}^t} = -\operatorname{grad} f(\mathbf{X}^t), \quad \eta \in T_{\mathbf{X}^t}\mathcal{M}_r,$$

which exactly becomes (31) in our setting. Meanwhile, it is not difficult to show that the Fisher scoring algorithm here is equivalent to the Riemannian Gauss–Newton method for solving nonlinear least squares; see Lange (2010, section 14.6) and Absil et al. (2008, section 8.4). Thus, η that solves Equation (31) is also the Riemannian Gauss–Newton direction.

It turns out that the update direction η^t (25) of RISRO solves the Fisher scoring or Riemannian Gauss–Newton equation (31).

Theorem 2. *Let $\{\mathbf{X}^t\}$ be the sequence generated by RISRO under the same assumptions as in Theorem 1. Then, for all $t \geqslant 0$, the implicitly encoded update direction η^t in (25) solves the Riemannian Gaus–Newton equation (31).*

Theorem 2 together with the retraction explanation in (26) establishes the connection of RISRO and Riemannian manifold optimization. Following this connection, we further show that each η_t is always a decent direction in the next Proposition 3. This fact is useful in boosting the local convergence of RISRO to the global convergence discussed in Remark 11.

Proposition 3. *For all $t \geqslant 0$, the update direction $\eta^t \in T_{\mathbf{X}^t}\mathcal{M}_r$ in (25) satisfies $\langle \operatorname{grad} f(\mathbf{X}^t), \eta^t\rangle < 0$, that is, η_t is a descent direction. If $\mathcal{A}$ satisfies the 2r-RIP, then the direction sequence $\{\eta^t\}$ is gradient related.*

Remark 9. The convergence of Riemannian Gauss–Newton is studied in a recent work (Breiding and Vannieuwenhoven 2018). Our results are significantly different from and offer improvements to Breiding and Vannieuwenhoven (2018) in the following ways. First, Breiding and Vannieuwenhoven (2018) consider a more general Riemannian Gauss–Newton setting, whereas their convergence results are established for a local minimum, which is a stronger and less practical requirement than

the stationary point assumption we need. Second, the convergence rate in Breiding and Vannieuwenhoven (2018) includes several unspecified constants, whereas we manage to work out all constants explicitly in our statement. Third, the local convergence radius in Breiding and Vannieuwenhoven (2018) does not specify the dependence on the rth singular value of the target matrix, whereas our result does. Fourth, our recursive importance sketching framework provides new sketching interpretations for several classic algorithms for rank-constrained least squares. Finally, in Section 6, we also apply RISRO in popular statistical models and show RISRO achieves quadratic convergence in terms of estimation. It is, however, not immediately clear how to utilize the results in Breiding and Vannieuwenhoven (2018) in these statistical settings.

Remark 10. In addition to providing an interpretation of the superiority of RISRO, the Riemannian Gauss–Newton perspective developed in this section can inspire algorithmic developments in more general settings. For example, consider a general constrained optimization programming: $\min_{\mathbf{X}\in\mathcal{M}} f(\mathbf{X})$, where $\mathcal{M}$ is an embedded submanifold of R^N and f is the restriction of a general twice-differentiable objective in the ambient space to $\mathcal{M}$. Although importance sketching is hard to define for this setting, the Riemannian Gauss–Newton equation inspires to compute $\eta \in T_{\mathbf{X}^t}\mathcal{M}$ by solving $P_{T_{\mathbf{X}^t}}\nabla^2 f(\mathbf{X}^t)[\eta] = -\operatorname{grad} f(\mathbf{X}^t)$ and then updating the iterate as $\mathbf{X}^{t+1} = R(\mathbf{X}^t, \eta)$, where $R(\cdot,\cdot)$ is a retraction operator onto $\mathcal{M}$. It is interesting to investigate the behavior of this algorithm from both optimization and statistical perspectives.

Meanwhile, the recursive sketching perspective also provides solutions to a wider range of constrained optimization problems. For example, one can replace the l_2 loss, that is, the least squares in Equation (5), by other loss functions, such as the l_1 loss, Huber loss, or logistic loss, to handle different types of error corruptions and develop more robust algorithms.

Remark 11 (Global Convergence of RISRO). By the classic theory of Riemannian optimization, the established connection of RISRO and Riemannian Gauss–Newton implies that vanilla RISRO may not converge when the RIP or initialization condition fails. On the other hand, Absil et al. (2008, section 8.4) suggests that, by adding or modifying the algorithm with certain line search or trust-region schemes, global convergence of Riemannian Gauss–Newton from any initialization to a stationary point can be guaranteed under proper assumptions. To be more specific, based on the Riemannian Gauss–Newton equation in (31) and Theorem 2, the Riemannian Gauss–Newton direction at iteration t satisfies

$$\begin{aligned}\eta^t &= \arg\min_{\eta\in T_{\mathbf{X}^t}\mathcal{M}_r} \|\mathcal{A}P_{T_{\mathbf{X}^t}}(\mathbf{X}^t+\eta)-\mathbf{y}\|_2^2 \\ &= (P_{T_{\mathbf{X}^t}}\mathcal{A}^*\mathcal{A}P_{T_{\mathbf{X}^t}})^{-1}P_{T_{\mathbf{X}^t}}\mathcal{A}^*(\mathbf{y}-\mathcal{A}^*(\mathbf{X}^t)). \qquad (32)\end{aligned}$$

After calculating η^t, we can update $\mathbf{X}^t$ to $\mathbf{X}^t+\eta^t$.

We can equip the algorithm with line search and update $\mathbf{X}^t$ to $\mathbf{X}^t+\alpha_t\eta^t$, where α_t is determined by some line search scheme, such as the Armijo method (Absil et al. 2008, section 4.3). Because the update direction η^t is gradient-related as shown in Proposition 3 under the RIP condition, this modified line search method has a guaranteed global convergence property as shown in Absil et al. (2008, theorem 4.3.1).

We can also apply the trust region method to achieve global convergence. Specifically, we calculate the update direction as

$$\tilde{\eta}^t = \arg\min_{\eta\in T_{\mathbf{X}^t}\mathcal{M}_r,\,\eta\leqslant\Delta_t} \|\mathcal{A}P_{T_{\mathbf{X}^t}}(\mathbf{X}^t+\eta)-\mathbf{y}\|_2^2$$

for some radius $\Delta_t > 0$. Then, if Δ_t is properly chosen such that $\tilde{\eta}^t$ guarantees sufficient decrease, the global convergence of this trust region method can be achieved under proper assumptions (Absil et al. 2008, theorem 7.4.2).

5. Computational Complexity of RISRO

In this section, we discuss the computational complexity of RISRO. Suppose $p_1 = p_2 = p$ and the computational complexity of RISRO per iteration is $O(np^2r^2 + (pr)^3)$ in the general setting. A comparison of the computational complexity of RISRO and other common algorithms is provided in Table 1. Here, the main complexity of RISRO and alter mini is from solving the least squares. The main complexity of the singular value projection (SVP) (Jain et al. 2010) and gradient descent (Tu et al. 2016) is from computing the gradient. From Table 1, we can see that RISRO has the same per-iteration complexity as alter mini and comparable complexity with SVP and GD when $n \geqslant pr$, and r is much less than n and p. On the other hand, RISRO and alter mini are tuning-free, whereas a proper step size is crucial for SVP and GD to have fast convergence: the convergence theory of SVP and GD are often established when the step size is chosen to be smaller than a hard-to-find threshold; there are several practical ways to determine this step size, and one needs to select the best one based on the

Table 1. Computational Complexity per Iteration and Convergence Rate for Alter Mini (Jain et al. 2013), SVP (Jain et al. 2010), GD (Tu et al. 2016), and RISRO

	Alter mini	SVP	GD	RISRO (this work)
Complexity per iteration	$O(np^2r^2+(pr)^3)$	$O(np^2)$	$O(np^2)$	$O(np^2r^2+(pr)^3)$
Convergence rate	Linear	Linear	Linear	Quadratic-(linear)

data (Zheng and Lafferty 2015), which may cost extra time. Finally, RISRO enjoys a high-order convergence as we show in Section 3, and the convergence rates of all other algorithms are limited to being linear.

The main computational bottleneck of RISRO is solving the least squares, which can be alleviated by using iterative linear system solvers, such as the (preconditioned) conjugate gradient method when the linear operator $\mathcal{A}$ has special structures. Such special structures occur, for example, in the matrix completion problem ($\mathcal{A}$ is sparse) (Vandereycken 2013), phase retrieval for X-ray crystallography imaging ($\mathcal{A}$ involves fast Fourier transforms) (Huang et al. 2017b), and blind deconvolution for imaging deblurring ($\mathcal{A}$ involves fast Fourier transforms and Haar wavelet transforms) (Huang and Hand 2018).

To utilize these structures, we introduce an intrinsic representation of tangent vectors in $\mathcal{M}_r$: if $\mathbf{U}, \mathbf{V}$ are the left and right singular vectors of a rank-r matrix $\mathbf{X}$, an orthonormal basis of $T_{\mathbf{X}}\mathcal{M}_r$ can be

$$\begin{aligned}
&\left\{[\mathbf{U} \quad \mathbf{U}_\perp]\begin{bmatrix} \mathbf{e}_i\mathbf{e}_j^\top & \mathbf{0}_{r\times(p-r)} \\ \mathbf{0}_{(p-r)\times r} & \mathbf{0}_{(p-r)\times(p-r)} \end{bmatrix}[\mathbf{V} \quad \mathbf{V}_\perp]^\top, \right. \\
&\qquad\qquad \left. i = 1,\dots,r, j = 1,\dots,r \right\} \cup \\
&\left\{[\mathbf{U} \quad \mathbf{U}_\perp]\begin{bmatrix} \mathbf{0}_{r\times r} & \mathbf{e}_i\tilde{\mathbf{e}}_j^\top \\ \mathbf{0}_{(p-r)\times r} & \mathbf{0}_{(p-r)\times(p-r)} \end{bmatrix}[\mathbf{V} \quad \mathbf{V}_\perp]^\top, \right. \\
&\qquad\qquad \left. i = 1,\dots,r, j = 1,\dots,p-r \right\} \cup \\
&\left\{[\mathbf{U} \quad \mathbf{U}_\perp]\begin{bmatrix} \mathbf{0}_{r\times r} & \mathbf{0}_{r\times(p-r)} \\ \tilde{\mathbf{e}}_i\mathbf{e}_j^\top & \mathbf{0}_{(p-r)\times(p-r)} \end{bmatrix}[\mathbf{V} \quad \mathbf{V}_\perp]^\top, \right. \\
&\qquad\qquad \left. i = 1,\dots,p-r, j = 1,\dots,r \right\},
\end{aligned}$$

where $\mathbf{e}_i$ and $\tilde{\mathbf{e}}_i$ denote the ith canonical basis of $\mathbb{R}^r$ and $\mathbb{R}^{p-r}$, respectively. It follows that any tangent vector in $T_{\mathbf{X}}\mathcal{M}_r$ can be uniquely represented by a coefficient vector in $\mathbb{R}^{(2p-r)r}$ via this basis. This representation is called the intrinsic representation (Huang et al. 2017a). Computing the intrinsic representations of a Riemannian gradient can be computationally efficient. For example, the complexity of computing the Riemannian gradient in matrix completion is $O(nr + pr^2)$, and its intrinsic representation can be computed by additional $O(pr^2)$ operations (Vandereycken 2013). The complexities of computing intrinsic representations of the Riemannian gradients of the phase retrieval and blind deconvolution are both $O(n\log(n)r + pr^2)$ (Huang et al. 2017b, Huang and Hand 2018).

By Theorem 2, the least squares problem (5) of RISRO is equivalent to solving $\eta \in T_{\mathbf{X}^t}\mathcal{M}_r$ such that $P_{T_{\mathbf{X}^t}}\mathcal{A}^*(\mathcal{A}(\eta)) = -\text{grad}f(\mathbf{X}^t)$. Reformulating this equation by intrinsic representation yields

$$-\text{grad}\, f(\mathbf{X}^t) = P_{T_{\mathbf{X}^t}}\mathcal{A}^*(\mathcal{A}(\eta)) \Rightarrow -u = \mathcal{B}_{\mathbf{X}}^*(\mathcal{A}^*(\mathcal{A}(\mathcal{B}_{\mathbf{X}}v))), \quad (33)$$

where u, v are the intrinsic representations of $\text{grad}f(\mathbf{X}^t)$ and η, the mapping $\mathcal{B}_{\mathbf{X}} : \mathbb{R}^{(2p-r)r} \to T_{\mathbf{X}}\mathcal{M}_r \subset \mathbb{R}^{p\times p}$ converts an intrinsic representation to the corresponding tangent vector, and $\mathcal{B}_{\mathbf{X}}^* : \mathbb{R}^{p\times p} \to \mathbb{R}^{(2p-r)r}$ is the adjoint operator of $\mathcal{B}_{\mathbf{X}}$. The computational complexity of using the conjugate gradient method to solve (33) is determined by the complexity of evaluating the operator $\mathcal{B}_{\mathbf{X}}^* \circ (\mathcal{A}^*\mathcal{A}) \circ \mathcal{B}_{\mathbf{X}}$ on a given vector. With the intrinsic representation, it can be shown that this evaluation costs $O(nr + pr^2)$ in matrix completion and $O(n\log(n)r + pr^2)$ in phase retrieval and blind deconvolution. Thus, when solving (33) via the conjugate gradient method, the complexity is $O(k(nr + pr^2))$ in the matrix completion and $O(k(n\log(n)r + pr^2))$ in the phase retrieval and blind deconvolution, where k is the number of conjugate gradient iterations and is provably at most $(2p - r)r$. Hence, for special applications, such as matrix completion, phase retrieval, and blind deconvolution, by using the conjugate gradient method with the intrinsic representation, the per-iteration complexity of RISRO can be greatly reduced. This point will be further exploited in our future research.

6. Recursive Importance Sketching Under Statistical Models

In this section, we study the applications of RISRO in machine learning and statistics. We specifically investigate the low-rank matrix trace regression and phase retrieval, whereas our key ideas can be applied to more problems. For the execution of RISRO, we assume that some estimate for the rank of the target parameter matrix, denoted by r, is available. In many statistical applications, such as phase retrieval and blind deconvolution, this assumption trivially holds as the parameter matrix is known to be rank-one. In other applications, whereas the rank of the parameter is unknown, it is generally not difficult to obtain a rough estimate given the domain knowledge. Then, we can optimize over the set of fixed-rank matrices using the formulation of (1) and dynamically update the selected rank (see, e.g., Vandereycken and Vandewalle 2010, Zhou et al. 2016).

6.1. Low-Rank Matrix Trace Regression

Consider the low-rank matrix trace regression model:

$$\mathbf{y}_i = \langle \mathbf{A}_i, \mathbf{X}^* \rangle + \boldsymbol{\epsilon}_i, \quad \text{for } 1 \leqslant i \leqslant n, \quad (34)$$

where $\mathbf{X}^* \in \mathbb{R}^{p_1\times p_2}$ is the true model parameter to be estimated. We estimate $\mathbf{X}^*$ by solving (1), where r in the rank constraint satisfies $r \leqslant \text{rank}(\mathbf{X}^*)$, that is, r is an estimate of $\text{rank}(\mathbf{X}^*)$.

The following Theorem 3 shows RISRO converges quadratically to the best rank-r approximation of $\mathbf{X}^*$, that is, $\mathbf{X}^*_{\max(r)}$ up to some statistical error given a proper initialization. Under the Gaussian ensemble design, RISRO

with spectral initialization achieves the minimax optimal estimation error rate.

Theorem 3 (RISRO in Matrix Trace Regression). *Consider the low-rank matrix trace regression problem (34). Define* $\tilde{\epsilon}_i := \epsilon_i + \langle \mathbf{A}_i, \mathbf{X}^* - \mathbf{X}^*_{\max(r)} \rangle$ *for* $i = 1, \dots, n$. *Suppose that* $\mathcal{A}$ *satisfies the* $2r$*-RIP, the initialization of RISRO satisfies*

$$\|\mathbf{X}^0 - \mathbf{X}^*_{\max(r)}\|_{\mathrm{F}} \leqslant \left(\frac{1}{4} \wedge \frac{1 - R_{2r}}{2\sqrt{5} R_{3r}}\right) \sigma_r(\mathbf{X}^*), \tag{35}$$

and

$$\sigma_r(\mathbf{X}^*) \geqslant \left(16\sqrt{5} \vee \frac{40\sqrt{2} R_{3r}}{1 - R_{2r}}\right) \frac{\|(\mathcal{A}^*(\tilde{\epsilon}))_{\max(r)}\|_{\mathrm{F}}}{1 - R_{2r}}. \tag{36}$$

Then, the iterations of RISRO converge as follows $\forall\, t \geqslant 0$:

$$\begin{aligned}
&\|\mathbf{X}^{t+1} - \mathbf{X}^*_{\max(r)}\|_{\mathrm{F}}^2 \\
&\leqslant 10 \frac{R_{3r}^2 \|\mathbf{X}^t - \mathbf{X}^*_{\max(r)}\|_{\mathrm{F}}^4}{(1 - R_{2r})^2 \sigma_r^2(\mathbf{X}^*)} + \frac{20 \|(\mathcal{A}^*(\tilde{\epsilon}))_{\max(r)}\|_{\mathrm{F}}^2}{(1 - R_{2r})^2} \\
&\leqslant 10 \frac{R_{3r}^2 \|\mathbf{X}^t - \mathbf{X}^*\|_{\mathrm{F}}^4}{(1 - R_{2r})^2 \sigma_r^2(\mathbf{X}^*)} \\
&\quad + \frac{20 (\|(\mathcal{A}^*(\epsilon))_{\max(r)}\|_{\mathrm{F}} + \|(\mathcal{A}^* \mathcal{A}(\mathbf{X}^* - \mathbf{X}^*_{\max(r)}))_{\max(r)}\|_{\mathrm{F}})^2}{(1 - R_{2r})^2}.
\end{aligned} \tag{37}$$

The overall convergence of RISRO shows two phases:

- *(Phase I) When* $\|\mathbf{X}^t - \mathbf{X}^*_{\max(r)}\|_{\mathrm{F}}^2 \geqslant \frac{\sqrt{2}}{R_{3r}} \|(\mathcal{A}^*(\tilde{\epsilon}))_{\max(r)}\|_{\mathrm{F}} \sigma_r(\mathbf{X}^*)$,

$$\|\mathbf{X}^{t+1} - \mathbf{X}^*_{\max(r)}\|_{\mathrm{F}} \leqslant 2\sqrt{5} \frac{R_{3r} \|\mathbf{X}^t - \mathbf{X}^*_{\max(r)}\|_{\mathrm{F}}^2}{(1 - R_{2r}) \sigma_r(\mathbf{X}^*)},$$

$$\|\mathbf{X}^{t+1} - \mathbf{X}^*\|_{\mathrm{F}} \leqslant 2\sqrt{5} \frac{R_{3r} \|\mathbf{X}^t - \mathbf{X}^*_{\max(r)}\|_{\mathrm{F}}^2}{(1 - R_{2r}) \sigma_r(\mathbf{X}^*)} + \|\mathbf{X}^*_{-\max(r)}\|_{\mathrm{F}},$$

where $\mathbf{X}^*_{-\max(r)} = \mathbf{X}^* - \mathbf{X}^*_{\max(r)}$.

- *(Phase II) When* $\|\mathbf{X}^t - \mathbf{X}^*_{\max(r)}\|_{\mathrm{F}}^2 \leqslant \frac{\sqrt{2}}{R_{3r}} \|(\mathcal{A}^*(\tilde{\epsilon}))_{\max(r)}\|_{\mathrm{F}} \sigma_r(\mathbf{X}^*)$,

$$\|\mathbf{X}^{t+1} - \mathbf{X}^*_{\max(r)}\|_{\mathrm{F}} \leqslant \frac{2\sqrt{10} \|(\mathcal{A}^*(\tilde{\epsilon}))_{\max(r)}\|_{\mathrm{F}}}{1 - R_{2r}},$$

$$\|\mathbf{X}^{t+1} - \mathbf{X}^*\|_{\mathrm{F}} \leqslant \frac{2\sqrt{10} \|(\mathcal{A}^*(\tilde{\epsilon}))_{\max(r)}\|_{\mathrm{F}}}{1 - R_{2r}} + \|\mathbf{X}^*_{-\max(r)}\|_{F}.$$

Moreover, we assume $\mathrm{rank}(\mathbf{X}^*) = r$, $(\mathbf{A}_i)_{[j,k]}$ *are independent sub-Gaussian random variables with mean zero and variance* $1/n$ *and* ϵ_i *are independent sub-Gaussian random variables with mean zero and variance* σ^2/n *(i.e.,* $\mathbb{E}(\mathbf{A}_i)_{[j,k]} = \mathbb{E}(\epsilon_i) = 0$, $\mathrm{Var}((\mathbf{A}_i)_{[j,k]}) = 1/n$, $\mathrm{Var}(\epsilon_i) = \sigma^2/n$, $\sup_{q \geqslant 1} (n/q)^{1/2} (\mathbb{E}|(\mathbf{A}_i)_{[j,k]}|^q)^{1/q} \leqslant C$, $\sup_{q \geqslant 1} (n/(q\sigma^2))^{1/2} (\mathbb{E}|\epsilon_i|^q)^{1/q} \leqslant C$ *for some fixed* $C > 0$*). Then, there exist universal constants* $C_1, C_2, C', c > 0$ *such that as long as* $n \geqslant C_1(p_1 + p_2) r \left(\frac{\sigma^2}{\sigma_r^2(\mathbf{X}^*)} \vee r\kappa^2\right)$ *(here,* $\kappa = \frac{\sigma_1(\mathbf{X}^*)}{\sigma_r(\mathbf{X}^*)}$ *is the condition number of* $\mathbf{X}^*$*) and* $t_{\max} \geqslant C_2 \log\log\left(\frac{\sigma_r^2(\mathbf{X}^*) n}{r(p_1+p_2)\sigma^2}\right) \vee 1$, *the output of RISRO with spectral initialization* $\mathbf{X}^0 = (\mathcal{A}^*(\mathbf{y}))_{\max(r)}$ *satisfies* $\|\mathbf{X}^{t_{\max}} - \mathbf{X}^*\|_{\mathrm{F}}^2 \leqslant c \frac{r(p_1+p_2)}{n} \sigma^2$ *with probability at least* $1 - \exp(-C'(p_1 + p_2))$.

Remark 12 (Quadratic Convergence, Two-Phase Convergence, Statistical Error, and Robustness). The upper bound of $\|\mathbf{X}^t - \mathbf{X}^*_{\max(r)}\|_{\mathrm{F}}^2$ in (37) includes two terms: the optimization error term $O(\|\mathbf{X}^t - \mathbf{X}^*_{\max(r)}\|_F^4)$ quadratically decreases over iteration t, and the statistical error term $O(\|(\mathcal{A}^*(\tilde{\epsilon}))_{\max(r)}\|_{\mathrm{F}}^2)$ is static through iterations. Moreover, RISRO includes two phases in its convergence. In phase I with large $\|\mathbf{X}^t - \mathbf{X}^*_{\max(r)}\|_{\mathrm{F}}^2$, RISRO converges quadratically toward $\mathbf{X}^*_{\max(r)}$; in phase II with moderate $\|\mathbf{X}^t - \mathbf{X}^*_{\max(r)}\|_{\mathrm{F}}^2$, the estimator returned by one more iteration of RISRO achieves the best possible statistical error rate $O(\|(\mathcal{A}^*(\tilde{\epsilon}))_{\max(r)}\|_{\mathrm{F}}^2)$ as suggested by the $d = 2$ case in Luo and Zhang (2021, theorem 2). Therefore, although the convergence rate of RISRO may decelerate to be linear in phase II, Theorem 3 suggests there is no need to run further iterations as the estimator is already statistically optimal after one additional iteration. Such performance of "quadratic convergence + one-iteration optimality" is unique, which does not appear in common first order methods.

Finally, (37) shows the error contraction factor is independent of the condition number κ, which demonstrates the robustness of RISRO to the ill-conditioning of the underlying low-rank matrix. We further demonstrate this point by simulation studies in Section 7.2.

Remark 13 (Optimal Statistical Error). Under the Gaussian ensemble design and when $\mathrm{rank}(\mathbf{X}^*) = r$, RISRO with spectral initialization achieves the rate of estimation error $cr(p_1 + p_2)\sigma^2/n$ after a double-logarithmic number of iterations when $n \geqslant C_1(p_1 + p_2)\, r\left(\frac{\sigma^2}{\sigma_r^2(\mathbf{X}^*)} \vee r\kappa^2\right)$. Compared with the lower bound of the estimation error

$$\min_{\hat{\mathbf{X}}} \max_{\mathrm{rank}(\mathbf{X}^*) \leqslant r} \mathbb{E}\|\hat{\mathbf{X}} - \mathbf{X}^*\|_{\mathrm{F}}^2 \geqslant c' \frac{r(p_1 + p_2)\sigma^2}{n}$$

for some $c' > 0$ in Candès and Plan (2011), RISRO achieves the minimax optimal estimation error with near-optimal sample complexity. To the best of our knowledge, RISRO is the first provable algorithm that achieves the minimax rate-optimal estimation error with only a double-logarithmic number of iterations, and this is an exponential improvement over common first order methods in which a logarithmic number of iterations is needed.

6.2. Phase Retrieval

In this section, we consider RISRO for solving the following quadratic equation system:

$$\mathbf{y}_i = |\langle \mathbf{a}_i, \mathbf{x}^* \rangle|^2 \quad \text{for} \quad 1 \leqslant i \leqslant n, \tag{38}$$

where $\mathbf{y} \in \mathbb{R}^n$ and covariates $\{\mathbf{a}_i\}_{i=1}^n \in \mathbb{R}^p$ (or $\mathbb{C}^p$) are known, whereas $\mathbf{x}^* \in \mathbb{R}^p$ (or $\mathbb{C}^p$) are unknown. The goal is to recover $\mathbf{x}^*$ based on $\{\mathbf{y}_i, \mathbf{a}_i\}_{i=1}^n$. One important application is known as phase retrieval, arising from physical science because of the nature of optical sensors (Fienup 1982). In the literature, various approaches are proposed for phase retrieval with provable guarantees, such as convex relaxation (Candès et al. 2013, Waldspurger et al. 2015, Huang et al. 2017b) and nonconvex approaches (Netrapalli et al. 2013, Candès et al. 2015, Chen and Candès 2017, Gao and Xu 2017, Sanghavi et al. 2017, Wang et al. 2017a, Duchi and Ruan 2019, Ma et al. 2019).

For ease of exposition, we focus on the real-value model, that is, $\mathbf{x}^* \in \mathbb{R}^n$ and $\mathbf{a}_i \in \mathbb{R}^n$, whereas a simple trick in Sanghavi et al. (2017) can recast Problem (38) in the complex model into a rank-two real-value matrix recovery problem; then, our approach still applies. In the real-valued setting, we can rewrite Model (38) into a low-rank matrix recovery model:

$$\mathbf{y} = \mathcal{A}(\mathbf{X}^*) \text{ with } \mathbf{X}^* = \mathbf{x}^*\mathbf{x}^{*\top} \text{ and } [\mathcal{A}(\mathbf{X}^*)]_i = \langle \mathbf{a}_i\mathbf{a}_i^\top, \mathbf{x}^*\mathbf{x}^{*\top}\rangle. \tag{39}$$

There are two challenges in phase retrieval compared with the low-rank matrix trace regression considered previously. First, because of the symmetry of sensing matrices $\mathbf{a}_i\mathbf{a}_i^\top$ and $\mathbf{x}^*\mathbf{x}^{*\top}$ in phase retrieval, the importance covariates $\mathcal{A}_{D_1}$ and $\mathcal{A}_{D_2}$ in (4) are exactly the same, and an adaptation of Algorithm 1 is, thus, needed. Second, in phase retrieval, the mapping $\mathcal{A}$ no longer satisfies a proper RIP condition in general (Candès et al. 2013, Cai and Zhang 2015), so a new theory is needed. To this end, we introduce a modified RISRO for phase retrieval in Algorithm 2. Particularly in step 4 of Algorithm 2, we multiply the importance covariates $\mathbf{A}_2$ by an extra factor 2 to account for the duplicate importance covariates because of symmetry.

Algorithm 2 (RISRO for Phase Retrieval)

1: Input: design vectors $\{\mathbf{a}_i\}_{i=1}^n \in \mathbb{R}^p$, $\mathbf{y} \in \mathbb{R}^n$, initialization $\mathbf{X}^0$ that admits eigenvalue decomposition $\sigma_1^0\mathbf{u}^0\mathbf{u}^{0\top}$.
2: **for** $t = 0, 1, \ldots,$ **do**
3: Perform importance sketching on $\mathbf{a}_i$ and construct the covariates $\mathbf{A}_1 \in \mathbb{R}^n$, $\mathbf{A}_2 \in \mathbb{R}^{n\times(p-1)}$, where for $1 \leqslant i \leqslant n$, $(\mathbf{A}_1)_i = (\mathbf{a}_i^\top\mathbf{u}^t)^2$, $(\mathbf{A}_2)_{[i,:]} = \mathbf{u}_\perp^{t\top}\mathbf{a}_i\mathbf{a}_i^\top\mathbf{u}^t$.
4: Solve the unconstrained least squares problem $(b^{t+1}, \mathbf{d}^{t+1}) = \arg\min_{b\in\mathbb{R},\mathbf{d}\in\mathbb{R}^{(p-1)}} \|\mathbf{y} - \mathbf{A}_1 b - 2\mathbf{A}_2\mathbf{d}\|_2^2$.
5: Compute the eigenvalue decomposition of $[\mathbf{u}^t\ \mathbf{u}_\perp^t]\begin{bmatrix} b^{t+1} & \mathbf{d}^{t+1\top} \\ \mathbf{d}^{t+1} & \mathbf{0} \end{bmatrix}[\mathbf{u}^t\ \mathbf{u}_\perp^t]^\top$, and denote it as $[\mathbf{v}_1\ \mathbf{v}_2]\begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}[\mathbf{v}_1\ \mathbf{v}_2]^\top$ with $\lambda_1 \geqslant \lambda_2$.
6: Update $\mathbf{u}^{t+1} = \mathbf{v}_1$ and $\mathbf{X}^{t+1} = \lambda_1\mathbf{u}^{t+1}\mathbf{u}^{t+1\top}$.
7: **end for**

Next, we show, under Gaussian ensemble design, given the sample number $n = O(p \log p)$ and proper initialization, the sequence $\{\mathbf{X}^t\}$ generated by Algorithm 2 converges quadratically to $\mathbf{X}^*$.

Theorem 4 (Local Quadratic Convergence of RISRO for Phase Retrieval). *In the phase retrieval problem (38), assume that $\{\mathbf{a}_i\}_{i=1}^n$ are independently generated from $N(0, \mathbf{I}_p)$. Then, for any $\delta_1, \delta_2 \in (0,1)$, there exist $c, C(\delta_1), C' > 0$ such that, when $p \geqslant c\log n, n \geqslant C(\delta_1)p\log p$, if $\|\mathbf{X}^0 - \mathbf{X}^*\|_F \leqslant \frac{(1-\delta_1)}{C'(1+\delta_2)p}\|\mathbf{X}^*\|_F$ with probability at least $1 - C_1\exp(-C_2(\delta_1,\delta_2)n) - C_3 n^{-p}$, the sequence $\{\mathbf{X}^t\}$ generated by Algorithm 2 satisfies*

$$\|\mathbf{X}^{t+1} - \mathbf{X}^*\|_F \leqslant \frac{C'(1+\delta_2)p}{(1-\delta_1)\|\mathbf{X}^*\|_F}\|\mathbf{X}^t - \mathbf{X}^*\|_F^2, \quad \forall t \geqslant 0 \tag{40}$$

for some $C_1, C_2(\delta_1,\delta_2), C_3 > 0$.

To overcome the technical difficulties in establishing quadratic convergence without RIP for phase retrieval, Theorem 4 is established under the assumption $\|\mathbf{X}^0 - \mathbf{X}^*\|_F \leqslant O(\|\mathbf{X}^*\|_F/p)$. Although it is difficult to prove that the spectral initializer meets this assumption under the near-optimal sample size (e.g., $n = Cp\log p$), we find by simulation that the spectral initialization yields quadratic convergence for RISRO (Section 7). On the other hand, we can also run a few iterations of factorized gradient descent to achieve the initialization condition in Theorem 4 with near-optimal sample complexity guarantee (Candès et al. 2015, Chen and Candès 2017, Ma et al. 2019) and then switch to RISRO. Specifically, the initialization algorithm for RISRO in phase retrieval via factorized gradient descent is provided in Algorithm 3, and its guarantee is given in Proposition 4.

Algorithm 3 (RISRO for Phase Retrieval with Gradient Descent Initialization)

1: Input: design vectors $\{\mathbf{a}_i\}_{i=1}^n \in \mathbb{R}^p$ and $\mathbf{y} \in \mathbb{R}^n$.
2: Let $\lambda_1(\mathbf{Y})$ and $\mathbf{v}_1$ be the leading eigenvalue and eigenvector of $\mathbf{Y} = \frac{1}{n}\sum_{j=1}^n \mathbf{y}_j\mathbf{a}_j\mathbf{a}_j^\top$, respectively, and set $\tilde{\mathbf{x}}^0 = \sqrt{\lambda_1(\mathbf{Y})/3}\mathbf{v}_1$.
3: **for** $t = 0, 1, \ldots, T_0 - 1$ **do**
4: Update $\tilde{\mathbf{x}}^{t+1} = \tilde{\mathbf{x}}^t - \eta_t\nabla g(\tilde{\mathbf{x}}^t)$, where $g(\mathbf{x}) = \frac{1}{4n}\sum_{j=1}^n((\mathbf{a}_j^\top\mathbf{x})^2 - \mathbf{y}_j)^2$.
5: **end for**
6: Apply Algorithm 2 with initialization $\tilde{\mathbf{x}}^{T_0}\tilde{\mathbf{x}}^{T_0\top}$.

Proposition 4. *In phase retrieval (38), suppose $\{\mathbf{a}_i\}_{i=1}^n$ are independently drawn from $N(0, \mathbf{I}_p)$ and $n \geqslant Cp\log p$ for some sufficiently large constant $C > 0$. Assume the step size in Algorithm 3 obeys $\eta_t \equiv \eta = c_1/(\log p \cdot \|\tilde{\mathbf{x}}^0\|_2^2)$ for constant $c_1 > 0$, where $\tilde{\mathbf{x}}^0$ is given in the algorithm. Then, there exist absolute constants $c_2, c_3 > 0$ such that, when $T_0 \geqslant c_2\log p \cdot \log(\|\mathbf{x}^*\|_2 p)$, the initialization $\mathbf{X}^0 := \tilde{\mathbf{x}}^{T_0}\tilde{\mathbf{x}}^{T_0\top}$ in Algorithm 3 satisfies the initialization condition in Theorem 4, and the conclusion of Theorem 4 holds with probability at least $1 - c_3 np^{-5}$.*

7. Numerical Studies

In this section, we conduct simulation studies to investigate the numerical performance of RISRO. We specifically consider two settings:

- Matrix trace regression. Let $p = p_1 = p_2$ and $\mathbf{y}_i = \langle \mathbf{X}^*, \mathbf{A}_i \rangle + \boldsymbol{\epsilon}_i$, where $\mathbf{A}_i$s are constructed with independent standard normal entries and $\boldsymbol{\epsilon}_i \overset{i.i.d.}{\sim} N(0, \sigma^2)$. $\mathbf{X}^* = \mathbf{U}^* \boldsymbol{\Sigma}^* \mathbf{V}^{*\top}$, where $\mathbf{U}^*, \mathbf{V}^* \in \mathbb{O}_{p,r}$ are randomly generated, $\boldsymbol{\Sigma} = \mathrm{diag}(\lambda_1, \ldots, \lambda_r)$. Also, we set $\lambda_1 = 3$ and $\lambda_i = \frac{\lambda_1}{\kappa^{i/r}}$ for $i = 2, \ldots, r$, so the condition number of $\mathbf{X}^*$ is κ. We initialize $\mathbf{X}^0$ via $(\mathcal{A}^*(\mathbf{y}))_{\max(r)}$.
- Phase retrieval. Let $\mathbf{y}_i = \langle \mathbf{a}_i, \boldsymbol{x}^* \rangle^2$, where $\boldsymbol{x}^* \in \mathbb{R}^p$ is a randomly generated unit vector, $\mathbf{a}_i \overset{i.i.d.}{\sim} N(0, \mathbf{I}_p)$. We initialize $\mathbf{X}^0$ via truncated spectral initialization (Chen and Candès 2017).

Throughout the simulation studies, we consider errors in two metrics: (1) $\|\mathbf{X}^t - \mathbf{X}^{t_{\max}}\|_{\mathrm{F}} / \|\mathbf{X}^{t_{\max}}\|_{\mathrm{F}}$, which measures the convergence error, and (2) $\|\mathbf{X}^t - \mathbf{X}^*\|_{\mathrm{F}} / \|\mathbf{X}^*\|_{\mathrm{F}}$, which is the relative root mean-squared error (RMSE) that measures the estimation error for $\mathbf{X}^*$. The algorithm is terminated when it reaches the maximum number of iterations $t_{\max} = 300$ or the corresponding error metric is less than 10^{-12}. Unless otherwise noted, the reported results are based on the averages of 50 simulations and on a computer with Intel Xeon E5-2680 2.5 GHz CPU. Additional development and simulation results for RISRO in matrix completion and robust PCA can be found in Online Appendix A.

7.1. Properties of RISRO

We first study the convergence rate of RISRO. Specifically, set $p = 100, r = 3, n \in \{1{,}200, 1{,}500, 1{,}800, 2{,}100, 2{,}400\}$, $\kappa = 1, \sigma = 0$ for low-rank matrix trace regression and $p = 1{,}200, n \in \{4{,}800, 6{,}000, 7{,}200, 8{,}400, 9{,}600\}$ for phase retrieval. The convergence performance of RISRO (Algorithm 1 in low-rank matrix trace regression and Algorithm 2 in phase retrieval) is plotted in Figure 1. We can see that RISRO with the (truncated) spectral initialization converges quadratically to the true parameter $\mathbf{X}^*$ in both problems, which is in line with the theory developed in previous sections. Although our theory on phase retrieval in Theorem 4 is based on a stronger initialization assumption, the truncated spectral initialization achieves great empirical performance.

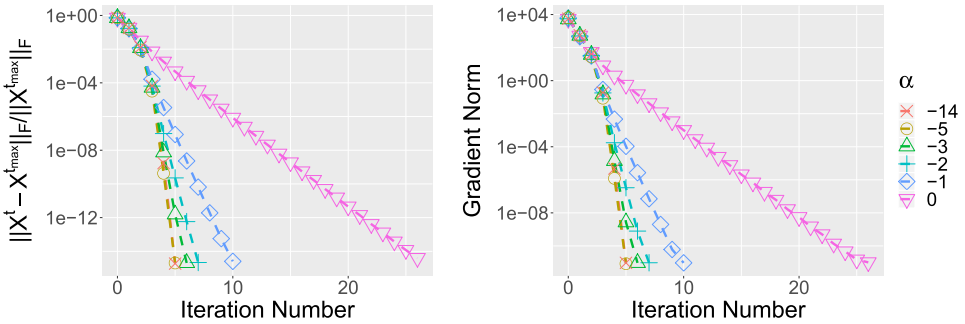
In another setting, we examine the quadratic-linear convergence for RISRO under the noisy setting. Consider the matrix trace regression problem, in which $\sigma = 10^{\alpha}$, $\alpha \in \{0, -1, -2, -3, -5, -14\}$, $n = 1{,}500$, and p, r, κ are the same as the previous setting. The simulation results in Figure 3 show the gradient norm $\|\mathrm{grad} f(\mathbf{X}^t)\|$ of the iterates converges to zero, which demonstrates the convergence of the algorithm. Meanwhile, because the observations are noisy, RISRO exhibits the quadratic-linear convergence as discussed in Remark 4: when $\alpha = 0$, that is, $\sigma = 1$, RISRO converges quadratically in the first two to three steps and then reduces to linear convergence afterward; as σ gets smaller, we can see RISRO enjoys a longer path of quadratic convergence, which matches our theoretical prediction in Remark 4.

Finally, we study the performance of RISRO under the large-scale setting of the matrix trace regression. Fix $n = 7{,}000, r = 3, \kappa = 1, \sigma = 0$ and let dimension p grow from 100 to 500. For the largest case, the space cost of storing $\mathcal{A}$ reaches $7{,}000 \cdot 500 \cdot 500 \cdot 8\mathrm{B} = 13.04\mathrm{GB}$. Figure 4 shows the relative RMSE of the output of RISRO and runtime versus dimension. We can clearly see the relative RMSE of the output is stable, and the runtime scales reasonably well as the dimension p grows.

7.2. Comparison of RISRO with Other Algorithms in Literature

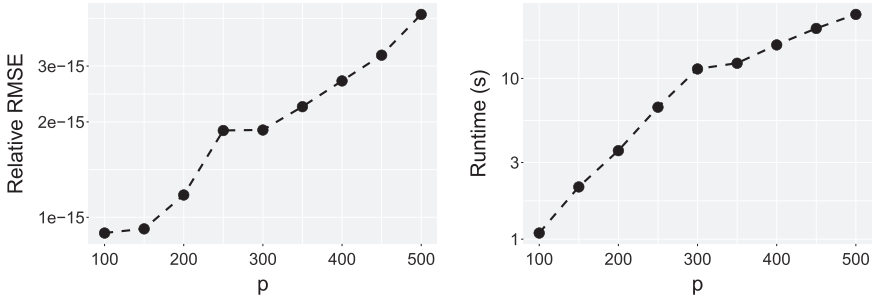
In this section, we further compare RISRO with existing algorithms in the literature. In the matrix trace regression, we compare our algorithm with SVP (Jain et al. 2010, Goldfarb and Ma 2011), alter mini (Jain et al. 2013, Zhao et al. 2015), GD (Zheng and Lafferty 2015, Tu et al. 2016, Park et al. 2018), and convex NNM (3) (Toh and Yun 2010). We consider the setting with $p = 100, r = 3$, $n = 1{,}500$, $\kappa \in \{1, 50, 500\}$, $\sigma = 0$ (noiseless case), or $\sigma = 10^{-6}$ (noisy case). Following Zheng and Lafferty (2015), in the implementation of GD and SVP, we evaluate three choices of step size, $\{5 \times 10^{-3}, 10^{-3}, 5 \times 10^{-4}\}$, then choose the best one. In phase retrieval, we compare

Figure 3. (Color online) Convergence Plot of RISRO in Matrix Trace Regression



Note. $p = 100, r = 3, n = 1{,}500, \kappa = 1, \sigma = 10^{\alpha}$ with varying α.

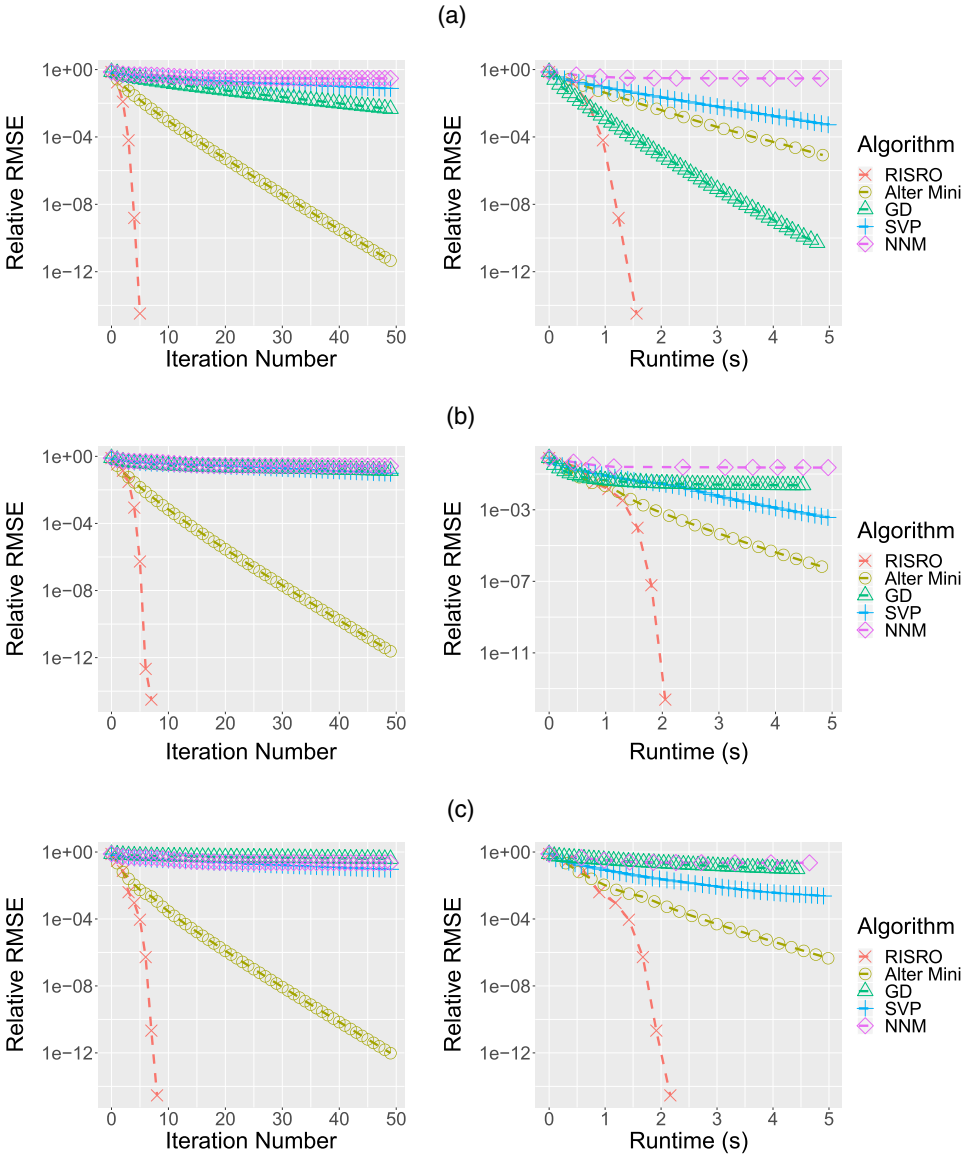
Figure 4. Relative RMSE and Runtime of RISRO in Matrix Trace Regression



Note. $p \in [100, 500], r = 3, n = 7{,}000, \kappa = 1, \sigma = 0$.

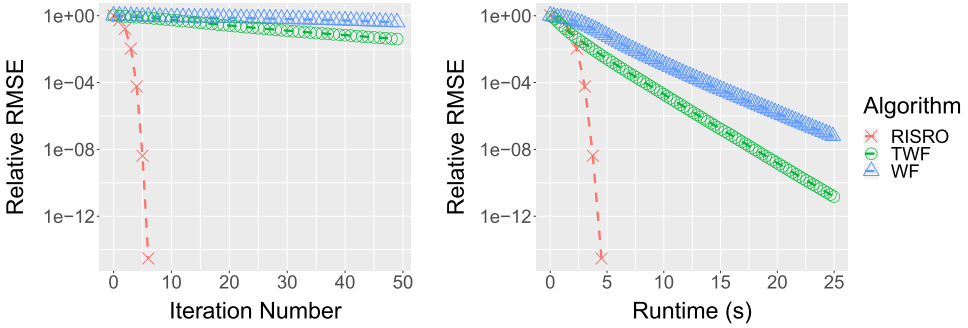
Algorithm 2 with Wirtinger flow (WF) (Candès et al. 2015) and truncated Wirtinger flow (TWF) (Chen and Candès 2017) with $p = 1{,}200, n = 6{,}000$. We use the codes of the accelerated proximal gradient for NNM, WF, and TWF from the corresponding authors' websites and implement the other algorithms by ourselves. The stopping criteria of all procedures are the same as RISRO mentioned in the previous simulation settings.

Figure 5. (Color online) Relative RMSE of RISRO, SVP, Alter Mini, GD, and NNM in Low-Rank Matrix Trace Regression



Notes. Here, $p = 100, r = 3, n = 1{,}500, \sigma = 0, \kappa \in \{1, 50, 500\}$. (a) $\kappa = 1$. (b) $\kappa = 50$. (c) $\kappa = 500$.

Figure 6. (Color online) Relative RMSE of RISRO, WF, and TWF in Phase Retrieval



Note. Here, $p = 1{,}200, n = 6{,}000$.

We compare the performance of various procedures on noiseless matrix trace regression in Figure 5. For all different choices of κ, RISRO converges quadratically to $\mathbf{X}^*$ in seven iterations with high accuracy, whereas the other baseline algorithms converge much slower at a linear rate. When κ (condition number of $\mathbf{X}^*$) increases from 1 to 50 and 500 so that the problem becomes more ill-conditioned, RISRO, alter mini, and SVP perform robustly, whereas GD converges more slowly. In Theorem 3, we show the quadratic convergence rate of RISRO is robust to the condition number (see Remark 12). As we expect, the nonconvex optimization methods converge much faster than the convex relaxation method. Moreover, to achieve a relative RMSE of 10^{-10}, RISRO only takes about five iterations and 1/5 runtime compared with other algorithms if $\kappa = 1$, and this factor is even smaller in the ill-conditioned cases that $\kappa = 50$ and 500.

The comparison of RISRO, WF, and TWF in phase retrieval is plotted in Figure 6. We can also see that RISRO can recover the underlying true signal with high accuracy in much less time than the other baseline methods.

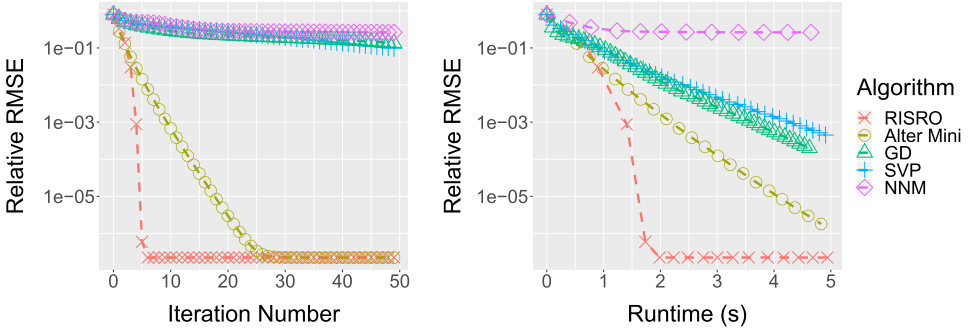
Next, we compare the performance of RISRO with other algorithms in the noisy setting, $\sigma = 10^{-6}$, in the low-rank matrix trace regression. We can see from the results in Figure 7 that, because of the noise, the estimation error first decreases and then stabilizes after reaching a certain level. Meanwhile, we can also find that RISRO converges at a much faster quadratic rate before reaching the stable level compared with all other algorithms.

Finally, we study the required sample size to guarantee successful recovery by RISRO and other algorithms. We set $p = 100, r = 3, \kappa = 5, n \in [600, 1{,}500]$ in the noiseless matrix trace regression and $p = 1{,}200$, $n \in [2{,}400, 6{,}000]$ in phase retrieval. We say the algorithm achieves successful recovery if the relative RMSE is less than 10^{-2} when the algorithm terminates. The simulation results in Figure 8 show RISRO requires the minimum sample size to achieve a successful recovery in both matrix trace regression and phase retrieval, alter mini has similar performance to RISRO, and both RISRO and alter mini require smaller sample sizes than the rest of the algorithms for successful recovery.

8. Conclusion and Discussion

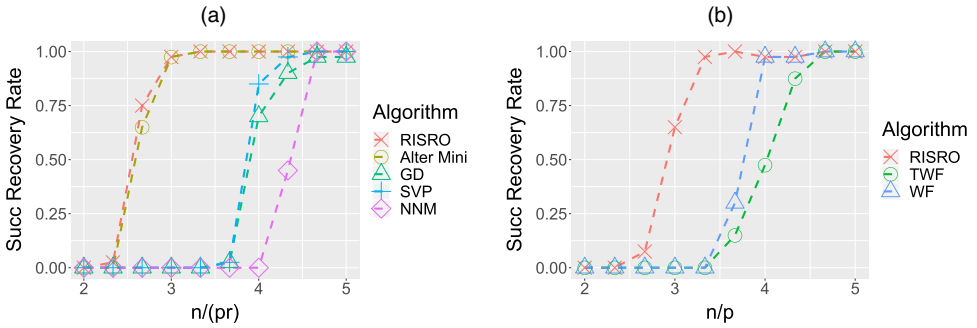
In this paper, we propose a new algorithm, RISRO, for solving rank-constrained least squares. RISRO is based on a novel algorithmic framework, recursive importance sketching, which also provides new sketching interpretations for several existing algorithms for rank-constrained least squares. RISRO is easy to implement and computationally efficient. Under some reasonable assumptions, local quadratic-linear and quadratic convergence are

Figure 7. (Color online) Relative RMSE of RISRO, SVP, Alter Mini, GD, and NNM in Low-Rank Matrix Trace Regression



Note. Here, $p = 100, r = 3, n = 1{,}500, \kappa = 5, \sigma = 10^{-6}$.

Figure 8. (Color online) Successful Recovery Rate Comparison



Notes. (a) Matrix trace regression ($p = 100, r = 3, \sigma = 0, \kappa = 5$). (b) Phase retrieval ($p = 1{,}200$).

established for RISRO. Simulation studies demonstrate the superior performance of RISRO.

The connection of recursive importance sketching and Riemannian Gauss–Newton discovered in this paper can be leveraged to other settings, such as in the low-rank tensor estimation problems (see a follow-up work in Luo and Zhang (2021) after the first preprint of this paper).

There are many interesting extensions to the results in this paper to be explored in the future. First, our current convergence theory on RISRO relies on the RIP assumption, which may not hold in many scenarios, such as phase retrieval, matrix completion, and robust PCA. In this paper, we give some theoretical guarantees of RISRO in phase retrieval with a strong initialization assumption. However, such an initialization requirement may be unnecessary, and spectral initialization is good enough to guarantee quadratic convergence as we observe in the simulation studies. Empirically, we also observe RISRO achieves quadratic convergence in the matrix completion and robust PCA examples; see their development in Online Appendix A. To improve and establish theoretical guarantees for RISRO in phase retrieval and matrix completion or robust PCA, we think more sophisticated analysis tools, such as the "leave-one-out" method, and some extra properties, such as "implicit regularization" (Ma et al. 2019), need to be incorporated into the analysis, and it will be interesting future work. Also, this paper focuses on the squared error loss in (1), whereas the other loss functions may be of interest in different settings, such as the ℓ_1 loss in robust low-rank matrix recovery (Li et al. 2020a,b; Charisopoulos et al. 2021), which is worth exploring.

Acknowledgments

The authors thank the editors and two anonymous reviewers for their suggestions and comments, which helped significantly improve the presentation of this paper.

References

Absil PA, Malick J (2012) Projection-like retractions on matrix manifolds. *SIAM J. Optim.* 22(1):135–158.

Absil PA, Mahony R, Sepulchre R (2008) *Optimization Algorithms on Matrix Manifolds* (Princeton University Press, Princeton, NJ).

Ahmed A, Recht B, Romberg J (2013) Blind deconvolution using convex programming. *IEEE Trans. Inform. Theory* 60(3): 1711–1732.

Bauch J, Nadler B, Zilber P (2021) Rank 2*r* iterative least squares: Efficient recovery of ill-conditioned low rank matrices from few entries. *SIAM J. Math. Data Sci.* 3(1):439–465.

Bhojanapalli S, Neyshabur B, Srebro N (2016) Global optimality of local search for low rank matrix recovery. *Adv. Neural Inform. Processing Systems* 30:3873–3881.

Boumal N, Absil PA (2011) RTRMC: A Riemannian trust-region method for low-rank matrix completion. *Adv. Neural Inform. Processing Systems* 24:406–414.

Breiding P, Vannieuwenhoven N (2018) Convergence analysis of Riemannian Gauss–Newton methods and its connection with the geometric condition number. *Appl. Math. Lett.* 78:42–50.

Burer S, Monteiro RD (2003) A nonlinear programming algorithm for solving semidefinite programs via low-rank factorization. *Math. Programming* 95(2):329–357.

Burke JV (1985) Descent methods for composite nondifferentiable optimization problems. *Math. Programming* 33(3):260–279.

Cai TT, Zhang A (2013) Sharp RIP bound for sparse signal and low-rank matrix recovery. *Appl. Comput. Harmonic Anal.* 35(1):74–93.

Cai TT, Zhang A (2014) Sparse representation of a polytope and recovery of sparse signals and low-rank matrices. *IEEE Trans. Inform. Theory* 60(1):122–132.

Cai TT, Zhang A (2015) ROP: Matrix recovery via rank-one projections. *Ann. Statist.* 43(1):102–138.

Cai TT, Zhou WX (2013) A max-norm constrained minimization approach to 1-bit matrix completion. *J. Machine Learn. Res.* 14(1):3619–3647.

Candès EJ (2008) The restricted isometry property and its implications for compressed sensing. *Comptes Rendus Mathematique* 346(9–10):589–592.

Candès EJ, Plan Y (2011) Tight oracle inequalities for low-rank matrix recovery from a minimal number of noisy random measurements. *IEEE Trans. Inform. Theory* 57(4):2342–2359.

Candès EJ, Tao T (2010) The power of convex relaxation: Near-optimal matrix completion. *IEEE Trans. Inform. Theory* 56(5): 2053–2080.

Candès EJ, Li X, Soltanolkotabi M (2015) Phase retrieval via Wirtinger flow: Theory and algorithms. *IEEE Trans. Inform. Theory* 61(4):1985–2007.

Candès EJ, Strohmer T, Voroninski V (2013) Phaselift: Exact and stable signal recovery from magnitude measurements via convex programming. *Comm. Pure Appl. Math.* 66(8):1241–1274.

Charisopoulos V, Chen Y, Davis D, Díaz M, Ding L, Drusvyatskiy D (2021) Low-rank matrix recovery with composite optimization:

Good conditioning and rapid convergence. *Foundations Comput. Math.* 21(6):1505–1593.

Chen Y, Candès EJ (2017) Solving random quadratic systems of equations is nearly as easy as solving linear systems. *Comm. Pure Appl. Math.* 70(5):822–883.

Chen Y, Wainwright MJ (2015) Fast low-rank estimation by projected gradient descent: General statistical and algorithmic guarantees. Preprint, submitted September 10, https://arxiv.org/abs/1509.03025.

Chen Y, Chi Y, Goldsmith AJ (2015) Exact and stable covariance estimation from quadratic sampling via convex programming. *IEEE Trans. Inform. Theory* 61(7):4034–4059.

Chi Y, Lu YM, Chen Y (2019) Nonconvex optimization meets low-rank matrix factorization: An overview. *IEEE Trans. Signal Processing* 67(20):5239–5269.

Clarkson KL, Woodruff DP (2017) Low-rank approximation and regression in input sparsity time. *J. ACM* 63(6):1–45.

Davenport MA, Romberg J (2016) An overview of low-rank matrix recovery from incomplete observations. *IEEE J. Selected Topics Signal Processing* 10(4):608–622.

Dobriban E, Liu S (2019) Asymptotics for sketching in least squares regression. *Adv. Neural Inform. Processing Systems* 33:3675–3685.

Drineas P, Magdon-Ismail M, Mahoney MW, Woodruff DP (2012) Fast approximation of matrix coherence and statistical leverage. *J. Machine Learn. Res.* 13:3475–3506.

Duchi JC, Ruan F (2019) Solving (most) of a set of quadratic equalities: Composite optimization for robust phase retrieval. *Inform. Inference* 8(3):471–529.

Erdogdu MA, Montanari A (2015) Convergence rates of sub-sampled Newton methods. *Adv. Neural Inform. Processing Systems* 28:3052–3060.

Fienup JR (1982) Phase retrieval algorithms: A comparison. *Appl. Optics* 21(15):2758–2769.

Fornasier M, Rauhut H, Ward R (2011) Low-rank matrix recovery via iteratively reweighted least squares minimization. *SIAM J. Optim.* 21(4):1614–1640.

Gao B, Xu Z (2017) Phaseless recovery using the Gauss–Newton method. *IEEE Trans. Signal Processing* 65(22):5885–5896.

Ge R, Jin C, Zheng Y (2017) No spurious local minima in nonconvex low rank problems: A unified geometric analysis. *Proc. 34th Internat. Conf. Machine Learn.*, 1233–1242.

Goldfarb D, Ma S (2011) Convergence of fixed-point continuation algorithms for matrix rank minimization. *Foundations Comput. Math.* 11(2):183–210.

Ha W, Liu H, Barber RF (2020) An equivalence between critical points for rank constraints vs. low-rank factorizations. *SIAM J. Optim.* 30(4):2927–2955.

Hardt M (2014) Understanding alternating minimization for matrix completion. *2014 IEEE 55th Annual Sympos. Foundations Comput. Sci.* (IEEE, Piscataway, NJ), 651–660.

Huang W, Hand P (2018) Blind deconvolution by a steepest descent algorithm on a quotient manifold. *SIAM J. Imaging Sci.* 11(4): 2757–2785.

Huang W, Absil PA, Gallivan KA (2017a) Intrinsic representation of tangent vectors and vector transports on matrix manifolds. *Numerische Mathematik* 136(2):523–543.

Huang W, Gallivan KA, Zhang X (2017b) Solving phaselift by low-rank Riemannian optimization methods for complex semidefinite constraints. *SIAM J. Sci. Comput.* 39(5):B840–B859.

Jain P, Meka R, Dhillon IS (2010) Guaranteed rank minimization via singular value projection. *Adv. Neural Inform. Processing Systems* 23:937–945.

Jain P, Netrapalli P, Sanghavi S (2013) Low-rank matrix completion using alternating minimization. *Proc. 45th Annual ACM Sympos. Theory Comput.* (ACM, New York), 665–674.

Jiang K, Sun D, Toh KC (2014) A partial proximal point algorithm for nuclear norm regularized matrix least squares problems. *Math. Programming Comput.* 6(3):281–325.

Keshavan RH, Oh S, Montanari A (2009) Matrix completion from a few entries. *2009 IEEE Internat. Sympos. Inform. Theory* (IEEE, Piscataway, NJ), 324–328.

Koltchinskii V, Lounici K, Tsybakov AB (2011) Nuclear-norm penalization and optimal rates for noisy low-rank matrix completion. *Ann. Statist.* 39(5):2302–2329.

Kümmerle C, Sigl J (2018) Harmonic mean iteratively reweighted least squares for low-rank matrix recovery. *J. Machine Learn. Res.* 19(1):1815–1863.

Lange K (2010) *Numerical Analysis for Statisticians* (Springer Science & Business Media).

Lee JM (2013) *Introduction to Smooth Manifolds* (Springer).

Lee JD, Recht B, Srebro N, Tropp J, Salakhutdinov RR (2010) Practical large-scale optimization for max-norm regularization. *Adv. Neural Inform. Processing Systems* 23:1297–1305.

Lewis AS, Wright SJ (2016) A proximal method for composite minimization. *Math. Programming* 158(1–2):501–546.

Li Q, Zhu Z, Tang G (2019a) The non-convex geometry of low-rank matrix optimization. *Inform. Inference* 8(1):51–96.

Li X, Ling S, Strohmer T, Wei K (2019b) Rapid, robust, and reliable blind deconvolution via nonconvex optimization. *Appl. Comput. Harmonic Anal.* 47(3):893–934.

Li X, Zhu Z, Man-Cho So A, Vidal R (2020a) Nonconvex robust low-rank matrix recovery. *SIAM J. Optim.* 30(1):660–686.

Li Y, Chi Y, Zhang H, Liang Y (2020b) Non-convex low-rank matrix recovery with arbitrary outliers via median-truncated gradient descent. *Inform. Inference* 9(2):289–325.

Luo Y, Zhang AR (2021) Low-rank tensor estimation via Riemannian Gauss-Newton: Statistical optimality and second-order convergence. Preprint, submitted April 24, https://arxiv.org/abs/2104.12031.

Ma C, Wang K, Chi Y, Chen Y (2019) Implicit regularization in nonconvex statistical estimation: Gradient descent converges linearly for phase retrieval, matrix completion, and blind deconvolution. *Foundations Comput. Math.* 20:451–632.

Mahoney MW (2011) Randomized algorithms for matrices and data. *Foundations Trends Machine Learn.* 3(2):123–224.

Meyer G, Bonnabel S, Sepulchre R (2011) Linear regression under fixed-rank constraints: A Riemannian approach. *Proc. 28th Internat. Conf. Machine Learn.* (ACM, New York), 545–552.

Miao W, Pan S, Sun D (2016) A rank-corrected procedure for matrix completion with fixed basis coefficients. *Math. Program.* 159(1): 289–338.

Mishra B, Meyer G, Bonnabel S, Sepulchre R (2014) Fixed-rank matrix factorizations and Riemannian low-rank optimization. *Comput. Statist.* 29(3–4):591–621.

Mohan K, Fazel M (2012) Iterative reweighted algorithms for matrix rank minimization. *J. Machine. Learn. Res.* 13(1):3441–3473.

Netrapalli P, Jain P, Sanghavi S (2013) Phase retrieval using alternating minimization. *Adv. Neural Inform. Processing Systems* 26: 2796–2804.

Park D, Kyrillidis A, Caramanis C, Sanghavi S (2018) Finding low-rank solutions via nonconvex matrix factorization, efficiently and provably. *SIAM J. Imaging Sci.* 11(4):2165–2204.

Pilanci M, Wainwright MJ (2016) Iterative Hessian sketch: Fast and accurate solution approximation for constrained least-squares. *J. Machine Learn. Res.* 17(1):1842–1879.

Pilanci M, Wainwright MJ (2017) Newton sketch: A near linear-time optimization algorithm with linear-quadratic convergence. *SIAM J. Optim.* 27(1):205–245.

Raskutti G, Mahoney MW (2016) A statistical perspective on randomized sketching for ordinary least-squares. *J. Machine Learn. Res.* 17(1):7508–7538.

Recht B, Fazel M, Parrilo PA (2010) Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM Rev.* 52(3):471–501.

Sanghavi S, Ward R, White CD (2017) The local convexity of solving systems of quadratic equations. *Results Math.* 71(3–4):569–608.

Shechtman Y, Eldar YC, Cohen O, Chapman HN, Miao J, Segev M (2015) Phase retrieval with application to optical imaging: A contemporary overview. *IEEE Signal Processing Magazine* 32(3):87–109.

Song Z, Woodruff DP, Zhong P (2017) Low rank approximation with entrywise l_1-norm error. *Proc. 49th Annual ACM Sympos. Theory Comput.* (ACM, New York), 688–701.

Sun R, Luo ZQ (2016) Guaranteed matrix completion via non-convex factorization. *IEEE Trans. Inform. Theory* 62(11):6535–6579.

Tanner J, Wei K (2013) Normalized iterative hard thresholding for matrix completion. *SIAM J. Sci. Comput.* 35(5):S104–S125.

Toh KC, Yun S (2010) An accelerated proximal gradient algorithm for nuclear norm regularized linear least squares problems. *Pacific J. Optim.* 6(3):615–640.

Tong T, Ma C, Chi Y (2021a) Accelerating ill-conditioned low-rank matrix estimation via scaled gradient descent. *J. Machine Learn. Res.* 22(150):1–63.

Tong T, Ma C, Chi Y (2021b) Low-rank matrix recovery with scaled subgradient methods: Fast and robust convergence without the condition number. *IEEE Trans. Signal Processing* 69:2396–2409.

Tran-Dinh Q (2021) Extended Gauss-Newton and ADMM-Gauss-Newton algorithms for low-rank matrix optimization. *J. Appl. Numerical Optim.* 3(1):115–150.

Tu S, Boczar R, Simchowitz M, Soltanolkotabi M, Recht B (2016) Low-rank solutions of linear matrix equations via Procrustes flow. *Internat. Conf. Machine Learn.*, 964–973.

Uschmajew A, Vandereycken B (2020) On critical points of quadratic low-rank matrix optimization problems. *IMA J. Numerical Anal.* 40(4):2626–2651.

Vandereycken B (2013) Low-rank matrix completion by Riemannian optimization. *SIAM J. Optim.* 23(2):1214–1236.

Vandereycken B, Vandewalle S (2010) A Riemannian optimization approach for computing low-rank solutions of Lyapunov equations. *SIAM J. Matrix Anal. Appl.* 31(5):2553–2579.

Waldspurger I, d'Aspremont A, Mallat S (2015) Phase recovery, maxcut and complex semidefinite programming. *Math. Programming* 149(1–2):47–81.

Wang G, Giannakis GB, Eldar YC (2017a) Solving systems of random quadratic equations via truncated amplitude flow. *IEEE Trans. Inform. Theory* 64(2):773–794.

Wang L, Zhang X, Gu Q (2017c) A unified computational and statistical framework for nonconvex low-rank matrix estimation. *Proc. 20th Internat. Conf. Artificial Intelligence Statist.* (PMLR, New York), 981–990.

Wang J, Lee JD, Mahdavi M, Kolar M, Srebro N (2017b) Sketching meets random projection in the dual: A provable recovery algorithm for big and high-dimensional data. *Electronic J. Statist.* 11(2):4896–4944.

Wei K, Cai JF, Chan TF, Leung S (2016) Guarantees of Riemannian optimization for low rank matrix recovery. *SIAM J. Matrix Anal. Appl.* 37(3):1198–1222.

Wen Z, Yin W, Zhang Y (2012) Solving a low-rank factorization model for matrix completion by a nonlinear successive over-relaxation algorithm. *Math. Programming Comput.* 4(4):333–361.

Woodruff DP (2014) Sketching as a tool for numerical linear algebra. *Foundations Trends Theoretical Comput. Sci.* 10(1–2):1–157.

Zhang RY, Sojoudi S, Lavaei J (2019) Sharp restricted isometry bounds for the inexistence of spurious local minima in nonconvex matrix recovery. *J. Machine Learn. Res.* 20(114):1–34.

Zhang AR, Luo Y, Raskutti G, Yuan M (2020) ISLET: Fast and optimal low-rank tensor regression via importance sketching. *SIAM J. Math. Data Sci.* 2(2):444–479.

Zhao T, Wang Z, Liu H (2015) A nonconvex optimization framework for low rank matrix estimation. *Adv. Neural Inform. Processing Systems* 28:559–567.

Zheng Q, Lafferty J (2015) A convergent gradient descent algorithm for rank minimization and semidefinite programming from random linear measurements. *Adv. Neural Inform. Processing Systems* 28:109–117.

Zheng Y, Liu G, Sugimoto S, Yan S, Okutomi M (2012) Practical low-rank matrix approximation under robust l_1-norm. *2012 IEEE Conf. Comput. Vision Pattern Recognition* (IEEE, Piscataway, NJ), 1410–1417.

Zhou G, Huang W, Gallivan KA, Van Dooren P, Absil PA (2016) A Riemannian rank-adaptive method for low-rank optimization. *Neurocomputing* 192:72–80.

Zhu Z, Li Q, Tang G, Wakin MB (2018) Global optimality in low-rank matrix optimization. *IEEE Trans. Signal Processing* 66(13): 3614–3628.

Yuetian Luo is a postdoctoral scholar at the Data Science Institute in University of Chicago. His research interests lie in high-dimensional statistics, nonconvex optimization, and tensor data analysis. He received the Institute of Mathematical Statistics Lawrence D. Brown PhD Student Award in 2023.

Wen Huang is a professor at School of Mathematical Sciences in Xiamen University. His research interest includes optimization on Riemannian manifolds and its applications. He received and International Conference on Machine Learning Outstanding Paper Award (2022).

Xudong Li is an associate professor at the School of Data Science, Fudan University. He focuses on matrix optimization, efficient algorithms for large-scale optimization problems and decision making under uncertainty. He received the Young Researcher Prize in Continuous Optimization of the Mathematical Optimization Society (2019), an International Conference on Machine Learning Outstanding Paper Award (2022), and the Young Researcher Prize of the Operations Research Society of China (2022).

Anru Zhang is the Eugene Anson Stead, Jr. M.D. Associate Professor at the Department of biostatistics & bioinformatics at Duke University. His research interest includes tensor learning, high-dimensional statistical inference, nonconvex optimization, and applications in electronic health records and microbiome studies. He won the NSF CAREER Award (2020), ASA Gottfried E. Noether Junior Award (2021), Bernoulli Society New Researcher Award (2021), and the IMS Tweedie Award (2022).