Neurocomputing 192 (2016) 72–80

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

A Riemannian rank-adaptive method for low-rank optimization ☆

Guifang Zhou [a], Wen Huang [b], Kyle A. Gallivan [a], Paul Van Dooren [b], Pierre-Antoine Absil [b,*]

[a] Department of Mathematics, Florida State University, 1017 Academic Way, Tallahassee, FL 32306-4510, USA
[b] ICTEAM Institute, Université catholique de Louvain, Avenue G. Lemaître 4, 1348 Louvain-la-Neuve, Belgium

ARTICLE INFO

Article history:
Received 12 July 2015
Received in revised form
31 January 2016
Accepted 2 February 2016
Available online 2 March 2016

Keywords:
Low-rank optimization
Rank-constrained optimization
Riemannian manifold
Fixed-rank manifold
Low-rank approximation

ABSTRACT

This paper presents an algorithm that solves optimization problems on a matrix manifold $\mathcal{M} \subseteq \mathbb{R}^{m\times n}$ with an additional rank inequality constraint. The algorithm resorts to well-known Riemannian optimization schemes on fixed-rank manifolds, combined with new mechanisms to increase or decrease the rank. The convergence of the algorithm is analyzed and a weighted low-rank approximation problem is used to illustrate the efficiency and effectiveness of the algorithm.

1. Introduction

We consider low-rank optimization problems of the following form:

$$\min_{X \in \mathcal{M}_{\leq k}} f(X), \tag{1}$$

where $\mathcal{M}$ is a submanifold of $\mathbb{R}^{m\times n}$,

$$\mathcal{M}_{\leq k} := \{X \in \mathcal{M} \mid \operatorname{rank}(X) \leq k\}$$

with $k \leq \min(m,n)$, and f is a real-valued function on $\mathcal{M}_{\leq k}$. The notation

$$\mathcal{M}_r := \{X \in \mathcal{M} \mid \operatorname{rank}(X) = r\} \tag{2}$$

will also be used frequently. Typical choices for $\mathcal{M}$ are $\mathbb{R}^{m\times n}$ itself and the Frobenius sphere, i.e., the set of all $m \times n$ matrices of fixed Frobenius norm.

Applications of (1) appear notably in machine learning, e.g., for collaborative filtering [1,2], multi-class classification [3], multi-response regression [4,5], learning a function over pairs of points [6], and learning a low-rank similarity measure [7]. Applications of low-rank optimization are also found in other areas such as systems and control [8,9] and computer vision [10,11].

An increasingly popular way to approach problem (1) is to consider the related but simpler problem $\min_{X \in \mathbb{R}_k^{m\times n}} f(X)$, where $\mathbb{R}_k^{m\times n} = \{X \in \mathbb{R}^{m\times n} \mid \operatorname{rank}(X) = k\}$ in view of the notation (2); see, e.g., [7,12–14]. Since $\mathbb{R}_k^{m\times n}$ is a submanifold of $\mathbb{R}^{m\times n}$ of dimension $(m+n-k)k$ (see [15, Chapter 5, Proposition 1.14]), this simpler problem can be solved using Riemannian optimization techniques such as those presented in [16–20]. However, a disadvantage is that the manifold $\mathbb{R}_k^{m\times n}$ is not closed in $\mathbb{R}^{m\times n}$, which jeopardizes the well-posedness of the optimization problem and complicates the convergence analysis of optimization methods if the iterates cannot be assumed to stay safely away from $\mathbb{R}_{\leq k-1}^{m\times n}$.

Very recently a more global view of a projected line-search method on $\mathbb{R}_{\leq k}^{m\times n} = \{X \in \mathbb{R}^{m\times n} \mid \operatorname{rank}(X) \leq k\}$ along with a convergence analysis has been developed in [21]. In [22], the results of [21] have been exploited to propose an algorithm that successively increases the rank by a given constant. Its convergence to critical points can be deduced from [21, Theorem 3.9]; it relies on the assumption, often satisfied in practice, that the limit points have rank k. Under this assumption, a line-search method on $\mathbb{R}_{\leq k}^{m\times n}$ is ultimately the same as a line-search method on $\mathbb{R}_k^{m\times n}$.

In this paper, we develop a Riemannian rank-adaptive algorithm for the optimization problem (1). Its main features are as follows. First, the feasible set $\mathcal{M}_{\leq k}$ is more general than the set $\mathbb{R}_{\leq k}^{m\times n}$ considered in [21,22]. Second, the proposed algorithm increases or decreases the rank by an adaptively chosen amount as the iteration proceeds. The rank update mechanism is governed by parameters that the user can adjust to strike a balance between the goals of (i) saving on space and time complexity by reducing the rank and (ii) achieving higher accuracy by increasing the rank. Finally, theoretical convergence results are given, and the

☆This paper presents research results of the Belgian Network DYSCO (Dynamical Systems, Control, and Optimization), funded by the Interuniversity Attraction Poles Programme initiated by the Belgian Science Policy Office. This work was supported by the National Science Foundation under Grant NSF-1262476 and by FNRS under Grant PDR T.0173.13.

* Corresponding author.

http://dx.doi.org/10.1016/j.neucom.2016.02.030
0925-2312/

proposed method is shown on numerical experiments to outperform state-of-the-art methods on a weighted low-rank approximation problem.

The rest of this paper is organized as follows. Standing assumptions are gathered in the next section. The proposed method is presented in Section 3 and analyzed in Section 4. Implementation practicalities are discussed in Section 5. Numerical experiments are reported in Section 6, and conclusions are drawn in Section 7.

A preliminary version of this work can be found in [23].

2. Notation, definitions, and standing assumptions

The notation $\mathcal{M}_r$ and $\mathcal{M}_{\leq r}$ defined above will be used frequently. The notation f_{F} stands for an extension of f on $\mathcal{M}$ (see Assumption 3 below) and f_r denotes the restriction of f to $\mathcal{M}_r$.

Throughout the paper, the following assumptions are in force.

Assumption 1. $\overline{\mathcal{M}_r} \subseteq \mathcal{M}_{\leq r}$ for all positive integers $r \leq k$, where $\overline{\mathcal{M}_r}$ stands for the closure of $\mathcal{M}_r$.

Observe that, since the closure of an intersection is a subset of the intersection of the closures and since $\overline{\mathbb{R}_r^{m\times n}} = \mathbb{R}_{\leq r}^{m\times n}$, it follows that the above assumption holds whenever the submanifold $\mathcal{M}$ is a closed subset of $\mathbb{R}^{m\times n}$. It is useful to bear in mind that a sequence of rank-r matrices can converge to a lower-rank matrix but not to a larger-rank matrix.

The next assumption is crucial to the Riemannian aspect of the proposed Riemannian rank-adaptive method:

Assumption 2. $\mathcal{M}_r$ is a submanifold of $\mathbb{R}^{m\times n}$, for all positive integers $r \leq k$.

We need the cost function f to be sufficiently smooth for gradient-descent techniques to be applicable:

Assumption 3. The cost function f admits a continuously differentiable extension f_{F} on a neighborhood of $\mathcal{M}_{\leq k}$ in $\mathcal{M}$.

The reader will observe that neither the size of the neighborhood nor the choice of the extension will have an impact on the proposed method.

The tangent cone to a set $S \subseteq \mathbb{R}^{m\times n}$ at $X \in \mathbb{R}^{m\times n}$ is the set

$$\mathrm{T}_X S := \{\dot\gamma(0) \mid \gamma \in C^1, \gamma(0) = X, \exists \delta > 0 : \forall t \in (0,\delta) : \gamma(t) \in S\},$$

where $\dot\gamma(0)$ denotes the derivative of curve γ at 0. This definition of $\mathrm{T}_X S$ is motivated by the goal of conducting line searches along smooth (i.e., C^1) curves. Observe that $\mathrm{T}_X S = \varnothing$ when $X \notin \overline{S}$.

We point out that, for any $X \in \mathcal{M}_r$, the tangent cones are nested as follows: $\mathrm{T}_X\mathcal{M}_{\leq 0} \subseteq \mathrm{T}_X\mathcal{M}_{\leq 1} \subseteq \cdots \subseteq \mathrm{T}_X\mathcal{M}$. The tangent cones $\mathrm{T}_X\mathcal{M}_{\leq r}$ and $\mathrm{T}_X\mathcal{M}$ are actually linear spaces since $\mathcal{M}$ and $\mathcal{M}_r$ are manifolds and $\mathcal{M}_{\leq r}$ is identical to $\mathcal{M}_r$ locally around $X \in \mathcal{M}_r$. Moreover, we have $\mathrm{T}_X\mathcal{M}_{\leq s} = \varnothing$ for all $s < r$. This justifies the following definition.

Definition 1 (*update-rank*). Let $X \in \mathcal{M}$ and $\eta_X \in \mathrm{T}_X\mathcal{M}$. The *update-rank* of η_X is the unique integer r such that $\eta_X \in \mathrm{T}_X\mathcal{M}_{\leq r} \setminus \mathrm{T}_X\mathcal{M}_{\leq r-1}$, with $A\setminus B$ denoting the set difference $\{x \in A \mid x \notin B\}$.

For the purpose of conducting line searches along given directions while keeping the rank under control, we will need $\mathcal{M}$ to be endowed with a curves-selection mechanism defined as follows, where $\mathrm{T}\mathcal{M} := \bigsqcup_{X\in\mathcal{M}} \mathrm{T}_X\mathcal{M}$ denotes the tangent bundle of $\mathcal{M}$.

Definition 2 (*Rank-related retraction*). In the context of problem (1), a mapping $\tilde R : \mathrm{T}\mathcal{M} \to \mathcal{M}$ is a *rank-related retraction* if, for all $X_* \in \mathcal{M}_{\leq k}$, there exist $\delta_{X_*} > 0$ and a neighborhood $\mathcal{U}$ of X_* in $\mathcal{M}_{\leq k}$ such that, for all $X \in \mathcal{U}$ and all $\xi_X \in \mathrm{T}_X\mathcal{M}_{\leq k}$ with $\|\xi_X\| = 1$, it holds that (i) $\tilde R_X(0) = X$, where $\tilde R_X$ denotes the restriction of $\tilde R$ to $\mathrm{T}_X\mathcal{M}$ and 0 stands for the zero vector in $\mathrm{T}_X\mathcal{M}$, (ii) $[0, \delta_{X_*}) \ni t \mapsto \tilde R_X(t\xi_X)$ is smooth and $\tilde R_X(t\xi_X) \in \mathcal{M}_{\leq \tilde r}$ for all $t \in [0, \delta_{X_*})$, where $\tilde r$ is the update-rank of ξ_X, (iii) $\frac{d}{dt}\tilde R_X(t\xi_X)\|_{t=0} = \xi_X$.

Note that $\tilde R_X$ is not necessarily a retraction on $\mathcal{M}$ in the sense given in [24,16], since it may not be smooth on the tangent bundle $\mathrm{T}\mathcal{M}$. A specific rank-related retraction is given in Section 5.

Observe that in point (ii) of Definition 2, we require $\tilde R_X(t\xi_X)$ to belong to $\mathcal{M}_{\leq \tilde r}$ but not necessarily to $\mathcal{M}_{\tilde r}$. Indeed we found that the condition $\tilde R_X(t\xi_X) \in \mathcal{M}_{\tilde r}$ would be cumbersome to enforce while being unnecessary for the convergence analysis.

We let $\operatorname{grad} f_{\mathrm{F}}(X)$ denote the Riemannian gradient of f_{F} at $X \in \mathcal{M}$. It can be obtained by considering any smooth extension of f_{F} around X in $\mathbb{R}^{m\times n}$ and taking the projection to $\mathrm{T}_X\mathcal{M}$ of its Euclidean gradient at X; see [16, (3.37)]. Likewise, $\operatorname{grad} f_r(X)$ denotes the Riemannian gradient of f_r at $X \in \mathcal{M}_r$, and it is obtained by projecting $\operatorname{grad} f_{\mathrm{F}}(X)$ onto the tangent space $\mathrm{T}_X\mathcal{M}_r$.

Throughout the paper, $\|\cdot\|$ denotes the Frobenius norm and $\langle\cdot,\cdot\rangle$ the Frobenius inner product.

Consider $X \in \mathcal{M}$, $\xi \in \mathrm{T}_X\mathcal{M}$, and a positive integer r. The set of best approximations of ξ in $\mathrm{T}_X\mathcal{M}_{\leq r}$ is denoted by $\mathrm{P}_{\mathrm{T}_X\mathcal{M}_{\leq r}}(\xi)$. Note that this set may contain more than one point. (In the case where $\mathcal{M} = \mathbb{R}^{m\times n}$, this follows directly from (12) and the non-uniqueness of a best low-rank approximation.) However, as indicated in [21, Section 2.1] (or see Lemma 1 below), all its elements have the same norm, hence $\|\mathrm{P}_{\mathrm{T}_X\mathcal{M}_{\leq r}}(\xi)\|$ is well defined. We say that X is a *critical point* of f if $\|\mathrm{P}_{\mathrm{T}_X\mathcal{M}_{\leq k}}(\operatorname{grad} f_{\mathrm{F}}(X))\| = 0$. (It can be seen that this notion does not depend on the chosen extension f_{F} of f.)

3. A Riemannian rank-adaptive algorithm

The proposed method is listed in Algorithm 3, but we invite the reader to first read the more reader-friendly description in Section 3.1 and to refer to the pseudocode in Algorithm 3 when needed.

3.1. Algorithm description

We first discuss the two subprograms, Algorithms 1 and 2, called by Algorithm 3.

Algorithm 1. Rank reduction with threshold Δ.

Require: (X, Δ), where $X \in \mathbb{R}^{m\times n}$ and $\Delta > 0$.
1: Find the singular values $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{\min\{m,n\}} \geq 0$ of matrix X;
2: Set r to be the largest integer r such that $\sigma_r/\sigma_1 \geq \Delta$;
3: Choose $\hat X \in \arg\min_{Y\in\mathcal{M}_{\leq r}} \|Y - X\|$;
4: Return $(\hat X, r)$.

Algorithm 2. Rank-related Armijo backtracking.

1: Inherit $\tilde R, X_n, \beta, \overline{\alpha}, \eta^*, \mathcal{M}_{\tilde r}, f, \sigma$ from Algorithm 3 where Algorithm 2 is called;
2: Compute the smallest nonnegative integer m such that
(i) $\tilde R_{X_n}(\beta^m\overline{\alpha}\eta^*)$ belongs to $\mathcal{M}_{\leq \tilde r}$, and
(ii) $f(X_n) - f(\tilde R_{X_n}(\beta^m\overline{\alpha}\eta^*)) \geq \sigma\langle -\operatorname{grad} f_{\mathrm{F}}(X_n), \beta^m\overline{\alpha}\eta^*\rangle_{X_n}$;
3: Return $t^* \leftarrow \beta^m\overline{\alpha}$.

The output $\hat X$ of Algorithm 1 is a best approximation of X in $\mathcal{M}_{\leq r}$, where r is the number (counting multiplicities) of singular values of X that are larger than $\sigma_1\Delta$, with σ_1 being the largest singular value of X. Observe that $\hat X$ is simply X in the plausible case