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 [2,4,5,6,10]. The proposed algorithms generalize our recent work on learning feed-forward symmetric positive semi-definite matrices [2].

Problem formulation

Given data matrix instances $X \in \mathbb{R}^{d \times n}$, observations $y \in \mathbb{R}$, and a linear regression model $y = \text{Tr}(W^T X)$, solve

$$\min_{W \in \mathbb{R}^{d \times r}} \text{Ex}_{(i,y)}(\ell(i,y)) \text{ subject to } \text{rank}(W) = r.$$ 

The loss function is the quadratic loss $\ell(i,y) = |y - (\text{Tr}(W^T X))|^2$.

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

$$f_r(W) = \frac{1}{n} \sum_{i=1}^{n} \ell(i,y),$$

or the instantaneous cost

$$f(W) = \ell(i,y),$$

for the batch algorithms.

Driving applications

- Low-rank matrix completion [3,4,5,10]. Completing the missing entries of a matrix $W$ given a subset of its entries fits in the considered regression framework. Observations $y_{ij}$ are the known entries and $X_{ij} \sim \mathcal{N}(0,1)$.
- Learning on pairs [7]. Given triplets $(x_i, x_j, y_{ij})$ with $x_i, x_j \in \mathbb{R}^d$ and $y_{ij} \in \mathbb{R}$, learn a regression model $y_{ij} = \text{Tr}(W^T X_{ij}) = x_i^T W x_j$.
- Multi-task regressions [8]. Learning of a parameter $W \in \mathbb{R}^{d \times r}$ that is shared among $p$ related regression problems. The model is given by $y_p = \text{Tr}(W^T X_{ij})$, where $X_{ij} \in \mathbb{R}^{d \times r}$ and $y_p \in \mathbb{R}$ 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 [9]. Compute a relevance score $(u_i, s_i, \hat{W}(X_{ij}))$ such that $(u_i, s_i) \sim \hat{W}(X_{ij})$. A common feature of these problems is that the input matrix $X$ is rank-one.

Matrix completion (synthetic data)

The proposed algorithms compete with the state-of-the-art: MMMF [3], OptSpace [8], SVD [6], ADiMA [2], SVM [11].

Experimental setup:
- Random rank-$1$ matrices $W \in \mathbb{R}^{d \times r}$ 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 $\leq 0.01$ is achieved.

A numerical benefit of balancing

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

$$G_{i+1} = G_i - \nabla \ell_i \quad \text{and} \quad H_{i+1} = H_i - \nabla \ell_i / \lambda$$

converges slowly when the factorization is unbalanced (e.g. $\|G_i\| \approx 3\|H_i\|$).

Experimental setup:
- Regression model: $y = \text{Tr}(GH^T X)$.
- Random data: $(x_i, x_j, y_{ij}) \sim \mathcal{N}(0,1)$.
- Random data: $(x_i, x_j, y_{ij}) \sim \mathcal{N}(0,1)$.

References