SVD with missing values

Tianxi Li

Assume the missing values for $A$ are indexed by $\Omega$. We just use the notation $P_{\Omega}$ defined in the usual way in many matrix completion literature. The low-rank approximation of $A$ with missing values by SVD can be defined as

$$\min \| P_{\Omega}A - P_{\Omega}M \|_F^2$$

s.t. $\text{rank}(M) = q$.

This is a very natural generalization of the classical SVD problem. It is hard to know who first proposed this but perhaps the most famous application of this was from Simon Funk in Netflix competition. This simple low-rank approximation achieved ninth place in the contest. There haven’t been a formal description of the algorithm except the descriptive blog of Simon Funk, so I came up with a real algorithm following the high level idea.

We can write $M = UV^T$ for some full column rank $U \in \mathbb{R}^{n \times q}$ and $V \in \mathbb{R}^{m \times q}$. Then the problem is

$$\min \| P_{\Omega}A - P_{\Omega}UV^T \|_F^2$$

s.t. $U \in \mathbb{R}^{n \times q}, V \in \mathbb{R}^{m \times q}$

$$\text{rank}(U) = \text{rank}(V) = q.$$  

As one workaround, we can first drop the full rank constraints, as in most cases, the solution satisfies the constraint. This is simply a matrix factorization problem. If the rank constraints are not satisfied, it already tells you that $q' < q$ is the intrinsic rank of the matrix. One can further manipulate the $q'$-SVD and include orthogonal columns to make it artificially rank $q$. So instead we are trying to solve

$$\min \| P_{\Omega}A - P_{\Omega}UV^T \|_F^2$$

s.t. $U \in \mathbb{R}^{n \times q}, V \in \mathbb{R}^{m \times q}$.

Let $u_i$ be the $i$th row of $U$ and $v_j$ be the $j$th row of $V$. The objective can be written as

$$\sum_{(i,j) \in \Omega} (A_{ij} - u_i^T v_j)^2.$$  

So it is easy to calculate the derivative for each element $u_{ij}$ and $v_{ij}$ thus gradient descent can be done. Instead, here we try to derive it in a more structural way.

For now assume we have complete data. It is clear that fixing $U$, the problem of finding $V^T$ is a multiple (response) least square problem and of course the solution is

$$\hat{V}^T = (U^T U)^{-1} U^T A$$

and the counterpart solution for $U$ when $V$ is given can be written as

$$\hat{U}^T = (V^T V)^{-1} V^T A^T.$$
A seemingly unnecessary alternative is to calculate the gradient and solve it by gradient descent:

$$\frac{\partial L}{\partial V^T} = -2U^T(A - UV^T) = -2U^TR$$

where $R = A - UV^T$ is the residual matrix in the regression. Similarly, we have

$$\frac{\partial L}{\partial U} = -2U^T(A - UV^T) = -2RV.$$

So the gradient descent updates starting with $t$th iteration with step size $\alpha$ are

$$U(t+1) = U(t) + 2\alpha \cdot R(t)V(t)$$

and

$$V^T(t+1) = V^T(t) + 2\alpha \cdot U^T(t)R(t).$$

Since we have explicit solution, we don’t have to resort to gradient descent actually. However, when there are missing data, the closed solution is not achievable. But we can modify the gradient descent slightly for a valid solution. Notice that after projection by $P_\Omega$, all the missing positions are taken to be 0 for both $A$ and $UV^T$. Thus in this case, we can define the residual as

$$\tilde{R}_{ij} = I\{(i, j) \in \Omega\}(A_{ij} - (UV^T)_{ij}) = (P_\Omega R)_{ij}.$$

The rationale is the differentiation can be passed into $P_\Omega$, since the derivative on 0 is still 0. Thus we can directly modify the gradient descent in complete case to this missing data case, which is described below:

**Algorithm 1.** Given $A$, $\Omega$, and step size $\alpha \in (0, 1)$. Initialize $U(0), V(0)$. Let $e(0) = \infty$. $A(0) = U(0)V^T(0)$ and $R(0) = P_\Omega A - P_\Omega A(0)$. For $t = 1, 2, \ldots, T_{\text{max}}$:

1. The error $e(t) = \|R(t)\|_F^2$. If $e_t > e_{t-1}$, let $\alpha := \alpha/2$.

2. Update $U(t+1) = U(t) + 2\alpha R(t)V(t)$ and $V^T(t+1) = V^T(t) + 2\alpha U^T(t)R(t)$.

3. $A(t+1) = U(t+1)V^T(t+1)$ and $R(t+1) = P_\Omega A - P_\Omega A(t)$.

4. If $\|A(t+1) - A(t)\|_F < \epsilon\|A(t)\|_F$, break;

Output $A(t)$ in the final iteration.

From the above derivation, it can be seen that the key leverage is the fact that the residuals on the missing positions can be ignored. This observation leads to another ad hoc iterative algorithm to solve the problem. Suppose we arrive a stationary point in a iterative algorithm with iteration operator $F$, such that $F(M) = M$. Since missing positions are not considered as having errors, we can simply treat the missing entries in $\Omega^c$ as the elements of $M_{\Omega^c}$. Thus if $M$ is a stationary point for $\|P_\Omega A - P_\Omega M\|^2_F$, it must be a stationary point of

$$\|P_\Omega A + P_{\Omega^c} M - M\|^2_F.$$

But we know that the latter algorithm has SVD as the solution, so this leads to the following iterative SVD algorithm:
Algorithm 2. Initialize $M(0)$. Iterate the following two steps for $t = 1, 2, \cdots$ until convergence:

1. Let $A(t) = P_{I\Omega}A + P_{\Omega^c}M(t)$.

2. Set $M(t+1)$ to be the rank-$q$ SVD of $A(t)$.

This is very straightforward derivation of the algorithm but it turns out that this is exactly hard-impute algorithm in [1].

References