A Scalable Second Order Method for Ill-Conditioned Matrix Completion from Few Samples

Christian Kümmerle [1] **Claudio Mayrink Verdun** [2]

Abstract

We propose an iterative algorithm for low-rank matrix completion that can be interpreted as an iteratively reweighted least squares (IRLS) algorithm, a saddle-escaping smoothing Newton method or a variable metric proximal gradient method applied to a non-convex rank surrogate. It combines the favorable data-efficiency of previous IRLS approaches with an improved scalability by several orders of magnitude. We establish the first local convergence guarantee from a minimal number of samples for that class of algorithms, showing that the method attains a local quadratic convergence rate. Furthermore, we show that the linear systems to be solved are well-conditioned even for very ill-conditioned ground truth matrices. We provide extensive experiments, indicating that unlike many state-of-the-art approaches, our method is able to complete very ill-conditioned matrices with a condition number of up to 10^{10} from few samples, while being competitive in its scalability.

1. Introduction

In different areas of machine learning and signal processing, low-rank models have turned out to be a powerful tool for the acquisition, storage and computation of information. In many of these applications, an important sub-problem is to infer the low-rank model from partial or incomplete data (Davenport & Romberg, 2016; Chi et al., 2019).

This problem is called *low-rank matrix completion*: Given a matrix $\mathbf{X}^0 \in \mathbb{R}^{d_1 \times d_2}$ of rank-r and an index set $\Omega \subset [d_1] \times [d_2]$, the task is to reconstruct $\mathbf{X}^0$ just from the knowledge of Ω and $P_\Omega(\mathbf{X}^0)$, where $P_\Omega : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}^m$ is the subsampling operator that maps a matrix to the set of entries indexed by Ω. It is well-known that this can be reformulated (Recht et al., 2010) as the NP-hard *rank minimization* problem

$$\min_{\mathbf{X} \in \mathbb{R}^{d_1 \times d_2}} \operatorname{rank}(\mathbf{X}) \quad \text{subject to } P_\Omega(\mathbf{X}) = P_\Omega(\mathbf{X}^0). \quad (1)$$

From an optimization point of view, (1) is particularly difficult to handle due to two properties: its *non-convexity* and its *non-smoothness*. A widely studied approach in the literature replaces the $\operatorname{rank}(\mathbf{X})$ by the (convex) nuclear norm $\|\mathbf{X}\|_* = \sum_{i=1}^{d} \sigma_i(\mathbf{X})$ (Fazel et al., 2003), which is the tightest convex envelope of the rank, as an objective. For this approach, a mature theory has been developed that includes performance guarantees for a near-optimal sample complexity (Candès & Tao, 2010; Chen, 2015) and robustness to noise (Candès & Plan, 2010; Chen et al., 2020b).

However, from a practical point of view, using such a convex relaxation to find a low-rank completion is *computationally very demanding*, as even first-order solvers have an per-iteration arithmetic complexity that is at least cubic in the dimensions of $\mathbf{X}^0$ (Chi et al., 2019). Thus, convex relaxations are of little use in large-scale applications of the model such as in recommender systems (Koren et al., 2009), where even storing the dense matrix $\mathbf{X}^0 \in \mathbb{R}^{d_1 \times d_2}$ is prohibitive. Another important, but less well-known issue is that a convex relaxation is *typically not as data efficient* as certain other algorithms (Tanner & Wei, 2013; Bauch et al., 2021), i.e., nuclear norm minimization typically necessitates a larger amount of samples m than other methods, measured by the quotient $\rho := m/(d_1 + d_2 - r)$ (oversampling ratio) between m and the number of degrees of freedom of $\mathbf{X}^0$, to identify $\mathbf{X}^0$ correctly (Amelunxen et al., 2014).

To overcome these drawbacks, a variety of alternative approaches have been proposed and studied. Among the most popular ones are "non-convex" algorithms based on matrix factorization (Burer & Monteiro, 2003) with objective

$$J(\mathbf{U}, \mathbf{V}) := \left\| P_\Omega(\mathbf{U}\mathbf{V}^*) - P_\Omega(\mathbf{X}^0) \right\|_F^2 + \frac{\lambda}{2} \left(\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2 \right) \quad (2)$$

for $\lambda \geq 0$, which use (projected) gradient descent on the

[1] Department of Applied Mathematics & Statistics, Johns Hopkins University, Baltimore, USA [2] Department of Mathematics and Department of Electrical and Computer Engineering, Technical University of Munich, Munich, Germany, e-mail: claudio.verdun@tum.de. Correspondence to: Christian Kümmerle <kuemmerle@jhu.edu>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021. Copyright 2021 by the author(s).

two factor matrices (Sun & Luo, 2016; Zheng & Lafferty, 2016; Ma et al., 2020), or related methods. These methods are much more scalable than those optimizing a convex rank surrogate, while also allowing for a theoretical analysis, see (Chi et al., 2019) for a recent survey. Furthermore, among the most data-efficient methods for low-rank completion are those that minimize a smooth objective over the Riemannian manifold of fixed rank matrices (Vandereycken, 2013; Wei et al., 2020; Boumal & Absil, 2015; Bauch et al., 2021). These approaches are likewise scalable and often able to reconstruct the low-rank matrix from fewer samples m than a convex formulation, but strong performance guarantees have remained elusive so far.

In many instances of our problem, such as in the discretization of PDE-based inverse problems with Fredholm equations (Cloninger et al., 2015) or in spectral estimation problems modeled by structured low-rank matrices, it is an additional difficulty that the matrix of interest $\mathbf{X}^0$ is severely *ill-conditioned*, i.e., $\kappa = \sigma_1(\mathbf{X}^0)/\sigma_r(\mathbf{X}^0)$ might be very large (up to $\kappa = 10^{15}$ in spectral estimation (Fannjiang & Liao, 2012)).

Our contribution. In this paper, we propose and analyze the algorithm *Matrix Iteratively Reweighted Least Squares* (`MatrixIRLS`) that is designed to find low-rank completions that are potentially very ill-conditioned, allowing for a scalable implementation. It is based on the minimization of quadratic models of a sequence of continuously differentiable, *non-convex "relaxations"* of the rank function $\text{rank}(\mathbf{X})$. We note that, while being severely non-convex, our method is fundamentally different from a typical non-convex approach with an objective such as eq. (2).

Let $D = \max(d_1, d_2)$ and $d = \min(d_1, d_2)$. From a theoretical angle, we establish that if the m sampled entries are distributed uniformly at random and if $m = \Omega(\mu_0 r D \log(D))$, with high probability, `MatrixIRLS` *exhibits local convergence* to $\mathbf{X}^0$ *with a local quadratic convergence rate*, where μ_0 is an incoherence factor. This sample complexity does not depend on the condition number κ, is *optimal* under the sampling model and improves, to the best of our knowledge, on the state-of-the-art of any algorithmic sample complexity result for low-rank matrix completion—albeit, with the caveat that unlike many other results, our guarantee is inherently *local*.

Furthermore, we show that the algorithm can be implemented in a per-iteration cost that is sub-quadratic in D, without the need of storing dense $(d_1 \times d_2)$ matrices. We show that under the random sampling model, the linear systems to be solved in the main computational step of `MatrixIRLS` are well-conditioned even close to the ground truth, unlike the systems of comparable IRLS algorithms in the literature (Daubechies et al., 2010; Fornasier et al., 2011; Mohan & Fazel, 2012; Kümmerle & Sigl, 2018).

The data-efficiency and scalability of our method compared to several state-of-the-art methods is finally explored in numerical experiments involving simulated data.

2. MatrixIRLS for log-det rank surrogate

The starting point of the derivation of our method is the observation that minimizing a *non-convex surrogate* objective F with more regularity than $\text{rank}(\mathbf{X})$ can lead to effective methods for solving (1) that may combine some of the aforementioned properties, e.g., if F is chosen as a *log-determinant* (Fazel, 2002; Candès et al., 2013), Schatten-p quasi-norm (with $0 < p < 1$) (Giampouras et al., 2020) or a smoothed clipped absolute deviation (SCAD) of the singular values (Mazumder et al., 2020). In particular, it has been observed in several works (Fazel, 2002; Candès et al., 2013) that optimizing the smoothed log-det objective $\sum_{i=1}^d \log(\sigma_i(\mathbf{X} + \epsilon\mathbf{I}))$ for some $\epsilon > 0$ can lead to less biased solutions than a nuclear norm minimizer—very generally, it can be shown that a minimizer of non-convex spectral functions such as the smoothed log-det objective coincides as least as often with the rank minimizer as the convex nuclear norm minimizer (Foucart, 2018). Relevant algorithmic approaches to minimize non-convex rank surrogates include iterative thresholding methods (Mazumder et al., 2020), iteratively reweighted least squares (Fornasier et al., 2011; Mohan & Fazel, 2012; Kümmerle & Sigl, 2018) and iteratively reweighted nuclear norm (Lu et al., 2015) algorithms.

However, finding the global minimizer of a non-convex and non-smooth rank surrogate can be very challenging, as the existence of sub-optimal local minima and saddle points might deter the success of many local optimization approaches. Furthermore, applications such as in recommender systems (Koren et al., 2009) require solving very high-dimensional problem instances so that it is impossible to store full matrices, let alone to calculate many singular values of these matrices, ruling out the applicability of many of the existing methods for non-convex surrogates. A major shortcoming is, finally, also that the available convergence theory for such algorithms is still very immature—a convergence theory quantifying the sample complexity or convergence rates is, to the best of our knowledge, not available for any method of this class.

To derive our method, let now $\epsilon > 0$ and $F_\epsilon : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}$ be the *smoothed log-det objective* defined as $F_\epsilon(\mathbf{X}) := \sum_{i=1}^d f_\epsilon(\sigma_i(\mathbf{X}))$ with $d = \min(d_1, d_2)$ and

$$f_\epsilon(\sigma) = \begin{cases} \log|\sigma|, & \text{if } \sigma \geq \epsilon, \\ \log(\epsilon) + \frac{1}{2}\left(\frac{\sigma^2}{\epsilon^2} - 1\right), & \text{if } \sigma < \epsilon. \end{cases} \quad (3)$$

It can be shown that that F_ϵ is continuously differentiable with ϵ^{-2}-Lipschitz gradient

$$\nabla F_{\epsilon_k}(\mathbf{X}) = \mathbf{U}\,\mathrm{dg}\left(\frac{\sigma_i(\mathbf{X})}{\max(\sigma_i(\mathbf{X}),\epsilon_k)^2}\right)_{i=1}^{d}\mathbf{V}^*,$$

where $\mathbf{X}$ has a singular value decomposition $\mathbf{X} = \mathbf{U}\,\mathrm{dg}\big(\sigma(\mathbf{X})\big)\mathbf{V}^* = \mathbf{U}\,\mathrm{dg}\big(\sigma\big)\mathbf{V}^*$. It is clear that the optimization landscape of F_ϵ crucially depends on the smoothing parameter ϵ. Instead of minimizing F_{ϵ_k} directly, our method minimizes, for $k \in \mathbb{N}$, $\epsilon_k > 0$ and $\mathbf{X}^{(k)}$ a *quadratic model*

$$\begin{aligned} Q_{\epsilon_k}(\mathbf{X}|\mathbf{X}^{(k)}) = F_{\epsilon_k}(\mathbf{X}^{(k)}) + \langle \nabla F_{\epsilon_k}(\mathbf{X}^{(k)}), \mathbf{X} - \mathbf{X}^{(k)}\rangle \\ + \frac{1}{2}\langle \mathbf{X} - \mathbf{X}^{(k)}, W^{(k)}(\mathbf{X} - \mathbf{X}^{(k)})\rangle \end{aligned}$$

under the data constraint $P_\Omega(\mathbf{X}) = P_\Omega(\mathbf{X}^0)$, where $W^{(k)}$ is the following operator, describing the precise shape of the quadratic model.

Definition 2.1 (Optimal weight operator). *Let $\epsilon_k > 0$ and $\mathbf{X}^{(k)} \in \mathbb{R}^{d_1 \times d_2}$ be a matrix with singular value decomposition $\mathbf{X}^{(k)} = \mathbf{U}_k\,\mathrm{dg}(\sigma^{(k)})\mathbf{V}_k^*$, i.e., $\mathbf{U}_k \in \mathbb{R}^{d_1 \times d_1}$ and $\mathbf{V}_k \in \mathbb{R}^{d_2 \times d_2}$ are orthonormal matrices. Then we call the linear operator $W^{(k)} : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}^{d_1 \times d_2}$ the* optimal weight operator *of the ϵ_k-smoothed log-det objective F_{ϵ_k} of* (3) *at $\mathbf{X}^{(k)}$ if for $\mathbf{Z} \in \mathbb{R}^{d_1 \times d_2}$,*

$$W^{(k)}(\mathbf{Z}) = \mathbf{U}_k\left[\mathbf{H}_k \circ (\mathbf{U}_k^*\mathbf{Z}\mathbf{V}_k)\right]\mathbf{V}_k^*, \tag{4}$$

where $\mathbf{H}_k \in \mathbb{R}^{d_1 \times d_2}$ is a matrix with positive entries such that $(\mathbf{H}_k)_{ij} := \Big(\max(\sigma_i^{(k)}, \epsilon_k)\max(\sigma_j^{(k)}, \epsilon_k)\Big)^{-1}$ and $\mathbf{H}_k \circ (\mathbf{U}_k^*\mathbf{Z}\mathbf{V}_k)$ denotes the entrywise product of $\mathbf{H}_k$ and $\mathbf{U}_k^*\mathbf{Z}\mathbf{V}_k$.

The weight operator $W^{(k)}$ is a positive, self-adjoint operator with strictly positive eigenvalues that coincide with the entries of the matrix $\mathbf{H}_k \in \mathbb{R}^{d_1 \times d_2}$, and it is easy to verify that $W^{(k)}(\mathbf{X}^{(k)}) = \nabla F_{\epsilon_k}(\mathbf{X}^{(k)})$. Based on this, it follows that the minimization of the quadratic model $Q_{\epsilon_k}(\mathbf{X}|\mathbf{X}^{(k)})$ boils down to a minimization of a quadratic form weighted by $W^{(k)}$. This enables us to design the iterative method *Matrix Iteratively Reweighted Least Squares (MatrixIRLS)*, which we describe in Algorithm 1.

Apart from the weighted least squares step (5), which minimizes the quadratic model $Q_{\epsilon_{k-1}}(\cdot|\mathbf{X}^{(k-1)})$ of $F_{\epsilon_{k-1}}$ for fixed ϵ_{k-1}, an indispensable ingredient of our scheme is the *update of the smoothing parameter* ϵ_k, which is performed in the spirit of smoothing methods for non-smooth objectives (Chen, 2012). In particular, the update rule eq. (6), which is similar to the update rule of (Kümmerle & Sigl, 2018), makes sure that if the rank estimate $\widetilde{r}$ is chosen such that $\widetilde{r} \geq r$, the smoothing parameter ϵ_k converges to 0 as the iterates approach a rank-r solution.

Algorithm 1 `MatrixIRLS` for low-rank matrix completion

```
Input: Set $\Omega$, observations $\mathbf{y} \in \mathbb{R}^m$, rank estimate $\widetilde{r}$.
Initialize $k = 0$, $\epsilon^{(0)} = \infty$ and $W^{(0)} = \mathrm{Id}$.
for $k = 1$ to $K$ do
  Solve weighted least squares: Use a conjugate gradient method to solve
      $\mathbf{X}^{(k)} = \arg\min_{\mathbf{X}:P_\Omega(\mathbf{X})=\mathbf{y}} \langle \mathbf{X}, W^{(k-1)}(\mathbf{X})\rangle.$   (5)
  Update smoothing: Compute $\widetilde{r}+1$-th singular value of $\mathbf{X}^{(k)}$ to update
      $\epsilon_k = \min\Big(\epsilon_{k-1}, \sigma_{\widetilde{r}+1}(\mathbf{X}^{(k)})\Big).$   (6)
  Update weight operator: For $r_k := |\{i \in [d] : \sigma_i(\mathbf{X}^{(k)}) > \epsilon_k\}|$, compute the first $r_k$ singular values $\sigma_i^{(k)} := \sigma_i(\mathbf{X}^{(k)})$ and matrices $\mathbf{U}^{(k)} \in \mathbb{R}^{d_1 \times r_k}$ and $\mathbf{V}^{(k)} \in \mathbb{R}^{d_2 \times r_k}$ with leading $r_k$ left/ right singular vectors of $\mathbf{X}^{(k)}$ to update $W^{(k)}$ defined in Equation (4).
end for
Output: $\mathbf{X}^{(K)}$.
```

We note that *Iteratively Reweighted Least Squares (IRLS)* methods with certain similarities to Algorithm 1 had been proposed (Fornasier et al., 2011; Mohan & Fazel, 2012; Kümmerle & Sigl, 2018) for the minimization of Schatten-p quasi-norms for $0 < p \leq 1$. Comparing the gradients of smoothed Schatten-p quasi-norms and of eq. (3), minimizing a smoothed log-det objective can be considered as a limit case for $p \to 0$. Most importantly, however, our algorithm has two distinct, conceptual differences compared to these methods: Firstly, the weight operator of Definition 2.1 is able capture the *second-order information* of F_{ϵ_k}, allowing for an interpretation of `MatrixIRLS` as a saddle-escaping smoothing Newton method, cf. Section 4.2, unlike the methods of (Fornasier et al., 2011; Mohan & Fazel, 2012; Kümmerle & Sigl, 2018) due to the different structure of their weight operators. Secondly, the interplay of F_{ϵ_k} and the weight operator $W^{(k)}$ in Algorithm 1 is designed to allow for efficient numerical implementations, cf. Section 3.

Finally, we note that it is non-trivial to show that the quadratic model $Q_{\epsilon_k}(\cdot|\mathbf{X}^{(k)})$ induced by $W^{(k)}$ from Definition 2.1 is actually a *majorant* of $F_{\epsilon_k}(\cdot)$ such that $F_{\epsilon_k}(\mathbf{X}) \leq Q_{\epsilon_k}(\mathbf{X}|\mathbf{X}^{(k)})$ for all $\mathbf{X} \in \mathbb{R}^{d_1 \times d_2}$. We defer a proof of this and a proof of the "optimality" of the majorant to an upcoming paper.

3. Computational Complexity

A crucial property of Algorithm 1 is that due to the structure of the weight operator (4) and the smoothing update rule (6), in fact, the weighted least squares step eq. (5) can be computed by solving a positive definite linear system of size $(r_k(d_1 + d_2 - r_k)) \times (r_k(d_1 + d_2 - r_k))$, where r_k is the number of singular values of $\mathbf{X}^{(k)}$ that are larger than ϵ_k, which is typically equal or very close to $\widetilde{r}$ (cf. Appendix A). Conceptually, this corresponds to a linear system in the tangent space T_k of the rank-r_k matrix manifold at the best rank-r_k approximation of $\mathbf{X}^{(k)}$, $T_k = \left\{ \left[\mathbf{U}^{(k)} \mathbf{U}_\perp^{(k)}\right] \begin{bmatrix} \mathbb{R}^{r_k \times r_k} & \mathbb{R}^{r_k(d_2 - r_k)} \\ \mathbb{R}^{(d_1 - r_k) r_k} & \mathbf{0} \end{bmatrix} \left[\mathbf{V}^{(k)} \mathbf{V}_\perp^{(k)}\right]^* \right\}$.

We note that in our implementation, it is never necessary to compute more than r_k singular vector pairs and singular values of $\mathbf{X}^{(k)}$, and $\mathbf{X}^{(k)}$ can be represented as a sum of a sparse matrix and a matrix in T_k, cf. Theorem 3.1. Thus, when using an iterative solver such as *conjugate gradients* to solve the linear system, we obtain an implementation of `MatrixIRLS` with a time and space complexity of the same order as for state-of-the-art first-order algorithms based on matrix factorization (i.e., of Burer-Monteiro type) (Chen & Chi, 2018). We refer to the supplementary materials (Appendix A) for details and a proof.

Theorem 3.1. *Let $\mathbf{X}^{(k)} \in \mathbb{R}^{d_1 \times d_2}$ be the k-th iterate of* `MatrixIRLS` *for an observation vector $\mathbf{y} \in \mathbb{R}^m$ and $\widetilde{r} = r$. Assume that $\sigma_i^{(k)} \le \epsilon_k$ for all $i > r$ and $\sigma_r^{(k)} > \epsilon_k$. Then an implicit representation of the new iterate $\mathbf{X}^{(k+1)} \in \mathbb{R}^{d_1 \times d_2}$ can be calculated in a* time complexity *of*

$$O\left((mr + r^2 D) \cdot N_{CG\_inner}\right),$$

where N_{CG_inner} is the number of inner iterations used in the conjugate gradient method and $D = \max(d_1, d_2)$. More precisely, $\mathbf{X}^{(k+1)}$ can be represented as

$$\mathbf{X}^{(k+1)} = P_\Omega^*(\mathbf{r}_{k+1}) + \mathbf{U}^{(k)} \mathbf{M}_1^{(k+1)*} + \mathbf{M}_2^{(k+1)} \mathbf{V}^{(k)*},$$

where $\mathbf{r}_{k+1} \in \mathbb{R}^m$, $\mathbf{M}_1^{(k+1)} \in \mathbb{R}^{d_2 \times r}$ and $\mathbf{M}_2^{(k+1)} \in \mathbb{R}^{d_1 \times r}$, i.e., with a space complexity *of $O(m + rD)$.*

Theorem 3.1 illustrates the computational advantage of `MatrixIRLS` compared to previous iteratively reweighted least squares algorithms for low-rank matrix recovery problems (Fornasier et al., 2011; Mohan & Fazel, 2012; Kümmerle & Sigl, 2018), which all require the storage and updates of full $(d_1 \times d_2)$-matrices and the calculation of singular value decompositions of these.

According to Theorem 3.1, since $P_\Omega^*(\mathbf{r}_{k+1}) \in \mathbb{R}^{d_1 \times d_2}$ is m-sparse, $\mathbf{X}^{(k+1)}$ can be seen a sum of a sparse and two rank-r matrices. Intuitively, this representation is possible as the weight operator $W^{(k)}$ of Definition 2.1 can be written as "identity + diagonal on T_k", and due to the Sherman-Morrison-Woodbury formula applied to the inverse in $\mathbf{X}^{(k+1)} = (W^{(k)})^{-1} P_\Omega^* \left(P_\Omega (W^{(k)})^{-1} P_\Omega^*\right)^{-1} (\mathbf{y})$, which is an explicit representation of the solution of eq. (5).

As a result, fast matrix-vector multiplications can be used in methods such as Lanczos bidiagonalization or randomized Block Krylov (Musco & Musco, 2015) to compute r_{k+1} singular values and vectors of $\mathbf{X}^{(k+1)}$ in step 3 of Algorithm 1.

4. Theoretical Analysis

This section sheds light on several theoretical aspects of Algorithm 1.

4.1. Local Convergence with Superlinear Rate & Conditioning of System Matrix

In order to obtain a theoretical understanding of the generic behavior of `MatrixIRLS`, we consider the canonical uniform random sampling model (Candès & Recht, 2009; Recht, 2011; Chen, 2015) where the sampling set $\Omega = (i_\ell, j_\ell)_{\ell=1}^m \subset [d_1] \times [d_2]$ consists of m double indices that are drawn uniformly at random without replacement. Not *each* rank-r matrix $\mathbf{X}^0 \in \mathbb{R}^{d_1 \times d_2}$ is expected to be identifiable from a small number of samples m under this sampling model. We quantify the alignment of a matrix with the standard basis of $\mathbb{R}^{d_1 \times d_2}$ by the following notion of incoherence, which is slightly weaker than related conditions of (Recht, 2011; Chen, 2015).

Definition 4.1. *We say that a rank-r matrix $\mathbf{X} \in \mathbb{R}^{d_1 \times d_2}$ with singular value decomposition $\mathbf{X} = \mathbf{U}\,\mathrm{dg}(\sigma)\mathbf{V}^*$, $\mathbf{U} \in \mathbb{R}^{d_1 \times r}$, $\mathbf{V} \in \mathbb{R}^{d_2 \times r}$, is* μ_0-incoherent *if there exists a constant $\mu_0 \ge 1$ such that*

$$\max_{1 \le i \le d_1, 1 \le j \le d_2} \|\mathcal{P}_T(e_i e_j^*)\|_F \le \sqrt{\mu_0 r \frac{d_1 + d_2}{d_1 d_2}}, \qquad (7)$$

where $T = T_{\mathbf{X}} = \{\mathbf{U}\mathbf{M}^* + \widetilde{\mathbf{M}}\mathbf{V}^* : \mathbf{M} \in \mathbb{R}^{d_2 \times r}, \widetilde{\mathbf{M}} \in \mathbb{R}^{d_1 \times r}\}$ is the tangent space onto the rank-r matrix manifold at $\mathbf{X}$ and $\mathcal{P}_T$ is the projection operator onto T.

With the notation that $\|\mathbf{X}\|_{S_\infty} = \sigma_1(\mathbf{X})$ denotes the *spectral norm* or *Schatten-∞ norm* of a matrix $\mathbf{X}$, we obtain the following local convergence result.

Theorem 4.1 (Local convergence of `MatrixIRLS` with Quadratic Rate)**.** *Let $\mathbf{X}^0 \in \mathbb{R}^{d_1 \times d_2}$ be a matrix of rank r that is μ_0-incoherent, and let $P_\Omega : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}^m$ be the subsampling operator corresponding to an index set $\Omega = (i_\ell, j_\ell)_{\ell=1}^m \subset [d_1] \times [d_2]$ that is drawn uniformly without replacement. If the sample complexity fulfills $m \gtrsim \mu_0 r(d_1 + d_2)\log(d_1 + d_2)$, then with high probability, the following holds: If the output matrix $\mathbf{X}^{(k)} \in \mathbb{R}^{d_1 \times d_2}$ of the k-th iteration of* `MatrixIRLS` *with inputs P_Ω, $\mathbf{y} = P_\Omega(\mathbf{X}^0)$ and $\widetilde{r} = r$ updates the smoothing parameter in (6) such that $\epsilon_k = \sigma_{r+1}(\mathbf{X}^{(k)})$ and fulfills*

$$\|\mathbf{X}^{(k)} - \mathbf{X}^0\|_{S_\infty} \lesssim \min\left(\sqrt{\frac{\mu_0 r}{d}}, \frac{\mu_0}{d \log(D) \kappa}\right) \sigma_r(\mathbf{X}^0), \qquad (8)$$

where $\kappa = \sigma_1(\mathbf{X}^0)/\sigma_r(\mathbf{X}^0)$, then the local convergence rate is quadratic *in the sense that $\|\mathbf{X}^{(k+1)} - \mathbf{X}^0\|_{S_\infty} \le \min(\mu \|\mathbf{X}^{(k)} - \mathbf{X}^0\|_{S_\infty}^2, \|\mathbf{X}^{(k)} - \mathbf{X}^0\|_{S_\infty})$ with $\mu \le \frac{d \log(D)}{\mu_0 \sigma_r(\mathbf{X}^0)} \kappa$, and furthermore*

$\mathbf{X}^{(k+\ell)} \xrightarrow{\ell\to\infty} \mathbf{X}^0$ *if additionally* $\|\mathbf{X}^{(k)} - \mathbf{X}^0\|_{S_\infty} \lesssim \min\left(\sqrt{\frac{\mu_0 r}{d}}, \frac{\mu_0^{3/2} r^{1/2}}{d^2 \log(D)^{3/2}\kappa}\right) \sigma_r(\mathbf{X}^0)$.

While a comparable local convergence result had been obtained for an IRLS algorithm for (non-convex) Schatten-p minimization (Kümmerle & Sigl, 2018), that result is *not* applicable for matrix completion, as the proof relied on a *null space property* (Recht et al., 2011) of the measurement operator, which is not fulfilled by P_Ω since there are always rank-ones matrices in the null space of the entry-wise operator P_Ω.

Unlike the theory of other algorithms, the sample complexity assumption of Theorem 4.1 is *optimal* as it matches a well-known lower bound for this sampling model (Candès & Tao, 2010) that is necessary for unique identifiability. Among the weakest sufficient conditions for existing algorithms are $m \gtrsim \mu_0 r(d_1+d_2)\log^2(d_1+d_2)$ for nuclear norm minimization (Chen, 2015), $m \gtrsim \mu_0\kappa^{14} r^2(d_1+d_2)\log^2(d_1+d_2)$ for gradient descent (Chen et al., 2020a) on a variant of (2) and $m \gtrsim \kappa^6(d_1+d_2)r^2\log(d_1+d_2)$ required random samples for the Riemannian gradient descent algorithm of (Wei et al., 2020). On the other hand, in contrast to other results, Theorem 4.1 only quantifies *local* convergence.

The following theorem implies that iterative solvers are indeed able to efficiently solve the linear system underlying (5) up to high accuracy in few iterations. It suggests that $N_{\text{CG\_inner}}$ of Theorem 3.1 can be chosen as an absolute constant.

Theorem 4.2 (Well-conditioning of system matrices of `MatrixIRLS`)**.** *In the setup and sampling model of Theorem 3.1, if* $m \gtrsim \mu_0 r(d_1+d_2)\log(d_1+d_2)$, *the following holds with high probability: If* $\epsilon_k = \sigma_{r+1}(\mathbf{X}^{(k)}) < \sigma_r(\mathbf{X}^{(k)})$ *and if* $\|\mathbf{X}^{(k)} - \mathbf{X}^0\|_{S_\infty} \lesssim \min\left(\sqrt{\frac{\mu_0 r}{d}}, \frac{1}{4}\right)\sigma_r(\mathbf{X}^0)$, *the spectrum* $\lambda(\mathbf{A}_k)$ *of the linear system matrix* $\mathbf{A}_k \in \mathbb{R}^{r(d_1+d_2-r)\times r(d_1+d_2-r)}$ *of the weighted least squares step* (5) *of* `MatrixIRLS` *satisfies* $\lambda(\mathbf{A}_k) \subset \frac{m}{d_1 d_2}\left[\frac{6}{10};\frac{24}{10}\right]$, *and thus, the condition number of* $\mathbf{A}_k$ *fulfills* $\kappa(\mathbf{A}_k) \le 4$.

Theorem 4.2 shows that `MatrixIRLS` is able to overcome a common problem of many IRLS algorithms for related problems: Unlike the methods of (Daubechies et al., 2010; Fornasier et al., 2016; Mohan & Fazel, 2012; Fornasier et al., 2011; Kümmerle & Sigl, 2018), does not suffer from ill-conditioned linear systems close to a low-rank (or sparse) solution.

4.2. MatrixIRLS as saddle-escaping smoothing Newton method

From a theoretical point of view, the local quadratic convergence rate is an inherently local property that does not explain the numerically observed global convergence behavior (see Section 5), which is remarkable due to the non-convexity of the objective function.

A possible avenue to explain this is to interpret `MatrixIRLS` as a *saddle-escaping smoothing Newton method*. Smoothing Newton methods minimize a non-smooth and possibly non-convex function F by using derivatives of certain smoothings of F (Chen et al., 1998; Chen, 2012). Interpreting the optimization problem $\min_{\mathbf{X}:P_\Omega(X)=\mathbf{y}} F_{\epsilon_k}(\mathbf{X})$ as an unconstrained optimization problem over the null space of P_Ω, we can write

$$\mathbf{X}^{(k+1)} = \mathbf{X}^{(k)} - P_{\Omega^c}^*\left(P_{\Omega^c} W^{(k)} P_{\Omega^c}^*\right)^{-1} P_{\Omega^c} W^{(k)}(\mathbf{X}^{(k)})$$
$$= \mathbf{X}^{(k)} - P_{\Omega^c}^*\left(P_{\Omega^c}\overline{\nabla^2 F_{\epsilon_k}(\mathbf{X}^{(k)})} P_{\Omega^c}^*\right)^{-1} P_{\Omega^c}\nabla F_{\epsilon}(\mathbf{X}^{(k)}),$$

if $\Omega^c = [d_1]\times[d_2]\backslash\Omega$ corresponds to the unobserved indices, where $\overline{\nabla^2 F_{\epsilon_k}(\mathbf{X}^{(k)})} : \mathbb{R}^{d_1\times d_2} \to \mathbb{R}^{d_1\times d_2}$ is a *modified* Hessian of F_{ϵ_k} at $\mathbf{X}^{(k)}$ that replaces negative eigenvalues of the Hessian $\nabla^2 F_{\epsilon_k}(\mathbf{X}^{(k)})$ by positive ones and slightly increases small eigenvalues. We refer to the supplementary material for more details. In (Paternain et al., 2019), it has been proved that for a fixed smooth function F_{ϵ_k}, similar modified Newton-type steps are able to escape the first-order saddle points at a rate that is independent of the problem's condition number.

4.3. MatrixIRLS as variable metric forward-backward method

Another instructive angle to understand our method comes from the framework of *variable metric forward-backward methods* (Bonnans et al., 1995; Chouzenoux et al., 2014; Frankel et al., 2015).

A forward-backward method can be seen as a combination of a gradient descent method and a proximal point algorithm (Combettes & Pesquet, 2011) that can be used to minimize the sum of a non-smooth function and a function with Lipschitz continuous gradients. In particular, if F is a proper, lower semi-continuous function, G is differentiable with Lipschitz gradient ∇G and $(\alpha_k)_k$ a sequence of step sizes, the iterations of the forward-backward algorithm (Attouch et al., 2013) are such that $\mathbf{X}^{(k+1)} \in \operatorname{prox}_{\alpha_k F}\left(\mathbf{X}^{(k)} - \alpha_k \nabla G(\mathbf{X}^{(k)})\right)$, where $\operatorname{prox}_{\alpha_k F}(\cdot)$ is the proximity operator of $\alpha_k F$. Typically, in such an algorithm, F would be chosen as the structure-promoting objective (such as the smoothed log-det objective F_ϵ above) and G as a data-fit term such as $G(\mathbf{X}) = \|P_\Omega(\mathbf{X}) - \mathbf{y}\|_2^2/\lambda$, leading to thresholding-type algorithms. Algorithm 1, however, fits into this framework if we choose, for $\epsilon_k > 0$, the non-smooth part F as the indicator function $F := \chi_{P_\Omega^{-1}(\mathbf{y})} : \mathbb{R}^{d_1\times d_2} \to \mathbb{R}$ of the constraint set $P_\Omega^{-1}(\mathbf{y}) := \{\mathbf{X} \in \mathbb{R}^{d_1\times d_2} : P_\Omega(\mathbf{X}) = \mathbf{y}\}$ and the smooth part G such that

$G := F_{\epsilon_k} : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}$ as in (3), while offsetting the distortion induced by the non-Euclidean nature of the level sets of F_{ϵ_k} via an appropriate choice of a *variable metric* $d_{A_k}(\mathbf{X}, \mathbf{Z}) = \sqrt{\langle \mathbf{X} - \mathbf{Z}, A_k(\mathbf{X} - \mathbf{Z}) \rangle_F}$ for a positive definite linear operator $A_k : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}^{d_1 \times d_2}$, such that

$$\mathbf{X}^{(k+1)} \in \mathrm{prox}_{\alpha_k F}^{A_k} \left(\mathbf{X}^{(k)} - \alpha_k A_k^{-1} (\nabla G(\mathbf{X}^{(k)})) \right),$$

where $\mathrm{prox}_F^{A_k}(\mathbf{X}) := \arg\min_{\mathbf{Z} \in \mathbb{R}^{d_1 \times d_2}} F(\mathbf{Z}) + \frac{1}{2} d_{A_k}(\mathbf{X}, \mathbf{Z})^2$ is the proximity operator of F *scaled in the metric* d_{A_k} at $\mathbf{X}$ (Chouzenoux et al., 2014). Specifically, if we choose the metric induced by the weight operator of (4) such that $A_k := W^{(k)}$ and unit step sizes $\alpha_k = 1$, we obtain

$$\begin{aligned}
&\mathrm{prox}_{\alpha_k F}^{A_k} \left(\mathbf{X}^{(k)} - \alpha_k A_k^{-1} (\nabla G(\mathbf{X}^{(k)})) \right) \\
&= \mathrm{prox}_{\chi_{P_\Omega^{-1}}}^{W^{(k)}} \left(\mathbf{X}^{(k)} - W_k^{-1} (\nabla F_{\epsilon_k}(\mathbf{X}^{(k)})) \right) \\
&= \mathrm{prox}_{\chi_{P_\Omega^{-1}}}^{W^{(k)}} \left(\mathbf{X}^{(k)} - W_k^{-1} W_k(\mathbf{X}^{(k)})) \right) = \mathrm{prox}_{\chi_{P_\Omega^{-1}}}^{W^{(k)}} (\mathbf{0}) \\
&= \operatorname*{arg\,min}_{\mathbf{X} : P_\Omega(\mathbf{X}) = \mathbf{y}} \frac{1}{2} d_{A_k}(\mathbf{X}, \mathbf{0})^2 = \operatorname*{arg\,min}_{\mathbf{X} : P_\Omega(\mathbf{X}) = \mathbf{y}} \langle \mathbf{X}, W^{(k)}(\mathbf{X}) \rangle,
\end{aligned}$$

where we used that $W_k(\mathbf{X}^{(k)}) = \nabla F_{\epsilon_k}(\mathbf{X}^{(k)})$ in the third line. This shows that this update rule for $\mathbf{X}^{(k+1)}$ coincides with (5).

Thus, `MatrixIRLS` can be considered as a forward-backward method with a variable metric induced by the weight operator $W^{(k)}$, using a unit step size $\alpha_k = 1$ for each k. One advantage of our method is therefore also that unlike many methods in this family, there is no step size to be tuned. A crucial difference, which makes existing theory (as, e.g., (Frankel et al., 2015)) for splitting methods not directly applicable for a convergence analysis of `MatrixIRLS`, is that the smooth function $G = F_{\epsilon_k}$ *is changing* at each iteration due to the smoothing parameter update (6). On the other hand, the results of (Frankel et al., 2015) already imply the finite sequence length of $(\mathbf{X}^{(k)})_k$ in the case that the smoothing parameter ϵ_k stagnates for $k \geq k_0$, using a Kurdyka-Łojasiewicz property (Bolte et al., 2007) of $F_{\epsilon_k} + \chi_{P_\Omega^{-1}(\mathbf{y})}$. We leave a detailed discussion of this for future work.

Finally, we note that previous IRLS methods (Fornasier et al., 2011; Mohan & Fazel, 2012; Kümmerle & Sigl, 2018) would also fit in the presented splitting framework, however, without fully capturing the underlying geometry as their weight operator has no strong connection to the Hessian $\nabla^2 F_{\epsilon_k}(\mathbf{X}^{(k)})$ of F_{ϵ_k}, as explained in the supplementary material.

5. Numerical Experiments

We explore the performance of `MatrixIRLS` for the completion of synthetic low-rank matrices in terms of statistical and computational efficiency in comparison to state-of-the-art algorithms in the literature. We base our choice on the desire to obtain a representative picture of state-of-the-art algorithms for matrix completion, including in particular those that are scalable to problems with dimensionality in the thousands or more, those that come with the best theoretical guarantees, and those that claim to perform particularly well to complete *ill-conditioned* matrices. All the methods are provided with the true rank r of $\mathbf{X}^0$ as an input parameter. If possible, we use the MATLAB implementation provided by the authors of the respective papers.

The algorithms being tested against `MatrixIRLS` can be grouped into three main categories: the non-convex matrix factorization ones which includes `LMaFit` (Wen et al., 2012), `ScaledASD` (Tanner & Wei, 2016) and `ScaledGD` (Tong et al., 2020), the Riemannian optimization on the manifold of fixed rank matrices ones which includes `LRGeomCG` (Vandereycken, 2013), `RTRMC` (Boumal & Absil, 2015) and `R3MC` (Mishra & Sepulchre, 2014), one alternating projection method on the manifold of fixed rank matrices, `NIHT` (Tanner & Wei, 2013) (see (Wei et al., 2020) for a connection between NIHT and Riemannian methods), and the recent `R2RILS` (Bauch et al., 2021) which can be seen as a factorization based method but also contains ideas from the Riemannian optimization family of algorithms. In the supplementary material we provide a description of each algorithm as well as the parameters used in the numerical section.

As for the numerical experiments, it is important to note that we are interested to find low-rank completions from a sampling set Ω of sample size $|\Omega| =: m = \lfloor \rho r(d_1 + d_2 - r) \rfloor$, where ρ is an oversampling ratio since $r(d_1 + d_2 - r)$ is just the number of degrees of freedom of an $(d_1 \times d_2)$-dimensional rank-r matrix. For a given Ω, the solution of (1) might *not* coincide with $\mathbf{X}^0$, or the solution might not be unique, even if the sample set Ω is chosen uniformly at random. In particular, this will be the case if Ω is such that there is a row or a column with *fewer than* r revealed entries, which a necessary condition for uniqueness of the (1) (Pimentel-Alarcón et al., 2016). To mitigate this problem that is rather related to the structure of the sampling set than to the performance of a certain algorithm, we, in fact, adapt the sampling model of uniform sampling without replacement. For a given factor $\rho \geq 1$, we sample a set $\Omega \subset [d_1] \times [d_2]$ of size $m = \lfloor \rho r(d_1 + d_2 - r) \rfloor$ indices randomly without replacement. Then we check whether the condition such that each row and each column in Ω has at least r observed entries, and resample Ω if this condition is not fulfilled. This procedure is repeated up to a maximum of 1000 resamplings.

We consider the following setup: we sample a pair of random matrices $\mathbf{U} \in \mathbb{R}^{d_1 \times r}$ and $\mathbf{V} \in \mathbb{R}^{d_2 \times r}$ with r orthonormal columns, and define the diagonal matrix $\Sigma \in \mathbb{R}^{r\times r}$ such that $\Sigma_{ii} = \kappa \exp(-\log(\kappa)\frac{i-1}{r-1})$ for $i \in [r]$. With this definition, we define a ground truth matrix $\mathbf{X}^0 = \mathbf{U}\Sigma\mathbf{V}^*$ of rank r that has exponentially decaying singular values between κ and 1.

5.1. Data-efficient recovery of ill-conditioned matrices

First, we run MatrixIRLS and the algorithms R2RILS , RTRMC, LRGeomCG, LMaFit, ScaledASD, ScaledGD, NIHT and R3MC to complete $\mathbf{X}^0$ from $P_\Omega(\mathbf{X}^0)$ where Ω corresponds to different oversampling factors ρ between 1 and 4, and where the condition number of $\mathbf{X}^0$ is $\kappa = \sigma_1(\mathbf{X}^0)/\sigma_r(\mathbf{X}^0) = 10$. In Figure 1, we report the median Frobenius errors $\|\mathbf{X}^{(K)} - \mathbf{X}^0\|_F/\|\mathbf{X}^0\|_F$ of the respective algorithmic outputs $\mathbf{X}^{(K)}$ across 100 independent realizations.

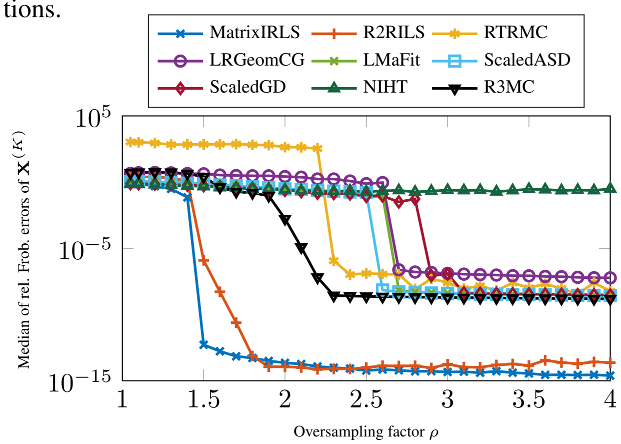


Figure 1. Performance of matrix completion algorithms for 1000 × 1000 matrices of rank $r = 5$ with condition number $\kappa = 10$, given $m = \lfloor \rho r(d_1 + d_2 - r)\rfloor$ random samples. Median of Frobenius errors $\|\mathbf{X}^{(K)} - \mathbf{X}^0\|_F/\|\mathbf{X}^0\|_F$ of 100 independent realizations.

We see that MatrixIRLS and R2RILS are the only algorithms that are able to complete $\mathbf{X}^0$ already for $\rho = 1.5$. In our experiment, R3MC completes $\mathbf{X}^0$ in a majority of instances starting from $\rho = 2.0$, whereas the other algorithms, except from NIHT, are able to reconstruct the matrix most of the times if ρ is at least between 2.4 and 3.0. This confirms the findings of (Bauch et al., 2021) which show that even for quite well-conditioned matrices, fewer samples are required if second-order methods such as R2RILS or MatrixIRLS are used.

We repeat this experiment for ill-conditioned matrices $\mathbf{X}^0$ with $\kappa = 10^5$. In Figure 2, we see that current state-of-the-art methods are *not able* to achieve exact recovery of $\mathbf{X}^0$. This is in particular true as given the exponential decay of the singular values, in order to recover the subspace corresponding to the smallest singular value of $\mathbf{X}^0$, a relative Frobenius error of 10^{-5} or even several orders of magnitude smaller needs to be achieved. We observe that MatrixIRLS is the only method that is able to complete $\mathbf{X}^0$ for any of the considered oversampling factors.

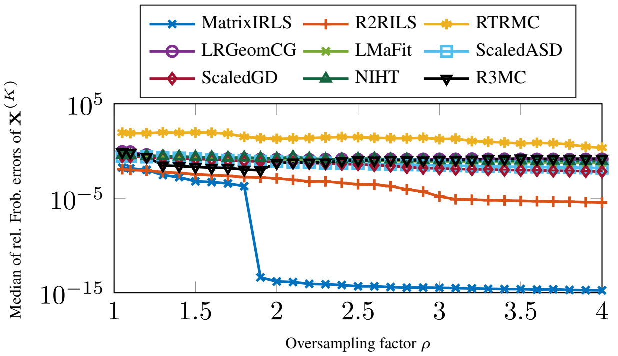


Figure 2. Performance of matrix completion algorithms as in Figure 1, but with $\kappa = 10^5$. Median of 50 realizations.

5.2. Running time for ill-conditioned problems

In Figure 3, for an oversampling ratio of $\rho = 4$, we illustrate the completion of one single extremely ill-conditioned 1000 × 1000 matrix with rank = 10 and $\kappa = 10^{10}$ and exponentially interpolated singular values as described above. We again can see that only second-order methods such as R2RILS or MatrixIRLS are able to achieve a relative Frobenius error $\approx 10^{-5}$ or smaller. MatrixIRLS goes beyond that and attains a relative Frobenius error of the order of the machine precision and, remarkably, exactly recover all the singular values up to 15 digits. This also shows that the conjugated gradient and the randomized block Krylov method used at the inner core of our implementation can be extremely precise when properly adjusted. R2RILS is also able to obtain relatively low Frobenius error but unlike our method, it is not able to retrieve all the singular values with high accuracy. Other methods were observed to lead to a meaningful error decrease for the ill-conditioned matrix of interest.

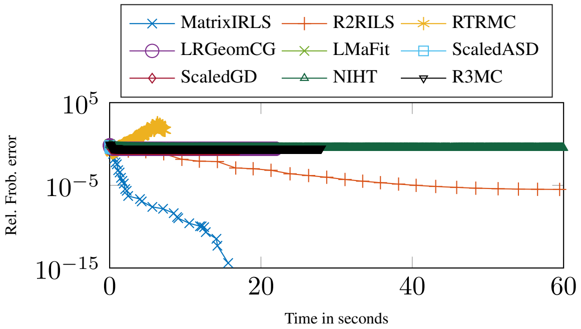


Figure 3. Completion task for a highly ill-conditioned 1000 × 1000 matrix of rank $r = 10$ with $\kappa = 10^{10}$ ($\rho = 4$).

In Figure 4, we compare the execution time of R2RILS and MatrixIRLS for a range of ground truth matrices with increasing dimension, for an oversampling ratio of $\rho = 2.5$, whose singular values are linearly interpolated between κ and 1. We observe that the larger the dimensions are, the larger is the discrepancy in the running time of the two algorithms. Other algorithms are not considered in this experiment because they typically do not reach a relative error below 10^{-4} for $\kappa \gg 10^2$.

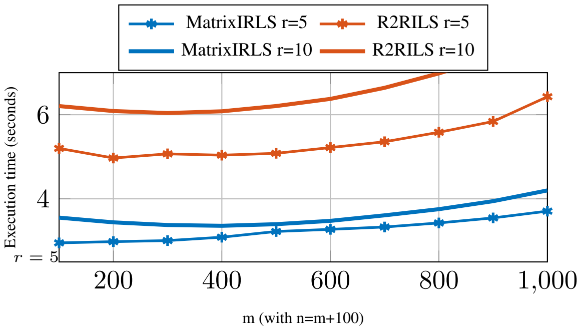


Figure 4. Execution time of `R2RILS` and `MatrixIRLS` for completion of rank $r \in \{5, 10\}$ matrices of size $m \times (m + 100)$ and condition number $\kappa = 10^2$, averaged across 50 independent realizations.

5.3. MatrixIRLS vs. rank-adaptive strategies

In Section 5.1, all methods were provided with the correct rank r of the ground truth, which was used to determine the size of the matrix factors or the rank of the fixed rank manifold. Even in this case, we illustrated numerically that most of the methods are not able to recover highly ill-conditioned matrices. To handle such ill-conditioned completion problems, (Mishra & Sepulchre, 2014; Uschmajew & Vandereycken, 2015; Tan et al., 2014) proposed rank-adaptive variants of the methods `R3MC` and `LRGeomCG`. These variants, which we call `LRGeomCG Pursuit`[1] (Uschmajew & Vandereycken, 2015; Tan et al., 2014) and `R3MC w/ Rank Update` (Mishra & Sepulchre, 2014), respectively, combine fixed-rank optimization with outer iterations that increase $\widetilde{r}$ from 1 to a target rank r, while warm starting each outer iteration with the output of the previous iteration. To compare the data efficiency of `MatrixIRLS` with the one of these three algorithms, we repeat the experiments of Section 5.1 for these methods and report the median Frobenius errors for the completion of 1000×1000 matrices of rank $r = 5$ with condition numbers $\kappa = 10$ and $\kappa = 10^5$, respectively, with those of `MatrixIRLS` in Figures 5 and 6.

In Figure 5, we observe that in the presence of a relatively small condition number of $\kappa = 10$, `MatrixIRLS` is more data efficient than the two rank-adaptaive methods as their phase transition occurs for a larger oversampling factor ($\rho = 1.8$ vs. $\rho = 1.5$).

On the other hand, it can be seen in Figure 6 that the rank-adaptive strategies `LRGeomCG Pursuit` and `R3MC w/ Rank Update` shine when completing matrices with large condition number such as $\kappa = 10^5$, as their phase transition occurs at around $\rho = 1.8$ and $\rho = 1.7$, where it occurs at $\rho = 1.9$ for `MatrixIRLS`. This shows that for large condition number, rank adaptive strategies can outperform the data efficiency of `MatrixIRLS`, and in both experiments, the phase transitions are considerably better than for their fixed rank versions `LRGeomCG` and `R3MC`, cf. Figures 1 and Figure 2.

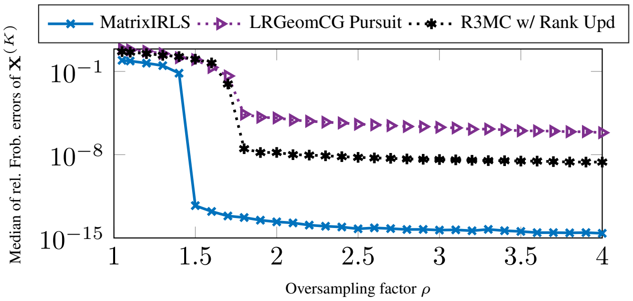


Figure 5. Completion of 1000×1000 matrices of rank $r = 5$ with condition number $\kappa = 10$, experiment as in Figure 1.

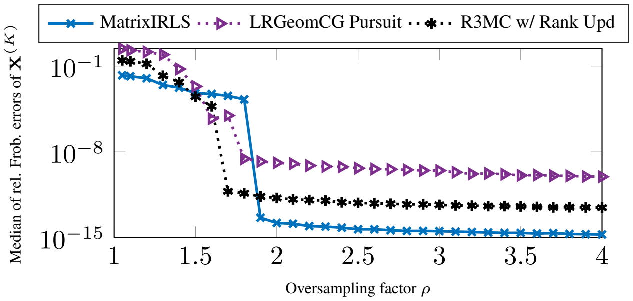


Figure 6. Completion of 1000×1000 matrices of rank $r = 5$ with condition number $\kappa = 10^5$, experiment as in Figure 2.

In all experiments so far, we have considered low-rank matrices with r singular values that exponentially decrease from κ to 1, as described in the beginning of this section. This might be a setting that is particularly suitable for rank-adaptive strategies that increase the rank parameter $\widetilde{r}$ one-by-one, as the singular subspaces are all one-dimensional and well-separated. For this reason, in a last experiment, we change this setup and consider ground truth matrices $\mathbf{X}^0$ that have a *plateau* in the set of singular values, potentially presenting a larger challenge for completion methods due to a higher dimensional subspace spanned by a set of multiple singular vectors. In particular, we consider the completion of a 1000×1000 matrix $\mathbf{X}^0$ with 10 singular values equal to $10^{10} \cdot \exp(-10 \cdot \log(10) \frac{14}{29})$, and with 10 singular values linearly interpolated on a logarithmic scale between this value and 10^{10} and, and another 10 between this value and 1 (see also Appendix E.2 for an illustration). For a random instance of such a matrix, we report the relative Frobenius error

[1] The MATLAB code containing the rank update was provided by B. Vandereycken in a private communication.

vs. execution time for the methods `MatrixIRLS` against the rank-adaptive variants of `LRGeomCG` and `R3MC`, here denoted by `LRGeomCG Pursuit` and `R3MC w/ Rank Update` in Figure 7, from random samples with a small oversampling factor of $\rho = 1.5$.

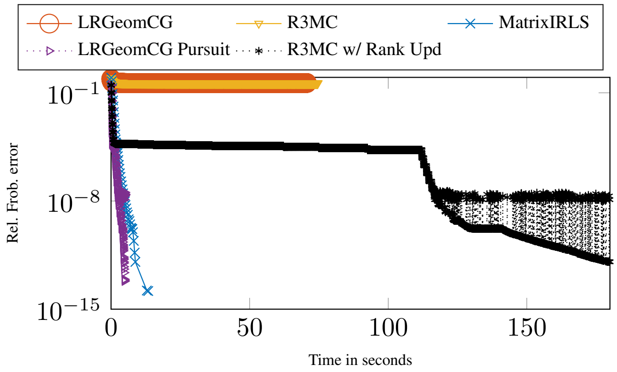


Figure 7. Comparison of matrix completion algorithms for 1000 × 1000 matrices of rank $r = 30$ with condition number $\kappa = 10^{10}$ and 10 equal singular values, oversampling factor of $\rho = 1.5$.

We observe that the fixed-rank variants `LRGeomCG` and `R3MC` are not able to complete the matrix, which is in line with the experiment of Section 5.2. `R3MC w/ Rank Update` exhibits a quick error decrease to a range around $6 \cdot 10^{-5}$, after which it just decreases very slowly for around 110 seconds, before converging to $\mathbf{X}^0$ up to an error of around 10^{-12} within another 70 seconds. The stagnation phase presumably corresponds to the learning of the 10-dimensional singular space of $\mathbf{X}^0$ in the central part of its spectrum. `LRGeomCG Pursuit`, on the other hand, reaches an error of around 10^{-12} already after 5 seconds, albeit without monotonicity with a fluctuation phase between errors of 10^{-8} and 10^{-12} from seconds 3 to 5. For `MatrixIRLS`, we use a tolerance parameter for the relative residual in the conjugate gradient method of $\text{tol}_{\text{inner}} = 10^{-3}$ and a maximal number of 3 iterations for the randomized Block Krylov method (cf. E for the default parameters), and observe that the method successfully converges to $\mathbf{X}^0$ slightly slower with a convergence within 13 seconds, but, remarkably, unlike `LRGeomCG Pursuit`, with a monotonous error decrease.[2]

6. Conclusion and Outlook

We formulated `MatrixIRLS`, a second order method that is able to efficiently complete large, highly ill-conditioned matrices from few samples, a problem for which most state-of-the-art methods fail. It improves on previous approaches for the optimization of non-convex rank objectives by applying a suitable smoothing strategy combined with saddle-escaping Newton-type steps.

As one goal of our investigation has been also to provide an efficient implementation, we focused on the matrix completion problem, leaving the extension of the ideas to other low-rank matrix estimation problems to future work including the case of inexact data or measurement errors. Furthermore, while we establish a local convergence guarantee for the algorithm, a precise analysis of its global convergence behavior might be of interest.

Software

An implementation of `MatrixIRLS` including scripts to reproduce the presented experiments can be found at `https://github.com/ckuemmerle/MatrixIRLS`.

Acknowledgements

We thank Bart Vandereycken and Boaz Nadler for fruitful discussions and for providing very valuable feedback. C.K. is grateful for support by the NSF under the grant NSF-IIS-1837991. C.M.V. is supported by the German Science Foundation (DFG) within the Gottfried Wilhelm Leibniz Prize under Grant BO 1734/20-1, under contract number PO-1347/3-2, under Emmy Noether junior research group KR 4512/1-1 and within Germany's Excellence Strategy EXC-2111 390814868.

References

Amelunxen, D., Lotz, M., McCoy, M. B., and Tropp, J. A. Living on the edge: Phase transitions in convex programs with random data. *Information and Inference: A Journal of the IMA*, 3(3):224–294, 2014.

Andersson, F., Carlsson, M., and Perfekt, K.-M. Operator-Lipschitz estimates for the singular value functional calculus. *Proc. Amer. Math. Soc.*, 144(5):1867–1875, 2016.

Attouch, H., Bolte, J., and Svaiter, B. F. Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward–backward splitting, and regularized Gauss–Seidel methods. *Mathematical Programming*, 137(1):91–129, 2013.

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

Björck, Å. *Numerical Methods for Least Squares Problems*. Society for Industrial and Applied Mathematics, 1996.

[2] For the default choice of algorithmic parameters as described in Appendix E, we obtain a qualitatively similar behavior for `MatrixIRLS`, but with a small runtime multiple due to the higher required precision at each iteration.

Bolte, J., Daniilidis, A., and Lewis, A. The Łojasiewicz inequality for nonsmooth subanalytic functions with applications to subgradient dynamical systems. *SIAM Journal on Optimization*, 17(4):1205–1223, 2007.

Bonnans, J. F., Gilbert, J. C., Lemaréchal, C., and Sagastizábal, C. A. A family of variable metric proximal methods. *Mathematical Programming*, 68(1):15–47, 1995.

Boumal, N. An introduction to optimization on smooth manifolds. *Available online at http://sma.epfl.ch/~nboumal/book/IntroOptimManifolds_Boumal_2020.pdf, November*, 2020.

Boumal, N. and Absil, P.-A. Low-rank matrix completion via preconditioned optimization on the grassmann manifold. *Linear Algebra and its Applications*, 15(475): 200–239, 2015.

Boumal, N., Mishra, B., Absil, P.-A., and Sepulchre, R. Manopt, a Matlab toolbox for optimization on manifolds. *J. Mach. Learn. Res.*, 15(42):1455–1459, 2014.

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

Candès, E. and Plan, Y. Matrix completion with noise. *Proceedings of the IEEE*, 98(6):925–936, 2010.

Candès, E. J. and Recht, B. Exact matrix completion via convex optimization. *Found. Comput. Math.*, 9(6):717–772, 2009.

Candès, E. J. and Tao, T. The Power of Convex Relaxation: Near-Optimal Matrix Completion. *IEEE Trans. Inf. Theory*, 56(5):2053–2080, 2010.

Candès, E. J., Eldar, Y., Strohmer, T., and Voroninski, V. Phase Retrieval via Matrix Completion. *SIAM J. Imag. Sci.*, 6(1):199–225, 2013.

Chen, J., Liu, D., and Li, X. Nonconvex rectangular matrix completion via gradient descent without $\ell_{2,\infty}$-regularization. *IEEE Trans. Inf. Theory*, 66(9):5806–5841, 2020a.

Chen, X. Smoothing methods for nonsmooth, nonconvex minimization. *Math. Program.*, 134(1):71–99, 2012.

Chen, X., Qi, L., and Sun, D. Global and superlinear convergence of the smoothing newton method and its application to general box constrained variational inequalities. *Mathematics of computation*, 67(222):519–540, 1998.

Chen, Y. Incoherence-Optimal Matrix Completion. *IEEE Trans. Inf. Theory*, 61(5):2909–2923, 2015. doi: 10.1109/TIT.2015.2415195.

Chen, Y. and Chi, Y. Harnessing Structures in Big Data via Guaranteed Low-Rank Matrix Estimation: Recent Theory and Fast Algorithms via Convex and Nonconvex Optimization. *IEEE Signal Process. Mag.*, 35(4):14–31, 2018.

Chen, Y., Chi, Y., Fan, J., Ma, C., and Yan, Y. Noisy matrix completion: Understanding statistical guarantees for convex relaxation via nonconvex optimization. *SIAM Journal on Optimization*, 30(4):3098–3121, 2020b.

Chi, Y., Lu, Y. M., and Chen, Y. Nonconvex Optimization Meets Low-Rank Matrix Factorization: An Overview. *IEEE Trans. Signal Process.*, 67(20):5239–5269, 2019.

Chouzenoux, E., Pesquet, J.-C., and Repetti, A. Variable metric forward–backward algorithm for minimizing the sum of a differentiable function and a convex function. *Journal of Optimization Theory and Applications*, 162(1): 107–132, 2014.

Cloninger, A., Czaja, W., Bai, R., and Basser, P. J. Solving 2D Fredholm Integral from Incomplete Measurements Using Compressive Sensing. *SIAM J. Imaging Sci.*, 7(3): 1775–1798, 2015.

Combettes, P. L. and Pesquet, J.-C. Proximal splitting methods in signal processing. In Bauschke, H. H., Burachik, R. S., Combettes, P. L., Elser, V., Luke, D. R., and Wolkowicz, H. (eds.), *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, pp. 185–212. Springer New York, New York, NY, 2011.

Daubechies, I., DeVore, R., Fornasier, M., and Güntürk, C. Iteratively reweighted least squares minimization for sparse recovery. *Commun. Pure Appl. Math.*, 63:1–38, 2010.

Dauphin, Y. N., Pascanu, R., Gulcehre, C., Cho, K., Ganguli, S., and Bengio, Y. Identifying and attacking the saddle point problem in high-dimensional non-convex optimization. In *Advances in Neural Information Processing Systems (NIPS)*, pp. 2933–2941, 2014.

Davenport, M. A. and Romberg, J. An Overview of Low-Rank Matrix Recovery From Incomplete Observations. *IEEE J. Sel. Topics Signal Process.*, 10:608–622, 2016.

Ding, C., Sun, D., Sun, J., and Toh, K.-C. Spectral operators of matrices. *Math. Program.*, 168(1):509–531, 2018.

Fannjiang, A. and Liao, W. Coherence Pattern–Guided Compressive Sensing with Unresolved Grids. *SIAM J. Imaging Sciences*, 5(1):179–202, 2012.

Fazel, M. *Matrix rank minimization with applications*. Ph.D. Thesis, Electrical Engineering Department, Stanford University, 2002.

Fazel, M., Hindi, H., and Boyd, S. P. Log-det heuristic for matrix rank minimization with applications to Hankel and Euclidean distance matrices. In *Proceedings of the American Control Conference*, volume 3, pp. 2156–2162, 2003.

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

Fornasier, M., Peter, S., Rauhut, H., and Worm, S. Conjugate gradient acceleration of iteratively re-weighted least squares methods. *Comput. Optim. Appl.*, 65(1):205–259, 2016.

Foucart, S. Concave Mirsky Inequality and Low-Rank Recovery. *SIAM J. Matrix Anal. Appl.*, 39(1):99–103, 2018.

Frankel, P., Garrigos, G., and Peypouquet, J. Splitting methods with variable metric for Kurdyka–Łojasiewicz functions and general convergence rates. *Journal of Optimization Theory and Applications*, 165(3):874–900, 2015.

Giampouras, P., Vidal, R., Rontogiannis, A., and Haeffele, B. A novel variational form of the Schatten-p quasi-norm. In *NeurIPS*, 2020.

Gohberg, I., Goldberg, S., and Krupnik, N. *Traces and determinants of linear operators*, volume 116 of *Operator Theory: Advances and Applications*. Springer, Basel, 2000.

Golub, G. H. and Underwood, R. The block lanczos method for computing eigenvalues. In *Mathematical software*, pp. 361–377. Elsevier, 1977.

Hestenes, M. R. and Stiefel, E. Methods of conjugate gradients for solving linear systems. *Journal of research of the National Bureau of Standards*, 49(1), 1952.

Horn, R. and Johnson, C. *Topics in Matrix Analysis*. Cambridge University Press, 1st edition edition, 1991.

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

Koren, Y., Bell, R., and Volinsky, C. Matrix Factorization Techniques for Recommender Systems. *Computer*, 42(8): 30–37, 2009.

Kümmerle, C. and Sigl, J. Harmonic Mean Iteratively Reweighted Least Squares for Low-Rank Matrix Recovery. *J. Mach. Learn. Res.*, 19(47):1–49, 2018.

Lai, M., Xu, Y., and Yin, W. Improved iteratively reweighted least squares for unconstrained smoothed ℓ_q minimization. *SIAM J. Numer. Anal*, pp. 2013, 2013.

Lewis, A. S. and Sendov, H. S. Nonsmooth Analysis of Singular Values. Part I: Theory. *Set-Valued Analysis*, 13 (3):213–241, 2005.

Lu, C., Tang, J., Yan, S., and Lin, Z. Nonconvex nonsmooth low rank minimization via iteratively reweighted nuclear norm. *IEEE Trans. Image Process.*, 25(2):829–839, 2015.

Luo, Y., Huang, W., Li, X., and Zhang, A. R. Recursive importance sketching for rank constrained least squares: Algorithms and high-order convergence. *arXiv preprint arXiv:2011.08360*, 2020.

Ma, C., Wang, K., Chi, Y., and Chen, Y. Implicit regularization in nonconvex statistical estimation: Gradient descent converges linearly for phase retrieval, matrix completion, and blind deconvolution. *Foundations of Computational Mathematics*, 20:451–632, 2020.

Mazumder, R., Saldana, D., and Weng, H. Matrix completion with nonconvex regularization: Spectral operators and scalable algorithms. *Statistics and Computing*, pp. 1–26, 2020.

Meurant, G. *The Lanczos and Conjugate Gradient Algorithms: From Theory to Finite Precision Computations*. Society for Industrial and Applied Mathematics,, 2006.

Mirsky, L. Symmetric Gauge Functions And Unitarily Invariant Norms. *The Quarterly Journal of Mathematics*, 11(1):50–59, 1960.

Mishra, B. and Sepulchre, R. R3MC: A Riemannian three-factor algorithm for low-rank matrix completion. In *53rd IEEE Conference on Decision and Control, CDC 2014, Los Angeles, CA, USA, December 15-17, 2014*, pp. 1137–1142. IEEE, 2014.

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

Murray, W. Newton-Type Methods. *Wiley Encyclopedia of Operations Research and Management Science*, 2010.

Musco, C. and Musco, C. Randomized Block Krylov Methods for Stronger and Faster Approximate Singular Value Decomposition. In *Advances in Neural Information Processing Systems (NIPS)*, pp. 1396–1404, 2015.

Nocedal, J. and Wright, S. *Numerical optimization*. Springer Science & Business Media, 2006.

Noferini, V. A Formula for the Fréchet Derivative of a Generalized Matrix Function. *SIAM J. Matrix Anal. Appl.*, 38(2):434–457, 2017.

Paternain, S., Mokhtari, A., and Ribeiro, A. A Newton-Based Method for Nonconvex Optimization with Fast Evasion of Saddle Points. *SIAM J. Optim.*, 29(1):343–368, 2019.

Pimentel-Alarcón, D. L., Boston, N., and Nowak, R. D. A Characterization of Deterministic Sampling Patterns for Low-Rank Matrix Completion. *IEEE J. Sel. Topics Signal Process.*, 10(4):623–636, 2016.

Recht, B. A Simpler Approach to Matrix Completion. *J. Mach. Learn. Res.*, 12:3413–3430, 2011.

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

Recht, B., Xu, W., and Hassibi, B. Null space conditions and thresholds for rank minimization. *Math. Program.*, 127(1):175–202, 2011.

Stewart, M. Perturbation of the SVD in the presence of small singular values. *Linear Algebra Appl.*, 419(1):53–77, 2006.

Sun, R. and Luo, Z. Q. Guaranteed Matrix Completion via Non-Convex Factorization. *IEEE Trans. Inf. Theory*, 62 (11):6535–6579, 2016.

Tan, M., Tsang, I. W., Wang, L., Vandereycken, B., and Pan, S. J. Riemannian pursuit for big matrix recovery. In *International Conference on Machine Learning (ICML)*, pp. 1539–1547. PMLR, 2014.

Tanner, J. and Wei, K. Normalized Iterative Hard Thresholding for Matrix Completion. *SIAM J. Sci. Comput.*, 35 (5):S104–S125, 2013.

Tanner, J. and Wei, K. Low rank matrix completion by alternating steepest descent methods. *Appl. Comput. Harmon. Anal.*, 40(2):417–429, 2016.

Tong, T., Ma, C., and Chi, Y. Accelerating ill-conditioned low-rank matrix estimation via scaled gradient descent. *arXiv preprint arXiv:2005.08898*, 2020.

Uschmajew, A. and Vandereycken, B. Greedy rank updates combined with Riemannian descent methods for low-rank optimization. In *Sampling Theory and Applications (SampTA), 2015 International Conference on*, pp. 420–424. IEEE, 2015.

Vandereycken, B. Low-Rank Matrix Completion by Riemannian Optimization. *SIAM J. Optim.*, 23(2), 2013.

Vershynin, R. *High-Dimensional Probability: An Introduction with Applications in Data Science*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2018. doi: 10.1017/9781108231596.

Voronin, S. *Regularization of linear systems with sparsity constraints with applications to large scale inverse problems*. Ph.D. Thesis, Princeton University, 2012.

Wedin, P.-Å. Perturbation bounds in connection with singular value decomposition. *BIT*, 12(1):99–111, 1972.

Wei, K., Cai, J.-F., Chan, T. F., and Leung, S. Guarantees of Riemannian Optimization for Low Rank Matrix Recovery. *SIAM J. Matrix Anal. Appl.*, 37(3):1198–1222, 2016.

Wei, K., Cai, J.-F., Chan, T. F., and Leung, S. Guarantees of Riemannian optimization for low rank matrix completion. *Inverse Probl. Imaging*, 14(2):233–265, 2020.

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

Woodbury, M. A. Inverting modified matrices. *Memorandum report*, 42(106):336, 1950.

Yang, Z. *A study on nonsymmetric matrix-valued functions*. Master's thesis, Department of Mathematics, National University of Singapore, 2009. URL https://www.polyu.edu.hk/ama/profile/dfsun/Main_YZ.pdf.

Yuan, Q., Gu, M., and Li, B. Superlinear convergence of randomized block lanczos algorithm. In *2018 IEEE International Conference on Data Mining (ICDM)*, pp. 1404–1409. IEEE, 2018.

Zheng, Q. and Lafferty, J. Convergence Analysis for Rectangular Matrix Completion Using Burer-Monteiro Factorization and Gradient Descent. *arXiv preprint arXiv:1605.07051*, 2016.

Supplementary material for *A Scalable Second Order Method for Ill-Conditioned Matrix Completion from Few Samples*

This supplementary material is divided into three parts: In Appendix A, we provide details for an efficient implementation of `MatrixIRLS` and furthermore, show Theorem 3.1. Then, we show in Appendix B and in Appendix C the theoretical results of Section 4. In Appendix D, we provide some details about the interpretation of `MatrixIRLS` as a saddle-escaping smoothing Newton method, cf. Section 4.2. Finally, we present a detailed description of the algorithmic parameters of the experiments of Section 5 in Appendix E, as well as a remark on an experiment of Section 5.3.

A. Computational Aspects of MatrixIRLS

We first introduce some notation that is suitable to describe an efficient implementation of Algorithm 1. Whenever we write $\|\mathbf{X}\|_{S_\infty}$, we refer to the Schatten-∞ or spectral norm $\|\mathbf{X}\|_{S_\infty} = \sigma_1(\mathbf{X})$ of a matrix $\mathbf{X} \in \mathbb{R}^{d_1 \times d_2}$. For $d_1, d_2 \in \mathbb{N}$, $d = \min(d_1, d_2)$ and a vector $\sigma \in \mathbb{R}^d$, we denote the $(d_1 \times d_2)$-matrix with the entries of σ on its diagonal (an 0 otherwise) as $\mathrm{dg}(\sigma) \in \mathbb{R}^{d_1 \times d_2}$, i.e., $\mathrm{dg}(\sigma)_{ij} = \sigma_i \delta_{ij}$ for all $i \in [d_1]$ and $j \in [d_2]$. If $\mathbf{X}^{(k)} \in \mathbb{R}^{d_1 \times d_2}$ is a matrix with singular values $\sigma_i^{(k)} := \sigma_i(\mathbf{X}^{(k)})$, for $\epsilon_k > 0$, let $r_k := |\{i \in [d] : \sigma_i(\mathbf{X}^{(k)}) > \epsilon_k\}| = |\{i \in [d] : \sigma_i^{(k)} > \epsilon_k\}|$. Then we write a singular value decomposition of $\mathbf{X}^{(k)}$ such that

$$\mathbf{X}^{(k)} = \mathbf{U}_k \,\mathrm{dg}(\sigma^{(k)}) \mathbf{V}_k^* = \begin{bmatrix} \mathbf{U}^{(k)} & \mathbf{U}_\perp^{(k)} \end{bmatrix} \begin{bmatrix} \mathbf{\Sigma}^{(k)} & 0 \\ 0 & \mathbf{\Sigma}_\perp^{(k)} \end{bmatrix} \begin{bmatrix} \mathbf{V}^{(k)*} \\ \mathbf{V}_\perp^{(k)*} \end{bmatrix}, \tag{9}$$

where $\mathbf{U}_k \in \mathbb{R}^{d_1 \times d_1}$ and $\mathbf{V}_k \in \mathbb{R}^{d_2 \times d_2}$, and corresponding submatrices $\mathbf{U}^{(k)} \in \mathbb{R}^{d_1 \times r_k}$, $\mathbf{U}_\perp^{(k)} \in \mathbb{R}^{d_1 \times (d_1 - r_k)}$, $\mathbf{V}^{(k)} \in \mathbb{R}^{d_2 \times r_k}$, $\mathbf{V}_\perp^{(k)} \in \mathbb{R}^{d_2 \times (d_2 - r_k)}$, $\mathbf{\Sigma}^{(k)} := \mathrm{diag}(\sigma_1^{(k)}, \ldots \sigma_{r_k}^{(k)})$ and $\mathbf{\Sigma}_\perp^{(k)} := \mathrm{dg}(\sigma_{r_k+1}^{(k)}, \ldots \sigma_d^{(k)})$. Furthermore, we denote by $\mathcal{T}_{r_k}(\mathbf{X}^{(k)})$ the *best rank-r_k approximation* of $\mathbf{X}^{(k)}$, i.e.,

$$\mathcal{T}_{r_k}(\mathbf{X}^{(k)}) := \underset{\mathbf{Z}:\mathrm{rank}(\mathbf{Z}) \leq r_k}{\arg\min} \|\mathbf{Z} - \mathbf{X}^{(k)}\| = \mathbf{U}^{(k)} \mathbf{\Sigma}^{(k)} \mathbf{V}^{(k)*}, \tag{10}$$

where $\|\cdot\|$ can be any unitarily invariant norm, due to the Eckardt-Young-Mirsky theorem (Mirsky, 1960). Let $\mathcal{M}_r := \{\mathbf{X} \in \mathbb{R}^{d_1 \times d_2} : \mathrm{rank}(\mathbf{X}) = r\}$ the manifold of rank-r matrices. Given these definitions, let

$$\begin{aligned} T_k := T_{\mathcal{T}_{r_k}(\mathbf{X}^{(k)})} \mathcal{M}_{r_k} &:= \left\{ \begin{bmatrix} \mathbf{U}^{(k)} \mathbf{U}_\perp^{(k)} \end{bmatrix} \begin{bmatrix} \mathbb{R}^{r_k \times r_k} & \mathbb{R}^{r_k (d_2 - r_k)} \\ \mathbb{R}^{(d_1 - r_k) r_k} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{V}^{(k)} \mathbf{V}_\perp^{(k)} \end{bmatrix}^* \right\} \\ &= \left\{ \begin{bmatrix} \mathbf{U}^{(k)} \mathbf{U}_\perp^{(k)} \end{bmatrix} \begin{bmatrix} \mathbf{M}_1 & \mathbf{M}_2 \\ \mathbf{M}_3 & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{V}^{(k)} \mathbf{V}_\perp^{(k)} \end{bmatrix}^* : \mathbf{M}_1 \in \mathbb{R}^{r_k \times r_k}, \mathbf{M}_2 \in \mathbb{R}^{r_k \times (d_2 - r_k)}, \mathbf{M}_3 \in \mathbb{R}^{(d_1 - r_k) \times r_k} \text{ arbitrary} \right\} \\ &= \{ \mathbf{U}^{(k)} \Gamma_1 \mathbf{V}^{(k)*} + \mathbf{U}^{(k)} \Gamma_2 \left(\mathbf{I} - \mathbf{V}^{(k)} \mathbf{V}^{(k)*} \right) + (\mathbf{I} - \mathbf{U}^{(k)} \mathbf{U}^{(k)*}) \Gamma_3 \mathbf{V}^{(k)*} : \Gamma_1 \in \mathbb{R}^{r_k \times r_k}, \Gamma_2 \in \mathbb{R}^{r_k \times d_2}, \Gamma_3 \in \mathbb{R}^{d_1 \times r_k} \} \end{aligned} \tag{11}$$

be tangent space of the rank-r_k matrix manifold at $\mathcal{T}_{r_k}(\mathbf{X}^{(k)})$, see also (Vandereycken, 2013) and Chapter 7.5 of (Boumal, 2020). While it is often more advantageous to computationally represent elements of T_k as in the last representation, it becomes clear from the first equality that T_k is an $r_k(d_1 + d_2 - r_k) = r_k^2 + r_k(d_1 - r_k) + r_k(d_2 - r_k)$-dimensional subspace of $\mathbb{R}^{d_1 \times d_2}$. If $\{B_i\}_{i=1}^{r_k(d_1+d_2-r_k)}$ is the standard orthonormal basis of the subspace T_k and $S_k := \mathbb{R}^{r_k(d_1+d_2-r_k)}$, let $P_{T_k} : S_k \to T_k$ be the parametrization operator such that for $\gamma \in S_k$,

$$P_{T_k}(\gamma) := \sum_{i=1}^{r_k(d_1+d_2-r_k)} \gamma_i B_i = \mathbf{U}^{(k)} \Gamma_1 \mathbf{V}^{(k)*} + \mathbf{U}^{(k)} \Gamma_2 \left(\mathbf{I} - \mathbf{V}^{(k)} \mathbf{V}^{(k)*} \right) + (\mathbf{I} - \mathbf{U}^{(k)} \mathbf{U}^{(k)*}) \Gamma_3 \mathbf{V}^{(k)*}$$

where $\Gamma_1 \in \mathbb{R}^{r_k \times r_k}, \Gamma_2 \in \mathbb{R}^{r_k \times d_2}, \Gamma_2 \mathbf{V}^{(k)} = 0, \Gamma_3 \in \mathbb{R}^{d_1 \times r_k}, \mathbf{U}^{(k)*}\Gamma_3 = 0$. The representation $\gamma \in S_k$ of an element of T_k is called an *intrinsic representation* (Huang et al., 2017), and while this is not trivial observation, we note that it is possible to calculate the matrices Γ_1, Γ_2 and Γ_3 of $P_{T_k}(\gamma)$ from just from the knowledge of γ, $\mathbf{U}^{(k)} \in \mathbb{R}^{d_1 \times r_k}$ and $\mathbf{V}^{(k)} \in \mathbb{R}^{d_2 \times r_k}$ in just $8(d_1 + d_2)r^2 = O(Dr^2)$ flops, see Algorithm 7 of (Huang et al., 2017). Furthermore, we denote by $P_{T_k}^*$ the adjoint operator of $P_{T_k} : S_k \to \mathbb{R}^{d_1 \times d_2}$, and this operator $P_{T_k}^* : \mathbb{R}^{d_1 \times d_2} \to S_k$ maps a matrix $\mathbf{X} \in \mathbb{R}^{d_1 \times d_2}$ first to $\{\Gamma_1, \Gamma_2, \Gamma_3\} := \left\{\mathbf{U}^{(k)*}\mathbf{X}\mathbf{V}^{(k)}, \mathbf{U}^{(k)*}\mathbf{X}(\mathbf{I} - \mathbf{V}^{(k)}\mathbf{V}^{(k)*}), (\mathbf{I} - \mathbf{U}^{(k)}\mathbf{U}^{(k)*})\mathbf{X}\mathbf{V}^{(k)}\right\}$ and then via Algorithm 6 of (Huang et al., 2017) to $\gamma = P_{T_k}^*(\mathbf{X}) \in S_k$. The computational complexity of this operator is $O(r\|\mathbf{X}\|_0 + Dr^2)$.

Together with P_{T_k} and $P_{T_k}^*$, the definition of the following diagonal operator $\mathbf{D}_{S_k} : S_k \to S_k$ allows us to reformulate and simplify the optimal weight operator $W^{(k)} : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}^{d_1 \times d_2}$ of Definition 2.1 as follows. Recalling that $\mathbf{H}_k \in \mathbb{R}^{d_1 \times d_2}$ is such that $(\mathbf{H}_k)_{ij} := \left(\max(\sigma_i^{(k)}, \epsilon_k)\max(\sigma_j^{(k)}, \epsilon_k)\right)^{-1}$ for all i and j, we write for each $\mathbf{Z} \in \mathbb{R}^{d_1 \times d_2}$

$$\begin{aligned} W^{(k)}(\mathbf{Z}) &= \mathbf{U}_k \left[\mathbf{H}_k \circ (\mathbf{U}_k^* \mathbf{Z} \mathbf{V}_k)\right] \mathbf{V}_k^* \\ &= \begin{bmatrix} \mathbf{U}^{(k)} & \mathbf{U}_\perp^{(k)} \end{bmatrix} \left(\mathbf{H}_k \circ \begin{bmatrix} \mathbf{U}^{(k)*}\mathbf{Z}\mathbf{V}^{(k)} & \mathbf{U}^{(k)*}\mathbf{Z}\mathbf{V}_\perp^{(k)} \\ \mathbf{U}_\perp^{(k)*}\mathbf{Z}\mathbf{V}^{(k)} & \mathbf{U}_\perp^{(k)*}\mathbf{Z}\mathbf{V}_\perp^{(k)} \end{bmatrix} \right) \begin{bmatrix} \mathbf{V}^{(k)*} \\ \mathbf{V}_\perp^{(k)*} \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{U}^{(k)} & \mathbf{U}_\perp^{(k)} \end{bmatrix} \left(\begin{bmatrix} \mathbf{H}^{(k)} & \mathbf{H}_{1,2}^{(k)} \\ \mathbf{H}_{2,1}^{(k)} & \epsilon_k^{-2}\mathbf{1} \end{bmatrix} \circ \begin{bmatrix} \mathbf{U}^{(k)*}\mathbf{Z}\mathbf{V}^{(k)} & \mathbf{U}^{(k)*}\mathbf{Z}\mathbf{V}_\perp^{(k)} \\ \mathbf{U}_\perp^{(k)*}\mathbf{Z}\mathbf{V}^{(k)} & \mathbf{U}_\perp^{(k)*}\mathbf{Z}\mathbf{V}_\perp^{(k)} \end{bmatrix} \right) \begin{bmatrix} \mathbf{V}^{(k)*} \\ \mathbf{V}_\perp^{(k)*} \end{bmatrix} \\ &= \left(P_{T_k}\mathbf{D}_{S_k}P_{T_k}^* + \epsilon_k^{-2}\left(\mathbf{I} - P_{T_k}P_{T_k}^*\right)\right)\mathbf{Z}, \end{aligned} \tag{12}$$

where the matrices $\mathbf{H}^{(k)} \in \mathbb{R}^{r_k \times r_k}$, $\mathbf{H}_{1,2}^{(k)} \in \mathbb{R}^{r_k \times (d_2 - r_k)}$ and $\mathbf{H}_{2,1}^{(k)} \in \mathbb{R}^{(d_1 - r_k) \times r_k}$ are defined such that

$$\mathbf{H}_{ij}^{(k)} = \left(\sigma_i^{(k)}\sigma_j^{(k)}\right)^{-1} \text{ for all } i, j \in [r_k], \tag{13}$$

$\left(\mathbf{H}_{1,2}^{(k)}\right)_{ij} = \left(\sigma_i^{(k)}\epsilon_k\right)^{-1}$ for all $i \in [r_k]$ and $j \in [d_2 - r_k]$ and $\left(\mathbf{H}_{2,1}^{(k)}\right)_{ij} = \left(\epsilon_k\sigma_j^{(k)}\right)^{-1}$ for all $i \in [d_1 - r_k]$ and $j \in [r_k]$. Furthermore, $\mathbf{1}$ in the third line is the $((d_1 - r_k) \times (d_2 - r_k))$-matrix of ones $\mathbf{1}$, and $\mathbf{I}$ is the identity operator. Defining $\mathbf{D}_{S_k} : S_k \to S_k$ implicitly through the last equality, we observe that $\mathbf{D}_{S_k}$ is a diagonal matrix with the entries of $\mathbf{H}^{(k)}$, $\mathbf{H}_{1,2}^{(k)}$ and $\mathbf{H}_{2,1}^{(k)}$ enumerated on its diagonal. Recall that the set of indices corresponding to provided entries is defined as $\Omega = \{(i_\ell, j_\ell)\} \subset [d_1] \times [d_2]$, and $P_\Omega : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}^m$ is the subsampling operator

$$P_\Omega(\mathbf{Z}) = \sum_{(i_\ell, j_\ell) \in \Omega} \langle e_{i_\ell}, \mathbf{Z} e_{j_\ell} \rangle,$$

where e_{i_ℓ} and e_{j_ℓ} are the i_ℓ-th and j_ℓ-th standard basis vectors of $\mathbb{R}^{d_1}$ and $\mathbb{R}^{d_2}$, respectively.

A.1. Implementation of MatrixIRLS

The definitions above now enable us to implement the weighted least squares step of (5) efficiently.

In the following lemma, we show that Algorithm 2 indeed computes the solution of the weighted least squares step (5) of Algorithm 1.

Lemma A.1. *If* $\mathbf{r}_{k+1} \in \mathbb{R}^m$ *and* $\gamma_k \in S_k$ *is the output of Algorithm 2, then* $\mathbf{X}^{(k+1)}$ *as in step* (5) *of Algorithm 1 fulfills*

$$\mathbf{X}^{(k+1)} = P_\Omega^*(\mathbf{r}_{k+1}) + P_{T_k}(\gamma_k).$$

Proof of Lemma A.1. Let $W^{(k)} : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}^{d_1 \times d_2}$ be the weight operator of Definition 2.1. As $W^{(k)}$ is a positive definite operator, it holds that the minimizer

$$\mathbf{X}^{(k+1)} = \operatorname*{arg\,min}_{P_\Omega(\mathbf{X}) = \mathbf{y}} \langle \mathbf{X}, W^{(k)}(\mathbf{X}) \rangle$$

of (5) is unique, and it is well-known (Björck, 1996) that the solution of this linearly constrained weighted least squares problem can be written such that

$$\mathbf{X}^{(k+1)} = (W^{(k)})^{-1} P_\Omega^* \left(P_\Omega (W^{(k)})^{-1} P_\Omega^*\right)^{-1} (\mathbf{y}), \tag{15}$$

Algorithm 2 Practical implementation of weighted least squares step of `MatrixIRLS`

```
Input: Set $\Omega$, observations $\mathbf{y} \in \mathbb{R}^m$, left and right singular vectors $\mathbf{U}^{(k)} \in \mathbb{R}^{d_1 \times r_k}$, $\mathbf{V}^{(k)} \in \mathbb{R}^{d_2 \times r_k}$ and singular values $\sigma_1^{(k)}, \ldots, \sigma_{r_k}^{(k)}$, smoothing parameter $\epsilon_k$, projection $\gamma_k^{(0)} = P_{T_k}^* P_{T_{k-1}}(\gamma_{k-1}) \in S_k$ of solution $\gamma_{k-1} \in S_{k-1}$ of linear system eq. (14) for previous iteration $k-1$.
1: Compute $\mathbf{h}_k^0 := P_{T_k}^* P_\Omega^* (\mathbf{y}) - \left(\epsilon_k^2 \left(\mathbf{D}_{S_k}^{-1} - \epsilon_k^2 \mathbf{I}_{S_k}\right)^{-1} + P_{T_k}^* P_\Omega^* P_\Omega P_{T_k}\right) \gamma_k^{(0)} \in S_k$.
2: Solve
       $\left(\epsilon_k^2 \left(\mathbf{D}_{S_k}^{-1} - \epsilon_k^2 \mathbf{I}_{S_k}\right)^{-1} + P_{T_k}^* P_\Omega^* P_\Omega P_{T_k}\right) \Delta\gamma_k = \mathbf{h}_k^0$        (14)
   for $\Delta\gamma_k \in S_k$ by the *conjugate gradient* method (Hestenes & Stiefel, 1952; Meurant, 2006).
3: Compute $\gamma_k = \gamma_k^{(0)} + \Delta\gamma_k$.
4: Compute residual $\mathbf{r}_{k+1} := \mathbf{y} - P_\Omega P_{T_k}(\gamma_k) \in \mathbb{R}^m$.
Output: $\mathbf{r}_{k+1} \in \mathbb{R}^m$ and $\gamma_k \in S_k$.
```

where $(W^{(k)})^{-1} : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}^{d_1 \times d_2}$ is the *inverse* of the weight operator $W^{(k)}$: This inverse exists as $W^{(k)}$ is self-adjoint and positive definite, which can be shown by realizing that its $(d_1 d_2 \times d_1 d_2)$-matrix representation has eigenvectors $v_i^{(k)} \otimes u_j^{(k)}$, where $v_i^{(k)} \in \mathbb{R}^{d_1}$ and $u_j^{(k)} \in \mathbb{R}^{d_2}$ are columns of $\mathbf{V}_k$ and $\mathbf{U}_k$, respectively, and that the eigenvalues of $W^{(k)}$ are just the entries of $\mathbf{H}^{(k)}$.

From this eigendecomposition of $W^{(k)}$ if further follows that the action of $(W^{(k)})^{-1} : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}^{d_1 \times d_2}$ is such that for each $\mathbf{Z} \in \mathbb{R}^{d_1 \times d_2}$,

$$(W^{(k)})^{-1}(\mathbf{Z}) = \mathbf{U}_k \left[\mathbf{H}_k^{-1} \circ (\mathbf{U}_k^* \mathbf{Z} \mathbf{V}_k)\right] \mathbf{V}_k^*,$$

where $\mathbf{H}_k^{-1} \in \mathbb{R}^{d_1 \times d_2}$ is the matrix whose entries are the *entrywise* inverse of the entries of $\mathbf{H}_k$. With the same argument as in (12), we can rewrite $(W^{(k)})^{-1}$ such that

$$(W^{(k)})^{-1} = P_{T_k} \mathbf{D}_{S_k}^{-1} P_{T_k}^* + \epsilon_k^2 \left(\mathbf{I} - P_{T_k} P_{T_k}^*\right) = P_{T_k} \left(\mathbf{D}_{S_k}^{-1} - \epsilon_k^2 \mathbf{I}_{S_k}\right) P_{T_k}^* + \epsilon_k^2 \mathbf{I},$$

where $\mathbf{I}_{S_k}$ is the identity on S_k and $\mathbf{D}_{S_k}^{-1}$ is the diagonal matrix that is the inverse of $\mathbf{D}_{S_k}$ in (12).

Since $P_\Omega P_\Omega^* = \mathbf{I}_m$, we can use this representation of $(W^{(k)})^{-1}$ to write

$$\left(P_\Omega (W^{(k)})^{-1} P_\Omega^*\right)^{-1} = \left(P_\Omega P_{T_k} \left(\mathbf{D}_{S_k}^{-1} - \epsilon_k^2 \mathbf{I}_{S_k}\right) P_{T_k}^* P_\Omega^* + \epsilon_k^2 \mathbf{I}_m\right)^{-1}.$$

Using the *Sherman-Morrison-Woodbury* formula (Woodbury, 1950)

$$(\mathbf{E}\mathbf{C}\mathbf{F}^* + \mathbf{B})^{-1} = \mathbf{B}^{-1} - \mathbf{B}^{-1}\mathbf{E}(\mathbf{C}^{-1} + \mathbf{F}^*\mathbf{B}^{-1}\mathbf{E})^{-1}\mathbf{F}^*\mathbf{B}^{-1}$$

for $\mathbf{B} := \epsilon_k^2 \mathbf{I}_m$, $\mathbf{C} := \left(\mathbf{D}_{S_k}^{-1} - \epsilon_k^2 \mathbf{I}_{S_k}\right)$ and $\mathbf{E} = \mathbf{F} := P_\Omega P_{T_k}$, we obtain that

$$\left(P_\Omega (W^{(k)})^{-1} P_\Omega^*\right)^{-1} = \epsilon_k^{-2}\mathbf{I} - \epsilon_k^{-2} P_\Omega P_{T_k} \left(\epsilon_k^2 \mathbf{C}^{-1} + P_{T_k}^* P_\Omega^* P_\Omega P_{T_k}\right)^{-1} P_{T_k}^* P_\Omega^* = \epsilon_k^{-2}\mathbf{I} - \epsilon_k^{-2} P_\Omega P_{T_k} \mathbf{M}^{-1} P_{T_k}^* P_\Omega^* \quad (16)$$

with linear system matrix $\mathbf{M} := \epsilon_k^2 \mathbf{C}^{-1} + P_{T_k}^* P_\Omega^* P_\Omega P_{T_k}$, noting that $\mathbf{C} = \left(\mathbf{D}_{S_k}^{-1} - \epsilon_k^2 \mathbf{I}_{S_k}\right)$ is invertible since $(\mathbf{H}_{ij}^{(k)})^{-1} = \sigma_i^{(k)} \sigma_j^{(k)} > \epsilon_k^2$ for all $i, j \in [r_k]$ and since $(\mathbf{D}_{ii}^{(k)})^{-1} = \sigma_i^{(k)} \epsilon_k > \epsilon_k^2$ for all $i \in [r_k]$.

Next, we note that the definition $\gamma_k = \gamma_k^{(0)} + \Delta\gamma_k$ and (14) implies that

$$\gamma_k = \gamma_k^{(0)} + \Delta\gamma_k = \gamma_k^{(0)} + \mathbf{M}^{-1}\mathbf{h}_k^0 = \gamma_k^{(0)} + \mathbf{M}^{-1}\left(P_{T_k}^* P_\Omega^* (\mathbf{y}) - \mathbf{M}\gamma_k^{(0)}\right) = \mathbf{M}^{-1} P_{T_k}^* P_\Omega^* (\mathbf{y}). \quad (17)$$

Inserting this into (16), we see that the residual $\mathbf{r}_{k+1}$ of Algorithm 2 satisfies

$$\mathbf{z} := \left(P_\Omega (W^{(k)})^{-1} P_\Omega^*\right)^{-1} (\mathbf{y}) = \epsilon_k^{-2}\mathbf{y} - \epsilon_k^{-2} P_\Omega P_{T_k} \mathbf{M}^{-1} P_{T_k}^* P_\Omega^* (\mathbf{y}) = \epsilon_k^{-2}\left(\mathbf{y} - P_\Omega P_{T_k}(\gamma_k)\right) = \epsilon_k^{-2}\mathbf{r}_{k+1}.$$

Thus, we obtain the representation

$$\begin{aligned}
\mathbf{X}^{(k+1)} &= (W^{(k)})^{-1} P_\Omega^* \left(P_\Omega (W^{(k)})^{-1} P_\Omega^* \right)^{-1} (\mathbf{y}) = (W^{(k)})^{-1} P_\Omega^*(\mathbf{z}) \\
&= \epsilon_k^2 \left[\mathbf{I} + \epsilon_k^{-2} P_{T_k} \mathbf{C} P_{T_k}^* \right] P_\Omega^*(\mathbf{z}) = \epsilon_k^2 P_\Omega^*(\mathbf{z}) + P_{T_k} \mathbf{C} P_{T_k}^* P_\Omega^*(\mathbf{z}) \\
&= \epsilon_k^2 P_\Omega^*(\mathbf{z}) + P_{T_k} \mathbf{C} \mathbf{C}^{-1} \left(\epsilon_k^2 \mathbf{C}^{-1} + P_{T_k}^* P_\Omega^* P_\Omega P_{T_k} \right)^{-1} P_{T_k}^* P_\Omega^*(\mathbf{y}) \\
&= P_\Omega^*(\mathbf{r}_{k+1}) + P_{T_k}(\gamma_k),
\end{aligned} \tag{18}$$

for $\mathbf{X}^{(k+1)}$, using (19) in the fifth equality, where (19) represents

$$\begin{aligned}
P_{T_k}^* P_\Omega^*(\mathbf{z}) &= \epsilon_k^{-2} P_{T_k}^* P_\Omega^*(\mathbf{r}_{k+1}) = \epsilon_k^{-2} \left(P_{T_k}^* P_\Omega^*(\mathbf{y}) - P_{T_k}^* P_\Omega^* P_\Omega P_{T_k}(\gamma_k) \right) \\
&= \epsilon_k^{-2} \left(P_{T_k}^* P_\Omega^*(\mathbf{y}) - P_{T_k}^* P_\Omega^* P_\Omega P_{T_k} \mathbf{M}^{-1} P_{T_k}^* P_\Omega^*(\mathbf{y}) \right) \\
&= \epsilon_k^{-2} \left(P_{T_k}^* P_\Omega^*(\mathbf{y}) - P_{T_k}^* P_\Omega^* P_\Omega P_{T_k} \left(\epsilon_k^2 \mathbf{C}^{-1} + P_{T_k}^* P_\Omega^* P_\Omega P_{T_k} \right)^{-1} P_{T_k}^* P_\Omega^*(\mathbf{y}) \right) \\
&= \epsilon_k^{-2} \left(P_{T_k}^* P_\Omega^*(\mathbf{y}) - \left(P_{T_k}^* P_\Omega^* P_\Omega P_{T_k} \pm \epsilon_k^2 \mathbf{C}^{-1} \right) \left(\epsilon_k^2 \mathbf{C}^{-1} + P_{T_k}^* P_\Omega^* P_\Omega P_{T_k} \right)^{-1} P_{T_k}^* P_\Omega^*(\mathbf{y}) \right) \\
&= \mathbf{C}^{-1} \left(\epsilon_k^2 \mathbf{C}^{-1} + P_{T_k}^* P_\Omega^* P_\Omega P_{T_k} \right)^{-1} P_{T_k}^* P_\Omega^*(\mathbf{y}),
\end{aligned} \tag{19}$$

which uses (17) in the third equality and the definition of $\mathbf{M}$ in the fourth equality.

This finishes the proof. □

With Lemma A.1, we are now able to prove Theorem 3.1.

Proof of Theorem 3.1. Based on Lemma A.1, we can compute the representation $\mathbf{X}^{(k+1)} = P_\Omega^*(\mathbf{r}_{k+1}) + P_{T_k}(\gamma_k)$ for $\mathbf{X}^{(k+1)}$ using Algorithm 2. By the assumption of Theorem 3.1, we know that $r = \widetilde{r} = r_k$.

The main computational cost in Algorithm 2 lies in the application of the operators $(P_\Omega P_{T_k})^* = P_{T_k}^* P_\Omega^* : \mathbb{R}^m \to S_k$, $P_\Omega P_{T_k} : S_k \to \mathbb{R}^m$ and $\left(\mathbf{D}_{S_k}^{-1} - \epsilon_k^2 \mathbf{I}_{S_k} \right)^{-1} : S_k \to S_k$. The application of $\left(\mathbf{D}_{S_k}^{-1} - \epsilon_k^2 \mathbf{I}_{S_k} \right)^{-1}$ has a time complexity of $r(d_1 + d_2 - r) = O(rD)$ as the operator is diagonal. The action of $P_{T_k}^* P_\Omega^*$ and $P_\Omega P_{T_k}$ can be computed as in Algorithm 3 and Algorithm 4, respectively.

Algorithm 3 Implementation of $P_{T_k}^* P_\Omega^* : \mathbb{R}^m \to S_k$

```
Input: Argument vector $\mathbf{y} \in \mathbb{R}^m$, index set $\Omega$, left and right singular vectors $\mathbf{U}^{(k)} \in \mathbb{R}^{d_1 \times r_k}$, $\mathbf{V}^{(k)} \in \mathbb{R}^{d_2 \times r_k}$.
$\mathbf{A}_1 = \mathbf{U}^{(k)*} P_\Omega^*(\mathbf{y}) \in \mathbb{R}^{r_k \times d_2}$.    ▷ $mr_k$ flops
$\mathbf{A}_2 = P_\Omega^*(\mathbf{y}) \mathbf{V}^{(k)} \in \mathbb{R}^{d_1 \times r_k}$.    ▷ $mr_k$ flops
$\Gamma_1 = \mathbf{A}_1 \mathbf{V}^{(k)} \in \mathbb{R}^{r_k \times r_k}$.    ▷ $d_2 r_k^2$ flops
$\Gamma_2 = \mathbf{A}_1 - \Gamma_1 \mathbf{V}^{(k)*} \in \mathbb{R}^{r_k \times d_2}$.    ▷ $2 d_2 r_k^2$ flops
$\Gamma_3 = \mathbf{A}_2 - \mathbf{U}^{(k)} \Gamma_1 \in \mathbb{R}^{d_1 \times r_k}$.    ▷ $d_1 r_k^2$ flops
Apply Algorithm 6 of (Huang et al., 2017) to compute[3] $\gamma \in S_k$ from $\{\Gamma_1, \Gamma_2, \Gamma_3\}$.    ▷ $4 r_k^2 (d_1 + d_2 - r_k) + 2 r_k^2$ flops
Output: $\gamma \in S_k$.
```

Using Algorithm 3 and Algorithm 4, we see that the first step of Algorithm 2 has a time complexity of $O(mr + r^2 D)$ flops, and each inner iteration of the conjugate gradient method in step 2 of Algorithm 2 has likewise a time complexity of $O(mr + r^2 D)$ flops. Finally, step 3 takes also $O(mr + r^2 D)$ flops.

We observe that the linear system (14) is positive definite, since $\epsilon_k^2 \left(\mathbf{D}_{S_k}^{-1} - \epsilon_k^2 \mathbf{I}_{S_k} \right)^{-1}$ is diagonal with positive entries and since $P_{T_k}^* P_\Omega^* P_\Omega P_{T_k}$ is a symmetric, positive definite operator. Thus, it is possible to use the conjugate gradient (CG) method, whose main step applies the three operators above at each iteration. It is known that in general, the CG method terminates with the exact solution γ_k after at most $N_{\text{CG\_inner}} = \dim(S_k) = r(d_1 + d_2 - r)$ iterations. However, if the system matrix

$$\frac{\epsilon_k^2 \mathbf{I}}{\mathbf{D}_{S_k}^{-1} - \epsilon_k^2 \mathbf{I}_{S_k}} + P_{T_k}^* P_\Omega^* P_\Omega P_{T_k}$$

[3] In the notation of (Huang et al., 2017), we have the correspondence that $U = \mathbf{U}^{(k)}$, $V = \mathbf{V}^{(k)}$, $\dot{S} = \Gamma_1$, $S\dot{V}^T = \Gamma_2$ and $\dot{U}S = \Gamma_3$.

Algorithm 4 Implementation of $P_\Omega P_{T_k} : S_k \to \mathbb{R}^m$

```
**Input:** Argument vector $\gamma \in S_k$, index set $\Omega$, left and right singular vectors $\mathbf{U}^{(k)} \in \mathbb{R}^{d_1 \times r_k}$, $\mathbf{V}^{(k)} \in \mathbb{R}^{d_2 \times r_k}$.
Apply Algorithm 7 of (Huang et al., 2017) to compute $\{\Gamma_1, \Gamma_2, \Gamma_3\}$ from $\gamma \in S_k$.    ▷ $8(d_1 + d_2)r_k^2$ flops
$\mathbf{N}_1 = \mathbf{U}^{(k)}\Gamma_1 \in \mathbb{R}^{d_1 \times r_k}$.    ▷ $d_1 r_k^2$ flops
$\mathbf{N}_2 = \Gamma_2 \mathbf{V}^{(k)} \in \mathbb{R}^{r_k \times r_k}$.    ▷ $d_2 r_k^2$ flops
$\mathbf{N}_3 = \mathbf{U}^{(k)*}\Gamma_3 \in \mathbb{R}^{r_k \times r_k}$.    ▷ $d_1 r_k^2$ flops
$\mathbf{N}_4 = \mathbf{N}_1 - \mathbf{U}^{(k)}\mathbf{N}_2 + \Gamma_3 - \mathbf{U}^{(k)}\mathbf{N}_3 \in \mathbb{R}^{d_1 \times r_k}$.    ▷ $d_1 r_k^2$ flops
Set $\mathbf{y}$ such that $\mathbf{y}_\ell = P_\Omega(\mathbf{N}_4 \mathbf{V}^{(k)*})_\ell = \sum_{k=1}^r (\mathbf{N}_4)_{i_\ell,k} (\mathbf{V}^{(k)})_{j_\ell,k}$ for each $\ell \in [m]$.    ▷ $2mr_k$ flops
Set $\mathbf{y}$ such that $\mathbf{y}_\ell = \mathbf{y}_\ell + P_\Omega(\mathbf{U}^{(k)}\Gamma_2)_\ell = \sum_{k=1}^r (\mathbf{U}^{(k)})_{i_\ell,k} (\Gamma_2^*)_{j_\ell,k}$ for each $\ell \in [m]$.    ▷ $2mr_k$ flops
**Output:** $\mathbf{y} \in \mathbb{R}^m$.
```

is well-conditioned (for example, with a condition number bounded by a small constant), the CG method can be used as an inexact solver of (14), returning very high precision approximate solutions *after a constant number* of iterations $N_{\text{CG\_inner}}$, thus, amounting to a time complexity of $O\left((mr + r^2 D) \cdot N_{\text{CG\_inner}}\right)$. We refer to Theorem 4.2 for a result that ensures this well-conditioning of the system matrix under certain conditions.

Since we have obtained the representation $\mathbf{X}^{(k+1)} = P_\Omega^*(\mathbf{r}_{k+1}) + P_{T_k}(\gamma_k)$ of $\mathbf{X}^{(k+1)}$ (which is approximate by nature if an iterative solver such as CG is used), we can now apply Algorithm 7 of (Huang et al., 2017) to $\gamma_k \in S_k$ to compute $\mathbf{M}_1^{(k+1)} \in \mathbb{R}^{d_2 \times r}$ and $\mathbf{M}_2^{(k+1)} \in \mathbb{R}\, d_1 \times r$ such that

$$\mathbf{X}^{(k+1)} = P_\Omega^*(\mathbf{r}_{k+1}) + \mathbf{U}^{(k)}\mathbf{M}_1^{(k+1)*} + \mathbf{M}_2^{(k+1)}\mathbf{V}^{(k)*},$$

by setting $\mathbf{M}_1^{(k+1)} = \mathbf{V}^{(k)*}\Gamma_1^* + \Gamma_2^* \in \mathbb{R}^{d_2 \times r}$ and $\mathbf{M}_2^{(k+1)} = \Gamma_3 \in \mathbb{R}^{d_1 \times r}$ if $\{\Gamma_1, \Gamma_2, \Gamma_3\}$ is the output of Algorithm 7 of (Huang et al., 2017). This last step has a time complexity of $8(d_1 + d_2)r^2 + r^2 d_2 + r d_2 = O(r^2 D)$ flops, so that we obtain a total time complexity of $O\left((mr + r^2 D) \cdot N_{\text{CG\_inner}}\right)$ for computing $\mathbf{X}^{(k+1)}$.

Including $\mathbf{r}_{k+1} \in \mathbb{R}^m$, $\mathbf{U}^{(k)} \in \mathbb{R}^{d_1 \times r}$ and $\mathbf{V}^{(k)} \in \mathbb{R}^{d_2 \times r}$ this amounts to a representation of $\mathbf{X}^{(k+1)}$ with a space complexity of $m + 2r(d_1 + d_2) = O(m + rD)$. Since also the space requirement of the intermediate variables does not exceed $O(m + rD)$, this finishes the proof of Theorem 3.1. □

Based on Theorem 3.1, we see that in `MatrixIRLS`, it is *never* necessary to work with full $(d_1 \times d_2)$-matrices. In order to update the smoothing parameter ϵ_{k+1} as in (6), we need the $\widetilde{r} + 1$-th singular value of $\mathbf{X}^{(k+1)}$. Furthermore, to update the information to define the weight operator $W^{(k+1)}$, we need to find the number r_{k+1} of singular values of $\mathbf{X}^{(k+1)}$ that are larger than ϵ_{k+1}, and their corresponding left and right singular vectors. Due to the definition of (6), it is clear that $r_{k+1} \geq \widetilde{r}$, and $r_{k+1} = \widetilde{r}$ for each iteration k when the smoothing parameter decreases, i.e., for each k with $\epsilon_{k+1} < \epsilon_k$.

In our experiments on exact completion of rank-r matrices with oracle knowledge of the rank r, we chose $\widetilde{r} = r$, and we observe that in most iterations of most of our experiments with `MatixIRLS`, it holds that $r_k = r$. In rare cases, we observe that $r_k = cr$ with a small constant $c > 1$ for a small number of the iterations k. On the other hand, we do not have a theoretical statement that bounds r_k in general.

The r_k singular values and vector pairs of $\mathbf{X}^{(k+1)}$ can be computed efficiently using any suitable method that uses matrix-matrix or matrix-vector products, as, for example, matrix-vector products with $\mathbf{X}^{(k+1)}$ can be calculated in $m + 2r_k(d_2 + d_1) = O(m + r_k D)$ due to its “sparse + low-rank” structure. Such a suitable method can be a (randomized) block Lanczos method (Golub & Underwood, 1977; Musco & Musco, 2015; Yuan et al., 2018). In our implementation we used a version of the method described in (Musco & Musco, 2015), which allows us to compute a good approximation of the needed singular triplets in a time complexity of $O(mr_k + r_k^2 D)$ (Yuan et al., 2018).

Remark A.1. *The authors of the recent preprint (Luo et al., 2020) propose an iterative method for rank-constrained least squares that has certain similarities to ours from a computational point of view. In particular, it can be shown that Algorithm 1 (called `RISRO`) of (Luo et al., 2020) is equivalent to solving the equation*

$$\left(P_{T_k}^* P_\Omega^* P_\Omega P_{T_k}\right) \mathbf{z}^{(k)} = P_{T_k}^* P_\Omega^*(\mathbf{y})$$

for $\mathbf{z}^{(\mathbf{k})} \in S_k$ in our notation, if specialized to matrix completion (see Theorem 2 and equation (58) of (Luo et al., 2020)), if *the parameter r_k coincides with the rank parameter of the rank-constraint least squares. Comparing this to (14), it can be observed that the Riemannian Gauss-Newton step of (Luo et al., 2020) corresponds to choosing the weights such that the entries of $\mathbf{H}^{(k)}$, $\mathbf{H}^{(k)}_{1,2}$ and $\mathbf{H}^{(k)}_{2,1}$ are all chosen equal and the smoothing parameter is chosen such that $\epsilon_k = 0$, rendering $\epsilon_k^2 \left(\mathbf{D}_{S_k}^{-1} - \epsilon_k^2 \mathbf{I}_{S_k}\right)^{-1} = 0$ in (14).*

On the other hand, the interpretation of `MatrixIRLS` and `RISRO` is quite different, as `MatrixIRLS` can be interpreted as a majorize-minimize method for smoothed log-det objectives eq. (3) with updated smoothing, whereas `RISRO` is harder to be interpreted with respect to a rank surrogate objective function, but rather follows a rank-constrained least squares framework.

B. Proof of Theorem 4.1

In this section, we prove under a random sampling model on the location of the provided entries, `MatrixIRLS` converges locally to a low-rank completion of the data with high probability, and that the convergence rate is quadratic, as described in Theorem 4.1.

First, we shortly elaborate on our notion of *incoherence* (see Definition 4.1), which quantifies the alignment of the standard basis $(e_i e_j^*)_{i=1,j=1}^{d_1,d_2}$ of $\mathbb{R}^{d_1 \times d_2}$ with the *tangent space* onto the rank-r manifold at rank-r matrix at a specific rank-r matrix, that we use in Theorem 4.1.

Remark B.1. *We note that the assumption that a rank-r matrix $\mathbf{X} \in \mathbb{R}^{d_1 \times d_2}$ is μ_0-incoherent according to Definition 4.1 is* weaker *than similar assumptions described in Definition 1.2, A0 and A1 of (Candès & Recht, 2009) and Definition 1 and Theorem 2 of (Recht, 2011), and even than the assumption (2) of (Chen, 2015), which is the weakest available incoherence condition in the literature that is used for showing successful completion by nuclear norm minimization. More precisely, (Chen, 2015) calls a matrix $\mathbf{X}$ μ_0-incoherent if*

$$\max_{1 \leq i \leq d_1} \|\mathbf{U}^* e_i\|_2 \leq \sqrt{\frac{\mu_0 r}{d_1}} \quad \text{and} \quad \max_{1 \leq j \leq d_2} \|\mathbf{V}^* e_j\|_F \leq \sqrt{\frac{\mu_0 r}{d_2}}. \tag{20}$$

In fact, condition (20) is stronger *than (7). If $\mathbf{U} \in \mathbb{R}^{d_1 \times r}$ and $\mathbf{V} \in \mathbb{R}^{d_2 \times r}$ are the left and right singular matrices corresponding to the r non-zero singular values of $\mathbf{X}$, we can write the projection operator $\mathcal{P}_T : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}^{d_1 \times d_2}$ that projects onto the tangent space T such $\mathcal{P}_T(\mathbf{Z}) = \mathbf{U}\mathbf{U}^*\mathbf{Z} + \mathbf{Z}\mathbf{V}\mathbf{V}^* - \mathbf{U}\mathbf{U}^*\mathbf{Z}\mathbf{V}\mathbf{V}^*$. Therefore, it can be seen that*

$$\begin{aligned}
\|\mathcal{P}_T(e_i e_j^*)\|_F^2 &= \|\mathbf{U}\mathbf{U}^* e_i e_j^* + e_i e_j^* \mathbf{V}\mathbf{V}^* - \mathbf{U}\mathbf{U}^* e_i e_j^* \mathbf{V}\mathbf{V}^*\|_F^2 = \|\mathbf{U}\mathbf{U}^* e_i e_j^* (\mathbf{I} - \mathbf{V}\mathbf{V}^*) + e_i e_j^* \mathbf{V}\mathbf{V}^*\|_F^2 \\
&= \|\mathbf{U}\mathbf{U}^* e_i e_j^* (\mathbf{I} - \mathbf{V}\mathbf{V}^*)\|_F^2 + \|e_i e_j^* \mathbf{V}\mathbf{V}^*\|_F^2 \leq \|\mathbf{U}\mathbf{U}^* e_i e_j^*\|_F^2 \|\mathbf{I} - \mathbf{V}\mathbf{V}^*\|^2 + \|e_i e_j^* \mathbf{V}\mathbf{V}^*\|_F^2 \\
&\leq \|\mathbf{U}^* e_i e_j^*\|_F^2 + \|e_i e_j^* \mathbf{V}\|_F^2 = \|\mathbf{U}^* e_i\|_2^2 + \|\mathbf{V}^* e_j\|_2^2 \leq \frac{\mu_0 r}{d_1} + \frac{\mu_0 r}{d_2} \leq \frac{\mu_0 r (d_1 + d_2)}{d_1 d_2}
\end{aligned}$$

for any $i \in [d_1]$, $j \in [d_2]$, if (20) is fulfilled, which holds since

$$\|\mathbf{U}^* e_i e_j^*\|_F^2 = \mathrm{tr}(e_j e_i^* \mathbf{U}\mathbf{U}^* e_i e_j^*) = \mathrm{tr}(e_i^* \mathbf{U}\mathbf{U}^* e_i) = e_i^* \mathbf{U}\mathbf{U}^* e_i = \|\mathbf{U}^* e_i\|_2^2$$

and similarly $\|e_i e_j^* \mathbf{V}\|_F^2 = \|\mathbf{V}^* e_j\|_2^2$.

B.1. Interplay between sampling operator and tangent space

In the statement of Theorem 4.1, we assume that the index set Ω is *drawn uniformly at random without replacement*. In our proof below, however, we use a sampling model on the locations $\Omega = (i_\ell, j_\ell)_{\ell=1}^m$ corresponding to *independent sampling with replacement*. It is well-known (see, e.g., Proposition 3 of (Recht, 2011)) that the statement then carries over to the above model of sampling without replacement.

As a preparation for our proof, we recall well-known result from (Recht, 2011) that bounds the number of repetitions of each location in Ω under the random sampling model with replacement.

Lemma B.1 (Proposition 5 of (Recht, 2011)). *Let $D = \max(d_1, d_2)$ and $\beta > 1$, let $\Omega = (i_\ell, j_\ell)_{\ell=1}^m$ be a multiset of double indices from $[d_1] \times [d_2]$ fulfilling $m < d_1 d_2$ that are sampled independently with replacement. Then with probability at least $1 - D^{2-2\beta}$, the maximal number of repetitions of any entry in Ω is less than $\frac{8}{3}\beta \log(D)$ for $D \geq 9$ and $\beta > 1$. Consequently,* *we have that with probability of at least* $1 - D^{2-2\beta}$, *the operator* $\mathcal{R}_\Omega : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}^{d_1 \times d_2}$ *defined such that*

$$\mathcal{R}_\Omega(\mathbf{X}) := P_\Omega^*(P_\Omega(\mathbf{X})) = \sum_{\ell=1}^{m} \langle e_{i_\ell} e_{j_\ell}^*, \mathbf{X} \rangle e_{i_\ell} e_{j_\ell}^* \tag{21}$$

fulfills

$$\|\mathcal{R}_\Omega\|_{S_\infty} \leq \frac{8}{3}\beta \log(D).$$

Next, we use a lemma of (Recht, 2011) that can be seen as a result of a *local restricted isometry property*. While the proof is fairly standard, we provide it for completeness since we use the weaker incoherence definition of Definition 4.1 instead of the incoherence notions of (Recht, 2011; Chen, 2015).

Lemma B.2 (Theorem 6 of (Recht, 2011))**.** *Let* $0 < \epsilon \leq \frac{1}{2}$, *let* $\mathbf{X}^0 \in \mathbb{R}^{d_1 \times d_2}$ *be a* μ_0*-incoherent matrix whose tangent space* $T_0 = T_{\mathbf{X}^0}$ *onto the rank-*r *manifold* $T_0 = T_{\mathbf{X}^0}\mathcal{M}_r$ *(see (11)) fulfills (7) and* $\mathcal{R}_\Omega : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}^{d_1 \times d_2}$ *be defined as in (21) from* m *independent uniformly sampled locations. Let* $\mathcal{P}_{T_0} : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}^{d_1 \times d_2}$ *be the projection operator associated to* T_0. *Then*

$$\left\| \frac{d_1 d_2}{m} \mathcal{P}_{T_0} \mathcal{R}_\Omega \mathcal{P}_{T_0} - \mathcal{P}_{T_0} \right\|_{S_\infty} \leq \varepsilon \tag{22}$$

holds with probability at least $1 - (d_1 + d_2)^{-2}$ *provided that*

$$m \geq \frac{7}{\varepsilon^2} \mu_0 r (d_1 + d_2) \log(d_1 + d_2). \tag{23}$$

Proof of Lemma B.2. First we define the family of operators $\mathcal{Z}_\ell, \widetilde{\mathcal{Z}}_\ell : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}^{d_1 \times d_2}$ such that for $\mathbf{X} \in \mathbb{R}^{d_1 \times d_2}$,

$$\mathcal{Z}_\ell(\mathbf{X}) := \frac{d_1 d_2}{m} \langle e_{i_\ell} e_{j_\ell}^*, \mathcal{P}_{T_0}(\mathbf{X}) \rangle \mathcal{P}_{T_0}(e_{i_\ell} e_{j_\ell}^*) - \frac{1}{m}\mathcal{P}_{T_0}(\mathbf{X}) := \frac{d_1 d_2}{m} \widetilde{\mathcal{Z}}_\ell(\mathbf{X}) - \frac{1}{m}\mathcal{P}_{T_0}(\mathbf{X})$$

for any $\ell \in [m]$. Then

$$\mathbb{E}[\mathcal{Z}_\ell] = \frac{1}{d_1 d_2} \sum_{i=1}^{d_1} \sum_{j=1}^{d_2} \frac{d_1 d_2}{m} \langle e_i e_j^*, \mathcal{P}_{T_0}(\cdot) \rangle \mathcal{P}_{T_0}(e_i e_j^*) - \frac{1}{m}\mathcal{P}_{T_0} = \frac{1}{d_1 d_2} \frac{d_1 d_2}{m} \mathcal{P}_{T_0} \mathbf{I} \mathcal{P}_{T_0} - \frac{1}{m}\mathcal{P}_{T_0} = 0. \tag{24}$$

Since for $\mathbf{X} \in \mathbb{R}^{d_1 \times d_2}$

$$\langle e_{i_\ell} e_{j_\ell}^*, \mathcal{P}_{T_0}(\mathbf{X}) \rangle \mathcal{P}_{T_0}(e_{i_\ell} e_{j_\ell}^*) = \langle \mathcal{P}_{T_0}(e_{i_\ell} e_{j_\ell}^*), \mathbf{X} \rangle \mathcal{P}_{T_0}(e_{i_\ell} e_{j_\ell}^*),$$

we obtain

$$\|\langle e_{i_\ell} e_{j_\ell}^*, \mathcal{P}_{T_0}(\mathbf{X}) \rangle \mathcal{P}_{T_0}(e_{i_\ell} e_{j_\ell}^*)\|_F \leq \left| \langle \mathcal{P}_{T_0}(e_{i_\ell} e_{j_\ell}^*), \mathbf{X} \rangle \right| \|\mathcal{P}_{T_0}(e_{i_\ell} e_{j_\ell}^*)\|_F \leq \|\mathcal{P}_{T_0}(e_{i_\ell} e_{j_\ell}^*)\|_F^2 \|X\|_F$$

by Cauchy-Schwartz, and thus the norm bound

$$\begin{aligned} \frac{d_1 d_2}{m} \left\| \widetilde{\mathcal{Z}}_\ell \right\|_{S_\infty} &\leq \frac{d_1 d_2}{m} \|\mathcal{P}_{T_0}(e_{i_\ell} e_{j_\ell}^*)\|_F^2 \leq \frac{d_1 d_2}{m} \max_{i \in [d_1], j \in [d_2]} \|\mathcal{P}_{T_0}(e_i e_j^*)\|_F^2 \\ &\leq \frac{d_1 d_2}{m} \frac{\mu_0 r (d_1 + d_2)}{d_1 d_2} = \frac{\mu_0 r (d_1 + d_2)}{m} \end{aligned} \tag{25}$$

using the incoherence assumption (7) in the last inequality. Similarly,

$$\left\| \frac{1}{m} \mathcal{P}_{T_0} \right\|_{S_\infty} = \left\| \frac{1}{m} \mathcal{P}_{T_0} \mathbf{I} \mathcal{P}_{T_0} \right\|_{S_\infty} \leq \frac{1}{m} \sum_{i=1}^{d_1} \sum_{j=1}^{d_2} \left\| \langle \mathcal{P}_{T_0}(e_i e_j^*), (\cdot) \rangle \mathcal{P}_{T_0}(e_i e_j^*) \right\|_{S_\infty} \leq \frac{\mu_0 r (d_1 + d_2)}{m}. \tag{26}$$

We note that if operators $\mathcal{A}$ and $\mathcal{B}$ are positive semidefinite, then $\|\mathcal{A} - \mathcal{B}\|_{S_\infty} \leq \max(\|\mathcal{A}\|_{S_\infty}, \|\mathcal{B}\|_{S_\infty})$, and as both $\widetilde{\mathcal{Z}}_\ell$ and $\mathcal{P}_{T_0}$ are positive semidefinite,

$$\|\mathcal{Z}_\ell\|_{S_\infty} \leq \max\left(\frac{d_1 d_2}{m} \left\| \widetilde{\mathcal{Z}}_\ell \right\|_{S_\infty}, \frac{1}{m} \|\mathcal{P}_{T_0}\|_{S_\infty} \right) = \frac{\mu_0 r (d_1 + d_2)}{m}$$

for all $\ell \in [m]$. For the expectation of the squares of $\mathcal{Z}_\ell$, we obtain

$$\begin{aligned}\mathbb{E}\,\mathcal{Z}_\ell\mathcal{Z}_\ell^* &= \frac{(d_1d_2)^2}{m^2}\,\mathbb{E}\left[(\widetilde{\mathcal{Z}}_\ell)^*\widetilde{\mathcal{Z}}_\ell\right] - \frac{d_1d_2}{m^2}\,\mathbb{E}\left[\widetilde{\mathcal{Z}}_\ell\right]\mathcal{P}_{T_0} - \frac{d_1d_2}{m^2}\mathcal{P}_{T_0}\,\mathbb{E}\left[\widetilde{\mathcal{Z}}_\ell\right] + \frac{1}{m^2}\mathcal{P}_{T_0} \\ &= \frac{(d_1d_2)^2}{m^2}\,\mathbb{E}\left[(\widetilde{\mathcal{Z}}_\ell)^*\widetilde{\mathcal{Z}}_\ell\right] + (1-2)\frac{1}{m^2}\mathcal{P}_{T_0},\end{aligned}$$

as $\mathcal{P}_{T_0}^2 = \mathcal{P}_{T_0}$ and $\mathbb{E}[\widetilde{\mathcal{Z}}_\ell] = \frac{1}{d_1d_2}\mathcal{P}_{T_0}$. Thus,

$$\begin{aligned}\left\|\sum_{\ell=1}^m \mathbb{E}\,\mathcal{Z}_\ell\mathcal{Z}_\ell^*\right\|_{S_\infty} &\leq \sum_{\ell=1}^m \|\mathbb{E}\,\mathcal{Z}_\ell\mathcal{Z}_\ell^*\|_{S_\infty} = \sum_{\ell=1}^m \left\|\frac{(d_1d_2)^2}{m^2}\,\mathbb{E}\left[(\widetilde{\mathcal{Z}}_\ell)^2\right] - \frac{1}{m^2}\mathcal{P}_{T_0}\right\|_{S_\infty} \\ &\leq \sum_{\ell=1}^m \max\left(\frac{(d_1d_2)^2}{m^2}\left\|\mathbb{E}\left[(\widetilde{\mathcal{Z}}_\ell)^2\right]\right\|_{S_\infty}, \frac{1}{m^2}\|\mathcal{P}_{T_0}\|_{S_\infty}\right) \\ &\leq \sum_{\ell=1}^m \max\left(\frac{(d_1d_2)^2}{m^2}\left\|\mathbb{E}\left[\|\mathcal{P}_{T_0}(e_{i_\ell}e_{j_\ell}^*)\|_F^2\,\widetilde{\mathcal{Z}}_\ell\right]\right\|_{S_\infty}, \frac{1}{m^2}\right) \\ &\leq \sum_{\ell=1}^m \max\left(\frac{(d_1d_2)(d_1+d_2)\mu_0 r}{m^2}\left\|\mathbb{E}\,\widetilde{\mathcal{Z}}_\ell\right\|_{S_\infty}, \frac{1}{m^2}\right) \\ &\leq \sum_{\ell=1}^m \max\left(\frac{(d_1+d_2)\mu_0 r}{m^2}, \frac{1}{m^2}\right) = \frac{\mu_0 r(d_1+d_2)}{m},\end{aligned}$$

where we used that $\|\mathcal{P}_{T_0}\|_2 \leq 1$ since $\mathcal{P}_{T_0}$ is a projection in the third inequality, the definition of μ_0 in the fourth and the fact that $\mathbb{E}\,\widetilde{\mathcal{Z}}_\ell = \frac{1}{d_1d_2}\mathcal{P}_{T_0}$ (see (24)) in the fifth. As the $\mathcal{Z}_\ell$ are Hermitian, it follows by the matrix Bernstein inequality (see, e.g., Theorem 5.4.1 of (Vershynin, 2018)) that

$$\begin{aligned}&\mathbb{P}\left(\left\|\frac{d_1d_2}{m}\mathcal{P}_{T_0}\mathcal{R}_\Omega\mathcal{P}_{T_0} - \mathcal{P}_{T_0}\right\|_{S_\infty} \geq \varepsilon\right) \leq (d_1+d_2)\exp\left(-\frac{m\varepsilon^2/2}{\mu_0 r(d_1+d_2) + \mu_0 r(d_1+d_2)\epsilon/3}\right) \\ &\leq (d_1+d_2)\exp\left(-\frac{m\varepsilon^2}{2\mu_0 r(d_1+d_2) + \mu_0 r(d_1+d_2)/3}\right),\end{aligned} \tag{27}$$

using that $\varepsilon \leq \frac{1}{2}$ in the last inequality.

Furthermore, if (23) is fulfilled, then

$$(d_1+d_2)\exp\left(-\frac{m\varepsilon^2}{\frac{7}{3}\mu_0 r(d_1+d_2)}\right) \leq (d_1+d_2)^{-2},$$

which shows that (22) holds with a probability of at least $1-(d_1+d_2)^{-2}$. □

To prove our theorem, we will use the local restricted isometry statement of (22) for tangent spaces $T_{\mathbf{X}}$ corresponding to matrices $\mathbf{X} \in \mathbb{R}^{d_1\times d_2}$ *that are close* to $\mathbf{X}^0$. We show the following auxiliary result, which is a refinement of Lemma 4.2 (Wei et al., 2020) as we obtain a bound in the S_∞-norm in (c) instead of in the Frobenius norm.

Lemma B.3. *Let $\mathbf{X}^0, \mathbf{X} \in \mathbb{R}^{d_1\times d_2}$ be matrices and assume that $0 < \varepsilon < 1$ and that the following three conditions hold:*

(a) For $\mathcal{R}_\Omega : \mathbb{R}^{d_1\times d_2} \to \mathbb{R}^{d_1\times d_2}$ as in (21),

$$\|\mathcal{R}_\Omega\|_{S_\infty} \leq \frac{16}{3}\log(D).$$

(b) The tangent space $T_0 = T_{\mathbf{X}^0}$ onto the rank-r manifold $\mathcal{M}_r$ at $\mathbf{X}^0$ fulfills

$$\left\|\frac{d_1d_2}{m}\mathcal{P}_{T_0}\mathcal{R}_\Omega\mathcal{P}_{T_0} - \mathcal{P}_{T_0}\right\|_{S_\infty} \leq \varepsilon.$$

(c) The spectral norm distance between $\mathbf{X}$ *and* $\mathbf{X}^0$ *fulfills*

$$\|\mathbf{X} - \mathbf{X}^0\|_{S_\infty} \leq \frac{\sqrt{3}}{32\sqrt{\log(D)}\sqrt{(1+\varepsilon)}}\varepsilon\sqrt{\frac{m}{d_1 d_2}}\sigma_r(\mathbf{X}^0).$$

Then the tangent space $T = T_{\mathbf{X}}$ *onto the rank-*r *manifold at* $\mathbf{X}$ *fulfills*

$$\left\| \frac{d_1 d_2}{m}\mathcal{P}_T \mathcal{R}_\Omega \mathcal{P}_T - \mathcal{P}_T \right\|_{S_\infty} \leq 4\varepsilon. \tag{28}$$

Proof. For any $\mathbf{Z} \in \mathbb{R}^{d_1 \times d_2}$, we have

$$\begin{aligned}
\|\mathcal{R}_\Omega \mathcal{P}_{T_0}(\mathbf{Z})\|_F^2 &= \langle \mathcal{R}_\Omega \mathcal{P}_T(\mathbf{Z}), \mathcal{R}_\Omega \mathcal{P}_T(\mathbf{Z})\rangle \leq \frac{16}{3}\log(D)\,\langle \mathcal{P}_{T_0}(\mathbf{Z}), \mathcal{R}_\Omega \mathcal{P}_{T_0}(\mathbf{Z})\rangle \\
&= \frac{16}{3}\log(D)\,\langle \mathcal{P}_{T_0}(\mathbf{Z}), \mathcal{P}_{T_0}\mathcal{R}_\Omega \mathcal{P}_{T_0}(\mathbf{Z})\rangle \\
&= \frac{16}{3}\log(D)\left(\left\langle \mathcal{P}_{T_0}(\mathbf{Z}), \frac{m}{d_1 d_2}\mathcal{P}_{T_0}(\mathbf{Z})\right\rangle + \left\langle \mathcal{P}_{T_0}(\mathbf{Z}), \left(\mathcal{P}_{T_0}\mathcal{R}_\Omega \mathcal{P}_{T_0}(\mathbf{Z}) - \frac{m}{d_1 d_2}\mathcal{P}_{T_0}(\mathbf{Z})\right)\right\rangle\right) \\
&\leq \frac{16}{3}\log(D)\left(\frac{m}{d_1 d_2} + \varepsilon\frac{m}{d_1 d_2}\right)\|\mathcal{P}_{T_0}(\mathbf{Z})\|_F^2 \leq \frac{16}{3}\log(D)(1+\varepsilon)\frac{m}{d_1 d_2}\,\|\mathbf{Z}\|_F^2\,,
\end{aligned}$$

where the first inequality follows from condition (a) and the second one from condition (b). It follows that

$$\|\mathcal{R}_\Omega \mathcal{P}_{T_0}\| \leq \sqrt{\frac{16}{3}\log(D)(1+\varepsilon)\frac{m}{d_1 d_2}}. \tag{29}$$

Furthermore, if $\mathbf{U}, \mathbf{U}_0 \in \mathbb{R}^{d_1 \times r}$ and $\mathbf{V}, \mathbf{V}_0 \in \mathbb{R}^{d_2 \times r}$ are the matrices of first r left and right singular vectors of $\mathbf{X}$ and $\mathbf{X}^0$, respectively, it holds that for any $\mathbf{Z} \in \mathbb{R}^{d_1 \times d_2}$,

$$\begin{aligned}
(\mathcal{P}_T - \mathcal{P}_{T_0})(\mathbf{Z}) &= \mathbf{U}\mathbf{U}^*\mathbf{Z} + \mathbf{Z}\mathbf{V}\mathbf{V}^* - \mathbf{U}\mathbf{U}^*\mathbf{Z}\mathbf{V}\mathbf{V}^* - \mathbf{U}_0\mathbf{U}_0^*\mathbf{Z} - \mathbf{Z}\mathbf{V}_0\mathbf{V}_0^* + \mathbf{U}_0\mathbf{U}_0^*\mathbf{Z}\mathbf{V}_0\mathbf{V}_0^* \\
&= (\mathbf{U}\mathbf{U}^* - \mathbf{U}_0\mathbf{U}_0^*)\,\mathbf{Z}(\mathbf{I} - \mathbf{V}_0\mathbf{V}_0^*) + (\mathbf{I} - \mathbf{U}\mathbf{U}^*)\mathbf{Z}(\mathbf{V}\mathbf{V}^* - \mathbf{V}_0\mathbf{V}_0^*),
\end{aligned}$$

which we use to estimate

$$\begin{aligned}
\|(\mathcal{P}_T - \mathcal{P}_{T_0})(\mathbf{Z})\|_F &\leq \|\mathbf{U}\mathbf{U}^* - \mathbf{U}_0\mathbf{U}_0^*\|_{S_\infty}\|\mathbf{Z}\|_F\|\mathbf{I} - \mathbf{V}_0\mathbf{V}_0^*\|_{S_\infty} + \|\mathbf{I} - \mathbf{U}\mathbf{U}^*\|_{S_\infty}\|\mathbf{Z}\|_F\|\mathbf{V}\mathbf{V}^* - \mathbf{V}_0\mathbf{V}_0^*\|_{S_\infty} \\
&\leq \frac{\|\mathcal{T}_r(\mathbf{X}) - \mathbf{X}^0\|_{S_\infty}}{\sigma_r(\mathbf{X}^0)}\|\mathbf{Z}\|_F \cdot 1 + 1 \cdot \|\mathbf{Z}\|_F\frac{\|\mathcal{T}_r(\mathbf{X}) - \mathbf{X}^0\|_{S_\infty}}{\sigma_r(\mathbf{X}^0)} \\
&\leq 2\frac{\|\mathcal{T}_r(\mathbf{X}) - \mathbf{X}\|_{S_\infty} + \|\mathbf{X} - \mathbf{X}^0\|_{S_\infty}}{\sigma_r(\mathbf{X}^0)}\|\mathbf{Z}\|_F,
\end{aligned}$$

where $\mathcal{T}_r(\mathbf{X})$ is the best rank-r approximation (10). Here, we used the results

$$\|\mathbf{U}\mathbf{U}^* - \mathbf{U}_0\mathbf{U}_0^*\|_{S_\infty} \leq \frac{\|\mathcal{T}_r(\mathbf{X}) - \mathbf{X}^0\|_{S_\infty}}{\sigma_r(\mathbf{X}^0)}$$

and

$$\|\mathbf{V}\mathbf{V}^* - \mathbf{V}_0\mathbf{V}_0^*\|_{S_\infty} \leq \frac{\|\mathcal{T}_r(\mathbf{X}) - \mathbf{X}^0\|_{S_\infty}}{\sigma_r(\mathbf{X}^0)}$$

of Lemma 4.2, ineq. (4.3) of (Wei et al., 2016), which bound the distance between the projections onto the left and right singular subspaces of $\mathbf{X}$ and $\mathbf{X}^0$.

From the Eckardt-Young-Mirsky theorem (10), it then follows that

$$\|(\mathcal{P}_T - \mathcal{P}_{T_0})\|_{S_\infty} \leq \frac{4\|\mathbf{X} - \mathbf{X}^0\|_{S_\infty}}{\sigma_r(\mathbf{X}^0)}. \tag{30}$$

With this, we further bound

$$
\begin{aligned}
\|\mathcal{R}_\Omega \mathcal{P}_T\|_{S_\infty} &\leq \|\mathcal{R}_\Omega(\mathcal{P}_T - \mathcal{P}_{T_0})\|_{S_\infty} + \|\mathcal{R}_\Omega \mathcal{P}_{T_0}\|_{S_\infty} \\
&\leq \frac{16}{3}\log(D)\frac{4\left\|\mathbf{X}-\mathbf{X}^0\right\|_{S_\infty}}{\sigma_r(\mathbf{X}^0)} + \|\mathcal{R}_\Omega \mathcal{P}_{T_0}\|_{S_\infty} \\
&\leq \frac{16}{3}\log(D)\frac{\sqrt{3}}{8\sqrt{\log(D)}\sqrt{(1+\varepsilon)}}\varepsilon\sqrt{\frac{m}{d_1 d_2}} + \sqrt{\frac{16}{3}\log(D)(1+\varepsilon)\frac{m}{d_1 d_2}} \\
&= \frac{2}{\sqrt{3}}\sqrt{\log(D)}\frac{1}{\sqrt{(1+\varepsilon)}}\varepsilon\sqrt{\frac{m}{d_1 d_2}} + \sqrt{\frac{16}{3}\log(D)(1+\varepsilon)\frac{m}{d_1 d_2}} \\
&\leq 2\sqrt{3}\sqrt{\log(D)}\sqrt{1+\varepsilon}\sqrt{\frac{m}{d_1 d_2}},
\end{aligned}
\tag{31}
$$

where the second inequality follows from (30) and the third from condition (c). To prove the statement (28), we calculate

$$
\begin{aligned}
\left\|\frac{d_1 d_2}{m}\mathcal{P}_T \mathcal{R}_\Omega \mathcal{P}_T - \mathcal{P}_T\right\|_{S_\infty} &\leq \|\mathcal{P}_T - \mathcal{P}_{T_0}\|_{S_\infty} + \frac{d_1 d_2}{m}\|\mathcal{P}_T \mathcal{R}_\Omega \mathcal{P}_T - \mathcal{P}_T \mathcal{R}_\Omega \mathcal{P}_{T_0}\|_{S_\infty} \\
&\quad + \frac{d_1 d_2}{m}\|\mathcal{P}_T \mathcal{R}_\Omega \mathcal{P}_{T_0} - \mathcal{P}_{T_0} \mathcal{R}_\Omega \mathcal{P}_{T_0}\|_{S_\infty} + \left\|\mathcal{P}_{T_0} - \frac{d_1 d_2}{m}\mathcal{P}_{T_0}\mathcal{R}_\Omega \mathcal{P}_{T_0}\right\|_{S_\infty} \\
&\leq \|\mathcal{P}_T - \mathcal{P}_{T_0}\|_{S_\infty} + \frac{d_1 d_2}{m}\|\mathcal{R}_\Omega \mathcal{P}_T\|_{S_\infty}\|\mathcal{P}_T - \mathcal{P}_{T_0}\|_{S_\infty} \\
&\quad + \frac{d_1 d_2}{m}\|\mathcal{R}_\Omega \mathcal{P}_{T_0}\|_{S_\infty}\|\mathcal{P}_T - \mathcal{P}_{T_0}\|_{S_\infty} + \left\|\mathcal{P}_{T_0} - \frac{d_1 d_2}{m}\mathcal{P}_{T_0}\mathcal{R}_\Omega \mathcal{P}_{T_0}\right\|_{S_\infty} \\
&\leq \frac{4\left\|\mathbf{X}-\mathbf{X}^0\right\|_{S_\infty}}{\sigma_r(\mathbf{X}^0)} + \frac{d_1 d_2}{m}\|\mathcal{R}_\Omega \mathcal{P}_T\|_{S_\infty}\frac{4\left\|\mathbf{X}-\mathbf{X}^0\right\|_{S_\infty}}{\sigma_r(\mathbf{X}^0)} \\
&\quad + \frac{d_1 d_2}{m}\|\mathcal{R}_\Omega \mathcal{P}_{T_0}\|_{S_\infty}\frac{4\left\|\mathbf{X}-\mathbf{X}^0\right\|_{S_\infty}}{\sigma_r(\mathbf{X}^0)} + \left\|\mathcal{P}_{T_0} - \frac{d_1 d_2}{m}\mathcal{P}_{T_0}\mathcal{R}_\Omega \mathcal{P}_{T_0}\right\|_{S_\infty} \\
&\leq 4\varepsilon
\end{aligned}
$$

where in the second inequality, we utilized the fact $\mathcal{R}_\Omega^* = \mathcal{R}_\Omega$ so that $\|\mathcal{P}_T \mathcal{R}_\Omega\|_{S_\infty} = \|\mathcal{R}_\Omega \mathcal{P}_T\|_{S_\infty}$. The very last estimate follows from conditions (b) and (c) and the bounds (29) and (31) for $\|\mathcal{R}_\Omega \mathcal{P}_T\|_{S_\infty}$ and $\|\mathcal{R}_\Omega \mathcal{P}_{T_0}\|_{S_\infty}$. □

In the following lemma, we combine the previous results to show that under our sampling model, with high probability, a local restricted isometry property holds with respect to tangent spaces T_k that are in some sense close to $\mathbf{X}^0$.

Lemma B.4. *Let $\mathbf{X}^0 \in \mathbb{R}^{d_1 \times d_2}$ be a matrix of rank r that is μ_0-incoherent, and let $\Omega = (i_\ell, j_\ell)_{\ell=1}^m$ be a random index set of cardinality $|\Omega| = m$ that is sampled uniformly without replacement, or, alternatively, sampled independently with replacement. There exists constants $C, \widetilde{C}, C_1$ such that if*

$$
m \geq C\mu_0 r(d_1 + d_2)\log(d_1 + d_2), \tag{32}
$$

then, with probability at least $1 - 2D^{-2}$, the following holds: For each matrix $\mathbf{X}^{(k)} \in \mathbb{R}^{d_1 \times d_2}$ fulfilling

$$
\|\mathbf{X}^{(k)} - \mathbf{X}^0\|_{S_\infty} \leq C_1\sqrt{\frac{\mu_0 r}{d}}\sigma_r(\mathbf{X}^0), \tag{33}
$$

it follows that the projection $\mathcal{P}_{T_k} : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}^{d_1 \times d_2}$ onto the tangent space $T_k := T_{\mathcal{T}_r(\mathbf{X}^{(k)})}\mathcal{M}_r$ satisfies

$$
\left\|\frac{d_1 d_2}{m}\mathcal{P}_{T_k} P_\Omega^* P_\Omega \mathcal{P}_{T_k} - \mathcal{P}_{T_k}\right\|_{S_\infty} \leq \frac{2}{5},
$$

and furthermore,

$$
\|\eta\|_F \leq \sqrt{\frac{\widetilde{C} d\log(D)}{\mu_0 r}}\|\mathcal{P}_{T_k^\perp}(\eta)\|_F
$$

for each matrix $\eta \in \ker P_\Omega$ in the null space of the subsampling operator $P_\Omega : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}^m$.

Proof of Lemma B.4. Assume that there are m locations $\Omega = (i_\ell, j_\ell)_{\ell=1}^m$ in $[d_1] \times [d_2]$ sampled independently uniformly *with replacement*, where m fulfills (32) with $C := 7/\varepsilon^2$ and $\varepsilon = 0.1$. By Lemma B.1, it follows that the corresponding operator $\mathcal{R}_\Omega : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}^{d_1 \times d_2}$ from (21) fulfills

$$\|\mathcal{R}_\Omega\|_{S_\infty} \leq \frac{16}{3}\log(D) \tag{34}$$

on an event called E_Ω, which occurs with a probability of at least $1 - D^{-2}$, and by Lemma B.2, the tangent space $T_0 = T_{\mathbf{X}^0}\mathcal{M}_r$ corresponding to the μ_0-incoherent rank-r matrix $\mathbf{X}^0$ fulfills

$$\left\| \frac{d_1 d_2}{m} \mathcal{P}_{T_0} P_\Omega^* P_\Omega \mathcal{P}_{T_0} - \mathcal{P}_{T_0} \right\|_{S_\infty} \leq \varepsilon$$

on an event called E_{Ω,T_0}, which occurs with a probability of at least $1 - D^{-2}$. Let $\tilde{\epsilon} = \frac{1}{10}$. If $\mathbf{X}^{(k)} \in \mathbb{R}^{d_1 \times d_2}$ is such that $\|\mathbf{X}^{(k)} - \mathbf{X}^0\|_{S_\infty} \leq \widetilde{\xi}\sigma_r(\mathbf{X}^0)$ with

$$\widetilde{\xi} = \frac{\sqrt{3}}{32} \frac{\epsilon}{\sqrt{\log(D)(1+\epsilon)}} \sqrt{\frac{m}{d_1 d_2}} = \frac{\sqrt{3}}{32} \frac{1}{10\sqrt{\log(D)(11/10)}} \sqrt{\frac{m}{d_1 d_2}}, \tag{35}$$

it follows by Lemma B.3 that on the event $E_\Omega \cap E_{\Omega,T_0}$, the tangent space $T_k := \mathbf{X}^{(k)}$ onto the rank-r manifold at $\mathbf{X}^{(k)}$ fulfills

$$\left\| \frac{d_1 d_2}{m} \mathcal{P}_{T_k} \mathcal{R}_\Omega \mathcal{P}_{T_k} - \mathcal{P}_{T_k} \right\|_{S_\infty} \leq 4\tilde{\epsilon} = \frac{2}{5}. \tag{36}$$

Next, we claim that on the event $E_\Omega \cap E_{\Omega,T_0}$,

$$\|\eta\|_F \leq \sqrt{\frac{\widetilde{C} d \log(D)}{\mu_0 r}} \|\mathcal{P}_{T_k^\perp}(\eta)\|_F. \tag{37}$$

for any for each matrix $\eta \in \ker P_\Omega$ in the null space of the subsampling operator $P_\Omega : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}^m$.

Indeed, to show this claim, we first note that $\eta \in \ker P_\Omega$ if and only if $\eta \in \ker \mathcal{R}_\Omega : P_\Omega^* P_\Omega$. Let $\eta \in \ker \mathcal{R}_\Omega$. Then

$$\begin{aligned}
\|\mathcal{P}_{T_k}(\eta)\|_F^2 &= \langle \mathcal{P}_{T_k}(\eta), \mathcal{P}_{T_k}(\eta) \rangle \\
&= \left\langle \mathcal{P}_{T_k}(\eta), \frac{d_1 d_2}{m} \mathcal{P}_{T_k} \mathcal{R}_\Omega \mathcal{P}_{T_k}(\eta) \right\rangle + \left\langle \mathcal{P}_{T_k}(\eta), \mathcal{P}_{T_k}(\eta) - \frac{d_1 d_2}{m} \mathcal{P}_{T_k} \mathcal{R}_\Omega \mathcal{P}_{T_k}(\eta) \right\rangle \\
&\leq \left\langle \mathcal{P}_{T_k}(\eta), \frac{d_1 d_2}{m} \mathcal{P}_{T_k} \mathcal{R}_\Omega \mathcal{P}_{T_k}(\eta) \right\rangle + \|\mathcal{P}_{T_k}(\eta)\|_F \left\| \mathcal{P}_{T_k} - \frac{d_1 d_2}{m} \mathcal{P}_{T_k} \mathcal{R}_\Omega \mathcal{P}_{T_k} \right\|_{S_\infty} \|\mathcal{P}_{T_k}(\eta)\|_F \\
&\leq \left\langle \mathcal{P}_{T_k}(\eta), \frac{d_1 d_2}{m} \mathcal{P}_{T_k} \mathcal{R}_\Omega \mathcal{P}_{T_k}(\eta) \right\rangle + 4\epsilon \|\mathcal{P}_{T_k}(\eta)\|_F^2,
\end{aligned}$$

using (36) in the last inequality, which implies that

$$\begin{aligned}
\|\mathcal{P}_{T_k}(\eta)\|_F^2 &\leq \frac{1}{1-4\epsilon} \frac{d_1 d_2}{m} \langle \mathcal{P}_{T_k}(\eta), \mathcal{P}_{T_k} \mathcal{R}_\Omega^2 \mathcal{P}_{T_k}(\eta) \rangle = \frac{1}{1-4\epsilon} \frac{d_1 d_2}{m} \|\mathcal{R}_\Omega \mathcal{P}_{T_k}(\eta)\|_F^2 \\
&\leq \frac{2 d_1 d_2}{m} \|\mathcal{R}_\Omega \mathcal{P}_{T_k}(\eta)\|_F^2
\end{aligned}$$

using the fact that $\mathcal{R}_\Omega : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}^{d_1 \times d_2}$ is positive semidefinite and has eigenvalues that are 0 or larger or equal than 1 only. Furthermore, we used that $\epsilon \leq \frac{1}{10}$ in the last inequality.

Since $\eta \in \ker \mathcal{R}_\Omega$, it holds that

$$0 = \|\mathcal{R}_\Omega(\eta)\|_F = \left\| \mathcal{R}_\Omega \left(\mathcal{P}_{T_k}(\eta) + \mathcal{P}_{T_k^\perp}(\eta) \right) \right\|_F \geq \|\mathcal{R}_\Omega \mathcal{P}_{T_k}(\eta)\|_F - \|\mathcal{R}_\Omega \mathcal{P}_{T_k^\perp}(\eta)\|_F$$

so that

$$\|\mathcal{R}_\Omega \mathcal{P}_{T_k}(\eta)\|_F \leq \|\mathcal{R}_\Omega \mathcal{P}_{T_k^\perp}(\eta)\|_F \leq \frac{16}{3} \log(D) \|\mathcal{P}_{T_k^\perp}(\eta)\|_F,$$

where we used (34) in the last inequality. Inserting this above, we obtain

$$\begin{aligned}\|\eta\|_F^2 = \|\mathcal{P}_{T_k}(\eta)\|_F^2 + \|\mathcal{P}_{T_k^\perp}(\eta)\|_F^2 &\le \left(\frac{2d_1d_2}{m}\frac{16^2}{3^2}\log(D)^2+1\right)\|\mathcal{P}_{T_k^\perp}(\eta)\|_F^2\\ &\le \left(\frac{2d_1d_2}{C\mu_0 r(d_1+d_2)\log(d_1+d_2)}\frac{16^2}{3^2}\log(D)^2+1\right)\|\mathcal{P}_{T_k^\perp}(\eta)\|_F^2\\ &\le \frac{\widetilde{C}d\log(D)}{\mu_0 r}\|\mathcal{P}_{T_k^\perp}(\eta)\|_F^2,\end{aligned}$$

where we used the sample complexity condition (32) in the second inequality and the definition

$$\widetilde{C} := \frac{4\cdot 16^2}{C\cdot 3^2}$$

for the constant $\widetilde{C}$.

Moreover, we observe that for $C_1 := \frac{\sqrt{C}}{320}\sqrt{\frac{30}{11}}$ where C is the constant of (32), it holds that

$$C_1\sqrt{\frac{\mu_0 r}{d}} \le \frac{\sqrt{3}}{32}\frac{1}{10\sqrt{\log(D)(11/10)}}\sqrt{\frac{C\mu_0 r(d_1+d_2)\log(d_1+d_2)}{d_1d_2}} \le \widetilde{\xi},$$

implying that the two statements of Lemma B.4 are satisfied on the event $E_\Omega \cap E_{\Omega,T_0}$ if (33) holds. By the above mentioned probability bounds and a union bound, $E_\Omega \cap E_{\Omega,T_0}$ occurs with a probability of at least $1-2D^{-2}$, finishing the proof for the sampling with replacement model. By the argument of Proposition 3 of (Recht, 2011), the result extends to the model of sampling locations drawn uniformly at random without replacement, with the same probability bound. This concludes the proof of Lemma B.4. □

The following lemma will also play a role in the proof of Theorem 4.1.

Lemma B.5. *Let $C, \widetilde{C}, C_1$ be the constants of Lemma B.4 and μ_0 be the incoherence factor of a rank-r matrix $\mathbf{X}^0$. If*

$$m \ge C\mu_0 r(d_1+d_2)\log(d_1+d_2)$$

and if $\eta^{(k)} = \mathbf{X}^{(k)} - \mathbf{X}^0$ fulfills

$$\|\eta^{(k)}\|_{S_\infty} \le \xi\sigma_r(\mathbf{X}^0),$$

with

$$\xi := \min\left(C_1\sqrt{\frac{\mu_0 r}{d}}, \frac{\mu_0}{4(1+6\kappa)d\log(D)\widetilde{C}}\right)$$

then, on the event of Lemma B.4, it holds that

$$\|\eta^{(k)}\|_{S_\infty} < \sqrt{\frac{4\widetilde{C}d(d-r)\log(D)}{\mu_0 r}}\sigma_{r+1}(\mathbf{X}^{(k)}). \tag{38}$$

Proof. First, we compute that

$$\begin{aligned}\|\mathcal{P}_{T_k^\perp}(\eta^{(k)})\|_F \le \|\mathcal{P}_{T_k^\perp}(\mathbf{X}^{(k)})\|_F + \|\mathcal{P}_{T_k^\perp}(\mathbf{X}^0)\|_F &\le \sqrt{\sum_{i=r+1}^{d}\sigma_i^2(\mathbf{X}^{(k)})} + \left\|\mathbf{U}_\perp^{(k)}\mathbf{U}_\perp^{(k)*}\mathbf{X}^0\mathbf{V}_\perp^{(k)}\mathbf{V}_\perp^{(k)*}\right\|_F\\ &\le \sqrt{d-r}\sigma_{r+1}(\mathbf{X}^{(k)}) + \|\mathbf{U}_\perp^{(k)*}\mathbf{U}_0\|_{S_\infty}\|\mathbf{\Sigma}_0\|_F\|\mathbf{V}_0^*\mathbf{V}_\perp^{(k)}\|_{S_\infty}\\ &\le \sqrt{d-r}\sigma_{r+1}(\mathbf{X}^{(k)}) + \frac{2\|\eta^{(k)}\|_{S_\infty}^2}{(1-\zeta)^2\sigma_r^2(\mathbf{X}^0)}\sqrt{r}\sigma_1(\mathbf{X}^0) = \sqrt{d-r}\sigma_{r+1}(\mathbf{X}^{(k)}) + \frac{2\|\eta^{(k)}\|_{S_\infty}^2}{(1-\zeta)^2\sigma_r(\mathbf{X}^0)}\sqrt{r}\kappa,\end{aligned}$$

where $0<\zeta<1$ such that $\|\mathbf{X}^{(k)}-\mathbf{X}^0\|_{S_\infty} \le \zeta\sigma_r(\mathbf{X}^0)$, using Lemma B.6 twice in the fourth inequality and $\|\mathbf{AB}\|_F \le \|\mathbf{A}\|_{S_\infty}\|\mathbf{B}\|_F$ all matrices $\mathbf{A}$ and $\mathbf{B}$, referring to the notations of Lemma B.9 (see below) for $\mathbf{U}_0, \mathbf{\Sigma}_0, \mathbf{V}_0, \mathbf{U}_\perp^{(k)}$ and $\mathbf{V}_\perp^{(k)}$.

Using Lemma B.4 for $\eta^{(k)} = \mathbf{X}^{(k)} - \mathbf{X}^0$, we obtain on the event on which the statement of Lemma B.4 holds that

$$\begin{aligned}
\|\eta^{(k)}\|_{S_\infty} \leq \|\eta^{(k)}\|_F &\leq \sqrt{\frac{\widetilde{C} d \log(D)}{\mu_0 r}} \|\mathcal{P}_{T_k^\perp}(\eta^{(k)})\|_F \\
&\leq \sqrt{\frac{\widetilde{C} d \log(D)}{\mu_0 r}} \left(\sqrt{d-r}\sigma_{r+1}(\mathbf{X}^{(k)}) + \frac{8\sqrt{r}\kappa \|\eta^{(k)}\|_{S_\infty}^2}{\sigma_r(\mathbf{X}^0)} \right) \\
&\leq \sqrt{\frac{\widetilde{C} d \log(D)}{\mu_0 r}} \left(\sqrt{d-r}\sigma_{r+1}(\mathbf{X}^{(k)}) + \frac{8\sqrt{r}\kappa \mu_0 \sigma_r(\mathbf{X}^0)}{4(1+6\kappa) d \log(D) \widetilde{C} \sigma_r(\mathbf{X}^0)} \|\eta^{(k)}\|_{S_\infty} \right) \\
&= \sqrt{\frac{\widetilde{C} d (d-r) \log(D)}{\mu_0 r}} \sigma_{r+1}(\mathbf{X}^{(k)}) + \frac{1}{3}\sqrt{\frac{\mu_0}{\widetilde{C} d \log(D)}} \|\eta^{(k)}\|_{S_\infty}.
\end{aligned}$$

Since $\mu_0 \leq \frac{d}{r}$, we have that $\frac{1}{3}\sqrt{\frac{\mu_0}{\widetilde{C} d \log(D)}} < \frac{1}{2}$, and therefore we obtain, after rearranging,

$$\left(1 - \frac{1}{2}\right) \|\eta^{(k)}\|_{S_\infty} < \left(1 - \frac{1}{3}\sqrt{\frac{\mu_0}{\widetilde{C} d \log(D)}}\right) \|\eta^{(k)}\|_{S_\infty} \leq \sqrt{\frac{\widetilde{C} d (d-r) \log(D)}{\mu_0 r}} \sigma_{r+1}(\mathbf{X}^{(k)}),$$

which implies the statement of this lemma. □

B.2. Weight operator and matrix perturbation

In the following, we use a well-known bound on perturbations of the singular value decomposition, which is originally due to (Wedin, 1972). The result bounds the alignment of the subspaces spanned by the singular vectors of two matrices by their norm distance, given a gap between the first singular values of one matrix and the last singular values of the other matrix that is sufficiently pronounced.

Lemma B.6 (Wedin's bound (Stewart, 2006))**.** *Let $\mathbf{X}$ and $\widehat{\mathbf{X}}$ be two matrices of the same size and their singular value decompositions*

$$\mathbf{X} = \begin{pmatrix} \mathbf{U} & \mathbf{U}_\perp \end{pmatrix} \begin{pmatrix} \mathbf{\Sigma} & 0 \\ 0 & \mathbf{\Sigma}_\perp \end{pmatrix} \begin{pmatrix} \mathbf{V}^* \\ \mathbf{V}_\perp^* \end{pmatrix} \quad \text{and} \quad \widehat{\mathbf{X}} = \begin{pmatrix} \widehat{\mathbf{U}} & \widehat{\mathbf{U}}_\perp \end{pmatrix} \begin{pmatrix} \widehat{\mathbf{\Sigma}} & 0 \\ 0 & \widehat{\mathbf{\Sigma}}_\perp \end{pmatrix} \begin{pmatrix} \widehat{\mathbf{V}}^* \\ \widehat{\mathbf{V}}_\perp^* \end{pmatrix},$$

where the submatrices have the sizes of corresponding dimensions. Suppose that δ, α satisfying $0 < \delta \leq \alpha$ are such that $\alpha \leq \sigma_{\min}(\Sigma)$ and $\sigma_{\max}(\widehat{\Sigma}_\perp) < \alpha - \delta$. Then

$$\|\widehat{\mathbf{U}}_\perp^* \mathbf{U}\|_{S_\infty} \leq \sqrt{2}\frac{\|\mathbf{X} - \widehat{\mathbf{X}}\|_{S_\infty}}{\delta} \text{ and } \|\widehat{\mathbf{V}}_\perp^* \mathbf{V}\|_{S_\infty} \leq \sqrt{2}\frac{\|\mathbf{X} - \widehat{\mathbf{X}}\|_{S_\infty}}{\delta}. \tag{39}$$

We also use a lemma which provides an explicit formula for the calculation of the new iterate $\mathbf{X}^{(k)}$ of `MatrixIRLS` and its characterization by optimality conditions. It is well-known in the IRLS literature, see, e.g., Eq. (1.9) and Lemma 5.2 of (Daubechies et al., 2010) or Lemma 5.1 (Fornasier et al., 2011), and is very general as it holds for any positive definite weight operator.

Lemma B.7. *Let $P_\Omega : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}^m$ be the sampling operator, let $\mathbf{y} \in \mathbb{R}^m$. Let $W^{(k)} : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}^{d_1 \times d_2}$ be the weight operator of Definition 2.1 defined based on $\mathbf{X}^{(k)} \in \mathbb{R}^{d_1 \times d_2}$. Then the solution of the weighted least squares step (5) of Algorithm 1 is unique and*

$$\mathbf{X}^{(k+1)} = \underset{P_\Omega(\mathbf{X}) = \mathbf{y}}{\arg\min} \langle \mathbf{X}, W^{(k)}(\mathbf{X}) \rangle = (W^{(k)})^{-1} P_\Omega^* \left(P_\Omega (W^{(k)})^{-1} P_\Omega^* \right)^{-1} (\mathbf{y}), \tag{40}$$

where $(W^{(k)})^{-1} : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}^{d_1 \times d_2}$ is the inverse matrix operator of $W^{(k)}$.

Moreover, a matrix $\mathbf{X}^{(k+1)} \in \mathbb{R}^{d_1 \times d_2}$ coincides with the one of (40) if and only if

$$\langle W^{(k)}(\mathbf{X}^{(k+1)}), \eta \rangle = 0 \text{ for all } \eta \in \ker P_\Omega \quad \text{and} \quad P_\Omega(\mathbf{X}^{(k+1)}) = \mathbf{y}. \tag{41}$$

We show the following lemma. Wherever it appears, $\|\mathbf{X}\|_{S_1}$ denotes the nuclear norm $\|\mathbf{X}\|_{S_1} = \sum_{i=1}^{d} \sigma_i(\mathbf{X})$ of a matrix $\mathbf{X} \in \mathbb{R}^{d_1 \times d_2}$.

Lemma B.8. *Let $\mathbf{X}^0 \in \mathbb{R}^{d_1 \times d_2}$ be a matrix of rank r, let $\mathbf{X}^{(k)}$ be the k-th iterate of Algorithm 1 for input parameters Ω, $\mathbf{y} = P_\Omega(\mathbf{X}^0)$ and $\widetilde{r} = r$. Assume that $\epsilon_k = \sigma_{r+1}(\mathbf{X}^{(k)})$ and that*

$$\|\eta\|_F \leq c(\mu_0, r, d_1, d_2) \|\mathcal{P}_{T_k^\perp} \eta\|_F \qquad \textit{for all } \eta \in \ker P_\Omega \tag{42}$$

for some constant $c(\mu_0, r, d_1, d_2)$ that may depend on μ_0, r, d_1, d_2, where $T_k = T_{\mathcal{T}_r(\mathbf{X}^{(k)})}\mathcal{M}_r$ is tangent space onto the manifold of rank-r matrices at $\mathcal{T}_r(\mathbf{X}^{(k)})$. Then

$$\|\mathbf{X}^{(k+1)} - \mathbf{X}^0\|_{S_\infty} \leq c(\mu_0, r, d_1, d_2)^2 \epsilon_k^2 \|W^{(k)}(\mathbf{X}^0)\|_{S_1}, \tag{43}$$

if $W^{(k)} : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}^{d_1 \times d_2}$ is the optimal weight operator of Definition 2.1 corresponding to $\mathbf{X}^{(k)}$.

Proof of Lemma B.8. Let $\eta^{(k+1)} := \mathbf{X}^{(k+1)} - \mathbf{X}^0$. Since $\eta^{(k+1)}$ is in the nullspace $\ker P_\Omega$, it follows from (42) that

$$\|\eta^{(k+1)}\|_{S_\infty}^2 \leq \|\eta^{(k+1)}\|_F^2 \leq c(\mu_0, r, d_1, d_2)^2 \|\mathcal{P}_{T_k^\perp}(\eta^{(k+1)})\|_F^2. \tag{44}$$

Recalling the definition of the weight operator $W^{(k)} : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}^{d_1 \times d_2}$ from Definition 2.1 we see that, if

$$\mathbf{X}^{(k)} = \mathbf{U}_k \mathbf{\Sigma}_k \mathbf{V}_k^* = \begin{bmatrix} \mathbf{U}^{(k)} & \mathbf{U}_\perp^{(k)} \end{bmatrix} \begin{bmatrix} \mathbf{\Sigma}^{(k)} & 0 \\ 0 & \mathbf{\Sigma}_\perp^{(k)} \end{bmatrix} \begin{bmatrix} \mathbf{V}^{(k)*} \\ \mathbf{V}_\perp^{(k)*} \end{bmatrix} \tag{45}$$

is a singular value decomposition with $\mathbf{U}^{(k)} \in \mathbb{R}^{d_1 \times r}$, $\mathbf{U}_\perp^{(k)} \in \mathbb{R}^{d_1 \times (d_1 - r)}$, $\mathbf{V}^{(k)} \in \mathbb{R}^{d_2 \times r}$, $\mathbf{V}_\perp^{(k)} \in \mathbb{R}^{d_2 \times (d_2 - r)}$, we have that

$$\langle \mathbf{Z}, W^{(k)}(\mathbf{Z}) \rangle = \langle \mathbf{U}_k^* \mathbf{Z} \mathbf{V}_k, \mathbf{H}_k \circ (\mathbf{U}_k^* \mathbf{Z} \mathbf{V}_k) \rangle \tag{46}$$

where $\mathbf{H}_k \in \mathbb{R}^{d_1 \times d_2}$ is as in Definition 2.1.

If $\mathbf{Z} = \mathcal{P}_{T_k^\perp}(\eta^{(k+1)}) \in T_k^\perp$, we know that $\mathbf{U}^{(k)*}\mathbf{Z} = 0$ and $\mathbf{Z}\mathbf{V}^{(k)} = 0$, and therefore

$$\mathbf{U}_k^* \mathbf{Z} \mathbf{V}_k = \begin{bmatrix} \mathbf{U}^{(k)*} \\ \mathbf{U}_\perp^{(k)*} \end{bmatrix} \mathbf{Z} \begin{bmatrix} \mathbf{V}^{(k)} & \mathbf{V}_\perp^{(k)} \end{bmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & \mathbf{U}_\perp^{(k)*} \mathbf{Z} \mathbf{V}_\perp^{(k)} \end{pmatrix}$$

with $\mathbf{U}_\perp^{(k)*} \mathbf{Z} \mathbf{V}_\perp^{(k)} \in \mathbb{R}^{(d_1 - r) \times (d_2 - r)}$.

By assumption of Lemma B.8, we know that $\epsilon_k = \sigma_{r+1}(\mathbf{X}^{(k)})$, which means that $r_k := |\{i \in [d] : \sigma_i(\mathbf{X}^{(k)}) > \epsilon_k\}| = r$, and therefore $(\mathbf{H}_k)_{ij} = \epsilon_k^{-2}$ for all $i, j > r$. This entails with (46) that

$$\begin{aligned} \langle \mathcal{P}_{T_k^\perp}(\eta^{(k+1)}), W^{(k)}(\mathcal{P}_{T_k^\perp}(\eta^{(k+1)})) \rangle &= \epsilon_k^{-2} \langle \mathbf{U}_k^* \mathcal{P}_{T_k^\perp}(\eta^{(k+1)}) \mathbf{V}_k, \mathbf{U}_k^* \mathcal{P}_{T_k^\perp}(\eta^{(k+1)}) \mathbf{V}_k \rangle \\ &= \epsilon_k^{-2} \langle \mathcal{P}_{T_k^\perp}(\eta^{(k+1)}), \mathcal{P}_{T_k^\perp}(\eta^{(k+1)}) \rangle = \epsilon_k^{-2} \|\mathcal{P}_{T_k^\perp}(\eta^{(k+1)})\|_F^2, \end{aligned}$$

using the cyclicity of the trace and the fact that $\mathbf{U}_k$ and $\mathbf{V}_k$ are orthonormal matrices.

Inserting this into (44), we obtain

$$\begin{aligned} \|\eta^{(k+1)}\|_{S_\infty}^2 &\leq c(\mu_0, r, d_1, d_2)^2 \epsilon_k^2 \left\langle \mathcal{P}_{T_k^\perp}(\eta^{(k+1)}), W^{(k)}(\mathcal{P}_{T_k^\perp}(\eta^{(k+1)})) \right\rangle \\ &\leq c(\mu_0, r, d_1, d_2)^2 \epsilon_k^2 \left\langle \eta^{(k+1)}, W^{(k)}(\eta^{(k+1)}) \right\rangle, \end{aligned} \tag{47}$$

where the last inequality holds since $W^{(k)}$ is positive definite and since $\left\langle \mathcal{P}_{T_k^\perp}(\eta^{(k+1)}), W^{(k)}(\mathcal{P}_{T_k}(\eta^{(k+1)})) \right\rangle = 0$ due to the orthogonality of T_k and $T_k^\perp$. Due to Lemma B.7, we know that the new iterate $\mathbf{X}^{(k+1)}$ fulfills

$$0 = \langle W^{(k)}(\mathbf{X}^{(k+1)}), \eta^{(k+1)} \rangle = \langle W^{(k)}(\eta^{(k+1)} + \mathbf{X}^0), \eta^{(k+1)} \rangle,$$

and therefore

$$\left\langle \eta^{(k+1)}, W^{(k)}(\eta^{(k+1)}) \right\rangle = -\left\langle W^{(k)}(\mathbf{X}^0), \eta^{(k+1)} \right\rangle \leq \|W^{(k)}(\mathbf{X}^0)\|_{S_1} \|\eta^{(k+1)}\|_{S_\infty},$$

using Hölder's inequality for Schatten-p (quasi-)norms (cf. Theorem 11.2 of (Gohberg et al., 2000)). Dividing (47) by $\|\eta^{(k+1)}\|_{S_\infty}$ concludes the proof of Lemma B.8. □

In order to obtain a fast local convergence rate, it is crucial to bound $\|W^{(k)}(\mathbf{X}^0)\|_{S_1}$. For this, we split $\|W^{(k)}(\mathbf{X}^0)\|_{S_1}$ into three parts and estimate the parts separately by using the classical singular subspace perturbation result of Lemma B.6.

Lemma B.9. *Let $W^{(k)} : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}^{d_1 \times d_2}$ be the weight operator (4) of Definition 2.1 corresponding to $\mathbf{X}^{(k)}$, let $\epsilon_k = \sigma_{r+1}(\mathbf{X}^{(k)}) = \sigma_r^{(k)}$ and $\mathbf{X}^0 \in \mathbb{R}^{d_1 \times d_2}$ be a rank-r matrix. Assume that there exists $0 < \zeta < 1$ such that*

$$\|\mathbf{X}^{(k)} - \mathbf{X}^0\|_{S_\infty} \leq \zeta \sigma_r(\mathbf{X}^0). \tag{48}$$

Then

$$\left\|W^{(k)}(\mathbf{X}^0)\right\|_{S_1} \leq r(1-\zeta)^{-2}\sigma_r(\mathbf{X}^0)^{-1}\left(1 + 4\frac{\|\eta^{(k)}\|_{S_\infty}}{\epsilon_k}\frac{\sigma_1(\mathbf{X}^0)}{\sigma_r(\mathbf{X}^0)} + 2\frac{\|\eta^{(k)}\|_{S_\infty}^2}{\epsilon_k^2}\frac{\sigma_1(\mathbf{X}^0)}{\sigma_r(\mathbf{X}^0)}\right).$$

Proof. Recalling the notation $\sigma_\ell^{(k)} = \sigma_\ell(\mathbf{X}^{(k)})$ for the ℓ-th singular value of $\mathbf{X}^{(k)}$ and the decomposition

$$\mathbf{H}_k = \begin{bmatrix} \mathbf{H}^{(k)} & \mathbf{H}_{1,2}^{(k)} \\ \mathbf{H}_{2,1}^{(k)} & \epsilon_k^{-2}\mathbf{1} \end{bmatrix} \tag{49}$$

of (12), we bound the entries of the different blocks $\mathbf{H}^{(k)}$, $\mathbf{H}_{1,2}^{(k)}$ and $\mathbf{H}_{2,1}^{(k)}$ separately.

Since $(\mathbf{H}_k)_{ij} = \Big(\max(\sigma_i^{(k)}, \epsilon_k)\max(\sigma_j^{(k)}, \epsilon_k)\Big)^{-1}$ for each $i \in [d_1]$ and $j \in [d_2]$ due to definition of $\mathbf{H}_k$, we observe that

$$\max_{i \in [r], j \in [r]} (\mathbf{H}^{(k)})_{ij} \leq (\sigma_r^{(k)})^{-2}, \tag{50}$$

and

$$\max\left(\max_{i,j}((\mathbf{H}_{1,2}^{(k)})_{ij}), \max_{i,j}((\mathbf{H}_{2,1}^{(k)})_{ij})\right) = \max_{i \in [r], r+1 \leq j \leq d_2} (\mathbf{H}_{1,2}^{(k)})_{ij} \leq (\sigma_r^{(k)})^{-1}\epsilon_k^{-1}, \tag{51}$$

In view of these entrywise bounds on the submatrices of $\mathbf{H}_k$ and $\mathbf{H}_2^{(k)}$, we compute, using (12), that

$$\begin{aligned}
\left\|W^{(k)}(\mathbf{X}^0)\right\|_{S_1} &= \left\|\begin{bmatrix}\mathbf{U}^{(k)} & \mathbf{U}_\perp^{(k)}\end{bmatrix}\left(\begin{bmatrix} \mathbf{H}^{(k)} & \mathbf{H}_{1,2}^{(k)} \\ \mathbf{H}_{2,1}^{(k)} & \epsilon_k^{-2}\mathbf{1} \end{bmatrix} \circ \begin{bmatrix} \mathbf{U}^{(k)*}\mathbf{X}^0\mathbf{V}^{(k)} & \mathbf{U}^{(k)*}\mathbf{X}^0\mathbf{V}_\perp^{(k)} \\ \mathbf{U}_\perp^{(k)*}\mathbf{X}^0\mathbf{V}^{(k)} & \mathbf{U}_\perp^{(k)*}\mathbf{X}^0\mathbf{V}_\perp^{(k)} \end{bmatrix}\right)\begin{bmatrix}\mathbf{V}^{(k)*} \\ \mathbf{V}_\perp^{(k)*}\end{bmatrix}\right\|_{S_1} \\
&\leq \left\|\mathbf{U}^{(k)}[\mathbf{H}^{(k)} \circ (\mathbf{U}^{(k)*}\mathbf{X}^0\mathbf{V}^{(k)})]\mathbf{V}^{(k)*}\right\|_{S_1} + \left\|\mathbf{U}_k\begin{bmatrix} 0 & \mathbf{H}_{1,2}^{(k)} \circ (\mathbf{U}^{(k)*}\mathbf{X}^0\mathbf{V}_\perp^{(k)}) \\ \mathbf{H}_{2,1}^{(k)} \circ (\mathbf{U}_\perp^{(k)*}\mathbf{X}^0\mathbf{V}^{(k)}) & 0 \end{bmatrix}\mathbf{V}_k^*\right\|_{S_1} \\
&\quad + \epsilon_k^{-2}\left\|\mathbf{U}_\perp^{(k)}\mathbf{U}_\perp^{(k)*}\mathbf{X}^0\mathbf{V}_\perp^{(k)}\mathbf{V}_\perp^{(k)*}\right\|_{S_1} =: \text{(I)} + \text{(II)} + \text{(III)}.
\end{aligned}$$

We now bound the terms (I), (II) and (III) separately.

First, we see that

$$\begin{aligned}
\text{(I)} = \left\|\mathbf{H}^{(k)} \circ (\mathbf{U}^{(k)*}\mathbf{X}^0\mathbf{V}^{(k)})\right\|_{S_1} &\leq \sqrt{r}\left\|\mathbf{H}^{(k)} \circ (\mathbf{U}^{(k)*}\mathbf{X}^0\mathbf{V}^{(k)})\right\|_F \\
&\leq \sqrt{r}\left\|\mathbf{H}^{(k)} \circ (\mathbf{U}^{(k)*}\mathbf{X}^{(k)}\mathbf{V}^{(k)})\right\|_F + \sqrt{r}\left\|\mathbf{H}^{(k)} \circ (\mathbf{U}^{(k)*}\eta^{(k)}\mathbf{V}^{(k)})\right\|_F \\
&\leq \sqrt{r}\left\|\mathbf{H}^{(k)} \circ \mathbf{\Sigma}^{(k)}\right\|_F + \sqrt{r}(\sigma_r^{(k)})^{-2}\|\mathbf{U}^{(k)*}\eta^{(k)}\mathbf{V}^{(k)}\|_F,
\end{aligned}$$

where we used the Cauchy-Schwarz inequality in the first inequality, the notation $\eta^{(k)} = \mathbf{X}^{(k)} - \mathbf{X}^0$ and the triangle inequality in the second inequality, and finally, (50) in the third inequality. $\Sigma^{(k)}$ $\mathbb{R}^{r \times r}$ is here as in (9).

Since

$$\left\|\mathbf{H}^{(k)} \circ \mathbf{\Sigma}^{(k)}\right\|_F = \left(\sum_{i=1}^r (\sigma_i^{(k)})^{-2}\right)^{1/2} \leq \sqrt{r}(\sigma_r^{(k)})^{-1}$$

and

$$\|\mathbf{U}^{(k)*}\eta^{(k)}\mathbf{V}^{(k)}\|_F \leq \sqrt{r}\|\mathbf{U}^{(k)*}\eta^{(k)}\mathbf{V}^{(k)}\|_{S_\infty} \leq \sqrt{r}\|\eta^{(k)}\|_{S_\infty} \leq \sqrt{r}\zeta\sigma_r(\mathbf{X}^0)$$

from assumption (48), it follows then that

$$\text{(I)} \leq r(\sigma_r^{(k)})^{-2}\left(\sigma_r^{(k)} + \zeta\sigma_r(\mathbf{X}^0)\right).$$

We can use the proximity assumption (48) further to get rid of the dependence on k in the bound, as

$$\sigma_r(\mathbf{X}^0) = \sigma_r(\mathbf{X}^{(k)} - \eta^{(k)}) \leq \sigma_r^{(k)} + \sigma_1(\eta^{(k)}) = \sigma_r^{(k)} + \|\eta^{(k)}\|_{S_\infty} \leq \sigma_r^{(k)} + \zeta\sigma_r(\mathbf{X}^0),$$

using $\sigma_{i+j-1}(\mathbf{A}) \leq \sigma_i(\mathbf{A}+\mathbf{B}) + \sigma_j(\mathbf{B})$ for any i, j (cf. Theorem 3.3.16 of (Horn & Johnson, 1991)) with $\mathbf{A}+\mathbf{B} = X^{(k)} - \eta^{(k)}$ and $\mathbf{B} = \eta^k$ so that

$$\sigma_r^{(k)} \geq (1-\zeta)\sigma_r(\mathbf{X}^0), \tag{52}$$

and hence

$$\text{(I)} \leq r\sigma_r(\mathbf{X}^0)^{-2}(1-\zeta)^{-2}\left(\sigma_r(\mathbf{X}^0)(1-\zeta) + \zeta\sigma_r(\mathbf{X}^0)\right) = r(1-\zeta)^{-2}\sigma_r(\mathbf{X}^0)^{-1}. \tag{53}$$

For the term (II), we compute that

$$\begin{aligned}
\text{(II)} &\leq \sqrt{2r}\left\|\begin{bmatrix} 0 & \mathbf{H}_{1,2}^{(k)} \circ (\mathbf{U}^{(k)*}\mathbf{X}^0\mathbf{V}_\perp^{(k)}) \\ \mathbf{H}_{2,1}^{(k)} \circ (\mathbf{U}_\perp^{(k)*}\mathbf{X}^0\mathbf{V}^{(k)}) & 0 \end{bmatrix}\right\|_F \\
&\leq \sqrt{2r}(\sigma_r^{(k)})^{-1}\epsilon_k^{-1}\left(\left\|\mathbf{U}^{(k)*}\mathbf{X}^0\mathbf{V}_\perp^{(k)}\right\|_F + \left\|\mathbf{U}_\perp^{(k)*}\mathbf{X}^0\mathbf{V}^{(k)}\right\|_F\right) \\
&\leq \sqrt{2r}(\sigma_r^{(k)})^{-1}\epsilon_k^{-1}\left(\|\mathbf{U}^{(k)*}\mathbf{U}_0\mathbf{\Sigma}_0\|_F\|\mathbf{V}_0^*\mathbf{V}_\perp^{(k)}\|_{S_\infty} + \|\mathbf{U}_\perp^{(k)*}\mathbf{U}_0\|_{S_\infty}\|\mathbf{\Sigma}_0\mathbf{V}_0^*\mathbf{V}^{(k)}\|_F\right),
\end{aligned}$$

using the singular value decomposition $\mathbf{X}^0 = \mathbf{U}_0\mathbf{\Sigma}_0\mathbf{V}_0^*$ of the rank-r matrix $\mathbf{X}^0$ with $\mathbf{U}_0 \in \mathbb{R}^{d_1\times r}$, $\mathbf{V}_0 \in \mathbb{R}^{d_2\times r}$. This allows us to use the singular subspace perturbation result of Lemma B.6, so that $\|\mathbf{V}_0^*\mathbf{V}_\perp^{(k)}\|_{S_\infty}$ and $\|\mathbf{U}_\perp^{(k)*}\mathbf{U}_0\|_{S_\infty}$ can compensate for the negative power of the ϵ_k, avoiding a blow-up of term (II): Indeed, using Lemma B.6 with $\mathbf{X} = \mathbf{X}^0$, $\widehat{\mathbf{X}} = \mathbf{X}^{(k)}$, $\alpha = \sigma_r(\mathbf{X}^0)$ and $\delta = (1-\zeta)\sigma_r(\mathbf{X}^0)$ results in

$$\max(\|\mathbf{V}_0^*\mathbf{V}_\perp^{(k)}\|_{S_\infty}, \|\mathbf{U}_\perp^{(k)*}\mathbf{U}_0\|_{S_\infty}) \leq \frac{\sqrt{2}\|\eta^{(k)}\|_{S_\infty}}{(1-\zeta)\sigma_r(\mathbf{X}^0)},$$

and since $\|\mathbf{U}^{(k)*}\mathbf{U}_0\mathbf{\Sigma}_0\|_F \leq \|\mathbf{\Sigma}_0\|_F \leq \sqrt{r}\sigma_1(\mathbf{X}^0)$, $\|\mathbf{\Sigma}_0\mathbf{V}_0^*\mathbf{V}^{(k)}\|_F \leq \sqrt{r}\sigma_1(\mathbf{X}^0)$, we obtain with (52) that

$$\text{(II)} \leq 4r(1-\zeta)^{-2}\sigma_r(\mathbf{X}^0)^{-1}\frac{\|\eta^{(k)}\|_{S_\infty}}{\epsilon_k}\frac{\sigma_1(\mathbf{X}^0)}{\sigma_r(\mathbf{X}^0)}. \tag{54}$$

It remains to bound the last term (III). For (III), we can use the subspace perturbation lemma *twice* in the same summand such that

$$\begin{aligned}
\text{(III)} = \epsilon_k^{-2}\left\|\mathbf{U}_\perp^{(k)}\mathbf{U}_\perp^{(k)*}\mathbf{X}^0\mathbf{V}_\perp^{(k)}\mathbf{V}_\perp^{(k)*}\right\|_{S_1} &= \epsilon_k^{-2}\|\mathbf{U}_\perp^{(k)*}\mathbf{X}^0\mathbf{V}_\perp^{(k)}\|_{S_1} \leq \sqrt{r}\epsilon_k^{-2}\|\mathbf{U}_\perp^{(k)*}\mathbf{X}^0\mathbf{V}_\perp^{(k)}\|_F \\
&\leq \sqrt{r}\epsilon_k^{-2}\|\mathbf{U}_\perp^{(k)*}\mathbf{U}_0\|_{S_\infty}\|\mathbf{\Sigma}_0\|_F\|\mathbf{V}_0^*\mathbf{V}_\perp^{(k)}\|_{S_\infty} \\
&\leq \sqrt{r}\epsilon_k^{-2}\frac{\sqrt{2}\|\eta^{(k)}\|_{S_\infty}}{(1-\zeta)\sigma_r(\mathbf{X}^0)}\sqrt{r}\sigma_1(\mathbf{X}^0)\frac{\sqrt{2}\|\eta^{(k)}\|_{S_\infty}}{(1-\zeta)\sigma_r(\mathbf{X}^0)} = 2r(1-\zeta)^{-2}\sigma_r(\mathbf{X}^0)^{-1}\frac{\|\eta^{(k)}\|_{S_\infty}^2}{\epsilon_k^2}\frac{\sigma_1(\mathbf{X}^0)}{\sigma_r(\mathbf{X}^0)}.
\end{aligned} \tag{55}$$

Combining eqs. (53) to (55) finally yields the statement of Lemma B.9. □

B.3. Wrapping up the proof

We can now put Lemma B.4, Lemma B.8 and Lemma B.9 together to prove the local convergence statement of Theorem 4.1, showing also that we attain locally quadratic convergence.

Proof of Theorem 4.1. Let $k = k_0$ and $\mathbf{X}^{(k)}$ be the k-th iterate of `MatrixIRLS` with the parameters stated in Theorem 4.1. Under the sampling model of Theorem 4.1, if the number of samples m fulfills $m \geq C\mu_0 r(d_1+d_2)\log(d_1+d_2)$, where C is the constant of Lemma B.4, we know from Lemma B.4 that with a probability of at least $1-2D^{-2}$, inequality (42) is satisfied with $c(\mu_0, r, d_1, d_2) = \sqrt{\frac{\widetilde{C}d\log(D)}{\mu_0 r}}$, if furthermore $\eta^{(k)} := \mathbf{X}^{(k)} - \mathbf{X}^0$ fulfills

$$\|\eta^{(k)}\|_{S_\infty} \leq \xi\sigma_r(\mathbf{X}^0) \tag{56}$$

with

$$\xi \leq C_1\sqrt{\frac{\mu_0 r}{d}}, \tag{57}$$

and thus, by Lemma B.8,

$$\|\mathbf{X}^{(k+1)} - \mathbf{X}^0\|_{S_\infty} \leq \frac{\widetilde{C}d\log(D)}{\mu_0 r}\epsilon_k^2\|W^{(k)}(\mathbf{X}^0)\|_{S_1}. \tag{58}$$

We denote the event that this is fulfilled by E. Furthermore, on this event, if $\xi \leq 1/2$ in (56) and denoting the condition number by $\kappa = \sigma_1(\mathbf{X}^0)/\sigma_r(\mathbf{X}^0)$, it follows from Lemma B.9 that

$$\|\mathbf{X}^{(k+1)} - \mathbf{X}^0\|_{S_\infty} \leq \frac{\widetilde{C}d\log(D)}{\mu_0}4\sigma_r(\mathbf{X}^0)^{-1}\left(\epsilon_k^2 + 4\epsilon_k\|\eta^{(k)}\|_{S_\infty}\kappa + 2\|\eta^{(k)}\|_{S_\infty}^2\kappa\right)$$

Furthermore, if $\mathbf{X}_r^{(k)} \in \mathbb{R}^{d_1\times d_2}$ denotes the best rank-r approximation of $\mathbf{X}^{(k)}$ in any unitarily invariant norm, we estimate that

$$\epsilon_k \leq \sigma_{r+1}(\mathbf{X}^{(k)}) = \|\mathbf{X}^{(k)} - \mathbf{X}_r^{(k)}\|_{S_\infty} \leq \|\mathbf{X}^{(k)} - \mathbf{X}^0\|_{S_\infty} = \|\eta^{(k)}\|_{S_\infty},$$

Inserting these two bounds into (58), we obtain

$$\|\eta^{(k+1)}\|_{S_\infty} = \|\mathbf{X}^{(k+1)} - \mathbf{X}^0\|_{S_\infty} \leq \frac{\widetilde{C}d\log(D)}{\mu_0}4\sigma_r(\mathbf{X}^0)^{-1}(1+6\kappa)\|\eta^{(k)}\|_{S_\infty}^2.$$

Finally, if, additionally, (56) is satisfied for

$$\xi \leq \frac{\mu_0}{4(1+6\kappa)d\log(D)\widetilde{C}}, \tag{59}$$

we conclude that

$$\|\eta^{(k+1)}\|_{S_\infty} < \|\eta^{(k)}\|_{S_\infty}$$

and also, we observe a quadratic decay in the spectral error such that

$$\|\eta^{(k+1)}\|_{S_\infty} \leq \mu\|\eta^{(k)}\|_{S_\infty}^2$$

with a constant $\mu = \frac{4\widetilde{C}d\log(D)(1+6\kappa)}{\mu_0\sigma_r(\mathbf{X}^0)}$. This shows inequality (33) of Theorem 4.1.

To show the remaining statement, we can use Lemma B.5 to show that if $\mathbf{X}^{(k)}$ is close enough to $\mathbf{X}^0$, we can ensure that the $(r+1)$-st singular value $\sigma_{r+1}(\mathbf{X}^{(k)})$ of the current iterate is strictly decreasing. More precisely, assume now the stricter assumption of

$$\|\eta^{(k)}\|_{S_\infty} \leq \sqrt{\frac{\mu_0 r}{4\widetilde{C}d(d-r)\log(D)}}\xi\sigma_r(\mathbf{X}^0). \tag{60}$$

In fact, if ξ fulfills (57) and (59), we can conclude that on the event E,

$$\begin{aligned}\sigma_{r+1}(\mathbf{X}^{(k+1)}) &\leq \|\eta^{(k+1)}\|_{S_\infty} \leq \frac{\widetilde{C}d\log(D)}{\mu_0}4\sigma_r(\mathbf{X}^0)^{-1}(1+6\kappa)\|\eta^{(k)}\|_{S_\infty}\cdot\|\eta^{(k)}\|_{S_\infty} \\ &< \frac{\widetilde{C}d\log(D)}{\mu_0}4\sigma_r(\mathbf{X}^0)^{-1}(1+6\kappa)\sqrt{\frac{\mu_0 r}{4\widetilde{C}d(d-r)\log(D)}}\xi\sigma_r(\mathbf{X}^0)\sqrt{\frac{4\widetilde{C}d(d-r)\log(D)}{\mu_0 r}}\sigma_{r+1}(\mathbf{X}^{(k)}) \\ &\leq \sigma_{r+1}(\mathbf{X}^{(k)})\end{aligned}$$

using Lemma B.5 for one factor $\|\eta^{(k)}\|_{S_\infty}$ and (60) for the other factor $\|\eta^{(k)}\|_{S_\infty}$ in the third inequality, and (59) in the last inequality. Taking the update rule (6) for the smoothing parameter into account, this implies that $\epsilon_{k+1} = \sigma_{r+1}(\mathbf{X}^{(k+1)})$, which ensures that the first statement of Theorem 4.1 is fulfilled likewise for iteration $k+1$. By induction, this implies that $\mathbf{X}^{(k+\ell)} \xrightarrow{\ell\to\infty} \mathbf{X}^0$, which finishes the proof of Theorem 4.1. □

The presented proof of Theorem 4.1 has certain similarities with the proof of *local superlinear convergence* of Theorem 11 in (Kümmerle & Sigl, 2018) for a related IRLS algorithm designed for Schatten-p quasi-norm minimization. However, that proof is not applicable to the matrix completion setting, and furthermore, is not extendable to a log-determinant objective as used in this paper. As observed in (Kümmerle & Sigl, 2018), it is not possible to obtain superlinear (or quadratic) convergence rates for the IRLS methods of (Fornasier et al., 2011; Mohan & Fazel, 2012).

C. Proof of Theorem 4.2

In this section, we provide a result about the *spectrum* of the system matrix

$$\mathbf{A} := \mathbf{D}_k + P_{T_k}^* P_\Omega^* P_\Omega P_{T_k} := \epsilon_k^2 \left(\mathbf{D}_{S_k}^{-1} - \epsilon_k^2 \mathbf{I}_{S_k}\right)^{-1} + P_{T_k}^* P_\Omega^* P_\Omega P_{T_k} \tag{61}$$

of eq. (14) in Algorithm 2. We recall that solving a linear system with $\mathbf{A}$ constitutes the main computational step in our implementation of `MatrixIRLS`.

It is well-known that the shape of the spectrum of the system matrix $\mathbf{A}$ plays an important role in the convergence of the conjugate gradient iterations. In particular, the CG method terminates after ℓ iterations (in exact arithmetic) if $\mathbf{A}$ has ℓ distinct eigenvalues (cf. Theorem 5.4 of (Nocedal & Wright, 2006)) and a bound on the error $\gamma_\ell - \gamma^*$ of the ℓ-th iterate γ_ℓ to the exact solution γ^* of the linear system (5.36) of (Nocedal & Wright, 2006) can be provided by

$$\langle \gamma_\ell - \gamma^*, \mathbf{A}(\gamma_\ell - \gamma^*)\rangle \leq 2 \left(\frac{\sqrt{\kappa(\mathbf{A})} - 1}{\sqrt{\kappa(\mathbf{A})} + 1}\right)^\ell \langle \gamma_0 - \gamma^*, \mathbf{A}(\gamma_0 - \gamma^*)\rangle ,$$

where

$$\kappa(\mathbf{A}) := \frac{\lambda_{\max}(\mathbf{A})}{\lambda_{\min}(\mathbf{A})} \tag{62}$$

is the condition number of $\mathbf{A}$.

It has been a common problem for IRLS methods that the linear systems to be solved become ill-conditioned close to the desired (low-rank or sparse, depending on the problem) solution (Daubechies et al., 2010; Fornasier et al., 2011; 2016). Close to the solution the smoothing parameter ϵ_k is typically very small, resulting in “very large weights” on large parts of the domain induced by the quadratic form

$$\langle \mathbf{X}, W^{(k)}(\mathbf{X})\rangle.$$

For the sparse recovery problem, it has been observed (Voronin, 2012) that this blow-up can be a problem for an inexact solver of the weighted least squares system, and in (Fornasier et al., 2016), an analysis was pursued for an IRLS algorithm for the sparse recovery problem about with which precision the linear system for each outer iteration k needs to be solved by a conjugate gradient method to ensure overall convergence.

However, the underlying issue of bad conditioning of the IRLS system matrices was not addressed or solved in (Fornasier et al., 2016) (see Section 5.2 of (Fornasier et al., 2016) for a discussion).

Theorem 4.2, which we now show, argues that by computing the weighted least squares update via Algorithm 2, these issues do not arise for `MatrixIRLS` in the same manner.

Proof of Theorem 4.2. Recall the definitions $\mathbf{D}_k = \epsilon_k^2 \left(\mathbf{D}_{S_k}^{-1} - \epsilon_k^2 \mathbf{I}_{S_k}\right)^{-1}$ and $\mathbf{A} = \mathbf{D}_k + P_{T_k}^* P_\Omega^* P_\Omega P_{T_k}$. We know that the eigenvalues of $\mathbf{D}_{S_k}^{-1}$ are just the inverses of the entries of the matrices $\mathbf{H}^{(k)}$, $\mathbf{H}_{1,2}^{(k)}$ and $\mathbf{H}_{2,1}^{(k)}$ in the block decomposition of the matrix $\mathbf{H}_k \in \mathbb{R}^{d_1 \times d_2}$ that defines the weight operator $W^{(k)}$. By (50) and (51), we can lower bound these eigenvalues by $\sigma_r(\mathbf{X}^{(k)})\epsilon_k = \sigma_r^{(k)}\epsilon_k$, and therefore,

$$\|\mathbf{D}_k\|_{S_\infty} \leq \frac{\epsilon_k^2}{\sigma_r(\mathbf{X}^{(k)})\epsilon_k - \epsilon_k^2} = \frac{\epsilon_k}{\sigma_r(\mathbf{X}^{(k)}) - \epsilon_k} \leq \frac{\epsilon_k}{3/4\sigma_r(\mathbf{X}_0) - \epsilon_k}, \tag{63}$$

using that $\sigma_r(\mathbf{X}^{(k)}) \geq (1-1/4)\sigma_r(\mathbf{X}_0)$ since $\|\mathbf{X}^{(k)} - \mathbf{X}_0\|_{S_\infty} \leq \frac{1}{4}\sigma_r(\mathbf{X}_0)$, see also eq. (52). Also, since $\epsilon_k = \sigma_{r+1}(\mathbf{X}^{(k)}) \leq \|\mathbf{X}^{(k)} - \mathbf{X}_0\|_{S_\infty} \leq \frac{1}{4}\sigma_r(\mathbf{X}_0)$ and further $\epsilon_k = \sigma_{r+1}(\mathbf{X}^{(k)}) \leq C_1\left(\frac{\mu_0 r}{d}\right)\sigma_r(\mathbf{X}_0)$, we have that

$$\frac{\epsilon_k}{(3/4\sigma_r(\mathbf{X}_0)) - \epsilon_k} \leq \frac{C_1\frac{\mu_0 r}{d}}{(3/4-1/4)}\frac{\sigma_r(\mathbf{X}_0)}{\sigma_r(\mathbf{X}_0)} \leq 2C_1\frac{\mu_0 r}{d}.$$

This implies that

$$0 \leq \lambda_{\min}(\mathbf{D}_k) \leq \lambda_{\max}(\mathbf{D}_k) = \|\mathbf{D}_k\|_{S_\infty} \leq 2C_1\frac{\mu_0 r}{d} \leq 2C_1\frac{m}{CdD\log(D)} \leq \frac{m}{d_1 d_2},$$

if the constant $C_1 > 0$ is small enough, using the lower bound on the sample complexity m.

The second summand in $\mathbf{A}$, the matrix $P_{T_k}^* P_\Omega^* P_\Omega P_{T_k}$, is positive semidefinite already due to its factorized form. We note that by following the proof of Theorem 4.1, we see that under the assumptions of Theorem 4.2, we have that for $\mathcal{P}_{T_k}: \mathbb{R}^{d_1\times d_2} \to \mathbb{R}^{d_1\times d_2}, \mathbf{Z} \mapsto P_{T_k}P_{T_k}^*(\mathbf{Z})$ and $\mathcal{P}_\Omega: \mathbb{R}^{d_1\times d_2} \to \mathbb{R}^{d_1\times d_2}, \mathbf{Z} \mapsto P_\Omega^* P_\Omega(\mathbf{Z})$,

$$\begin{aligned}\frac{d_1 d_2}{m}\left\|P_{T_k}\left[P_{T_k}^* P_\Omega^* P_\Omega P_{T_k} - \frac{m}{d_1 d_2}\mathbf{I}\right]P_{T_k}^*\right\|_{S_\infty} &= \left\|\frac{d_1 d_2}{m}P_{T_k}P_{T_k}^* P_\Omega^* P_\Omega P_{T_k} P_{T_k}^* - P_{T_k}P_{T_k}^*\right\|_{S_\infty} \\ = \left\|\frac{d_1 d_2}{m}\mathcal{P}_T\mathcal{P}_\Omega\mathcal{P}_T - \mathcal{P}_T\right\|_{S_\infty} \leq \frac{4}{10}\end{aligned}$$

on an event E that holds with high probability.

As P_{T_k} is a matrix with orthonormal columns such that $P_{T_k}^* P_{T_k} = \mathbf{I}$, this implies that

$$\left\|P_{T_k}^* P_\Omega^* P_\Omega P_{T_k} - \frac{m}{d_1 d_2}\mathbf{I}\right\|_{S_\infty} \leq \frac{4m}{10 d_1 d_2}$$

on the event E. Thus, the bound on the spectrum of $\mathbf{A}$ follows from this and eq. (63) since

$$\begin{aligned}\left\|\mathbf{A} - \frac{3}{2}\frac{m}{d_1 d_2}\mathbf{I}\right\|_{S_\infty} &= \left\|\mathbf{D}_k + P_{T_k}^* P_\Omega^* P_\Omega P_{T_k} - \frac{3}{2}\frac{m}{d_1 d_2}\mathbf{I}\right\|_{S_\infty} \leq \left\|\mathbf{D}_k - \frac{1}{2}\frac{m}{d_1 d_2}\mathbf{I}\right\|_{S_\infty} + \left\|P_{T_k}^* P_\Omega^* P_\Omega P_{T_k} - \frac{m}{d_1 d_2}\mathbf{I}\right\|_{S_\infty} \\ &\leq \frac{1}{2}\frac{m}{d_1 d_2} + \frac{4}{10}\frac{m}{d_1 d_2} = \frac{9}{10}\frac{m}{d_1 d_2}.\end{aligned}$$

The condition number bound follows immediately since $\kappa(\mathbf{A}) = \frac{\lambda_{\max}(\mathbf{A})}{\lambda_{\min}(\mathbf{A})} \leq 4$. □

As a summary, since Theorem 4.2 gives a bound on the condition number of the linear system matrix $\mathbf{A}$ that is a *small constant*, the theory of the conjugate gradient methods suggests that very good solutions can be found already after few, in particular, after

$$N_{\text{CG\_inner}} = \text{cst}.$$

CG iterations (where cst. is small), for each IRLS iteration, at least in the neighborhood of a low-rank matrix $\mathbf{X}_0$ that is compatible with the measurements.

Taking into account the statement of Theorem 3.1, this suggests that at least locally, a new iterate $\mathbf{X}^{(k+1)}$ can be calculated with a time complexity of

$$O\left((mr + r^2 D)\cdot N_{\text{CG\_inner}}\right) = O\left(mr + r^2 D\right).$$

D. Remarks to MatrixIRLS as a saddle-escaping smoothing Newton method

We briefly elaborate on the interpretation of `MatrixIRLS` as a saddle-escaping smoothing method.

If $\epsilon_k > 0$ and if $F_{\epsilon_k} : \mathbb{R}^{d_1 \times d_2} \to R$ is the ϵ_k-smoothed log-det objective of (3), it can be shown that F_{ϵ_k} is continuously differentiable with ϵ_k^{-2}-Lipschitz gradient $\nabla F_{\epsilon_k}(\mathbf{X}) = \mathbf{U} \operatorname{dg}\left(\frac{\sigma_i(\mathbf{X})}{\max(\sigma_i(\mathbf{X}),\epsilon_k)^2}\right)_{i=1}^{d} \mathbf{V}^*$ for any matrix $\mathbf{X}$ with singular value decomposition $\mathbf{X} = \mathbf{U} \operatorname{dg}\big(\sigma(\mathbf{X})\big)\mathbf{V}^* = \mathbf{U} \operatorname{dg}\big(\sigma\big)\mathbf{V}^*$. This can be shown by using results from (Lewis & Sendov, 2005; Andersson et al., 2016). Additionally, it holds that ∇F_{ϵ_k} is differentiable at $\mathbf{X}$ if and only if the second derivative $f''_{\epsilon_k} : \mathbb{R} \to \mathbb{R}$ of f_{ϵ_k} from (3) exists at all $\sigma = \sigma_i(\mathbf{X})$, $i \in [d]$, which is the case if $\mathbf{X} \in \mathcal{D}_{\epsilon_k} := \big\{\mathbf{X} : \sigma_i(\mathbf{X}) \neq \epsilon_k \text{ for all } i \in [d]\big\}$. The latter statement follows from the calculus of *non-Hermitian Löwner functions* (Yang, 2009; Ding et al., 2018), also called *generalized matrix functions* (Noferini, 2017), as $\mathbf{X} \mapsto \nabla F_{\epsilon_k}(\mathbf{X})$ is such a function.

Let now $\mathbf{X}^{(k)} \in \mathcal{D}_{\epsilon_k} := \big\{\mathbf{X} : \sigma_i(\mathbf{X}) \neq \epsilon_k \text{ for all } i \in [d]\big\}$ with singular value decomposition as in (9), and $r_k := |\{i \in [d] : \sigma_i(\mathbf{X}^{(k)}) > \epsilon_k\}| = |\{i \in [d] : \sigma_i^{(k)} > \epsilon_k\}|$. In this case, it can be calculated that the Hessian $\nabla^2 F_{\epsilon_k}(\mathbf{X}^{(k)})$ at $\mathbf{X}^{(k)}$, which is a function that maps $\mathbb{R}^{d_1 \times d_2}$ to $\mathbb{R}^{d_1 \times d_2}$ matrices, satisfies in the case of $d_1 = d_2$

$$\nabla^2 F_{\epsilon_k}(\mathbf{X}^{(k)})(\mathbf{Z}) = \mathbf{U}_k \left[\mathbf{M}^{\mathrm{S}} \circ S(\mathbf{U}_k^* \mathbf{Z} \mathbf{V}_k) + \mathbf{M}^{\mathrm{T}} \circ T(\mathbf{U}_k^* \mathbf{Z} \mathbf{V}_k)\right] \mathbf{V}_k^*, \tag{64}$$

for any $\mathbf{Z} \in \mathbb{R}^{d_1 \times d_2}$, where $S : \mathbb{R}^{d \times d} \to \mathbb{R}^{d \times d}$ and $T : \mathbb{R}^{d \times d} \to \mathbb{R}^{d \times d}$ are the *symmetrization operator* and *antisymmetrization operator*, respectively, that map any $\mathbf{X} \in \mathbb{R}^{d \times d}$ to

$$S(\mathbf{X}) = \frac{1}{2}(\mathbf{X} + \mathbf{X}^*), \qquad \text{and} \qquad T(\mathbf{X}) = \frac{1}{2}(\mathbf{X} - \mathbf{X}^*)$$

for any $\mathbf{X} \in \mathbb{R}^{d \times d}$, and $\mathbf{M}^{\mathrm{S}}, \mathbf{M}^{\mathrm{T}} \in \mathbb{R}^{d_1 \times d_2}$ fulfill

$$\mathbf{M}^{\mathrm{S}} = \left[\begin{array}{c|c} -\mathbf{H}^{(k)} & \mathbf{M}_{1,2}^{-} \\ \hline \mathbf{M}_{2,1}^{-} & \epsilon_k^{-2}\mathbf{1} \end{array}\right] \qquad \mathbf{M}^{\mathrm{T}} = \left[\begin{array}{c|c} -\mathbf{H}^{(k)} & \mathbf{M}_{1,2}^{+} \\ \hline \mathbf{M}_{2,1}^{+} & \epsilon_k^{-2}\mathbf{1} \end{array}\right]$$

with $\mathbf{H}^{(k)} \in \mathbb{R}^{r_k \times r_k}$ as in (13) and the $(d_1 - r_k) \times (d_2 - r_k)$-matrix of ones $\mathbf{1}$. Furthermore, the matrices $\mathbf{M}_{1,2}^{-}, \mathbf{M}_{1,2}^{+} \in (d_1 - r_k) \times r_k$ are such that

$$\big(\mathbf{M}_{1,2}^{\pm}\big)_{ij} = \frac{(\sigma_i^{(k)})^{-1} \pm \sigma_{j+r_k}^{(k)} \epsilon_k^{-2}}{\sigma_i^{(k)} \pm \sigma_{j+r_k}^{(k)}}$$

for $i \in [r_k]$, $j \in [d_2 - r_k]$ and

$$\big(\mathbf{M}_{2,1}^{\pm}\big)_{ij} = \frac{(\sigma_j^{(k)})^{-1} \pm \sigma_{i+r_k}^{(k)} \epsilon_k^{-2}}{\sigma_j^{(k)} \pm \sigma_{i+r_k}^{(k)}}$$

for$j \in [r_k]$, $i \in [d_1 - r_k]$. The formula (64) for $\nabla^2 F_{\epsilon_k}(\mathbf{X}^{(k)})$ follows by inserting the operator ∇F_{ϵ_k} into Theorem 2.2.6 of (Yang, 2009), Corollary 3.10 (Noferini, 2017) or Theorem 4 of (Ding et al., 2018).

By realizing that $0 \leq \sigma_\ell^{(k)} \leq \epsilon_k$ for all $\ell > r_k$, we see that

$$\frac{1}{(\sigma_i^{(k)})^2} \leq \big(\mathbf{M}_{1,2}^{+}\big)_{ij} = \big(\mathbf{M}_{2,1}^{+}\big)_{ji} \leq \frac{1}{\sigma_i^{(k)} \epsilon_k}$$

and

$$-\frac{1}{\sigma_i^{(k)} \epsilon_k} \leq \big(\mathbf{M}_{1,2}^{-}\big)_{ij} = \big(\mathbf{M}_{2,1}^{-}\big)_{ji} \leq \frac{1}{(\sigma_i^{(k)})^2}$$

for all i and j.

Now, comparing M^{S} and M^{T} with $\mathbf{H}_k$, see (49), of the weight operator $W^{(k)}$, we see that the upper left blocks of M^{S} and M^{T} are just the *negative* of the upper left block $\mathbf{H}^{(k)}$ of $\mathbf{H}_k$, while the lower right blocks coincide. Furthermore, the lower left and the upper right blocks are related such that

$$\left|\big(\mathbf{M}_{1,2}^{\pm}\big)_{ij}\right| \leq \frac{1}{\sigma_i^{(k)} \epsilon_k} = (\mathbf{H}_{1,2}^{(k)})_{ij}$$

for all $i \in [r_k]$, $j \in [d_2 - r_k]$, and

$$\left|\left(\mathbf{M}^{\pm}_{2,1}\right)_{ij}\right| \leq \frac{1}{\sigma_j^{(k)}\epsilon_k} = (\mathbf{H}^{(k)}_{2,1})_{ij}$$

for all $i \in [d_2 - r_k]$, $j \in [r_k]$.

We now point out the relationship of these considerations to an analysis that was performed in (Paternain et al., 2019) for the case of an *unconstrained minimization* of F_{ϵ_k}, assuming furthermore that F_{ϵ_k} was smooth:

In this case, (Paternain et al., 2019) considers using *modified Newton steps*

$$\mathbf{X}^{(k+1)} := \mathbf{X}^{(k)} - \eta_k \left|\nabla^2 F_{\epsilon_k}(\mathbf{X}^{(k)})\right|_c^{-1} \nabla F_{\epsilon_k}(\mathbf{X}^{(k)})$$

where the Hessian $\nabla^2 F_{\epsilon_k}(\mathbf{X}^{(k)})$ is replaced by a positive definite truncated eigenvalue matrix $\left|\nabla^2 F_{\epsilon_k}(\mathbf{X}^{(k)})\right|_c$, which replaces the large negative eigenvalues of $\nabla^2 F_{\epsilon_k}(\mathbf{X}^{(k)})$ by their modulus for eigenvalues that have large modulus and eigenvalues of small modulus by an appropriate constant c. (Paternain et al., 2019) shows that such steps are, unlike conventional Newton steps, which often are *attracted by saddle points*, able to *escape* saddle points with an exponential rate that does *not* depend on the conditioning of the problem. Experimental observations of such behavior has been reported also in other works (Murray, 2010; Dauphin et al., 2014).

In view of this, we observe that the weight operator $W^{(k)}$ is nothing but a refined variant of $\left|\nabla^2 F_{\epsilon_k}(\mathbf{X}^{(k)})\right|_c$, as the eigenvalues of $\nabla^2 F_{\epsilon_k}(\mathbf{X}^{(k)})$ from (64) are simply $\{(\mathbf{M}^{\mathrm{S}}_{ij}, i \leq j\} \cup \{(\mathbf{M}^{\mathrm{T}}_{ij}, i < j\}$, c.f., e.g., Theorem 4.5 of (Noferini, 2017). In particular, the refinement is such that the small eigenvalues of $\nabla^2 F_{\epsilon_k}(\mathbf{X}^{(k)})$, which can be found in the entries of $\mathbf{M}^{\pm}_{1,2}$ and $\mathbf{M}^{\pm}_{2,1}$, are replaced not by a uniform constant, but by *different* upper bounds $(\sigma_i^{(k)}\epsilon_k)^{-1}$ and $(\sigma_j^{(k)}\epsilon_k)^{-1}$ that depend either on the row index i or the column index j.

Besides this connection, there are important differences of our algorithm to the algorithm analyzed in (Paternain et al., 2019). While that paper considers the minimization of a fixed smooth function, we update the smoothing parameter ϵ_k and thus the function F_{ϵ_k} at each iteration. Furthermore, Algorithm 1 of (Paternain et al., 2019) uses backtracking for each modified Newton step, which would be prohibitive to perform as evaluations of F_{ϵ_k} are very expensive for our smoothed log-det objectives, as they would require the calculation of all singular values. On the other hand, `MatrixIRLS` uses full modified Newton steps, and we can assure that these are always a descent direction in our case, as we explain in an upcoming paper. Lastly, we do not add noise to the iterates.

As mentioned in Section 2, `MatrixIRLS` is by no means the *first* algorithm for low-rank matrix recovery that can be considered as an iteratively reweighted least squares algorithm. However, the IRLS algorithms (Fornasier et al., 2011; Mohan & Fazel, 2012; Lai et al., 2013; Kümmerle & Sigl, 2018) are different from `MatrixIRLS` not only in their computational aspects, but also since they do *not* allow for a close relationship between their weight operator $W^{(k)}$ and the Hessian $\nabla^2 F_{\epsilon_k}(\mathbf{X}^{(k)})$ at $\mathbf{X}^{(k)}$ as described above.

E. Experimental Details

In this section, we specify some details of the setup and the algorithmic parameters for the experiments presented in Section 5. The sample complexity experiments of Figures 1, 2, 5 and 6 were conducted on a Linux node with Intel Xeon E5-2690 v3 CPU with 28 cores and 64 GB RAM, using MATLAB R2019a. The experiment of Figure 4 was conducted on a Windows 10 laptop with Intel i7 7660U with 2 cores and 8 GB RAM, also using MATLAB R2019a, and all other experiments were conducted on a iMac with 4 GHz Quad-Core Intel Core i7 CPU, using MATLAB R2020b. In the main text we divided the algorithms into three main categories according to the main optimization philosophy behind. But now, for the purpose of our experiment, we categorize the algorithms into algorithms of *first-order* type and of *second-order* type based on whether an algorithm exhibits empirically observed locally superlinear convergence rates or not.

E.1. Algorithmic Parameter Choice

All the methods are provided with the true rank r of $\mathbf{X}^0$ as an input parameter. If possible, we use the MATLAB implementation provided by the authors of the respective papers. Below we point out the links from which one can download such implementations. We do not make use of explicit parallelization for any of the methods, but most methods use complied

C subroutines to efficiently implement sparse evaluations of matrix factorizations. We base our choice of algorithms on the desire to obtain a representative picture of state-of-the-art algorithms for matrix completion, including in particular those that are scalable to problems with dimensionality in the thousands or more, those that come with the best theoretical guarantees, and those that claim to perform particularly well to complete *ill-conditioned* matrices.

We set a maximal number of outer iterations for the second-order methods as $N_0 = 400$. The second-order type algorithms considered for this paper, including their parameter choices, are:

- `MatrixIRLS`, as described in Algorithm 1 or, minutely, in Appendix A.1. As a stopping criterion, we choose a threshold of 10^{-9} for the relative change of the Frobenius norm $\frac{\|\mathbf{X}^{(k+1)}-\mathbf{X}^{(k)}\|_F}{\|\mathbf{X}^{(k)}\|_F}$. We use the CG method for solving the linear system (14) without any preconditioning. We terminate the CG method if a maximum number of $N_{\text{CG\_inner}} = 500$ inner iterations is reached or if a relative residual of $\text{tol}_{\text{inner}} = 10^{-9}$ is reached, whichever happens first.[4] For the weight operator update step, we use a variant of the randomized Block Krylov method (Musco & Musco, 2015) based on the implementation provided by the authors[5], setting the parameter for the maximal number of iterations to 20.
- `R2RILS` (Bauch et al., 2021) or *rank* $2r$ *iterative least squares*, a method that optimizes a least squares data fit objective $\|P_\Omega(\mathbf{X}_0) - P_\Omega(\mathbf{X})\|_F$ over $\mathbf{X} \in T_{\mathbf{Z}^{(k)}}\mathcal{M}_r$, where $T_{\mathbf{Z}^{(k)}}\mathcal{M}_r$ is a tangent space onto the manifold of rank-r matrices, while iteratively updating this tangent space. As above, we stop the outer iterations a threshold of 10^{-9} is reached for the relative change of the Frobenius norm $\frac{\|\mathbf{X}^{(k+1)}-\mathbf{X}^{(k)}\|_F}{\|\mathbf{X}^{(k)}\|_F}$. At each outer iteration, `R2RILS` solves an overdetermined least squares problem of size $(m \times r(d_1 + d_2))$ via the iterative solver LSQR, for which we choose the maximal number of inner iterations as $N_{\text{LSQR\_inner}} = 500$ and a termination criterion based on a relative residual of 10^{-10}. We use the implementation based on the code provided by the authors, but adapted for these stopping criteria.[6]
- `RTRMC`, the preconditioned Riemannian trust-region method called RTRMC 2p of (Boumal & Absil, 2015), which was reported to achieve the best performance among a variety of matrix completion algorithms for the task of completing matrices of a condition number of up to $\kappa = 150$. We use the implementation provided by the authors[7] with default options except from setting the maximal number of inner iterations to $N_{\text{inner}} = 500$ and setting the parameter for the tolerance on the gradient norm to 10^{-15}. Furthermore, as the algorithm otherwise would often run into certain submatrices that are not positive definite for ρ between 1 and 1.5, we set the regularization parameter $\lambda = 10^{-8}$, which is small enough not to deter high precision approximations of $\mathbf{X}^0$ if enough samples are provided.

Furthermore, we consider the following first-order algorithms, setting the maximal number of outer iterations to $N_0 = 4000$:

- `LRGeomCG` (Vandereycken, 2013), a local optimization method for a quadratic data fit term based on gradients with respect to the Riemannian manifold of fixed rank matrices. We use the author's implementation[8] while setting the parameters related to the stopping conditions `abs_grad_tol`, `rel_grad_tol`, `abs_f_tol`, `rel_f_tol`, `rel_tol_change_x` and `rel_tol_change_res` each to 10^{-9}. The rank-adaptive variant of `LRGeomCG`, called `LRGeomCG Pursuit` (Uschmajew & Vandereycken, 2015; Tan et al., 2014), is used with the same algorithmic parameters as `LRGeomCG` for the inner iterations, and with a rank increase of 1 each outer iteration.
- `LMaFit` or low-rank matrix fitting (Wen et al., 2012), a nonlinear successive over-relaxation algorithm based on matrix factorization. We use the implementation provided by the authors[9], setting the tolerance threshold for the stopping condition (which is based on a relative data fit error $\|P_\Omega(\mathbf{X}^{(k)}) - \mathbf{y}\|_2/\|\mathbf{y}\|_2$) to $5 \cdot 10^{-10}$.
- `ScaledASD` or scaled alternating steepest descent (Tanner & Wei, 2016), a gradient descent method based on matrix factorization which scales the gradients in a quasi-Newton fashion. We use the implementation provided by the authors[10] with the stopping condition of $\|P_\Omega(\mathbf{X}^{(k)}) - \mathbf{y}\|_2/\|\mathbf{y}\|_2 \leq 10^{-9}$.

[4] While this stopping condition uses the condition number κ, which will probably be unknown in practice, it can be generally chosen independently of κ without any problems of convergence.

[5] https://github.com/cpmusco/bksvd

[6] https://github.com/Jonathan-WIS/R2RILS

[7] RTRMC v3.2 from http://web.math.princeton.edu/~nboumal/RTRMC/index.html, together with the toolbox Manopt 6.0 (https://www.manopt.org/) (Boumal et al., 2014).

[8] http://www.unige.ch/math/vandereycken/matrix_completion.html

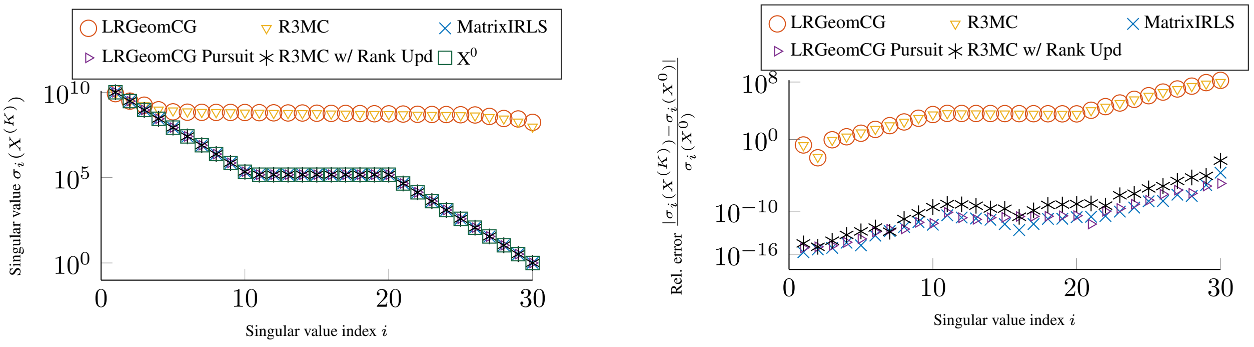
[9] http://lmafit.blogs.rice.edu

[10] http://www.sdspeople.fudan.edu.cn/weike/code/mc20140528.tar

- ScaledGD or scaled gradient descent (Tong et al., 2020), a method that is very similar to ScaledASD, but for which a non-asymptotic local convergence analysis has been achieved for the case of a matrix recovery problem related to matrix completion, and which has been investigated experimentally in (Tong et al., 2020) in the light of the completion of ill-conditioned low-rank matrices. We use an adapted version of the author's implementation[11]: We choose a step size of $\eta = 0.5$, but increase the normalization parameter p by a factor of 1.5 in case the unmodified algorithm ScaledGD leads to divergent algorithmic iterates, using the same stopping condition as for ScaledASD.
- NIHT or normalized iterative hard thresholding (Tanner & Wei, 2013), which performs iterative hard thresholding steps with adaptive step sizes. We use the implementation provided by the authors [10] with a stopping threshold of 10^{-9} for the relative data fit error $\|P_\Omega(\mathbf{X}^{(k)}) - \mathbf{y}\|_2/\|\mathbf{y}\|_2$ and the convergence rate threshold parameter $1 - 10^{-9}$.
- R3MC (Mishra & Sepulchre, 2014), a Riemannian nonlinear conjugate-gradient that also optimizes a least squares data fit objective $\|P_\Omega(\mathbf{X}_0) - P_\Omega(\mathbf{X})\|_F$ by exploiting a three-factor matrix factorization similar to the SVD and performs a search on a quotient manifold defined from the manifold of rank r matrices, this factorization and symmetries from the action of the orthogonal group. We use the author's implementation[12] while choosing the Polyak-Ribier rule for the nonlinear CG and the Armijo line search with a maximum of 50 line searches allowed at each iteration. Also, we set the tolerance parameter for stopping criterion to 10^{-9}. R3MC w/ Rank Upd corresponds to the method described in the section on *rank updating* of (Mishra & Sepulchre, 2014).

E.2. Remark to Experiment of Figure 7

Tracking the relative Frobenius error to gauge the performance of methods for the recovery of highly ill-conditioned matrices without taking account the condition number κ might not provide a full picture, as a recovery of the singular spaces corresponding to the smallest singular values can only be expected once the relative error is smaller than $1/\kappa$.



(a) Spectrum of output matrices $\mathbf{X}^{(K)}$ and $\mathbf{X}^0$

(b) Relative errors $\frac{|\sigma_i(\mathbf{X}^{(K)})-\sigma_i(\mathbf{X}^0)|}{\sigma_i(\mathbf{X}^0)}$

Figure 8. Spectrum information of algorithmic output $\mathbf{X}^{(K)}$ after convergence, experiment of Figure 7 (1000×1000 matrix, $r = 30$, $\kappa = 10^{10}$, oversampling factor $\rho = 1.5$)

For this reason, we report in Figure 8a the singular values of the recovered matrices $\mathbf{X}^{(K)}$ and the relative error on a basis of individual singular values in Figure 8b for the very experiment conducted in Section 5.3 and illustrated in Figure 7. We observe that MatrixIRLS, LRGeomCG Pursuit and R3MC w/ Rank Upd each are able to recover even the smallest singular values such (with indices $i = 28, 29, 30$) with a high precision of a relative error between 10^{-7} and 10^{-3}.

This shows that despite a not too restrictive choice of the tolerance on of the inner conjugate gradient iterations (such as $\text{tol}_{\text{inner}} = 10^{-3}$), MatrixIRLS is successful in recovering the complete spectrum of $\mathbf{X}^0$, indicating that an implementation of MatrixIRLS that solves (14) via conjugate gradient method together with weight updates based on a randomized block Krylov method can be very precise even without requiring an very high precision on the iterative linear system solver.

These observations suggest that MatrixIRLS and Riemannian optimization methods with adaptive rank updates such as LRGeomCG Pursuit and R3MC w/ Rank Upd are good alternatives to solve hard matrix recovery problems in a

[11] https://github.com/Titan-Tong/ScaledGD

[12] https://bamdevmishra.in/codes/r3mc/. We used the version from Sep. 2020 which already includes the rank updating strategy.

numerically efficient way, warranting further investigations for a better theoretical understanding.