Rank-constrained linear regression: a Riemannian approach

Gilles Meyer*, Silvère Bonnabel◇, and Rodolphe Sepulchre*

*Department of EECS, University of Liège, Belgium
{g.meyer, r.sepulchre}@ulg.ac.be

◇Robotics center, Mines ParisTech, France
silvere.bonnabel@mines-paristech.fr

Abstract

This poster presents novel algorithms for learning a linear regression model whose parameter is a real fixed-rank matrix.

The focus is on the non linear nature of the search space.

Because the set of fixed-rank matrices enjoys a rich Riemannian manifold structure, the theory of line-search algorithms on matrix manifolds can be applied [1].

The resulting algorithms scale to high-dimensional problems, enjoy local convergence properties, and connect with the recent contributions on learning fixed-rank matrices [3,4,5,6,10].

The proposed algorithms generalize our recent work on learning fixed-rank symmetric positive semidefinite matrices [2].

Problem formulation

Given data matrix instances $\mathbf{X} \in \mathbb{R}^{d_2 \times d_1}$, observations $y \in \mathbb{R}$, and a linear regression model $\hat{y} = \mathrm{Tr}(\mathbf{W}\mathbf{X})$, solve

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

The loss function is the quadratic loss $\ell(\hat{y}, y) = \frac{1}{2}(\hat{y} - y)^2$.

In practice, a surrogate cost function for the expectation above is

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

or the instantaneous cost

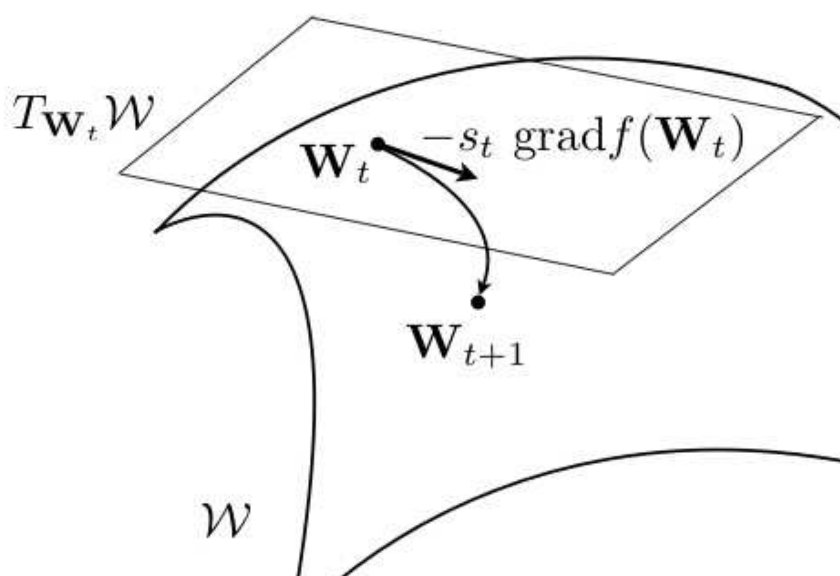
$$f_t(\mathbf{W}) = \ell(\hat{y}_t, y_t), \quad \text{(online algorithms)}.$$

Driving applications

- **Low-rank matrix completion** [3,4,5,10]. Completing the missing entries of a matrix $\mathbf{W}$ given a subset of its entries fits in the considered regression framework. Observations y_{ij} are the known entries and $\mathbf{X}_{ij} = \mathbf{e}_i\mathbf{e}_j^T$ such that $\hat{y}_{ij} = \mathrm{Tr}(\mathbf{W}\mathbf{X}_{ij}) = \mathbf{W}_{ij}$, whenever (i, j) belongs to the set of known entries.
- **Learning on pairs** [7]. Given triplets $(\mathbf{x}, \mathbf{z}, y)$ with $\mathbf{x} \in \mathbb{R}^{d_1}$, $\mathbf{z} \in \mathbb{R}^{d_2}$ and $y \in \mathbb{R}$, learn a regression model $\hat{y} = \mathrm{Tr}(\mathbf{W}\mathbf{z}\mathbf{x}^T) = \mathbf{x}^T\mathbf{W}\mathbf{z}$.
- **Multi-task regression** [8]. Learning of a parameter $\mathbf{W} \in \mathbb{R}^{d \times P}$ that is shared between P related regression problems. The model is given by $\hat{y}_{pi} = \mathrm{Tr}(\mathbf{W}\mathbf{e}_p\mathbf{x}_{pi}^T)$, where $\mathbf{e}_p \in \mathbb{R}^P$ and $\mathbf{x}_{pi} \in \mathbb{R}^d$ is the i-th data for the p-th problem. The cost function typically contains a data fitting term and a term that accounts for the information that is shared between the problems.
- **Ranking** [6]. Compute a relevance score $\hat{y}(\mathbf{x}_i, \mathbf{x}_j) = \mathrm{Tr}(\mathbf{W}\mathbf{x}_j\mathbf{x}_i^T)$ such that $\hat{y}(\mathbf{x}_i, \mathbf{x}_i^+) > \hat{y}(\mathbf{x}_i, \mathbf{x}_i^-)$, whenever $\mathbf{x}_i^+$ is more relevant to $\mathbf{x}_i$ than $\mathbf{x}_i^-$.

A common feature of these problems is that the input matrix $\mathbf{X}$ is rank-one.

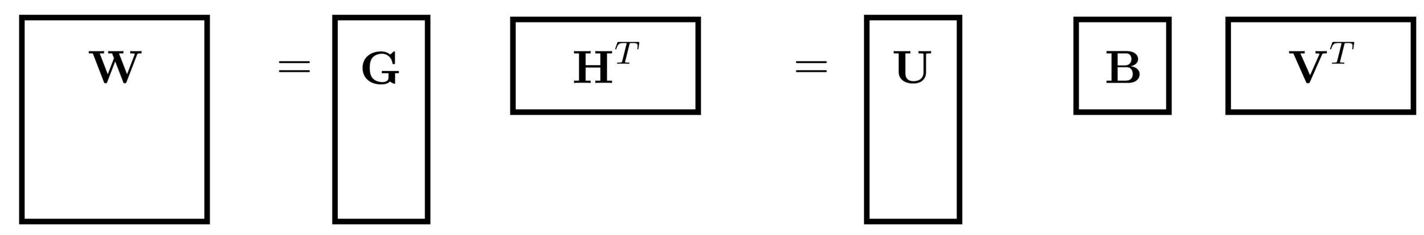
Line-search algorithms on matrix manifolds



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

Gradient iteration on a Riemannian manifold: the search direction $-\mathrm{grad}f(\mathbf{W}_t)$ belongs to the tangent space $T_{\mathbf{W}_t}\mathcal{W}$ and the updated point $\mathbf{W}_{t+1}$ automatically remains inside the manifold.

Fixed-rank factorizations and quotient geometries



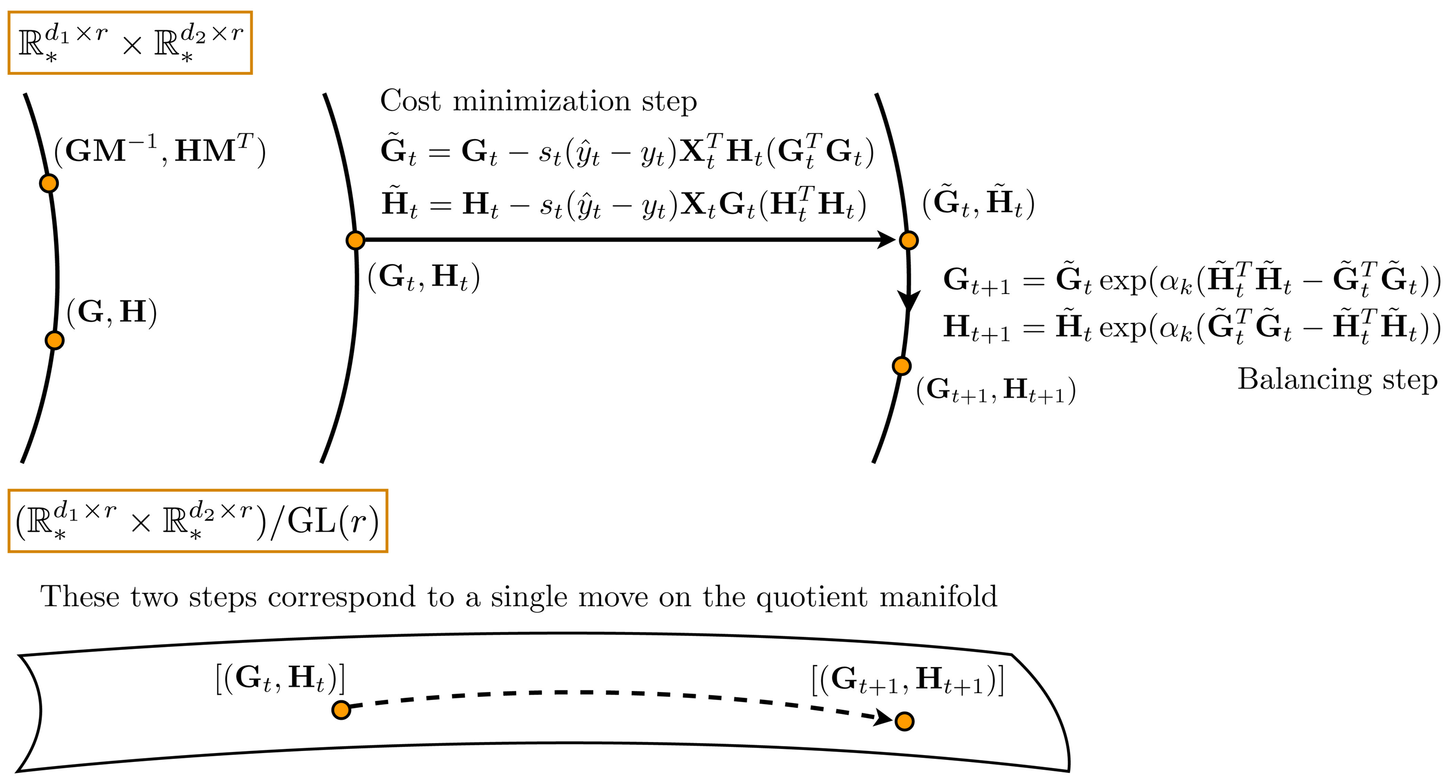
$\mathbb{R}^{d_1 \times d_2}$	$\mathbb{R}_*^{d_1 \times r}$	$\mathbb{R}_*^{d_2 \times r}$	$\mathrm{St}(r, d_1)$	$\mathrm{S}_+(r)$	$\mathrm{St}(r, d_2)$
$\mathrm{rank}(\mathbf{W}) = r$			$\mathbf{U}^T\mathbf{U} = \mathbf{I}$	$\mathbf{B} \succ 0$	$\mathbf{V}^T\mathbf{V} = \mathbf{I}$

$\mathcal{F}(r, d_1, d_2) \quad \simeq \quad (\mathbb{R}_*^{d_1 \times r} \times \mathbb{R}_*^{d_2 \times r})/\mathrm{GL}(r) \quad \simeq \quad (\mathrm{St}(r, d_1) \times \mathrm{S}_+(r) \times \mathrm{St}(r, d_2))/\mathrm{O}(r)$

Equivalence mapping: $(\mathbf{G}, \mathbf{H}) \mapsto (\mathbf{G}\mathbf{M}^{-1}, \mathbf{H}\mathbf{M}^T)$ where $\det(\mathbf{M}) \neq 0$; $(\mathbf{U}, \mathbf{B}, \mathbf{V}) \mapsto (\mathbf{U}\mathbf{O}, \mathbf{O}^T\mathbf{B}\mathbf{O}, \mathbf{V}\mathbf{O})$ where $\mathbf{O}^T\mathbf{O} = \mathbf{O}\mathbf{O}^T = \mathbf{I}$

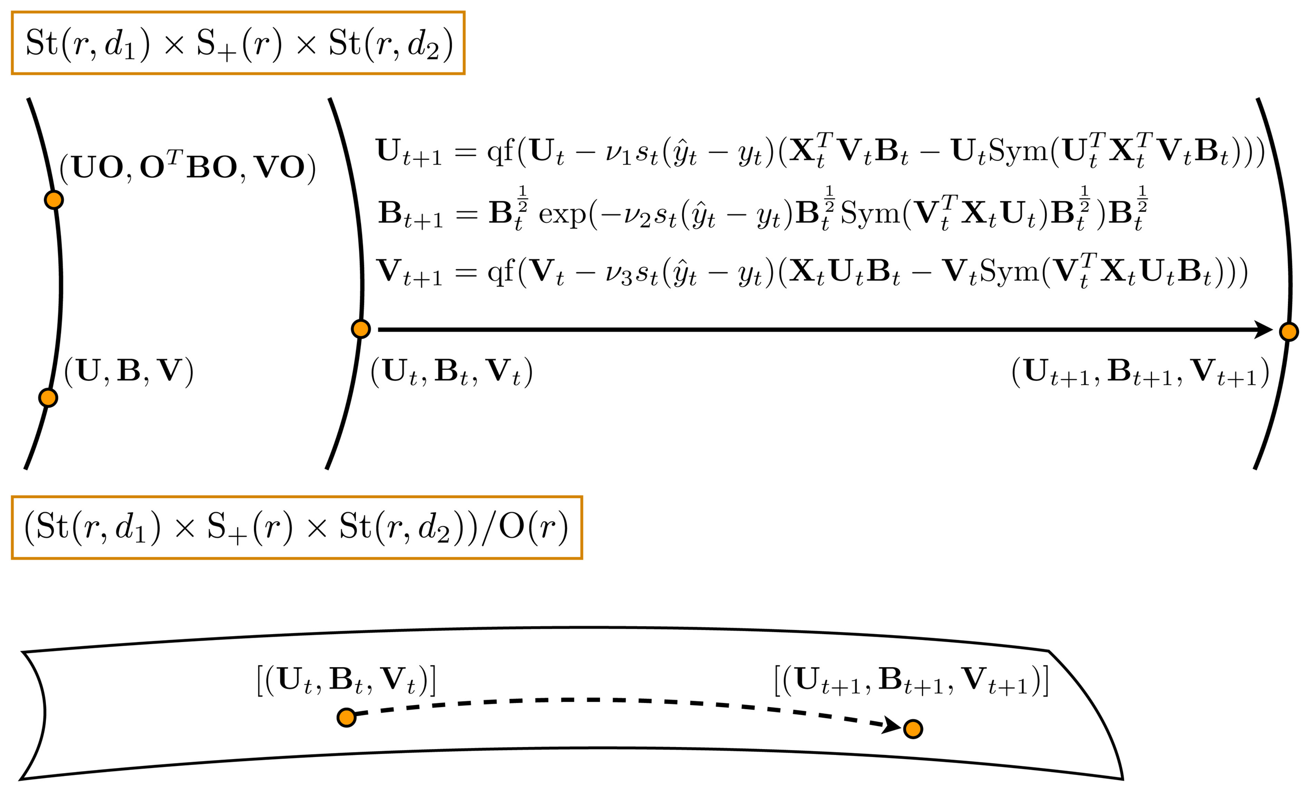
A given element $\mathbf{W}$ of the search space is represented by an entire equivalence class of matrices. Line-search algorithms on the quotient manifold moves from one equivalence class to another.

Linear regression with a balanced factorization



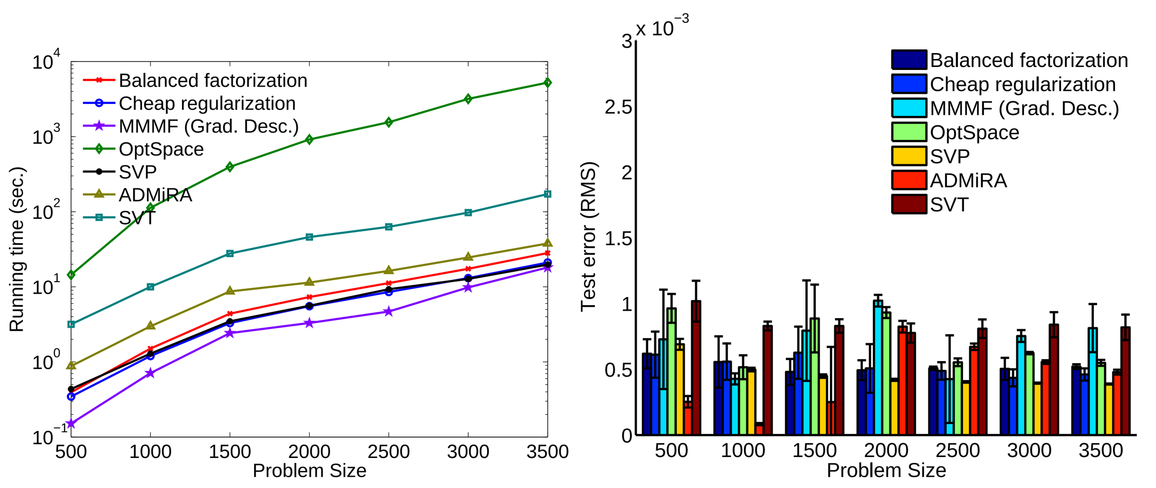
- The algorithm converges to a local minimum of the cost that is a balanced factorization $\mathbf{W} = \mathbf{G}\mathbf{H}^T$ with $\mathbf{G}^T\mathbf{G} = \mathbf{H}^T\mathbf{H}$.
- The balancing step minimizes the function $\Omega(\mathbf{G}, \mathbf{H}) = \|\mathbf{G}\|_F^2 + \|\mathbf{H}\|_F^2$ along a given fiber.
- Balanced factorizations ensure good numerical conditioning and robustness to noise [3].
- The computational complexity is $O(d_1 d_2 r)$, and $O((d_1 + d_2)r^2)$ when $\mathbf{X}$ is rank-one.

Linear regression with cheap regularization



- The algorithm converges to a local minimum of the cost, and the considered factorization automatically encodes the structure of a balanced factorization.
- A regularization on $\|\mathbf{W}\|_F^2$ is equivalent to a cheap regularization on $\|\mathbf{B}\|_F^2$.
- The parameters $\nu_1, \nu_2, \nu_3 \geq 0$ weight the learning of the different matrices $\mathbf{U}$, $\mathbf{B}$ and $\mathbf{V}$.
- The computational complexity is $O(d_1 d_2 r)$, and $O((d_1 + d_2)r^2)$ when $\mathbf{X}$ is rank-one.

Matrix completion (synthetic data)

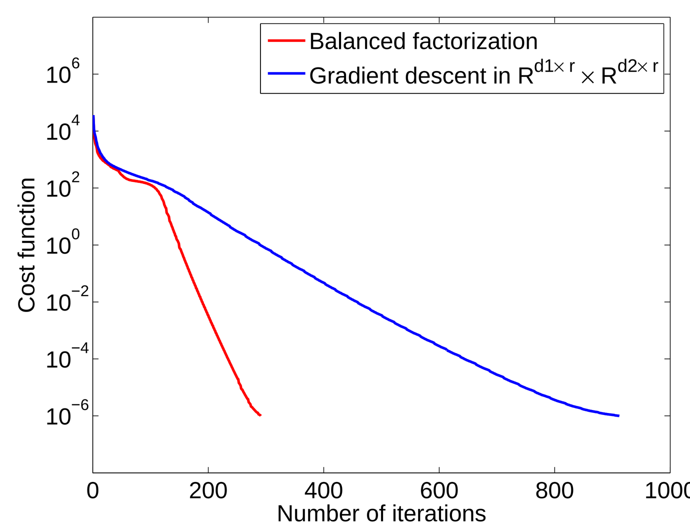


The proposed algorithms compete with the state-of-the-art: MMMF [3], OptSpace [5], SVP [4], ADMiRA [12], SVT [11].

Experimental setup:

- Random rank-2 matrices $\mathbf{W} \in \mathbb{R}^{d \times d}$ for various sizes d;
- A fraction $p = 0.1$ of entries are randomly selected for training (batch mode);
- The competing algorithms all stop when a RMSE ≤ 0.001 is achieved.

A numerical benefit of balancing



The classical gradient descent algorithm in $\mathbb{R}^{d_1 \times r} \times \mathbb{R}^{d_2 \times r}$:

$$\mathbf{G}_{t+1} = \mathbf{G}_t - s_t(\hat{y}_t - y_t)\mathbf{x}_t\mathbf{z}_t^T\mathbf{H}_t, \quad \mathbf{H}_{t+1} = \mathbf{H}_t - s_t(\hat{y}_t - y_t)\mathbf{z}_t\mathbf{x}_t^T\mathbf{G}_t,$$

converges slowly when the factorization is unbalanced (*e.g.* $\|\mathbf{G}\|_F \approx 2\|\mathbf{H}\|_F$).

Experimental setup:

- Regression model: $\hat{y} = \mathbf{x}^T\mathbf{G}\mathbf{H}^T\mathbf{z} + \epsilon, \quad \epsilon \sim \mathcal{N}(0, 10^{-3})$;
- Random data: $\{(\mathbf{x}_i, \mathbf{z}_i, y_i)\}_{i=1}^n$ with $\mathbf{x}_i \in \mathbb{R}^{50}$, $\mathbf{z}_i \in \mathbb{R}^{25}$, $n = 2500$.

References

[1] P.-A. Absil, R. Mahony and R. Sepulchre. *Optimization Algorithms on Matrix Manifolds.* Princeton University press, 2008.

[2] G. Meyer, S. Bonnabel and R. Sepulchre. *Regression on fixed-rank positive semidefinite matrices: a Riemannian approach.* JMLR, accepted pending minor revisions, 2010. http://arxiv.org/abs/1006.1288

[3] J. Rennie and N. Srebro. *Fast maximum margin matrix factorization for collaborative prediction.* ICML, 2005.

[4] R. Meka, P. Jain and I. Dhillon. *Guaranteed rank minimization via singular value projection.*, NIPS, 2010.

[5] R. H. Keshavan, A. Montanari and S. Oh. *Matrix completion from noisy entries.* JMLR, 11(Jul):2057-2078, 2010.

[6] U. Shalit, D. Weinshall and G. Chechik. *Online learning in the manifold of low-rank matrices.* NIPS, 2010.

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

[8] T. Evgeniou, C.A. Micchelli and M. Pontil. *Learning multiple tasks with kernel methods.* JMLR, 6(Apr):615-637, 2005.

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

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

[11] J.-F. Cai, E.J. Candès, and Z. Shen *A singular value thresholding algorithm for matrix completion.* SIAM Journal on Optimization, 20(4):1956â1982, 2010.

[12] K. Lee and Y. Bresler *ADMiRA: atomic decomposition for minimum rank approximation.* IEEE Transactions on Information Theory, Vol. 56 Issue 9, 2009.

Acknowledgements

Gilles Meyer is a research fellow of the Belgian National Fund for Scientific Research (FRS-FNRS). 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. The scientific responsibility rests with its authors.