Linear Regression under Fixed-Rank Constraints: A Riemannian Approach

Gilles Meyer G.MEYER@ULG.AC.BE
Department of EECS, University of Liège, B-4000 Liège, Belgium

Silvère Bonnabel SILVERE.BONNABEL@MINES-PARISTECH.FR
Robotics Center, Mines ParisTech, Boulevard Saint-Michel, 60, 75272 Paris, France

Rodolphe Sepulchre R.SEPULCHRE@ULG.AC.BE
Department of EECS, University of Liège, B-4000 Liège, Belgium

Abstract

In this paper, we tackle the problem of learning a linear regression model whose parameter is a fixed-rank matrix. We study the Riemannian manifold geometry of the set of fixed-rank matrices and develop efficient line-search algorithms. The proposed algorithms have many applications, scale to high-dimensional problems, enjoy local convergence properties and confer a geometric basis to recent contributions on learning fixed-rank matrices. Numerical experiments on benchmarks suggest that the proposed algorithms compete with the state-of-the-art, and that manifold optimization offers a versatile framework for the design of rank-constrained machine learning algorithms.

1. Introduction

Learning a low-rank matrix from data is a fundamental problem arising in many modern machine learning applications: collaborative filtering (Rennie & Srebro, 2005), classification with multiple classes (Amit et al., 2007), learning on pairs (Abernethy et al., 2009), dimensionality reduction (Cai et al., 2007), learning of low-rank distances (Meyer et al., 2011) and low-rank similarity measures (Shalit et al., 2010), multi-task learning (Evgeniou et al., 2005), just to name a few.

Parallel to the development of these new applications, the ever-growing size and number of large-scale datasets demands machine learning algorithms that can cope with very large matrices. Scalability to high dimensional problems is therefore a crucial issue in the design of algorithms.

Most of the recent algorithmic contributions on learning low-rank matrices have been proposed in the context of matrix completion. Convex relaxations based on the nuclear norm or trace norm heuristic (Fazel, 2002; Cai et al., 2008) have attracted a lot of attention as theoretical performance guarantees are available (Bach, 2008; Recht et al., 2010). However, an intrinsic limitation of the approach is that the rank of intermediate solutions cannot be bounded a priori. For large-scale problems, memory requirement may thus become prohibitively large. A different yet complementary approach that resolves this issue, assumes a fixed-rank factorization of the solution and optimize the corresponding non-convex optimization problem (Rennie & Srebro, 2005; Keshavan et al., 2010; Jain et al., 2010; Shalit et al., 2010). Despite the potential introduction of local minima, fixed-rank factorizations achieve very good performance in practice. Moreover, Keshavan et al. (2010) and Jain et al. (2010) show that performance guarantees are also possible when a good heuristic is available for the initialization.

In this paper, we pursue the research on fixed-rank factorizations and study the Riemannian geometry of two particular fixed-rank factorizations (Sections 2 and 3). We build on recent advances in optimization on Riemannian matrix manifolds (Absil et al., 2008) and exploit the manifold geometry of the search space. We design novel line-search algorithms for learning a linear regression model whose parameter is a matrix $\mathbf{W} \in \mathcal{F}(r, d_1, d_2)$, the set of d_1-by-d_2 matrices of a

Appearing in *Proceedings of the 28th International Conference on Machine Learning*, Bellevue, WA, USA, 2011.

given rank r. The resulting algorithms (Section 4) generalize our recent work on symmetric fixed-rank positive semidefinite matrices (Meyer et al., 2011), they scale to high dimensional problems and connect with the recent contributions on learning low-rank matrices. Numerical experiments (Section 5) are performed on benchmark problems for which the proposed algorithms compete with the state-of-the-art.

2. Quotient Geometry of Fixed-Rank Matrix Factorizations

We review two fixed-rank matrix factorizations and study the corresponding quotient manifold geometries. The quotient nature of the underlying search space stems from the fact that an element $\mathbf{W} \in \mathcal{F}(r, d_1, d_2)$ is represented by an entire equivalence class of matrices.

The factorizations of interest are rooted in the thin singular value decomposition (SVD)

$$\mathbf{W} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T,$$

where $\mathbf{U} \in \mathrm{St}(r, d_1) = \{\mathbf{U} \in \mathbb{R}^{d_1 \times r} : \mathbf{U}^T\mathbf{U} = \mathbf{I}\}$, $\mathbf{V} \in \mathrm{St}(r, d_2)$, and $\mathbf{\Sigma} \in \mathbb{R}^{r \times r}$ is diagonal with positive entries. The SVD exists for any $\mathbf{W} \in \mathcal{F}(r, d_1, d_2)$.

2.1. Balanced Factorization

The SVD can be rearranged as

$$\mathbf{W} = (\mathbf{U}\mathbf{\Sigma}^{\frac{1}{2}})(\mathbf{\Sigma}^{\frac{1}{2}}\mathbf{V}^T) = \mathbf{G}\mathbf{H}^T,$$

where $\mathbf{G} = \mathbf{U}\mathbf{\Sigma}^{\frac{1}{2}} \in \mathbb{R}_*^{d_1 \times r}$ and $\mathbf{H} = \mathbf{V}\mathbf{\Sigma}^{\frac{1}{2}} \in \mathbb{R}_*^{d_2 \times r}$ are full-rank matrices. The resulting factorization is not unique since the group action

$$(\mathbf{G}, \mathbf{H}) \mapsto (\mathbf{G}\mathbf{M}^{-1}, \mathbf{H}\mathbf{M}^T), \tag{1}$$

where $\mathbf{M} \in \mathrm{GL}(r) = \{\mathbf{M} \in \mathbb{R}^{r \times r} : \det(\mathbf{M}) \neq 0\}$, leaves the original matrix $\mathbf{W}$ unchanged.

The map (1) allows us to identify the search space of interest with the quotient space

$$\mathcal{F}(r, d_1, d_2) \simeq (\mathbb{R}_*^{d_1 \times r} \times \mathbb{R}_*^{d_2 \times r})/\mathrm{GL}(r), \tag{2}$$

which represents the set of equivalence classes

$$[(\mathbf{G}, \mathbf{H})] = \{(\mathbf{G}\mathbf{M}^{-1}, \mathbf{H}\mathbf{M}^T) : \mathbf{M} \in \mathrm{GL}(r)\}. \tag{3}$$

Among the set of representatives $(\mathbf{G}, \mathbf{H})$ in (3), balanced factorizations are of particular interest. A factorization $\mathbf{W} = \mathbf{G}\mathbf{H}^T$ is balanced if $\mathbf{G}^T\mathbf{G} = \mathbf{H}^T\mathbf{H}$. Balanced factorization are well-known in model reduction and system approximation (Helmke & Moore, 1996), they ensure good numerical conditioning and robustness to noise. Helmke & Moore (1996) show that balanced factorizations are characterized as the critical points of the cost function

$$b(\mathbf{G}, \mathbf{H}) = \|\mathbf{G}\|_F^2 + \|\mathbf{H}\|_F^2. \tag{4}$$

within an equivalence class (3).

2.2. Polar Factorization

A second interesting factorization is obtained by considering the following group action on the SVD,

$$(\mathbf{U}, \mathbf{\Sigma}, \mathbf{V}) \mapsto (\mathbf{U}\mathbf{O}, \mathbf{O}^T\mathbf{\Sigma}\mathbf{O}, \mathbf{V}\mathbf{O}),$$

where $\mathbf{O} \in \mathcal{O}(r)$, the set of r-by-r rotation matrices. Since $\mathbf{O}^T\mathbf{\Sigma}\mathbf{O}$ now represents a positive definite matrix, this gives us the fixed-rank factorization

$$\mathbf{W} = \mathbf{U}\mathbf{B}\mathbf{V}^T,$$

where $\mathbf{U} \in \mathrm{St}(d_1, r)$, $\mathbf{V} \in \mathrm{St}(d_2, r)$, and $\mathbf{B} \in S_+(r)$, the set of r-by-r positive definite matrices. The alternative choice $\mathbf{B}$ positive definite instead of $\mathbf{B}$ diagonal removes the discrete symmetries induced by the arbitrary order on the singular values. The search space is again a quotient manifold

$$\mathcal{F}(r, d_1, d_2) \simeq (\mathrm{St}(d_1, r) \times S_+(r) \times \mathrm{St}(d_2, r))/\mathcal{O}(r), \tag{5}$$

which represents the set of equivalence classes

$$[(\mathbf{U}, \mathbf{B}, \mathbf{V})] = \{(\mathbf{U}\mathbf{O}, \mathbf{O}^T\mathbf{B}\mathbf{O}, \mathbf{V}\mathbf{O}) : \mathbf{O} \in \mathcal{O}(r)\}. \tag{6}$$

Since $\mathbf{U}$ and $\mathbf{V}$ are orthogonal matrices, the polar factorization automatically encodes the property of a balanced factorization. Another nice property of the factorization is that $\|\mathbf{W}\|_F^2 = \|\mathbf{B}\|_F^2$. A regularization on $\|\mathbf{W}\|_F^2$ is thus very cheap because it only involves a matrix of size r, with typically $r \ll d_1, d_2$.

3. Geometry of Line-Search Algorithms with Fixed-Rank Constraints

This section studies the first-order quotient geometry of the factorizations $\mathbf{W} = \mathbf{G}\mathbf{H}^T = \mathbf{U}\mathbf{B}\mathbf{V}^T$. It also introduces the key concepts and notations that allow a systematic derivation of line-search algorithms on quotient manifolds. Proofs for all the propositions below are provided as supplementary material.

3.1. Line-Search Algorithms on Riemannian Matrix Manifolds

This section summarizes the exposition of Absil et al. (2008, Chapters 3 and 4). An abstract line-search algorithm on a manifold $\mathcal{W}$ is based on the update formula

$$\mathbf{W}_{t+1} = R_{\mathbf{W}_t}(s_t \xi_{\mathbf{W}_t}),$$

where the search direction $\xi_{\mathbf{W}_t}$ is an element of the tangent space $T_{\mathbf{W}_t}\mathcal{W}$ at $\mathbf{W}_t$. The scalar $s_t > 0$ is the step size. The retraction $R_{\mathbf{W}_t}$ is a local update mapping from the tangent space $T_{\mathbf{W}_t}\mathcal{W}$ to the manifold $\mathcal{W}$. Let $f : \mathcal{W} \to \mathbb{R}$ be a smooth real-valued function on the manifold. When the search direction $\xi_{\mathbf{W}_t}$ coincides with $-\mathrm{grad}\, f(\mathbf{W}_t)$, a gradient descent algorithm to minimize f on the manifold is obtained (Figure 1).

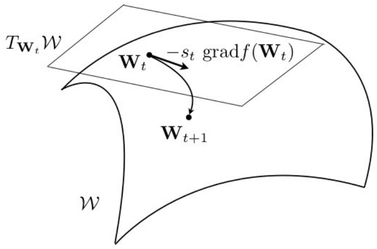

$T_{\mathbf{W}_t}\mathcal{W}$
$\mathbf{W}_t$
$-s_t \ \mathrm{grad} f(\mathbf{W}_t)$
$\mathbf{W}_{t+1}$
$\mathcal{W}$

Figure 1. Line-search algorithm on a manifold.

The manifold is endowed with a metric $g(\xi_{\mathbf{W}}, \zeta_{\mathbf{W}})$, which is an inner product between elements $\xi_{\mathbf{W}}, \zeta_{\mathbf{W}}$ of the tangent space $T_{\mathbf{W}}\mathcal{W}$ at $\mathbf{W}$.

The Riemannian gradient $\mathrm{grad}\, f(\mathbf{W}_t)$ at $\mathbf{W}_t$ is computed according to the chosen metric. It is defined as the unique element $\mathrm{grad}\, f(\mathbf{W}) \in T_{\mathbf{W}}\mathcal{W}$ that satisfies

$$Df(\mathbf{W})[\xi_{\mathbf{W}}] = g(\mathrm{grad}\, f(\mathbf{W}), \xi_{\mathbf{W}}), \forall \xi_{\mathbf{W}} \in T_{\mathbf{W}}\mathcal{W}. \quad (7)$$

The quantity $Df(\mathbf{W})[\xi_{\mathbf{W}}]$ is the directional derivative of $f(\mathbf{W})$ in the direction $\xi_{\mathbf{W}}$.

In a nutshell, a quotient manifold is a set of equivalence classes. A simple example is the Grassmann manifold $\mathrm{Gr}(r, d)$, the set of r-dimensional subspaces in $\mathbb{R}^d$, regarded as a set of r-dimensional orthogonal frames that cannot be superposed by a rotation.

For a quotient manifold $\mathcal{W} = \overline{\mathcal{W}}/\sim$, where $\overline{\mathcal{W}}$ is the total space and $\sim$ is the equivalence relation that defines the quotient, a tangent vector $\xi_{[\mathbf{W}]} \in T_{[\mathbf{W}]}\mathcal{W}$ at $[\mathbf{W}]$ is restricted to the directions that do not induce a displacement along the set of equivalence classes $[\mathbf{W}]$.

This is achieved by decomposing the tangent space in the total space $T_{\mathbf{W}}\overline{\mathcal{W}}$ into complementary spaces

$$T_{\mathbf{W}}\overline{\mathcal{W}} = \mathcal{V}_{\mathbf{W}}\overline{\mathcal{W}} \oplus \mathcal{H}_{\mathbf{W}}\overline{\mathcal{W}}.$$

The *vertical space* $\mathcal{V}_{\mathbf{W}}\overline{\mathcal{W}}$ is the set of directions that contains tangent vectors to the equivalence classes. The *horizontal space* $\mathcal{H}_{\mathbf{W}}\overline{\mathcal{W}}$ is a complement of the vertical space $\mathcal{V}_{\mathbf{W}}\overline{\mathcal{W}}$ in $T_{\mathbf{W}}\overline{\mathcal{W}}$, that allows us to represent tangent vectors to the quotient space. Indeed, with such a decomposition of $T_{\mathbf{W}}\overline{\mathcal{W}}$, a given tangent vector $\xi_{[\mathbf{W}]} \in T_{[\mathbf{W}]}\mathcal{W}$ at $[\mathbf{W}]$ is uniquely represented by a tangent vector $\bar{\xi}_{\mathbf{W}} \in \mathcal{H}_{\mathbf{W}}\overline{\mathcal{W}}$ that satisfies

$$D\pi(\mathbf{W})[\bar{\xi}_{\mathbf{W}}] = \xi_{[\mathbf{W}]}.$$

The mapping π is the *quotient map* $\pi : \mathbf{W} \mapsto [\mathbf{W}]$. The tangent vector $\bar{\xi}_{\mathbf{W}} \in \mathcal{H}_{\mathbf{W}}\overline{\mathcal{W}}$ is called the *horizontal lift* of $\xi_{[\mathbf{W}]}$ at $\mathbf{W}$. Provided that the metric $\bar{g}(\bar{\xi}_{\mathbf{W}}, \bar{\zeta}_{\mathbf{W}})$ in the total space is invariant along equivalence classes, it defines a metric on the quotient

$$g(\xi_{[\mathbf{W}]}, \zeta_{[\mathbf{W}]}) \triangleq \bar{g}(\bar{\xi}_{\mathbf{W}}, \bar{\zeta}_{\mathbf{W}}).$$

Natural displacements on the manifold are performed by following geodesics (paths of shortest length on the manifold) starting from $\mathbf{W}$ and tangent to $\xi_{\mathbf{W}}$. This is performed by means of the exponential map

$$\mathbf{W}_{t+1} = \mathrm{Exp}_{\mathbf{W}_t}(s_t \xi_{\mathbf{W}_t}),$$

which induces a line-search algorithm along geodesics. However, the geodesics are generally either expensive to compute or not available in closed-form.

A more general update formula is obtained if we relax the constraint of moving along geodesics. The retraction mapping $R_{\mathbf{W}_t}(s_t \xi_{\mathbf{W}_t})$, locally approximates the exponential mapping. It provides an attractive alternative to the exponential mapping in the design of optimization algorithms on manifolds, as it reduces the computational cost of the update while retaining the main properties that ensure convergence results.

3.2. Geometry of the Balanced Factorization

We endow the space $\mathbb{R}_*^{d_1 \times r} \times \mathbb{R}_*^{d_2 \times r}$ with the metric,

$$\begin{aligned} \bar{g}((\bar{\xi}_{\mathbf{G}}, \bar{\xi}_{\mathbf{H}}), (\bar{\zeta}_{\mathbf{G}}, \bar{\zeta}_{\mathbf{H}})) = \mathrm{Tr}((\mathbf{G}^T\mathbf{G})^{-1} \bar{\xi}_{\mathbf{G}}^T \bar{\zeta}_{\mathbf{G}}) \\ + \mathrm{Tr}((\mathbf{H}^T\mathbf{H})^{-1} \bar{\xi}_{\mathbf{H}}^T \bar{\zeta}_{\mathbf{H}}), \end{aligned} \quad (8)$$

which is chosen to be invariant along the set of equivalence classes (3).

Proposition 3.1. *The quotient manifold* (2) *endowed with the Riemannian metric* (8) *admits a set of horizontal vectors* $(\bar{\xi}_{\mathbf{G}}, \bar{\xi}_{\mathbf{H}}) \in \mathbb{R}^{d_1 \times r} \times \mathbb{R}^{d_2 \times r}$ *that satisfies*

$$\bar{\xi}_{\mathbf{G}}^T \mathbf{G}(\mathbf{H}^T\mathbf{H}) = (\mathbf{G}^T\mathbf{G})\mathbf{H}^T \bar{\xi}_{\mathbf{H}}. \quad (9)$$

The next proposition show that the chosen metric is invariant along the set of equivalence classes (3).

Proposition 3.2. *Let* $\xi_{[(\mathbf{G},\mathbf{H})]}$ *be a tangent vector to the quotient* (2) *at* $[(\mathbf{G}, \mathbf{H})]$. *The horizontal lifts of* $\xi_{[(\mathbf{G},\mathbf{H})]}$ *at* $(\mathbf{G}, \mathbf{H})$ *and at* $(\mathbf{G}\mathbf{M}^{-1}, \mathbf{H}\mathbf{M}^T)$ *are related as follow,* $\forall \mathbf{M} \in \mathrm{GL}(r)$,

$$(\bar{\xi}_{\mathbf{G}\mathbf{M}^{-1}}, \bar{\xi}_{\mathbf{H}\mathbf{M}^T}) = (\bar{\xi}_{\mathbf{G}} \cdot \mathbf{M}^{-1}, \bar{\xi}_{\mathbf{H}} \cdot \mathbf{M}^T).$$

Therefore, the metric (8) *is invariant along the set* (3).

A simple and efficient retraction is provided by the following formulas

$$\begin{aligned} R_{\mathbf{G}}(s\bar{\xi}_{\mathbf{G}}) &= \mathbf{G} + s\bar{\xi}_{\mathbf{G}}, \\ R_{\mathbf{H}}(s\bar{\xi}_{\mathbf{H}}) &= \mathbf{H} + s\bar{\xi}_{\mathbf{H}}. \end{aligned} \tag{10}$$

3.3. Geometry of the Polar Factorization

The space $\mathrm{St}(r, d_1) \times S_+(r) \times \mathrm{St}(r, d_2)$ is endowed with the metric

$$\begin{aligned} &\bar{g}((\bar{\xi}_{\mathbf{U}}, \bar{\xi}_{\mathbf{B}}, \bar{\xi}_{\mathbf{V}}), (\bar{\zeta}_{\mathbf{U}}, \bar{\zeta}_{\mathbf{B}}, \bar{\zeta}_{\mathbf{V}})) = \\ &\quad \mathrm{Tr}(\bar{\xi}_{\mathbf{U}}^T \bar{\zeta}_{\mathbf{U}}) + \mathrm{Tr}(\mathbf{B}^{-1}\bar{\xi}_{\mathbf{B}}\mathbf{B}^{-1}\bar{\zeta}_{\mathbf{B}}) + \mathrm{Tr}(\bar{\xi}_{\mathbf{V}}^T \bar{\zeta}_{\mathbf{V}}). \end{aligned} \tag{11}$$

Proposition 3.3. *The quotient manifold (5) endowed with the Riemannian metric (11) admits a set of horizontal vectors $(\bar{\xi}_{\mathbf{U}}, \bar{\xi}_{\mathbf{B}}, \bar{\xi}_{\mathbf{V}})$ defined as*

$$\begin{aligned} \bar{\xi}_{\mathbf{U}} &= \mathbf{U}\mathrm{Skew}(\mathbf{A}) + \mathbf{U}_\perp, \, \mathbf{A} \in \mathbb{R}^{r\times r}, \; \mathbf{U}_\perp^T \mathbf{U} = 0, \\ \bar{\xi}_{\mathbf{B}} &= \mathbf{B}^{\frac{1}{2}} \mathrm{Sym}(\mathbf{\Delta}) \mathbf{B}^{\frac{1}{2}}, \quad \mathbf{\Delta} \in \mathbb{R}^{r\times r}, \\ \bar{\xi}_{\mathbf{V}} &= \mathbf{V}\mathrm{Skew}(\mathbf{D}) + \mathbf{V}_\perp, \, \mathbf{D} \in \mathbb{R}^{r\times r}, \; \mathbf{V}_\perp^T \mathbf{V} = 0, \end{aligned}$$

with the additional condition

$$\mathbf{B}(\mathrm{Skew}(\mathbf{A}) + \mathrm{Skew}(\mathbf{D}))\mathbf{B} = \bar{\xi}_{\mathbf{B}}\mathbf{B} - \mathbf{B}\bar{\xi}_{\mathbf{B}}.$$

We have defined the functions $\mathrm{Sym}(\mathbf{\Delta}) = (\mathbf{\Delta} + \mathbf{\Delta}^T)/2$ and $\mathrm{Skew}(\mathbf{A}) = (\mathbf{A} - \mathbf{A}^T)/2$. We now show that metric (11) is invariant along the equivalence classes (6).

Proposition 3.4. *Let $\xi_{[(\mathbf{U},\mathbf{B},\mathbf{V})]}$ be a tangent vector to the quotient (5) at $[(\mathbf{U}, \mathbf{B}, \mathbf{V})]$. The horizontal lifts of $\xi_{[(\mathbf{U},\mathbf{B},\mathbf{V})]}$ at $(\mathbf{U}, \mathbf{B}, \mathbf{V})$ and at $(\mathbf{UO}, \mathbf{O}^T\mathbf{BO}, \mathbf{VO})$ are related as follow, $\forall \mathbf{O} \in \mathcal{O}(r)$,*

$$(\bar{\xi}_{\mathbf{UO}}, \bar{\xi}_{\mathbf{O}^T\mathbf{BO}}, \bar{\xi}_{\mathbf{VO}}) = (\bar{\xi}_{\mathbf{U}} \cdot \mathbf{O}, \mathbf{O}^T \cdot \bar{\xi}_{\mathbf{B}} \cdot \mathbf{O}, \bar{\xi}_{\mathbf{V}} \cdot \mathbf{O}).$$

Therefore, the metric (11) is invariant along (6).

We choose the following retraction

$$\begin{aligned} R_{\mathbf{U}}(s\bar{\xi}_{\mathbf{U}}) &= \mathrm{qf}(\mathbf{U} + s\bar{\xi}_{\mathbf{U}}), \\ R_{\mathbf{B}}(s\bar{\xi}_{\mathbf{B}}) &= \mathbf{B}^{\frac{1}{2}} \exp(s\mathbf{B}^{-\frac{1}{2}}\bar{\xi}_{\mathbf{B}}\mathbf{B}^{-\frac{1}{2}})\mathbf{B}^{\frac{1}{2}}, \\ R_{\mathbf{V}}(s\bar{\xi}_{\mathbf{V}}) &= \mathrm{qf}(\mathbf{V} + s\bar{\xi}_{\mathbf{V}}), \end{aligned} \tag{12}$$

where $\mathrm{qf}(\cdot)$ is a function that extracts the orthogonal factor of the QR-factorization of its argument.

4. Linear Regression under Fixed-Rank Constraints

In this section, we exploit the quotient geometries presented previously to develop line-search algorithms for the following regression problem. Given data matrix instances $\mathbf{X} \in \mathbb{R}^{d_2 \times d_1}$, scalar observations $y \in \mathbb{R}$, and a linear regression model expressed as $\hat{y} = \mathrm{Tr}(\mathbf{WX})$, solve

$$\min_{\mathbf{W}\in\mathbb{R}^{d_1\times d_2}} \mathbb{E}_{\mathbf{X},y}\{\ell(\hat{y}, y)\}, \quad \text{s.t.} \quad \mathrm{rank}(\mathbf{W}) = r. \tag{13}$$

The loss $\ell(\hat{y}, y)$ penalizes the discrepancy between the observed value y and the value predicted by the model $\hat{y}$. A typical choice for regression is $\ell(\hat{y}, y) = \frac{1}{2}(\hat{y} - y)^2$. Although the focus of this paper is on linear regression and on a quadratic cost function, more general cost functions can be treated equally.

Since $\mathbb{E}_{\mathbf{X},y}\{\cdot\}$ cannot generally be computed explicitely, classical iterative approaches such as batch gradient descent minimize instead the empirical risk

$$f_n(\mathbf{W}) = \frac{1}{n}\sum_{i=1}^{n} \ell(\hat{y}_i, y_i),$$

over a finite dataset $\{(\mathbf{X}_i, y_i)\}_{i=1}^n$. Online gradient descent algorithms consider datasets with possibly infinite number of samples $\{(\mathbf{X}_t, y_t)\}_{t\geq 0}$, and process the samples one at a time or by mini-batches. At time t, online algorithms minimize the instantaneous cost

$$f_t(\mathbf{W}) = \frac{1}{b}\sum_{\tau=t}^{t+b} \ell(\hat{y}_\tau, y_\tau),$$

where b is the mini-batch size. When $b = 1$, one recovers plain stochastic gradient descent. To shorten the exposition, the development are presented for stochastic gradient descent versions of algorithms only. We denote by f the function minimized at each iteration.

4.1. Regression with a Balanced Factorization

We now derive a line-search algorithm to solve (13) using a balanced factorization of $\mathbf{W} = \mathbf{GH}^T$. With this factorization, the cost function is

$$f(\mathbf{G}, \mathbf{H}) = \frac{1}{2}(\mathrm{Tr}(\mathbf{GH}^T\mathbf{X}) - y)^2.$$

Applying formula (7) to this cost function yields

$$\begin{aligned} \overline{\mathrm{grad}_{\mathbf{G}}\, f} &= (\hat{y} - y)\mathbf{X}^T\mathbf{H}(\mathbf{G}^T\mathbf{G}), \\ \overline{\mathrm{grad}_{\mathbf{H}}\, f} &= (\hat{y} - y)\mathbf{XG}(\mathbf{H}^T\mathbf{H}). \end{aligned}$$

Combining the horizontal gradient of this cost function with retraction (10) yields the online algorithm

$$\begin{aligned} \widetilde{\mathbf{G}}_t &= \mathbf{G}_t - s_t(\hat{y}_t - y_t)\mathbf{X}_t^T\mathbf{H}_t(\mathbf{G}_t^T\mathbf{G}_t), \\ \widetilde{\mathbf{H}}_t &= \mathbf{H}_t - s_t(\hat{y}_t - y_t)\mathbf{X}_t\mathbf{G}_t(\mathbf{H}_t^T\mathbf{H}_t). \end{aligned} \tag{14}$$

This update has computational complexity $O(d_1 d_2 r)$.

To balance a given factorization $\widetilde{\mathbf{W}}_t = \widetilde{\mathbf{G}}_t \widetilde{\mathbf{H}}_t^T$, Helmke & Moore (1996) propose the update,

$$\begin{aligned} \mathbf{G}_{t+1} &= \widetilde{\mathbf{G}}_t \exp(\alpha_t(\widetilde{\mathbf{H}}_t^T \widetilde{\mathbf{H}}_t - \widetilde{\mathbf{G}}_t^T \widetilde{\mathbf{G}}_t)), \\ \mathbf{H}_{t+1} &= \widetilde{\mathbf{H}}_t \exp(\alpha_t(\widetilde{\mathbf{G}}_t^T \widetilde{\mathbf{G}}_t - \widetilde{\mathbf{H}}_t^T \widetilde{\mathbf{H}}_t)), \end{aligned} \tag{15}$$

with a step size $\alpha_t = 1/(2\lambda_{max}(\widetilde{\mathbf{G}}_t^T \widetilde{\mathbf{G}}_t + \widetilde{\mathbf{H}}_t^T \widetilde{\mathbf{H}}_t))$. The complexity of a balancing step is $O((d_1 + d_2)r^2 + r^3)$. Notice that $\mathbf{G}_{t+1}\mathbf{H}_{t+1}^T = \widetilde{\mathbf{W}}_t$ and that the fixed points of (15) are balanced. A justification for the step size selection along with a detailed convergence proof is provided in Helmke & Moore (1996, Theorem 6.1).

The proposed cascaded algorithm asymptotically converges to a local minimum of the cost function with a balanced factorization. The insight comes from geometry: (14) is a gradient update on the quotient manifold $(\mathbb{R}_*^{d_1 \times r} \times \mathbb{R}_*^{d_2 \times r})/\mathrm{GL}(r)$, it is unaffected by the choice of the representative $(\mathbf{G}, \mathbf{H})$ in the fiber (3). In contrast, (15) is a gradient update in the fiber (3) for the cost function (4). In the quotient manifold, algorithm (14) is "blind" to the change of representative that results from (15). The sequence of iterates thus converges to a fiber that minimizes the cost function. But the balancing algorithm (15) guarantees that the asymptotic factorization also minimizes the cost (4), implying the balancing condition $\mathbf{G}^T\mathbf{G} = \mathbf{H}^T\mathbf{H}$.

Connection with Existing Work The proposed algorithm is closely related to the gradient descent version of MMMF (Rennie & Srebro, 2005). However, in contrast to the gradient descent version of MMMF, the proposed algorithm is invariant along an equivalence class (3). This resolves the issue of choosing an appropriate step size when there is a discrepancy between $\|\mathbf{G}\|_F$ and $\|\mathbf{H}\|_F$. This situation leads to a slow convergence of the MMMF algorithm, whereas the proposed algorithm is not affected (Figure 2). To illustrate this effect, the two algorithms are compared in batch mode with data generated from the model (18). For both algorithms, the step size is computed using the Armijo rule (Nocedal & Wright, 2006). The initial discrepancy between the factors is $\|\mathbf{G}_0\|_F \approx 5\|\mathbf{H}_0\|_F$.

The Loreta algorithm (Shalit et al., 2010) also fits in the considered optimization framework. This algorithm relies on an embedded geometry of $\mathcal{F}(r, d_1, d_2)$, while the focus of this paper is on quotient geometries. For rank-one data, Loreta has the same complexity as update (14) but a significantly larger constant factor.

Optimizing the Algorithm for Rank-One Data Updates (14)-(15) can be optimized in the setting $\mathbf{X}_t = \mathbf{x}_t\mathbf{z}_t^T$, with $\mathbf{x}_t \in \mathbb{R}^{d_2}$ and $\mathbf{z}_t \in \mathbb{R}^{d_1}$. This setting is important for applications (see Section 5).

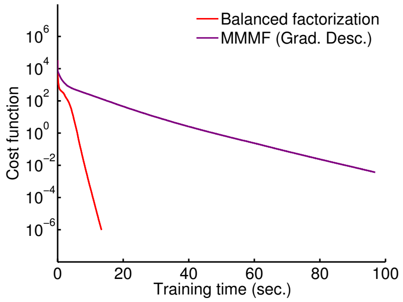


Figure 2. The proposed algorithm resolves the issue of choosing an appropriate step size when there is a discrepancy between $\|\mathbf{G}\|_F$ and $\|\mathbf{H}\|_F$, a situation that leads to a slow convergence of the MMMF algorithm.

Algorithm 1 summarizes the optimizations. The complexity of update (14) reduces to $O((d_1 + d_2)r)$. As we observe empirically that (15) rapidly converges to a balanced factorization, the balancing update is performed only every τ_B iterations to ensure a good numerical conditioning of the algorithm.

Regularization Although the rank constraint already performs a spectral regularization on $\mathbf{W}$, it is useful in practice to add a pointwise regularization

$$f(\mathbf{G}, \mathbf{H}) = \frac{1}{2}(\mathrm{Tr}(\mathbf{G}\mathbf{H}^T\mathbf{X}) - y)^2 + \frac{\lambda}{2}\|\mathbf{G}\mathbf{H}^T\|_F^2.$$

The regularizer $\|\mathbf{G}\mathbf{H}^T\|_F^2$ is chosen because it is invariant along the set of equivalence classes (3) as opposed to the common choice $\|\mathbf{G}\|_F^2 + \|\mathbf{H}\|_F^2$ (Rennie & Srebro, 2005). Update (14) becomes

$$\begin{aligned} \widetilde{\mathbf{G}}_t &= \mathbf{G}_t - s_t(\hat{y}_t - y_t)\mathbf{X}_t^T\mathbf{L}_t - \lambda\mathbf{R}_t(\mathbf{G}_t^T\mathbf{G}_t), \\ \widetilde{\mathbf{H}}_t &= \mathbf{H}_t - s_t(\hat{y}_t - y_t)\mathbf{X}_t\mathbf{R}_t - \lambda\mathbf{L}_t(\mathbf{H}_t^T\mathbf{H}_t), \end{aligned}$$

where $\mathbf{L}_t = \mathbf{H}_t(\mathbf{G}_t^T\mathbf{G}_t)$ and $\mathbf{R}_t = \mathbf{G}_t(\mathbf{H}_t^T\mathbf{H}_t)$. This modification does not increase significantly the computational cost, since the regularization term have common subexpressions with the gradient of the loss. However, a more efficient approach is obtained using the polar factorization $\mathbf{W} = \mathbf{U}\mathbf{B}\mathbf{V}^T$.

4.2. Regression with Cheap Regularization

With the factorization $\mathbf{W} = \mathbf{U}\mathbf{B}\mathbf{V}^T$, the cost becomes

$$f(\mathbf{U}, \mathbf{B}, \mathbf{V}) = \frac{1}{2}(\mathrm{Tr}(\mathbf{U}\mathbf{B}\mathbf{V}^T\mathbf{X}) - y)^2. \tag{16}$$

Applying formula (7) to this cost function and projecting onto the set of horizontal vectors gives us

$$\begin{aligned} \overline{\mathrm{grad}_{\mathbf{U}} f} &= (\hat{y} - y)(\mathbf{X}^T\mathbf{V}\mathbf{B} - \mathbf{U}\mathrm{Sym}(\mathbf{U}^T\mathbf{X}^T\mathbf{V}\mathbf{B})), \\ \overline{\mathrm{grad}_{\mathbf{B}} f} &= (\hat{y} - y)\mathbf{B}\mathrm{Sym}(\mathbf{V}^T\mathbf{X}\mathbf{U})\mathbf{B}, \\ \overline{\mathrm{grad}_{\mathbf{V}} f} &= (\hat{y} - y)(\mathbf{X}\mathbf{U}\mathbf{B} - \mathbf{V}\mathrm{Sym}(\mathbf{V}^T\mathbf{X}\mathbf{U}\mathbf{B})). \end{aligned}$$

Algorithm 1 Balanced factorization

```
Input: $\{(\mathbf{x}_t, \mathbf{z}_t, y_t)\}_{t\geq 0}$ (dataset), $(\mathbf{G}_0, \mathbf{H}_0)$ (initial factorization), $T \geq 0$ (number of iterations), $s_t > 0$ (sequence of step size), $\tau_B \geq 1$ (balancing period).
Output: a factorization $(\mathbf{G}_T, \mathbf{H}_T)$.
Set $\tau = \tau_B$, and compute $\mathbf{S}_0 = \mathbf{G}_0^T\mathbf{G}_0$, $\mathbf{R}_0 = \mathbf{H}_0^T\mathbf{H}_0$
for $t = 0$ to $T - 1$ do
  Pick a sample $(\mathbf{x}_t, \mathbf{z}_t, y_t)$
  Set $\bar{\mathbf{x}}_t = \mathbf{H}_t^T\mathbf{x}_t$, $\bar{\mathbf{z}}_t = \mathbf{G}_t^T\mathbf{z}_t$, $\mathbf{s}_t = \mathbf{S}_t\bar{\mathbf{x}}_t$, $\mathbf{r}_t = \mathbf{R}_t\bar{\mathbf{z}}_t$
  Predict $\hat{y}_t = \bar{\mathbf{x}}_t^T\bar{\mathbf{z}}_t$ and set $\beta_t = s_t(\hat{y}_t - y_t)$
  Update $\widetilde{\mathbf{G}}_t = \mathbf{G}_t - \beta_t\mathbf{z}_t\mathbf{s}_t^T$
  Update $\widetilde{\mathbf{H}}_t = \mathbf{H}_t - \beta_t\mathbf{x}_t\mathbf{r}_t^T$
  Update $\widetilde{\mathbf{S}}_t = \mathbf{S}_t - \beta_t\bar{\mathbf{z}}_t\mathbf{s}_t^T - \beta_t\mathbf{s}_t\bar{\mathbf{z}}_t^T + \beta_t^2\|\bar{\mathbf{z}}_t\|_2^2\mathbf{s}_t\mathbf{s}_t^T$
  Update $\widetilde{\mathbf{R}}_t = \mathbf{R}_t - \beta_t\bar{\mathbf{x}}_t\mathbf{r}_t^T - \beta_t\mathbf{r}_t\bar{\mathbf{x}}_t^T + \beta_t^2\|\bar{\mathbf{x}}_t\|_2^2\mathbf{r}_t\mathbf{r}_t^T$
  Set $\tau = \tau - 1$
  if $\tau \leq 0$ then
    Perform a balancing step with (15)
    Set $\mathbf{S}_{t+1} = \mathbf{G}_{t+1}^T\mathbf{G}_{t+1}$, $\mathbf{R}_{t+1} = \mathbf{H}_{t+1}^T\mathbf{H}_{t+1}$
    Set $\tau = \tau_B$
  else
    Set $\mathbf{G}_{t+1} = \widetilde{\mathbf{G}}_t$ and $\mathbf{H}_{t+1} = \widetilde{\mathbf{H}}_t$
    Set $\mathbf{S}_{t+1} = \widetilde{\mathbf{S}}_t$ and $\mathbf{R}_{t+1} = \widetilde{\mathbf{R}}_t$
  end if
end for
```

Combining this gradient with the retraction (12) yields

$$\begin{aligned}
\mathbf{U}_{t+1} &= \mathrm{qf}(\mathbf{U}_t - s_t e_t(\mathbf{Y}_t - \mathbf{U}_t\mathrm{Sym}(\mathbf{U}_t^T\mathbf{Y}_t))),\\
\mathbf{B}_{t+1} &= \mathbf{B}_t^{\frac{1}{2}}\exp(-s_t e_t\mathbf{B}_t^{\frac{1}{2}}\mathrm{Sym}(\mathbf{\Psi}_t)\mathbf{B}_t^{\frac{1}{2}})\mathbf{B}_t^{\frac{1}{2}}, \qquad (17)\\
\mathbf{V}_{t+1} &= \mathrm{qf}(\mathbf{V}_t - s_t e_t(\mathbf{Z}_t - \mathbf{U}_t\mathrm{Sym}(\mathbf{U}_t^T\mathbf{Z}_t))),
\end{aligned}$$

where we have defined $e_t = \hat{y}_t - y_t$, $\mathbf{Y}_t = \mathbf{X}_t^T\mathbf{V}_t\mathbf{B}_t$, $\mathbf{Z}_t = \mathbf{X}_t\mathbf{U}_t\mathbf{B}_t$, and $\mathbf{\Psi}_t = \mathbf{V}_t^T\mathbf{X}_t\mathbf{U}_t$.

Connection with Existing Work The OptSpace algorithm (Keshavan et al., 2010) also relies on the factorization $\mathbf{W} = \mathbf{U}\mathbf{B}\mathbf{V}^T$, but with $\mathbf{B} \in \mathbb{R}^{r\times r}$. It alternates between a gradient descent step on $\mathbf{U}$ and $\mathbf{V}$ in the Grassmann manifold for fixed $\mathbf{B}$, and a least-square estimation of $\mathbf{B}$ for fixed $\mathbf{U}$ and $\mathbf{V}$. The proposed algorithm is different from OptSpace in the choice $\mathbf{B}$ positive definite versus $\mathbf{B} \in \mathbb{R}^{r\times r}$. As a consequence, each step of the algorithm retains the geometry of a SVD factorization. Our algorithm also differs from OptSpace in the simultaneous and progressive nature of the updates. Furthermore, the choice $\mathbf{B} \succ 0$ allows us to derive alternative updates based on different metrics on the set $S_+(r)$. This flexibility is exploited in Meyer et al. (2011) to show that metrics of $S_+(r)$ are connected to Bregman divergences and information geometry.

Algorithm 2 Polar factorization

```
Input: $\{(\mathbf{x}_t, \mathbf{z}_t, y_t)\}_{t\geq 0}$ (dataset), $(\mathbf{U}_0, \mathbf{B}_0, \mathbf{V}_0)$ (initial factorization), $T \geq 0$ (number of iterations), $s_t > 0$ (sequence of step size).
Output: a factorization $(\mathbf{U}_T, \mathbf{B}_T, \mathbf{V}_T)$
for $t = 0$ to $T - 1$ do
  Pick a sample $(\mathbf{x}_t, \mathbf{z}_t, y_t)$
  Set $\bar{\mathbf{x}}_t = \mathbf{V}_t^T\mathbf{x}_t$, $\bar{\mathbf{z}}_t = \mathbf{U}_t^T\mathbf{z}_t$
  Set $\mathbf{s}_t = \mathbf{B}_t^{\frac{1}{2}}\bar{\mathbf{x}}_t$, $\mathbf{r}_t = \mathbf{B}_t^{\frac{1}{2}}\bar{\mathbf{z}}_t$, $\bar{\mathbf{s}}_t = \mathbf{B}_t^{\frac{1}{2}}\mathbf{s}_t$, $\bar{\mathbf{r}}_t = \mathbf{B}_t^{\frac{1}{2}}\mathbf{r}_t$
  Predict $\hat{y}_t = \mathbf{r}_t^T\mathbf{s}_t$ and set $e_t = \hat{y}_t - y_t$
  Set $\mathbf{U}_{t+1} = \mathrm{qf}(\mathbf{U}_t - s_t e_t(\mathbf{z}_t\bar{\mathbf{s}}_t^T - \mathbf{U}_t\mathrm{Sym}(\bar{\mathbf{z}}_t\bar{\mathbf{s}}_t^T)))$
  Set $\mathbf{B}_{t+1} = \mathbf{B}_t^{\frac{1}{2}}\exp(-s_t e_t\mathrm{Sym}(\mathbf{s}_t\mathbf{r}_t^T))\mathbf{B}_t^{\frac{1}{2}}$
  Set $\mathbf{V}_{t+1} = \mathrm{qf}(\mathbf{V}_t - s_t e_t(\mathbf{x}_t\bar{\mathbf{r}}_t^T - \mathbf{V}_t\mathrm{Sym}(\bar{\mathbf{x}}_t\bar{\mathbf{r}}_t^T)))$
end for
```

The SVP algorithm (Jain et al., 2010) is based on the SVD factorization $\mathbf{W} = \mathbf{U}\mathbf{B}\mathbf{V}^T$ with $\mathbf{B}$ diagonal. The algorithm can be interpreted in the considered framework as a gradient descent algorithm along with an efficient SVD-based retraction exploiting the sparse structure of the gradient.

Simonsson & Eldén (2010) considered the variant factorization $\mathbf{W} = \mathbf{U}\mathbf{Z}^T$, where $\mathbf{U} \in \mathrm{St}(r, d_1)$ and $\mathbf{Z} \in \mathbb{R}_*^{d_2\times r}$. Although they propose a Newton's algorithm, the corresponding gradient descent version directly fits into the considered optimization framework.

This variant factorization is also exploited by the GROUSE algorithm (Balzano et al., 2010). This online algorithm estimates the subspace $\mathbf{U}$ with a gradient descent on the Grassmann manifold while the remaining factor $\mathbf{Z}$ is computed using least-squares.

Optimizing the Algorithm for Rank-One Data Algorithm 2 summarizes the optimizations. The qf function can be implemented using rank-one updates of the QR factorization (Daniel et al., 1976). This reduces the cost of a QR factorization to $O(dr)$, compared to $O(dr^2)$ when it is computed from scratch.

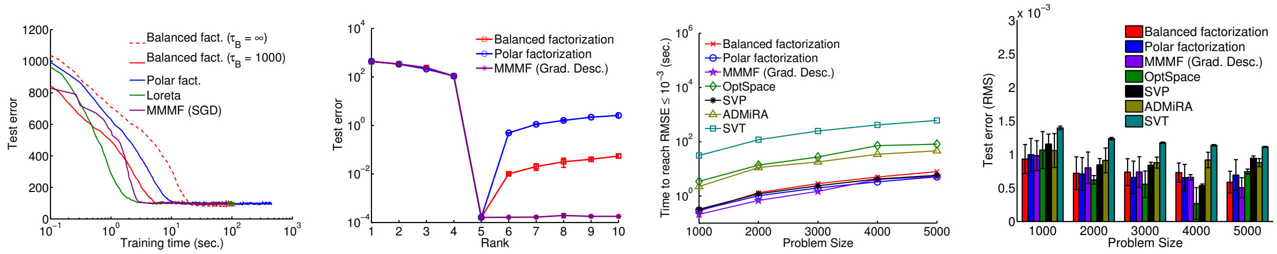
Each update of Algorithm 2 costs $O((d_1 + d_2)r + r^3)$.

Regularization With a regularizer $\|\mathbf{W}\|_F^2$, the cost function of interest becomes

$$f(\mathbf{U}, \mathbf{B}, \mathbf{V}) = \frac{1}{2}(\mathrm{Tr}(\mathbf{U}\mathbf{B}\mathbf{V}^T\mathbf{X}) - y)^2 + \frac{\lambda}{2}\|\mathbf{B}\|_F^2,$$

and only the update of $\mathbf{B}$ needs the cheap modification

$$\mathbf{B}_{t+1} = \mathbf{B}_t^{\frac{1}{2}}\exp(-s_t(\mathbf{B}_t^{\frac{1}{2}}(e_t\mathrm{Sym}(\mathbf{\Psi}_t) + \lambda\mathbf{B}_t)\mathbf{B}_t^{\frac{1}{2}})\mathbf{B}_t^{\frac{1}{2}}.$$



(a) (b) (c) (d)

Figure 3. **Left.** Learning on pairs (toy data): both online (a) and batch (b) algorithms perform well compared to the existing approaches. **Right.** Matrix completion on synthetic data. The proposed algorithms compete with state-of-the-art low-rank matrix completion algorithms, both in terms of time to reach convergence (c) and test error (d).

5. Experiments

We now focus on two applications and illustrate the behavior of the proposed algorithms on benchmarks.

5.1. Learning on Pairs

Given data $\mathbf{x} \in \mathbb{R}^{d_1}$ and $\mathbf{z} \in \mathbb{R}^{d_2}$ associated with two type of samples, learning on pairs amounts to learn a model $\hat{y} : \mathbb{R}^{d_1} \times \mathbb{R}^{d_2} \to \mathbb{R}$ from data $\{(\mathbf{x}_i, \mathbf{z}_i, y_i)\}_{i=1}^n$. When the regression model $\hat{y}$ is a fixed-rank bilinear form $\hat{y} = \mathbf{z}^T \mathbf{W} \mathbf{x}$, the problem boils down to the considered regression problem with the choice $\mathbf{X} = \mathbf{x}\mathbf{z}^T$.

Applications of this learning framework include graph inference (Bleakley & Yamanishi, 2009) and collaborative filtering with attributes (Abernethy et al., 2009).

Toy Data We generate data according to

$$y_i = \mathbf{z}_i^T \mathbf{W}^\star \mathbf{x}_i + \epsilon_i, \quad i = 1, ..., n, \tag{18}$$

where $\mathbf{W}^\star \in \mathcal{F}(50, 25, 5)$, $\mathbf{z}_i \in \mathbb{R}^{50}$ and $\mathbf{x}_i \in \mathbb{R}^{25}$ have entries drawn from a standard Gaussian distribution. Gaussian noise $\epsilon_i \sim \mathcal{N}(0, 10^{-2})$ is added to the data.

We first show that the proposed algorithms perform well in the online setting (Figure 3(a)). The dataset is generated from (18), 40,000 samples are used for learning and 10,000 for testing. At each iteration, the algorithms pick a sample at random and update the model. The algorithms all process the same set of samples. The step size is selected during a pre-training phase of 5,000 iterations, the step size leading to the smallest train error is retained. Figure 3(a) reports the test error as a function of the training time. The proposed algorithms compete with Loreta (Shalit et al., 2010) and with an online version of MMMF. Balancing reduces the time to achieve convergence.

We now test the proposed algorithms in batch mode. Using (18), we generate a dataset of 3,000 samples and compute the test error as a function of the approximation rank (Figure 3(b)). The validation protocol is 90/10 train/test split. The results are averaged over 10 random partitions. The regularization parameter λ is selected using cross-validation. Not surprisingly, the competing algorithms all achieve a minimal error when the rank equals the rank of the target model. When the rank further increases, the algorithms start overfitting. These observations suggest to increase progressively the value of the rank until satisfactory results are achieved or until performance degrades.

5.2. Low-Rank Matrix Completion

Let $\mathbf{W}^\star \in \mathbb{R}^{d_1 \times d_2}$ be a matrix whose entries $\mathbf{W}^\star_{ij}$ are given only for some $(i, j) \in \Omega$. The set Ω is a subset of the complete set of entries. Low-rank matrix completion amounts to solve

$$\min_{\mathbf{W} \in \mathbb{R}^{d_1 \times d_2}} \|\mathcal{P}_\Omega(\mathbf{W}) - \mathcal{P}_\Omega(\mathbf{W}^\star)\|_F^2, \text{ s.t. } \operatorname{rank}(\mathbf{W}) = r,$$

where $\mathcal{P}_\Omega(\mathbf{W})_{ij} = \mathbf{W}_{ij}$ if $(i, j) \in \Omega$, and 0 otherwise.

This problem is recast in the considered regression framework if each known entry $\mathbf{W}^\star_{ij}$ with $(i, j) \in \Omega$ is an observation y_{ij} and $\mathbf{X}_{ij} = \mathbf{e}_j \mathbf{e}_i^T$, where $\mathbf{e}_j \in \mathbb{R}^{d_2}$ and $\mathbf{e}_i \in \mathbb{R}^{d_1}$ are canonical basis vectors. This gives us the regression model $\hat{y}_{ij} = \operatorname{Tr}(\mathbf{W}\mathbf{e}_j\mathbf{e}_i^T) = \mathbf{W}_{ij}$.

The proposed algorithms are run in batch mode and are compared to: a gradient descent version of MMMF (Rennie & Srebro, 2005), OptSpace (Keshavan et al., 2010), SVP (Jain et al., 2010), ADMiRA (Lee & Bresler, 2010), a matching pursuit based algorithm, and SVT (Cai et al., 2007), a nuclear norm minimization based algorithm. We use Matlab code provided by the respective authors except for MMMF for which we use our own implementation.

Synthetic Data with Uniform Sampling Following Jain et al. (2010), we generate random rank-2 matrices $\mathbf{W}^\star \in \mathbb{R}^{d\times d}$ of various sizes d and sample a fraction $p = 0.1$ of entries for learning. Figure 3(c) reports the time taken by the algorithms to reach a root mean square error (RMSE) of 10^{-3} on the learning set. The corresponding RMSE on the test set is presented in Figure 3(d). Results are averaged over 10 runs. The proposed algorithms compete with the other methods both in terms of convergence speed and test error.

MovieLens Data Finally, we compare the fixed-rank factorization based algorithms on the 1M MovieLens collaborative filtering data, which contains one million ratings for 6,040 users and 3,952 movies. We average the test RMSE for different values of the rank over 10 random 90/10 train/test partitions. Results are presented in Table 1. The proposed algorithms compete with the other methods. The algorithm based on the polar factorization achieved the smallest RMSE for rank 10 and 12. Standard deviations of the errors are not reported since they were not significant.

Table 1. Test RMSE on Movielens data

r	Bal.	Pol.	MMMF	SVP	Opt.	ADMiRa
2	0.90	0.89	0.88	0.89	0.90	1.11
3	0.90	0.89	0.88	0.88	0.90	1.09
5	0.90	0.87	0.86	0.88	0.90	1.07
7	0.88	0.86	0.86	0.89	0.89	1.04
10	0.88	**0.85**	0.86	0.90	0.89	1.04
12	0.88	**0.85**	0.87	0.92	0.89	1.03

Acknowledgments

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 State, Science Policy Office. GM is supported as an FNRS research fellow (Belgian Fund for Scientific Research). The scientific responsibility rests with its authors.

References

Abernethy, J., Bach, F., Evgeniou, T., and Vert, J.-P. A new approach to collaborative filtering: Operator estimation with spectral regularization. *Journal of Machine Learning Research*, 10(Mar):803–826, 2009.

Absil, P.-A., Mahony, R., and Sepulchre, R. *Optimization Algorithms on Matrix Manifolds*. Princeton University Press, 2008.

Amit, Y., Fink, M., Srebro, N., and Ullman, S. Uncovering shared structures in multiclass classification. In *Proceedings of the 24th International Conference on Machine Learning (ICML)*, 2007.

Bach, F. Consistency of trace norm minimization. *Journal of Machine Learning Research*, 9(Jun):1019–1048, 2008.

Balzano, L., Nowak, R., and Recht, B. Online identification and tracking of subspaces from highly incomplete information. *arXiv:1006.4046v1*, 2010.

Bleakley, K. and Yamanishi, Y. Supervised prediction of drug-target interactions using bipartite local models. *Bioinformatics*, 25(18):2397–2403, 2009.

Cai, D., He, X., and Han, J. Efficient kernel discriminant analysis via spectral regression. *IEEE International Conference on Data Mining (ICDM)*, 2007.

Cai, J.-F., Candes, E. J., and Shen, Z. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 20(4):1956–1982, 2008.

Daniel, J., Gragg, W.B., Kaufman, L., and Stewart, G. W. Reorthogonalization and stable algorithms for updating the Gram-Schmidt QR factorization. *Math. Comp.*, 30: 772–795, 1976.

Evgeniou, T., Micchelli, C.A., and Pontil, M. Learning multiple tasks with kernel methods. *Journal of Machine Learning Research*, 6(Apr):615–637, 2005.

Fazel, M. *Matrix rank minimization with applications*. PhD thesis, Stanford University, 2002.

Helmke, U. and Moore, J. *Optimization and Dynamical Systems*. Springer, 1996.

Jain, P., Meka, R., and Dhillon, I. Guaranteed rank minimization via singular value projection. In *Advances in Neural Information Processing Systems (NIPS)*, 2010.

Keshavan, R., Montanari, A., and Oh, S. Matrix completion from noisy entries. *Journal of Machine Learning Research*, 11(Jul):2057–2078, 2010.

Lee, K. and Bresler, Y. Admira: Atomic decomposition for minimum rank approximation. *IEEE Trans. on Information Theory*, 56(9):4402–4416, 2010.

Meyer, G., Bonnabel, S., and Sepulchre, R. Regression on fixed-rank positive semidefinite matrices: a Riemannian approach. *Journal of Machine Learning Research*, 12 (Feb):593–625, 2011.

Nocedal, J. and Wright, S. J. *Numerical Optimization, Second Edition*. Springer, 2006.

Recht, B., Fazel, M., and Parrilo, P. A. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM Review*, 53(3):471–501, August 2010.

Rennie, J. and Srebro, N. Fast maximum margin matrix factorization for collaborative prediction. In *Proceedings of the 22nd International Conference on Machine Learning (ICML)*, 2005.

Shalit, U., Weinshall, D., and Chechik, G. Online learning in the manifold of low-rank matrices. In *Advances in Neural Information Processing Systems (NIPS)*, 2010.

Simonsson, L. and Eldén, L. Grassmann algorithms for low rank approximation of matrices with missing values. *BIT Numerical Math.*, 50(1):173–191, 2010.