DESINGULARIZATION OF BOUNDED-RANK MATRIX SETS

VALENTIN KHRULKOV* AND IVAN OSELEDETS*†

Abstract. Conventional ways to solve optimization problems on low-rank matrix sets which appear in great number of applications ignore its underlying structure of an algebraic variety and existence of singular points. This leads to appearance of inverses of singular values in algorithms and since they could be close to 0 it causes certain problems. We tackle this problem by utilizing ideas from the algebraic geometry and show how to desingularize these sets. Our main result is algorithm which uses only bounded functions of singular values and hence does not suffer from the issue described above.

Key words. low-rank matrices, algebraic geometry, optimization

AMS subject classifications. 65F30

1. Introduction.

Although low-rank matrices appear in many applications, the structure of the corresponding matrix variety[1] is not fully utilized in the computations, and theoretical investigation is complicated by the existence of the singular points on such manifold, which correspond to matrices of smaller rank. We tackle this problem by utilizing the modified Room-Kempf desingularization [2] of determinantal varieties that is classical in algebraic geometry, but has never been applied in the context of optimization over matrix manifolds. Our main contributions are

- We propose and analyze a modified Room-Kempf desingularization technique for the variety of matrices with rank bounded by r (subsection 2.2)
- We obtain the bounds on the curvature (subsection 2.2)
- We find effective low-dimensional parametrization of the tangent space (subsection 2.3)
- We implement effective realization of reduced Hessian method for the optimization over desingularized variety (section 3)
- We also briefly discuss few technical details in the implementation of the algorithm (section 4)

1.1. Idea of desingularization.

Before describing desingularization of bounded rank matrix sets we will introduce it's basic idea first. The low-rank matrix case will be described in next section. Let V be some variety (not neccessarily smooth) and f be some function

$$f : V \to \mathbb{R}.$$

To solve

$$f(x) \to \min, x \in V,$$

we often use methods involving tangent bundle of V. However due to the existence of the singular points where tangent space is not well-defined it is hard to prove correctness and convergence of those methods. To avoid this problem we construct a smooth variety $\widehat{V}$ and a surjective map π

$$\pi : \widehat{V} \to V.$$

*Skolkovo Institute of Science and Technology 143025, Nobel St. 3, Skolkovo Innovation Center Moscow, Russia.

†Institute of Numerical Mathematics of the Russian Academy of Sciences 119333, Gubkina St. 8 Moscow, Russia .

[1]We more or less freely use the word *variety* in the text. It should be understood as a zero set of a polynomial system in some vector space $\mathbb{R}^n$ and which is not neccessarily smooth.

Let $\widehat{f}$ be a pullback of f via map π i.e.

$$\widehat{f} : \widehat{V} \to \mathbb{R},$$

$$\widehat{f} = \pi \circ f.$$

It is obvious that

$$\min_{x \in V} f(x) = \min_{y \in \widehat{V}} \widehat{f}(y),$$

so we reduced our non-smooth minimization problem to a smooth one. Typically $\widehat{V}$ is a variety in a space of bigger dimension and is constructed to be of the same dimension as the smooth part of V. To have some geometrical idea one can think about the following example (see Figure 1). Let V be a cubic curve given by the following equation

$$y^2 = x^2(x+1),$$

and parametrized as

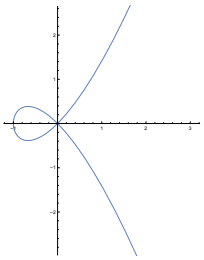
$$(x(t), y(t)) = (t^2 - 1, t(t^2 - 1)).$$

It easy to see that $(0, 0)$ is a singular point of V. Then its desingularization is given by

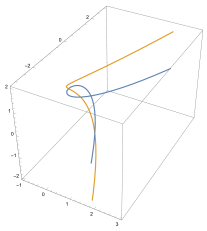
$$\widehat{V} = (x(t), y(t), z(t)) = (t^2 - 1, t(t^2 - 1), t) \subset \mathbb{R}^3,$$

which is clearly smooth. Projection is then just

$$\pi : (x(t), y(t), z(t)) = (x(t), y(t)).$$



(a) V



(b) Blue - V, Orange - $\widehat{V}$

Fig. 1: Desingularization of the cubic

2. Desingularization of low-rank matrix varieties via kernel.

2.1. 2×2 **matrices.** Let V be a variety of 2×2 matrices with rank ≤ 1. We have

$$V = \{x_{11}x_{22} - x_{12}x_{21} = 0, (x_{11}, x_{21}, x_{12}, x_{22}) \in \mathbb{R}^4\}, \tag{1}$$

so it is indeed an algebraic variety. In order to analyze its smoothness and compute the tangent space we recall the following result. Given the set M defined implicitly as

$$M = \{f_1(x) = 0, f_2(x) = 0 \dots f_k(x) = 0, x \in R^l\}.$$

and some point $p \in M$ we construct the matrix $N(p)$,

$$N(p) = \begin{bmatrix} \nabla f_1(p) \\ \nabla f_2(p) \\ \vdots \\ \nabla f_k(p) \end{bmatrix}.$$

Then the point p is called notsingular if $N(p)$ has full rank at p. In this case, the tangent space is defined as

$$T_p M = \{v \in \mathbb{R}^l, N(p)v = 0\}.$$

Applying this to V defined in (1) we obtain

$$N(x_{11}, x_{21}, x_{12}, x_{22}) = \begin{bmatrix} x_{22} & -x_{12} & -x_{21} & x_{11} \end{bmatrix},$$

and then $(0, 0, 0, 0)$ is a singular point of V.

We desingularize it by considering $\widetilde{V}$ which is defined as the set of pairs $(A, Y) \in \mathbb{R}^{2\times 2} \times \mathbb{R}^2$ with coordinates[2]

$$A = \begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{bmatrix},$$

and

$$Y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix},$$

satisfying

$$AY = 0,$$

and

$$Y^\top Y = 1.$$

More explicitly

$$\widetilde{V} = \{p : (x_{11}y_1 + x_{12}y_2 = 0, x_{21}y_1 + x_{22}y_2 = 0, y_1^2 + y_2^2 = 1)\},$$

$$p = (x_{11}, x_{21}, x_{12}, x_{22}, y_1, y_2) \in \mathbb{R}^6.$$

We find that the normal space at p is spanned by rows of the following matrix $N(p)$:

$$N(p) = \begin{bmatrix} y_1 & 0 & y_2 & 0 & x_{11} & x_{12} \\ 0 & y_1 & 0 & y_2 & x_{21} & x_{22} \\ 0 & 0 & 0 & 0 & 2y_1 & 2y_2 \end{bmatrix}. \tag{2}$$

Let us also denote $p_A = (x_{11}, x_{21}, x_{12}, x_{22})$ and $p_Y = (y_1, y_2)$. Since $y_1^2 + y_2^2 = 1$ the matrix (2) clearly has rank 3 at any point of $\widetilde{V}$ which proves that $\widetilde{V}$ is smooth. Projection π is just

$$\pi : (x_{11}, x_{21}, x_{12}, x_{22}, y_1, y_2) \to (x_{11}, x_{21}, x_{12}, x_{22})$$

whose image is the entire V. However we would also like to estimate how close are tangent spaces at close points. Recall that by the definition of Grassmanian metric the distance between planes C and D is given by

$$\|C - D\|_{Gr} := \|P_C - P_D\|,$$

[2]We assume Fortran order so A is vectorized as $(x_{11}, x_{21}, x_{12}, x_{22})$

where P_C and P_D are orthogonal projectors on corresponding planes. Obviously, then the distance between any two planes is equal to the distance between their orthogonal complements.
It is well known that the projection on the plane spanned by the rows of a matrix M is given by $M^\dagger M$, where $M^\dagger$ is a pseudoinverse defined as

$$M^\dagger = M^\top (MM^\top)^{-1}.$$

Hence for two different points a and b on the desingularized manifold we obtain

$$\|P_{N(a)} - P_{N(b)}\| = \|N(a)^\dagger N(a) - N(b)^\dagger N(b)\|.$$

We also have the following result [1]:

$$\|N(a)^\dagger N(a) - N(b)^\dagger N(b)\| \leq \max\{\|N(a)^\dagger\|, \|N(b)^\dagger\|\}\|N(a) - N(b)\|. \tag{3}$$

In order to estimate the smoothness we need to estimate how P_{N_p} changes under small changes of p. It is sufficient to estimate the gradient of P. Thus, we have to uniformly bound $\|N^\dagger\|$ from above, which is equivalent to bounding $\sigma_{\min}(N)$ from below. Taking into account defining equations of desingularized manifold we find that

$$N(p)N(p)^\top = \begin{bmatrix} 1 + x_{11}^2 + x_{12}^2 & x_{11}x_{21} + x_{12}x_{22} & 0 \\ x_{11}x_{21} + x_{12}x_{22} & 1 + x_{21}^2 + x_{22}^2 & 0 \\ 0 & 0 & 4 \end{bmatrix}. \tag{4}$$

Hence $\sigma_{\min}^2(N(a)) \geq 1$ and $\|N(a)^\dagger\| \leq 1$. From definition of $N(p)$ it follows that

$$\|N(a) - N(b)\| \leq 4\|a_Y - b_Y\| + \|a_A - b_A\|.$$

We will derive and prove similar estimates for the general case in the next section.

2.2. General construction and estimation of curvature. Consider a variety $\mathcal{M}_{\leq r}$ of $n \times m$, $n \geq m$ of matrices of rank not higher than r,

$$\mathcal{M}_{\leq r} = \{A \in \mathbb{R}^{n\times m} : \operatorname{rank}(A) \leq r\}.$$

Note that $\mathcal{M}_{\leq r}$ is a variety of dimension $(n+m)r - r^2$. The following classical result is given for the sake of completeness.

LEMMA 1. *If the matrix $A \in \mathcal{M}_{\leq r}$ has rank smaller than r, A is a singular point on this variety.*

Proof. Indeed, let A be an arbitrary matrix in $\mathbb{R}^{n\times m}$.

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & a_{2m} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nm} \end{bmatrix}.$$

Then

$$A \in \mathcal{M}_{\leq r},$$

if and only if

$$\det A[i_1, i_2, \dots, i_{r+1}; j_1, j_2, \dots, j_{r+1}] = 0, \tag{5}$$

$$\forall \quad 1 \le i_1 < i_2 < \cdots < i_{r+1} \le n, 1 \le j_1 < j_2 < \cdots < j_{r+1} \le m,$$

and

$$A[i_1, i_2, \ldots, i_{r+1}; j_1, j_2, \ldots, j_{r+1}],$$

is a submatrix

$$\begin{bmatrix} a_{i_1 j_1} & a_{i_1 j_2} & \ldots & a_{i_1 j_{r+1}} \\ a_{i_2 j_1} & a_{i_2 j_2} & \ldots & a_{i_2 j_{r+1}} \\ \vdots & \vdots & & \vdots \\ a_{i_{r+1} j_1} & a_{i_{r+1} j_2} & \ldots & a_{i_{r+1} j_{r+1}} \end{bmatrix},$$

of the matrix A.
Now suppose $\operatorname{rank} A = k < r$. This implies

$$\det A[i_1, i_2, \ldots, i_l; j_1, j_2, \ldots, j_l] = 0, \tag{6}$$

$$\forall l \ge k+1,$$

in particular for $l = r$. Moreover, for any matrix C_{ij} of size $n \times n$ we have the identity

$$\frac{\partial \det C}{\partial C_{ij}} = (-1)^{i+j} \det C[1 \ldots \hat{i} \ldots n; 1 \ldots \hat{j} \ldots n] \tag{7}$$

where $\hat{i}$ means that this index is omitted.
Using (6) and (7) we conclude that the Jacobian of (5) vanishes at A so it is indeed a singular point. □

Now we return to the main topic of the paper. For $\mathcal{M}_{\le r}$ we propose the following desingularization. Given a matrix $A \in \mathcal{M}_{\le r}$ we take all the elements

$$Y \in Gr(m-r, m),$$

such that

$$AY = 0, \tag{8}$$

and define a new manifold $\widehat{\mathcal{M}}_r$ as the set of pairs

$$(A, Y) \in \mathbb{R}^{n,m} \times Gr(m-r, m).$$

satisfying (8). However in order to do computations it is more convenient to choose representative in each element of the Grassmanian, and it makes sense to choose an orthonormal frame. This way also requires fixing ambiguity in such choice and to do that we will also introduce gauge condition. So henceforth we assume the following definition of $\widehat{\mathcal{M}}_r$:

$$\widehat{\mathcal{M}}_r = (A, Y) \in \mathbb{R}^{n \times m} \times R^{m \times (m-r)}$$

$$AY = 0, \quad (Y^\top Y)_{ij} = \delta_{ij}, \quad i = 1, \ldots, m-r, \quad j = 1, \ldots, i \tag{9}$$

It is still a $(n+m)r - r^2$ -dimensional variety.

Lemma 2. *If $e = (A, Y) \in \widehat{\mathcal{M}}_r$ then $A \in \mathcal{M}_{\le r}$.*

Proof. Follows directly from the fact that $AY = 0$. ∎

The main result of this paper is the following.

THEOREM 3. *Variety $\widehat{\mathcal{M}}_r$ is smooth.*

Proof. In order to prove the smoothness of the variety, we have to study the tangent space. Recall from subsection 2.1 that for a given $e = (A, Y)$ the tangent space is the nullspace of the gradient of the constraints:

$$\mathcal{T}_e\widehat{\mathcal{M}}_r = \{v : N(e)v = 0\},$$

where the gradient matrix $N(e)$ has the following block structure:

$$N(e) = \begin{bmatrix} Y^\top \otimes I_n & I_{m-r} \otimes A \\ 0 & I_{m-r} \otimes Y^\top \end{bmatrix}. \tag{10}$$

Here we used gauge condition[3]

$$(\delta Y)^\top Y = 0,$$

$$\forall (\delta A, \delta Y) \in \mathcal{T}_e\widehat{\mathcal{M}}_r.$$

For simplicity of notation we will omit e in $N(e)$. The projection onto the tangent space of a given vector z is given by the following formula (similar to 2×2 case)

$$v = (I - N^\top (NN^\top)^{-1} N)z = P(e)z, \tag{11}$$

where

$$P = (I - N^\dagger N),$$

is the orthogonal projector onto the row range of N. Using exactly the same idea as in previous section we estimate $\sigma_{\min}(N)$ from below. Consider the Gram matrix

$$Z = NN^\top = \begin{bmatrix} Y^\top \otimes I_n & I_{m-r} \otimes A \\ 0 & I_{m-r} \otimes Y^\top \end{bmatrix} \begin{bmatrix} Y \otimes I_n & 0 \\ I_{m-r} \otimes A^\top & I_{m-r} \otimes Y \end{bmatrix} =$$
$$= \begin{bmatrix} Y^\top Y \otimes I_n + AA^\top \otimes I_{m-r} & I_{m-r} \otimes AY \\ I_{m-r} \otimes Y^\top A^\top & I_{m-r} \otimes Y^\top Y \end{bmatrix}.$$

Now we recall that for each point at the manifold (9) holds, therefore

$$Z = \begin{bmatrix} I + AA^\top \otimes I_{m-r} & 0 \\ 0 & I \end{bmatrix}. \tag{12}$$

It is obvious that $\sigma_{\min}(Z) \geq 1$. Finally, $\sigma^2_{\min}(N) = \sigma_{\min}(Z) \geq 1$, therefore

$$\sigma_{\min}(N) \geq 1, \quad \|(N^\top)^\dagger\| \leq 1. \tag{13}$$

∎

Putting (13) into (3) we get

$$\|P - P'\| \leq \|N - N'\|.$$

[3]For simplicity of notation we will always assume that δx is the vectorized matrix δX and vice versa (shape of δX can always be easily recovered).

Finally, we need to estimate how N and N' changes under the change of A and Y. We have

$$N - N' = \begin{bmatrix} (Y - Y') \otimes I_n & 0 \\ (A^\top - (A')^\top) \otimes I_{m-r} & I_{m-r} \otimes (Y^\top - (Y^\top)') \end{bmatrix},$$

therefore

$$\|N - N'\| \leq 2\|Y - Y'\| + \|A - A'\|.$$

For small r

$$(m+n)r - r^2 \ll nm,$$

so to fully utilize the properties of $\widehat{\mathcal{M}}_r$ in computations we have to find an explicit basis in the tangent space first. This will be done in the next section.

2.3. Parametrization of the tangent space. To work with low rank matrices it is very convenient to represent them using reduced singular value decomposition (SVD). Namely for $A \in \mathcal{M}_{\leq r}$ we have

$$A = USV^\top,$$

with U and V having r orthonormal columns and S being a diagonal matrix. Using this notation we find that the following result holds:

THEOREM 4. *Orthogonal basis in the kernel of N from (10) is given by columns of the following matrix Q*

$$Q = \begin{bmatrix} V \otimes I_n & -Y \otimes (US_1) \\ 0 & I_{m-r} \otimes (VS_2) \end{bmatrix},$$

where S_1 and S_2 are diagonal matrices defined as

$$S_1 = \frac{S}{\sqrt{S^2 + 1}}, \quad S_2 = \frac{1}{\sqrt{S^2 + 1}}.$$

Proof. To theck that $Q^\top Q = I$ and $NQ = 0$ it is sufficient to do direct multiplication. The number of columns in Q is $nr + (m - r)r$ which is exactly the dimension of the tangent space. ∎

Now we will use smoothness of $\widehat{\mathcal{M}}_r$ to develop optimization algorithm over $\mathcal{M}_{\leq r}$.

3. Newton method.

3.1. Basic Newton method. Consider the optimization problem

$$F(A) \to \min, \quad \text{s.t. } A \in \mathcal{M}_{\leq r},$$

where F is twice differentiable. Using the idea described in subsection 1.1 this problem is equivivalent to

$$F(A) \to \min, \quad \text{s.t. } A \in \widehat{\mathcal{M}}_r.$$

To solve it we will rewrite it using Lagrange multipliers method. Taking into account defining equations of $\widehat{\mathcal{M}}_r$ (9) the Lagrangian for the constrained optimization problem reads

$$\mathcal{L}(A, Y, \Lambda, S) = F(A) + \langle AY, \Lambda \rangle + \frac{1}{2}\langle M, Y^\top Y - I \rangle,$$

where $\Lambda \in \mathbb{R}^{n \times m-r}$ and $M \in \mathbb{R}^{(m-r)\times(m-r)}$ are the Lagrange multipliers.

First-order optimality conditions give (for simplicity we denote the gradient of $F(A)$ with respect to A as a matrix $\nabla F(A)$).

$$\nabla F + \Lambda Y^\top = 0, \quad YM + A^\top \Lambda = 0, \quad AY = 0, \quad Y^\top Y = I.$$

Matrix M enters only the second equation, thus we always have

$$M = \arg\min_Z \|YZ + A^\top \Lambda\| = -Y^\top A^\top \Lambda = 0,$$

where we have used that Y has orthonormal columns and $AY = 0$. Thus, the first-order optimality conditions reduce to

$$\nabla F + \Lambda Y^\top = 0, \quad A^\top \Lambda = 0, \quad AY = 0, \quad Y^\top Y = I. \tag{14}$$

Now we can write down the Newton method for the system (14), which can be written in the saddle form

$$\begin{bmatrix} \widehat{G} & N^\top \\ N & 0 \end{bmatrix} \begin{bmatrix} \delta z \\ \delta \lambda \end{bmatrix} = \begin{bmatrix} f \\ 0 \end{bmatrix}, \tag{15}$$

and

$$f = -\mathrm{vec}(\nabla F + \Lambda Y^\top).$$

where we assumed that the initial point satisfies the constraints ($AY = 0$), the vector δz is

$$\delta z = \begin{bmatrix} \mathrm{vec}(\delta A) \\ \mathrm{vec}(\delta Y) \end{bmatrix}, \quad \delta \lambda = \mathrm{vec}(\delta \Lambda),$$

the constraints are written as

$$N \delta z = 0,$$

and

$$N = \begin{bmatrix} Y^\top \otimes I_n & I_{m-r} \otimes A \\ 0 & I_{m-r} \otimes Y^\top \end{bmatrix}$$

which means that δz is in the tangent space. Using parametrization via the matrix Q defined in Theorem 4 we obtain that $\delta z = Q \delta w$.
The first block row of system (15) reads

$$\widehat{G} Q \delta w + N^\top \delta \lambda = f.$$

Multiplying by $Q^\top$ we can eliminate $\delta \lambda$, which leads to the reduced Hessian equation

$$Q^\top \widehat{G} Q \delta w = Q^\top f.$$

The matrix $\widehat{G}$ in turn has a saddle-point structure:

$$\widehat{G} = \begin{bmatrix} H & C \\ C^\top & 0 \end{bmatrix},$$

where $H = \nabla^2 F$ is the ordinary Hessian, and C comes from the differentiation of the term $\Lambda Y^\top$ with respect to Y. In order to put C into a compact form, we introduce a "transposition" operator

$$T_{m,m-r} : \mathbb{R}^{m \times (m-r)} \times \mathbb{R}^{(m-r) \times m},$$

which acts as

$$T_{m,m-r}\mathrm{vec}(X) = \mathrm{vec}(X^\top),$$

for any $m \times (m-r)$ matrix X. Using this operator, matrix C is written as

$$C = (I_m \otimes \Lambda)T_{m,m-r}.$$

An important property of the matrix C is that if $Q_{12} = -Y \otimes (US_1)$ is the (1, 2) block of the matrix Q, then

$$Q_{12}C = 0,$$

if

$$A^\top \Lambda = 0,$$

which is again verified by the direct multiplication using the properties of the Kronecker product. The direct evaluation of the product

$$\widetilde{B} = Q^\top \widehat{G} Q,$$

(together with the property above) gives

$$\widetilde{G} = \begin{bmatrix} Q_{11}^\top H Q_{11} & Q_{11}^\top H Q_{12} + Q_{11}^\top C Q_{22} \\ Q_{12}^\top H Q_{11} + Q_{22}^\top C^\top Q_{11} & Q_{12}^\top H Q_{12} \end{bmatrix}, \tag{16}$$

and the system we need to solve has the form

$$\begin{bmatrix} Q_{11}^\top H Q_{11} & Q_{11}^\top H Q_{12} + Q_{11}^\top C Q_{22} \\ Q_{12}^\top H Q_{11} + Q_{22}^\top C^\top Q_{11} & Q_{12}^\top H Q_{12} \end{bmatrix} \begin{bmatrix} \delta u \\ \delta p \end{bmatrix} = \begin{bmatrix} Q_{11}^\top f \\ Q_{12}^\top f \end{bmatrix}. \tag{17}$$

We also need to estimate Λ. In order to get $Q_{12}C = 0$ we have to require that $A^\top \Lambda = 0$ exactly, thus

$$\Lambda = Z\Phi,$$

where Z is the orthonormal basis for the left nullspace of A, and Φ is defined from minimization of

$$\|\nabla F + Z\Phi Y^\top\| \to \min,$$

i.e.

$$\Phi = -Z^\top \nabla F Y,$$

and

$$\Lambda = -ZZ^\top \nabla F Y.$$

Note that f then is just a standard projection on the tangent space:

$$f = -\mathrm{vec}(\nabla F - ZZ^\top \nabla F Y Y^\top) = -\mathrm{vec}(\nabla F - (I - UU^\top)\nabla F (I - VV^\top)),$$

which is always a vectorization of a matrix with a rank not larger than $2r$. Moreover,

$$g_1 = Q_{11}^\top f = (V^\top \otimes I) f = \tag{18}$$

$$-\mathrm{vec}((\nabla F - (I - UU^\top)\nabla F (I - VV^\top))V) = \mathrm{vec}(-\nabla F V),$$

and the second component

$$g_2 = Q_{12}^\top f = -(Y^\top \otimes (US_1)^\top) f = \tag{19}$$

$$\mathrm{vec}((US_1)^\top(\nabla F - (I - UU^\top)\nabla F(I - VV^\top))Y) = \mathrm{vec}(S_1^\top U^\top \nabla F Y).$$

The solution is recovered from δu, δp as

$$\delta a = (V \otimes I_n)\delta u - (Y \otimes (US_1))\delta p,$$

or in the matrix form,

$$\delta A = \delta U V^\top - U S_1 \delta P Y^\top,$$

and the error in A (which we are interested in) is given by

$$\|\delta A\|_F^2 = \|\delta U\|_F^2 + \|S_1 \delta P\|_F^2.$$

We can further simplify the off-diagonal block. Consider

$$\widehat{C} = Q_{11}^\top C Q_{22} = (V^\top \otimes I)(I \otimes \Lambda)T(I \otimes V)(I \otimes S_2).$$

Consider the multiplication of this matrix by vector:

$$\mathrm{mat}(\widehat{C}\mathrm{vec}(\Phi)) = \Lambda(VS_2\Phi)^\top V = \Lambda\Phi^\top S_2^\top V^\top V = \Lambda\Phi^\top S_2^\top,$$

thus

$$\widehat{C} = (S_2 \otimes \Lambda)T_{(r,n-r)}.$$

3.2. Retraction. Note that since we assumed that initial points satisfies the constraints

$$AY = 0, \quad Y^\top Y = I \tag{20}$$

after doing each step of the Newton algorithm we have to perform the retraction to the manifold $\widehat{\mathcal{M}}_r$. One such possible retraction is the following. Given some pair (A_0, Y_0) not neccessarily satisfying (20) we first make columns of Y_0 orthonormal using QR-decomposition so

$$Y = Q(Y_0),$$

and then we define

$$A = A_0(I - YY^\top).$$

The basic Newton method on the manifold $\widehat{\mathcal{M}}_r$ is summarized in the following algorithm

Algorithm 1 Basic Newton method

Initial conditions A_0, Y_0, functional $F(A)$ and tolerance ε
Result: minimum of F on $\mathcal{M}_{\leq r}$
while $\|\delta U_i\|^2 + \|(S_1)_i \delta P\|^2 > \varepsilon$ **do**
  $U_i, S_i, V_i = \mathrm{svd}(A_i)$
  solve $\widetilde{A}\begin{bmatrix}\delta u \\ \delta p\end{bmatrix} = \begin{bmatrix} g_1 \\ g_2\end{bmatrix}$
  $\delta A_i = \delta U_i V_i^\top - U_i (S_1)_i \delta P Y_i^\top$
  $\delta Y_i = V_i (S_2)_i \delta P_i$
  $A_{i+1}, Y_{i+1} = \mathrm{retraction}(A_i + \delta A_i, Y + \delta Y_i)$
  $i = i + 1$
end while
return A_i

3.3. Semi-implicit parametrization of the tangent space. Storing and doing multiplications by Y are of order $O(nm)$. We can further simplify Algorithm 1 to reduce computational complexity. We introduce a new variable

$$\delta\phi^\top = Y\delta p^\top.$$

However this results in an implicit constraint on $\delta\phi$

$$\delta\phi V = 0.$$

In order to make arbitrary ϕ satisfy it we first multiply it by the projection operator $I - VV^\top$,

$$\phi' = \phi(I - VV^\top),$$

or in the matrix form

$$\begin{bmatrix}\delta u\\ \delta\phi'\end{bmatrix} = \begin{bmatrix}I & 0\\ 0 & I - VV^\top \otimes I\end{bmatrix}\begin{bmatrix}\delta u\\ \delta\phi\end{bmatrix}.$$

Notice also that

$$\delta p = \delta\phi Y,$$

and again using the properties of the Kronecker product we obtain

$$\begin{bmatrix}\delta u\\ \delta p\end{bmatrix} = \begin{bmatrix}I & 0\\ 0 & Y^\top \otimes I\end{bmatrix}\begin{bmatrix}\delta u\\ \delta\phi\end{bmatrix}.$$

Denote

$$\Pi = \begin{bmatrix}I & 0\\ 0 & I - VV^\top \otimes I\end{bmatrix},$$

$$W = \begin{bmatrix}I & 0\\ 0 & Y^\top \otimes I\end{bmatrix}.$$

The equations for the Newton method in the new variables take the following form:

$$\Pi^\top W^\top Q^\top \widetilde{G} Q W \Pi \begin{bmatrix}\delta u\\ \delta\phi\end{bmatrix} = \Pi^\top W^\top \begin{bmatrix}g_1\\ g_2\end{bmatrix} \tag{21}$$

where $g_1, g_2, Q, \widetilde{G}$ are as in (16), (18), and (19) and Theorem 4.

3.4. Iterative method. For a large n and m forming the full matrix (21) is computationally expensive, so we switch to iterative methods. To implement the matvec operation we need to simplify

$$\begin{bmatrix}l_1(\delta u, \delta\phi)\\ l_2(\delta u, \delta\phi)\end{bmatrix} = \Pi^\top W^\top Q^\top \widetilde{G} Q W \Pi \begin{bmatrix}\delta u\\ \delta\phi\end{bmatrix},$$

first.
Direct computation shows that

$$\begin{bmatrix}l_1(\delta u, \delta\phi)\\ l_2(\delta u, \delta\phi)\end{bmatrix} =$$

$$\Pi^\top W^\top \left[\begin{array}{c} (V^\top \otimes I)H(V\otimes I)\delta u- \\ (V^\top \otimes I)H(I\otimes U)\mathrm{vec}(S_1\delta\Phi(I-VV^\top))- \\ \mathrm{vec}((I-UU^\top)\nabla F(I-VV^\top)\delta\Phi^\top S_2) \\ \hline -(Y^\top\otimes I)(I\otimes S_1)(I\otimes U^\top)H(V\otimes I)\delta u+ \\ (Y^\top\otimes I)\mathrm{vec}(S_2(\delta U)^\top(-(I-UU^\top)\nabla F))+ \\ (Y^\top\otimes I)(I\otimes S_1)(I\otimes U^\top)H(I\otimes U)\mathrm{vec}(S_1\widehat{\delta\phi}(I-VV')) \end{array}\right],$$

and the right hand side has following form:

$$\begin{bmatrix} g_1 \\ g_2 \end{bmatrix} = \Pi^\top W^\top \begin{bmatrix} -\mathrm{vec}\nabla F V \\ (Y^\top\otimes I)\mathrm{vec}(S_1^\top U^\top \nabla F) \end{bmatrix}.$$

Since both Π and W only act on the second block it easy to check that in the final formula

$$l_2(\delta u,\delta\phi) = -(I-VV^\top\otimes I)(I\otimes S_1)(I\otimes U^\top)H(V\otimes I)\delta u+$$

$$\mathrm{vec}(S_2(\widehat{\delta u})^\top(-(I-UU^\top)\nabla F(I-VV^\top))+$$

$$(I-VV^\top\otimes I)(I\otimes S_1)(I\otimes U^\top)H(I\otimes U)\mathrm{vec}(S_1\widehat{\delta\Phi}(I-VV')),$$

$$g_2 = \mathrm{vec}(S_1^\top U^\top \nabla F(I-VV^\top)).$$

Note that in new variables we obtain

$$\delta A = \delta U V^\top - U S_1 \delta\Phi,$$

and

$$A+\delta A = U(SV^\top - S_1\delta\Phi) + \delta U V^\top.$$

Using this representation it is easy to compute SVD of $A+\delta A$ which has the rank no more than $2r$, and then truncate it to rank r. This allows us not to store the matrix A itself but only U, S and V we get from the SVD. We obtain the following algorithm

Algorithm 2 Newton method

Initial conditions U_0, S_0, V_0, functional $F(A)$ and tolerance ε
Result: minimum of F on $\mathcal{M}_{\leq r}$
while $\|\delta U_i\|^2 + \|(S_1)_i\delta\Phi\|^2 > \varepsilon$ **do**
  $(\delta u, \delta\phi) = \mathrm{gmres}(g_1, g_2, \mathrm{matvec})$
  $\delta A = \delta U_i V_i^\top - U_i(S_1)_i\delta\Phi$
  $U_{i+1}, S_{i+1}, V_{i+1} = \mathrm{retraction}(A+\delta A)$
  $i = i+1$
end while
return U_i, S_i, V_i

4. Technical aspects of the implementation.

4.1. Computation of the matvec and complexity. To efficiently compute the matvec for a given functional F one has to be able to evaluate the following expressions:

$$F(U,S,V) \tag{22}$$

(23) $$\nabla F V$$

(24) $$(\nabla F)^\top U$$

(25) $$(V^\top \otimes I) H (V \otimes I) \delta x$$

(26) $$(V^\top \otimes I) H (I \otimes U) \delta x$$

(27) $$(I \otimes U^\top) H (V \otimes I) \delta x$$

(28) $$(I \otimes U^\top) H (I \otimes U) \delta x$$

(29) $$\nabla F \delta x$$

(30) $$\delta x \nabla F$$

Computational complexity of Algorithm 2 depends heavily on whether we can effectively evaluate (22)–(30), which, however, any similar algorithm requires. Suppose now that for some functional F one can compute each of the expressions (22)–(30) in $O(\alpha(n, m))$ operations. In this case computing matvec in Algorithm 2 and total complexity are $O((n + m)r^2 + \alpha(n, m))$. Let us now consider two examples.

4.2. Matrix completion. Given some matrix B and set of indices Γ define

$$F(x) = \sum_{(i,j)\in\Gamma} (x_{i,j} - B_{i,j})^2 \to \min, x \in \mathcal{M}_{\leq r}.$$

Then under assumption that cardinality of Γ is small, operations (22)–(30) reduce to products of sparse matrices and hence $\alpha(n, m) = (n + m)r^2$.

4.3. Approximation of a sparse matrix. Consider approximation functional

$$F(x) = \|x - B\| \to \min, x \in \mathcal{M}_{\leq r},$$

and B is a sparse matrix. Expressions (22)–(30) can be heavily simplified by noticing that H in this case is the identity matix. Sparsness of B is used to evaluate $\nabla F \delta x$. Similarly $\alpha(n, m) = (n + m)r^2$.

5. Numerical results. Algorithm 2 was implemented in Python using `numpy` and `scipy` libraries. We tested it on the functional described in subsection 4.3 for B being the matrix constructed from the MovieLens 100K Dataset [**?**], so $n = 1000, m = 1700$ and B has 100000 non-zero elements. Since pure Newton method is only local, for a first test we choose small random perturbation of the solution obtained via SVD as initial condition. We got following results for various r (see Figure 2).

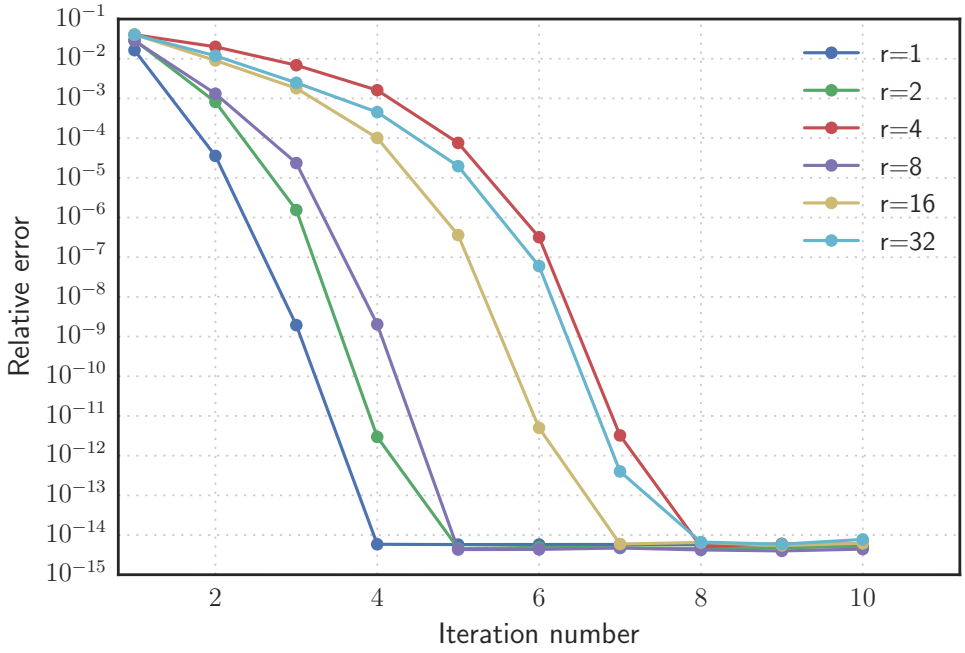
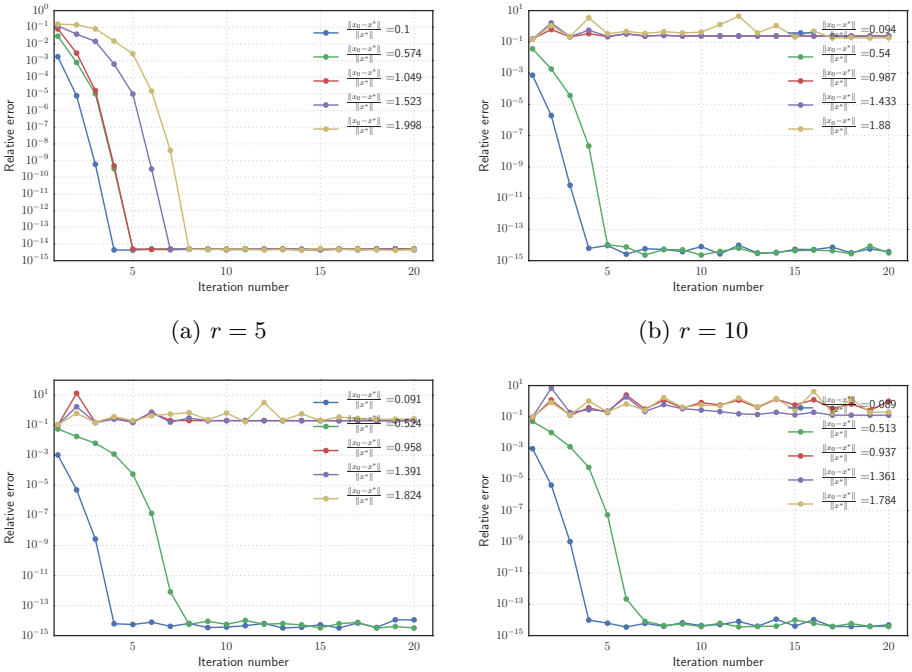


Fig. 2: Sparse matrix approximation: close initial guess

This shows quadratic convergence of our method. As a second test we fix the rank and test whether method converges, depending on the distance between the initial guess and the answer, obtained via SVD (see Figure 3).



(a) $r = 5$ (b) $r = 10$

(c) $r = 15$ (d) $r = 20$

Fig. 3: Sparse matrix approximation: varying distance to the initial guess.

REFERENCES

[1] Y. M. Chen, X. S. Chen, and W. Li, *On perturbation bounds for orthogonal projections*, Numerical Algorithms, (2016), pp. 1–12.

[2] S. Naldi, *Exact algorithms for determinantal varieties and semidefinite programming*, PhD thesis, Toulouse, INSA, 2015.