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

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

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

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

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.

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 $\mathrm{rank}(\mathbf{X})$. 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 $\mathrm{rank}(\mathbf{X})$ 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

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

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 $\mathbf{X} = \mathbf{U}\,\mathrm{dg}\,(\sigma(\mathbf{X}))\mathbf{V}^* = \mathbf{U}\,\mathrm{dg}\,(\sigma)\mathbf{V}^*$. 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*

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

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 $(\mathbf{H}_k)_{ij} := \left(\max(\sigma_i^{(k)},\epsilon_k)\max(\sigma_j^{(k)},\epsilon_k)\right)^{-1}$ 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.

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.

Algorithm 1 `MatrixIRLS` for low-rank matrix completion

Input: Set Ω, 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

$$\mathbf{X}^{(k)} = \operatorname*{arg\,min}_{\mathbf{X}:P_\Omega(\mathbf{X})=\mathbf{y}} \langle \mathbf{X}, W^{(k-1)}(\mathbf{X})\rangle. \tag{5}$$

Update smoothing: Compute $\widetilde{r}+1$-th singular value of $\mathbf{X}^{(k)}$ to update

$$\epsilon_k = \min\left(\epsilon_{k-1}, \sigma_{\widetilde{r}+1}(\mathbf{X}^{(k)})\right). \tag{6}$$

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: $\mathbf{X}^{(K)}$.

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 $\mathbf{X}^{(k)}$, $T_k = \left\{ \begin{bmatrix} \mathbf{U}^{(k)} \mathbf{U}_\perp^{(k)} \end{bmatrix} \begin{bmatrix} \mathbb{R}^{r_k \times r_k} & \mathbb{R}^{r_k (d_2 - r_k)} \\ \mathbb{R}^{(d_1 - r_k) r_k} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{V}^{(k)} \mathbf{V}_\perp^{(k)} \end{bmatrix}^* \right\}$.

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 $\sigma_i^{(k)} \leq \epsilon_k$ 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 $\mathbf{X} = \mathbf{U} \operatorname{dg}(\sigma) \mathbf{V}^*$, $\mathbf{U} \in \mathbb{R}^{d_1 \times r}$, $\mathbf{V} \in \mathbb{R}^{d_2 \times r}$, is* μ_0-incoherent *if there exists a constant $\mu_0 \geq 1$ such that*

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

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 $\|\mathbf{X}^{(k+1)} - \mathbf{X}^0\|_{S_\infty} \leq \min(\mu \|\mathbf{X}^{(k)} - \mathbf{X}^0\|_{S_\infty}^2, \|\mathbf{X}^{(k)} - \mathbf{X}^0\|_{S_\infty})$ with $\mu \leq \frac{d \log(D)}{\mu_0 \sigma_r(\mathbf{X}^0)} \kappa$, 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 $\Omega^c = [d_1]\times[d_2]\setminus\Omega$ 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

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

where $\operatorname{prox}^{A_k}_F(\mathbf{X}) := \arg\min_{\mathbf{Z} \in \mathbb{R}^{d_1 \times d_2}} F(\mathbf{Z}) + \frac{1}{2} d_{A_k}(\mathbf{X}, \mathbf{Z})^2$ 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

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

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 orthonor-

mal 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.

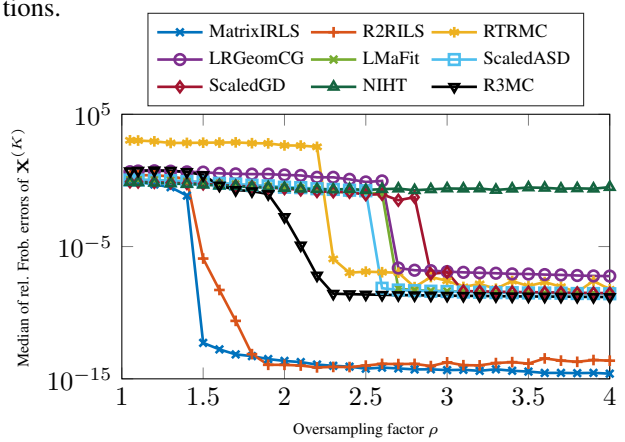

Figure 1. Performance of matrix completion algorithms for 1000×1000 matrices of rank $r = 5$ with condition number $\kappa = 10$, given $m = \lfloor \rho r(d_1 + d_2 - r)\rfloor$ random samples. Median of Frobenius errors $\|\mathbf{X}^{(K)} - \mathbf{X}^0\|_F/\|\mathbf{X}^0\|_F$ of 100 independent realizations.

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.

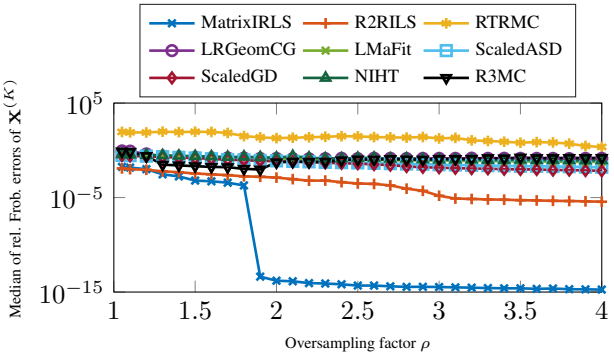

Figure 2. Performance of matrix completion algorithms as in Figure 1, but with $\kappa = 10^5$. Median of 50 realizations.

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 $\text{rank} = 10$ 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.

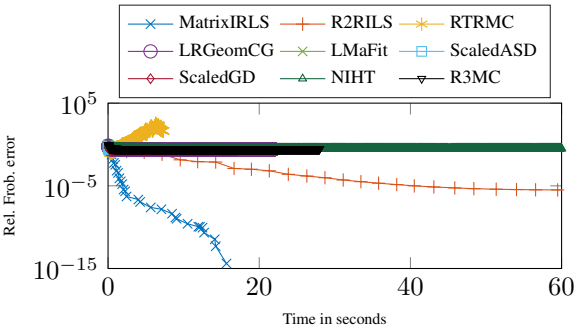

Figure 3. Completion task for a highly ill-conditioned 1000×1000 matrix of rank $r = 10$ with $\kappa = 10^{10}$ ($\rho = 4$).

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 $\kappa \gg 10^2$.

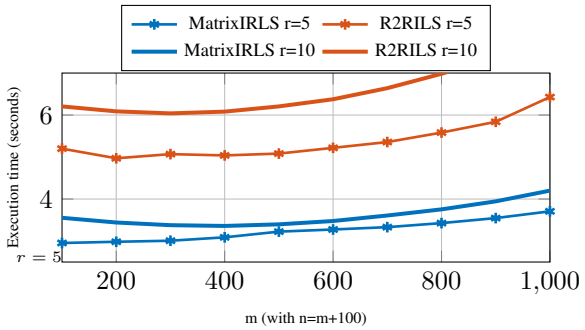

Figure 4. Execution time of `R2RILS` and `MatrixIRLS` for completion of rank $r \in \{5, 10\}$ matrices of size $m \times (m + 100)$ and condition number $\kappa = 10^2$, averaged across 50 independent realizations.

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.

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

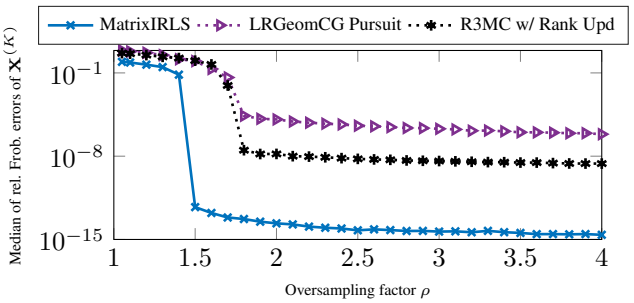

Figure 5. Completion of 1000×1000 matrices of rank $r = 5$ with condition number $\kappa = 10$, experiment as in Figure 1.

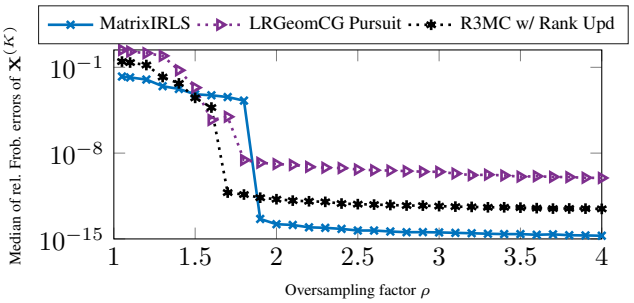

Figure 6. Completion of 1000×1000 matrices of rank $r = 5$ with condition number $\kappa = 10^5$, experiment as in Figure 2.

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

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 $\rho = 1.5$.

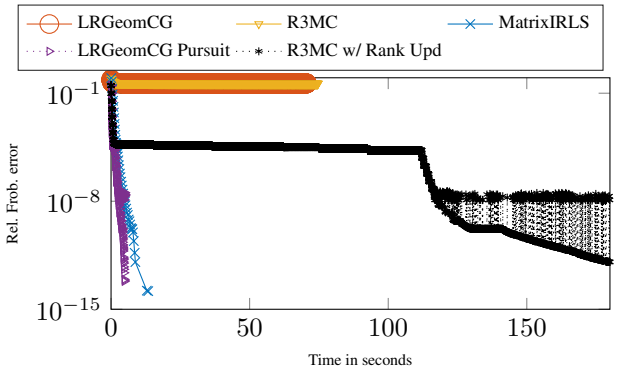

Figure 7. Comparison of matrix completion algorithms for 1000×1000 matrices of rank $r = 30$ with condition number $\kappa = 10^{10}$ and 10 equal singular values, oversampling factor of $\rho = 1.5$.

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.