

# 对称加权算法对数据矩阵补全的优化研究

刘云, 郑文凤, 张轶

(昆明理工大学信息工程与自动化学院, 昆明 650500)

**摘要:** 数据分析中存在数据集矩阵缺失, 可用数据矩阵补全缺失数据元素, 高效的补全数据矩阵算法可从算法精度等方面优化提升. 为此提出对称加权 (SW) 算法, 首先, 根据通用的矩阵补全模型, 用正则化方法进行低秩矩阵分解补全; 其次, 对分解后的矩阵因子用共同的对称矩阵加权, 得到新的矩阵补全模型和正则化加权函数; 最后, 结合块坐标下降和交替最小二乘法优化算法, 迭代得到目标函数最优解, 获得数据补全的最优补全矩阵. 仿真结果表明, 与 APALM, IRSVF 和 IRNN 算法相比, 对称加权算法在数据矩阵补全的精度和算法收敛速度方面均有较好提升.

**关键词:** 矩阵分解; 数据补全; 对称加权; 正则化

**中图分类号:** TP312 **文献标识码:** A **DOI:** 10.19907/j.0490-6756.2021.043001

## Optimization of data matrix completion by symmetric weighting algorithm

LIU Yun, ZHENG Wen-Feng, ZHANG Yi

(Faculty of Information Engineering and Automation, Kunming University of  
Science and Technology, Kunming 650500, China)

**Abstract:** The matrix completion algorithm can be used to solve the data missing problem in data analysis, and the efficiency of matrix completion algorithm can be optimized and improved from the aspects such as algorithm accuracy. In this paper, a Symmetric Weighting (SW) algorithm is proposed. Specifically, the regularization method is first used to complete the low-rank matrix decomposition according to the general matrix completion model. Then, the decomposed matrix factor is weighted by a common symmetric matrix to obtain a new matrix completion model and a regularization weighting function. Finally, the optimal solution of the objective function is obtained iteratively using block coordinate descent and alternate least square optimization algorithm, and the optimal completion matrix of data completion is obtained. The results of simulation showed that the Symmetric Weighting algorithm is better than the APALM, IRSVF and IRNN in terms of accuracy and convergence speed.

**Keywords:** Matrix decomposition; Data completion; Symmetric weighted; Regularization

## 1 引言

数据分析中经常存在数据值的缺失, 在数据预处理中补全丢失数据可进行更有效的数据分析, 广

泛应用于图像处理, 医疗疾病预测和电子商务推荐等领域<sup>[1]</sup>. 尤其高维数据的矩阵稀疏性很强, 解决数据矩阵稀疏问题可以实现数据补全, 用低秩矩阵分解 (Low-Rank Matrix Factorization, LRMF) 重

收稿日期: 2020-02-03

基金项目: 国家自然科学基金(61761025); 云南省重大科技专项计划(202002AD080002)

作者简介: 刘云(1973-), 男, 云南昆明人, 副教授, 研究方向为数据挖掘、人工智能研究. E-mail: liuyun@kmust.edu.cn

通讯作者: 郑文凤. E-mail: 2867328528@qq.com

构矩阵是解决数据矩阵补全 (Matrix Completion, MC) 问题的主流方法<sup>[2-3]</sup>. LRMF 引起了非凸优化问题, 传统方法用矩阵的秩凸近似解, 即核范数近似优化<sup>[4]</sup>.

Fan 等人<sup>[5]</sup>基于稀疏因子分解方法, 将不完整矩阵分解为密集矩阵和稀疏矩阵, 提出加速近端交替线性最小化 (Accelerated Proximal Alternating Linearized Minimization, APALM) 算法, 解决了稀疏分解方法的非凸优化问题, 更适合对从多个子空间提取的数据进行矩阵补全. Zhang 等人<sup>[6]</sup>根据非凸非光滑秩 (Nonconvex Nonsmooth Rank, NNR) 松弛函数, 建议使用双重 NNR (Double Nonconvex Nonsmooth Rank, DNNR) 函数近似低秩恢复函数, 提出通用的迭代重加权奇异值函数 (Iteratively Reweighted Singular Values Function, IRSVF) 算法, 闭式解决了 DNNR 函数最小化收敛问题, 性能上优于其他核范数加权方法, 更适合解决数据矩阵补全非凸优化问题. Canyi 等人<sup>[7]</sup>建议使用  $l_0$  范数的非凸替代矩阵的奇异值近似低秩函数, 提出迭代重加权核范数 (Iteratively Reweighted Nuclear Norm, IRNN) 算法来解决非凸非光滑低秩最小化问题, 并通过将各种稀疏性强加于奇异值向量上, 解决了加权奇异值阈值问题, 比传统方法更适合低秩矩阵恢复.

为降低传统 MC 算法的计算复杂度, 提高补全精度, 提出对称加权 (Symmetric Weighting, SW) 算法. 主要针对非凸 LRMF 的正则化补全模型, 利用稀疏性提出加权 MC 模型; 再对分解后的矩阵因子选择共同的对称矩阵加权, 通过正则化矩阵联合列稀疏性耦合矩阵因子, 利用列修剪减少矩阵元素, 促进矩阵秩最小化达到目标矩阵的秩; 结合块坐标下降 (Block Coordinate Descent, BCD)<sup>[8]</sup> 和交替最小二乘法 (Alternating Least Squares, ALS)<sup>[9]</sup>, 交替迭代得到最优低秩补全数据矩阵. 仿真结果, 相比于现有的算法, SW 算法可以更快更准确地补全数据, 并且适用于高维数据补全.

## 2 正则化矩阵分解补全

假设不完整矩阵  $\mathbf{Y}$  之间的元素有较高的连续性, 采样产生一个低秩结构化矩阵  $\mathbf{X}$ , 通过低秩最小化来恢复其缺失的元素, 如图 1.

为了优化低秩矩阵恢复, 首先提出一个通用的矩阵秩最小化问题, 如下式.

$$\min[\text{rank}(\mathbf{X})] \text{ s. t. } \mathcal{A}(\mathbf{X}) = \mathbf{b} \quad (1)$$

在式(1)中,  $\mathbf{X} \in \mathbf{R}^m \times n$ ,  $\mathbf{b} \in \mathbf{R}^l$ ,  $\text{rank}(\mathbf{X})$  表示矩阵的秩,  $\mathcal{A}$  表示  $\mathbf{X}$  映射到  $\mathbf{b}$  的线性算子<sup>[10]</sup>. MC 的一般模型如下式.

$$\min[\text{rank}(\mathbf{X})] \text{ s. t. } \mathcal{P}_\Omega(\mathbf{Y}) = \mathcal{P}_\Omega(\mathbf{X}) \quad (2)$$

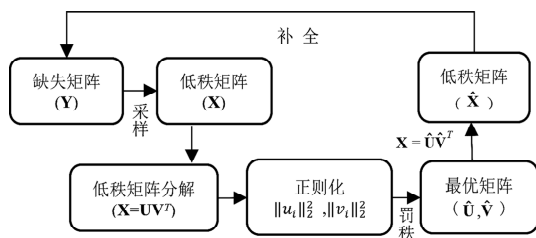


图 1 正则化低秩矩阵分解补全流程

Fig. 1 Regularization low-rank decomposition of matrix completion process

这里  $\mathcal{P}_\Omega$  表示集合  $\Omega$  上的采样运算符,  $\Omega$  为目标矩阵  $\mathbf{Y}$  中信息存在部分的下标集合, 数据不完整的矩阵  $\mathbf{Y}$  由低秩数据矩阵  $\mathbf{X}$  近似表示<sup>[2-4]</sup>.

为了优化低秩最小化问题, 通常对低秩数据矩阵  $\mathbf{X}$  进行分解, 如图 1 用矩阵  $\mathbf{U}_{m \times r}$  和矩阵  $\mathbf{V}_{n \times r}$  的转置相乘表示, 即  $\mathbf{X} = \mathbf{U}\mathbf{V}^T$ . 其中  $r \leq \min(m, n)$ , 这种方法在处理大规模或高维数据时可以减少相关变量的大小, 使计算复杂度  $O(mn)$  降低到  $O((m+n)r)$ <sup>[11]</sup>. 新的 LRMF 优化秩最小化的补全模型如下式.

$$\min[\text{rank}(\mathbf{U}\mathbf{V}^T)] \text{ s. t. } \mathcal{P}_\Omega(\mathbf{Y}) = \mathcal{P}_\Omega(\mathbf{U}\mathbf{V}^T) \quad (3)$$

通常矩阵  $\mathbf{X}$  的秩  $r$  未知, 高估重构矩阵  $\mathbf{X}$  的秩  $r$  为  $d$ , 有  $d \geq r$ , 因此要惩罚  $\mathbf{U}\mathbf{V}^T$  的秩, 找到实际的秩  $r$ . 考虑如下矩阵乘积  $\mathbf{U}\mathbf{V}^T$  的一阶分解, 如下式.

$$\mathbf{U}\mathbf{V}^T = \sum_{i=1}^d u_i v_i^T \quad (4)$$

根据秩的可加性, 降低式(4)右边项的一个秩会导致左边项  $\mathbf{U}\mathbf{V}^T$  的秩相应减少. 由此引入一个列修剪过程, 即通过迭代删除矩阵中被清零的列, 逐渐降低计算复杂度<sup>[12]</sup>.

其次, 考虑矩阵  $\mathbf{U}$  和  $\mathbf{V}$  的联合列稀疏性, 添加一个正则化过程, 用  $l_2$  范数表示惩罚项优化 MC 模型, 如下式.

$$\min_{\mathbf{U}, \mathbf{V}} \sum_{i=1}^d (\|u_i\|_2 + \|v_i\|_2)^{1/2} \text{ s. t. } \mathcal{P}_\Omega(\mathbf{Y}) = \mathcal{P}_\Omega(\mathbf{U}\mathbf{V}^T) \quad (5)$$

该正则化方法中,  $u_i$  和  $v_i$  分别是矩阵  $\mathbf{U}$  和  $\mathbf{V}$  的第  $i$  列元素, 通过正则化矩阵因子中的列元素降低模型的复杂度. 考虑矩阵  $\mathbf{Y}$  可能添加了独立同

分布的高斯噪声, 根据拉格朗日定理式(5)可以等效如下式.

$$\{\hat{\mathbf{U}}, \hat{\mathbf{V}}\} = \underset{\mathbf{U}, \mathbf{V}}{\operatorname{argmin}} \frac{1}{2} \|\mathcal{P}_{\Omega}(\mathbf{Y}) - \mathcal{P}_{\Omega}(\mathbf{U} \mathbf{V}^T)\|_F^2 + \lambda \sum_{i=1}^d (\|\mathbf{u}_i\|_2^2 + \|\mathbf{v}_i\|_2^2)^{1/2} \quad (6)$$

$\|\mathbf{X}\|_F = \sqrt{\sum_i \sum_j x_{ij}^2}$  为 Frobenius 范数的表达式,  $\lambda$  是拉格朗日参数, 可得到更优的补全数据矩阵的低秩矩阵因子<sup>[13]</sup>. 通过矩阵乘法得到优化的低秩矩阵, 即  $\hat{\mathbf{X}} = \hat{\mathbf{U}} \hat{\mathbf{V}}^T$ , 补全缺失数据矩阵  $\mathbf{Y}$ , 如下式.

$$\min[\operatorname{rank}(\hat{\mathbf{X}})] \text{ s. t. } \mathcal{P}_{\Omega}(\mathbf{Y}) = \mathcal{P}_{\Omega}(\hat{\mathbf{X}}) \quad (7)$$

如果在式(5)中给分解后的矩阵因子添加一个权重, 可以促进稀疏性, 更快得到式(7)中的低秩补全矩阵.

### 3 对称加权(SW)算法

目前已经有很多优化矩阵补全问题的加权方案. 其中, 重加权 Frobenius 范数优化矩阵补全模型如下式.

$$\min[\operatorname{rank}(\|\mathbf{X} \mathbf{W}^{\frac{1}{2}}\|_F^2)] \text{ s. t. } \mathcal{P}_{\Omega}(\mathbf{Y}) = \mathcal{P}_{\Omega}(\mathbf{X}) \quad (8)$$

其最小化等价于 Schatten-p 范数<sup>[14]</sup>, 其中,  $\|\mathbf{X}\|_{s_p} =$

$\|\boldsymbol{\sigma}(\mathbf{X})\|_p = (\sum_{i=1}^{\operatorname{rank}(\mathbf{X})} \sigma_i^p(\mathbf{X}))^{1/p}$ ,  $\boldsymbol{\sigma}(\mathbf{X})$  表示矩阵  $\mathbf{X}$  非递增排序的奇异值向量,  $0 < p \leq 1$ .

$$\begin{aligned} \|\mathbf{X}\|_{s_p} &= \operatorname{tr}\{(\mathbf{X}^T \mathbf{X})^{p/2}\} = \\ &= \operatorname{tr}\{(\mathbf{X}^T \mathbf{X})(\mathbf{X}^T \mathbf{X})^{p/2-1}\} = \\ &= \operatorname{tr}\{(\mathbf{X}^T \mathbf{X}) \mathbf{W}\} = \|\mathbf{X} \mathbf{W}^{\frac{1}{2}}\|_F^2 \end{aligned} \quad (9)$$

$\operatorname{tr}\{\mathbf{X}\}$  表示矩阵  $\mathbf{X}$  的迹, 可知  $\mathbf{W}$  是对称权重矩阵, 且每次迭代的  $\mathbf{W}$  值由上一次迭代后获得的矩阵  $\mathbf{X}$  计算得到,

$$\mathbf{W} = (\mathbf{X}^T \mathbf{X})^{p/2-1} \quad (10)$$

当  $p = 1$  时, Schatten-p 范数等价于核范数  $\|\mathbf{X}\|_*$ , 根据 LRMF 非凸性, 常用矩阵  $\mathbf{X}$  的核范数近似代替矩阵的秩解决非凸优化问题. 式(1)中秩最小化问题用加权核范数解决, 得到新的矩阵补全模型,

$$\min[\operatorname{rank}(\|\mathbf{X}\|_{*, \mathbf{W}})] \text{ s. t. } \mathcal{P}_{\Omega}(\mathbf{Y}) = \mathcal{P}_{\Omega}(\mathbf{X}) \quad (11)$$

其中, 矩阵  $\mathbf{X}$  的核范数  $\|\mathbf{X}\|_* = \operatorname{tr}\{(\mathbf{X}^T \mathbf{X})^{1/2}\} = \sum_{i=1}^{\operatorname{rank}(\mathbf{X})} \sigma_i(\mathbf{X})$ , 则矩阵  $\mathbf{X}$  的加权核范数  $\|\mathbf{X}\|_{*, \mathbf{W}} = \sum_{i=1}^{\operatorname{rank}(\mathbf{X})} w_i \sigma_i(\mathbf{X})$ ,  $w_i$  是权重矩阵  $\mathbf{W}$  上的向量元素.

该模型能有效处理秩最小化问题, 更快得到补全数据的低秩矩阵<sup>[15]</sup>. 同时, 根据矩阵范数的可乘性, 关系如下式.

$$\begin{aligned} \|\mathbf{X}\|_* &\leq \|\mathbf{U}\|_F^2 \|\mathbf{V}\|_F^2 \leq \\ &= \frac{1}{2} (\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2) \end{aligned} \quad (12)$$

因此, 可以定义一个具有严格上界的核范数来解决式(11)中的最小化问题, 如下式.

$$\|\mathbf{X}\|_* = \min_{\mathbf{U} \in \mathbf{R}^{m \times d}, \mathbf{V} \in \mathbf{R}^{n \times d}} \frac{1}{2} (\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2) \quad (13)$$

利用核范数在矩阵  $\mathbf{U}$  和  $\mathbf{V}$  产生一个平滑度, 促进了低秩最小化.

#### 3.1 SW 算法

基于正则化矩阵分解补全, 分别对数据矩阵因子  $\mathbf{U}$  和  $\mathbf{V}$  加权, 得到加权目标函数, 如图 2 所示. 其次, 利用 ALS 方法可以更好地处理稀疏矩阵的分解问题, 得到数据补全的最优矩阵.

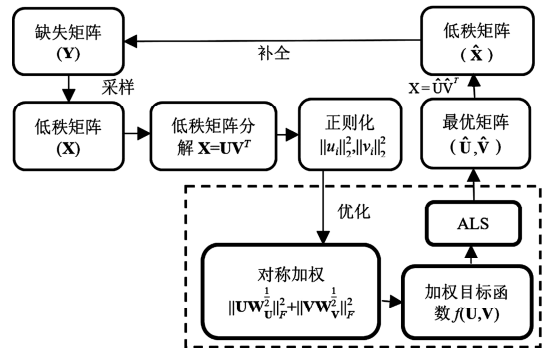


图 2 对称加权矩阵补全框图

Fig. 2 Symmetrical weighted matrix completion-block diagram

结合式(13)得到矩阵的重加权 Frobenius 范数和的最小化补全模型, 如下式.

$$\begin{aligned} \min [\operatorname{rank}(\|\mathbf{U} \mathbf{W}_U^{\frac{1}{2}}\|_F^2 + \|\mathbf{V} \mathbf{W}_V^{\frac{1}{2}}\|_F^2)] \\ \text{ s. t. } \mathcal{P}_{\Omega}(\mathbf{Y}) = \mathcal{P}_{\Omega}(\mathbf{U} \mathbf{V}^T) \end{aligned} \quad (14)$$

为了解决上述最小化问题, 定义一个低秩函数, 如下式.

$$g(\mathbf{U}, \mathbf{V}) = \frac{1}{2} (\|\mathbf{U} \mathbf{W}_U^{\frac{1}{2}}\|_F^2 + \|\mathbf{V} \mathbf{W}_V^{\frac{1}{2}}\|_F^2) \quad (15)$$

根据式(10)利用对称性设置权值  $\mathbf{W}_U = (\mathbf{U}^T \mathbf{U})^{p-1}$  和  $\mathbf{W}_V = (\mathbf{V}^T \mathbf{V})^{p-1}$ , 当  $p = 1$  时,  $\mathbf{W}_U = \mathbf{W}_V = \mathbf{I}_d$ , 可以得到式(13)中的核范数变分形式, 假设  $\mathbf{W}_U = \mathbf{W}_V = \mathbf{W}$ , 有

$$\mathbf{W} = \operatorname{diag}((\|\mathbf{u}_1\|_2^2 + \|\mathbf{v}_1\|_2^2)^{p/2-1}, (\|\mathbf{u}_2\|_2^2 + \|\mathbf{v}_2\|_2^2 + \dots))$$

$$\|v_2\|_2^{p/2-1}, \dots, (\|u_d\|_2^2 + \|v_d\|_2^2)^{p/2-1}) \quad (16)$$

式中,  $0 < p \leq 1$ ,  $\mathbf{W}$  是对称的对角权重矩阵, 矩阵中的元素是正则化的. 对矩阵  $\mathbf{U}$  和  $\mathbf{V}$  选择共同的权重矩阵  $\mathbf{W}$  进行加权处理, 可促进矩阵因子的稀疏性使列隐式耦合. 将式(16)代入(15)中得到新的加权目标函数.

$$g(\mathbf{U}, \mathbf{V}) = \frac{1}{2} \sum_{i=1}^d (\|u_i\|_2^2 + \|v_i\|_2^2)^{p/2} \quad (17)$$

正则化联合列稀疏性可以更快的惩罚秩, 但该方法具有不光滑性且不可以分离矩阵  $\mathbf{U}$  和  $\mathbf{V}$  中的元素. 为了使优化问题变得光滑, 根据函数中的正则化项, 添加一个很小的正常数  $\eta^2$  得到可微优化<sup>[14]</sup>.

$$\hat{g}(\mathbf{U}, \mathbf{V}) = \sum_{i=1}^d (\|u_i\|_2^2 + \|v_i\|_2^2 + \eta^2)^{p/2} \quad (18)$$

该简易方法缓解了梯度不连续的点, 可以得到使目标函数最小化的唯一平稳点. 则根据正则化矩阵分解补全模型定义的目标函数  $f(\mathbf{U}, \mathbf{V})$  如下式,

$$f(\mathbf{U}, \mathbf{V}) = \frac{1}{2} \|\mathcal{P}_\Omega(\mathbf{Y} - \mathbf{U}\mathbf{V}^T)\|_F^2 + \lambda \sum_{i=1}^d (\|u_i\|_2^2 + \|v_i\|_2^2 + \eta^2)^{p/2} \quad (19)$$

最后, 结合 BCD 和 ALS 处理  $\mathbf{U}$  和  $\mathbf{V}$ , 首先通过给定一个可用的初始化矩阵  $\mathbf{V}$  更新为矩阵  $\mathbf{V}_k$ , 每次迭代时最小化目标函数的局部上界得到  $\mathbf{U}_{k+1}$ ; 其次, 使用  $\mathbf{U}_{k+1}$  最小化另一个局部上界得到  $\mathbf{V}_{k+1}$ . 该方法解决了两个矩阵因子的不可分性, 主要步骤如下.

**步骤 1** 在点  $(\mathbf{U}_k, \mathbf{V}_k)$  对  $f(\mathbf{U}, \mathbf{V}_k)$  近似二阶泰勒展开, 该展开式定义为上界函数  $h(\mathbf{U}|\mathbf{U}_k, \mathbf{V}_k)$ , 求其最小化.  $h(\mathbf{U}|\mathbf{U}_k, \mathbf{V}_k)$  是严格凸的, 此松弛方法可以闭式表达, 得到最优的低秩补全矩阵.

$$\mathbf{U}_{k+1} = \arg \min_{\mathbf{U}} h(\mathbf{U}|\mathbf{U}_k, \mathbf{V}_k) \quad (20)$$

其中,

$$h(\mathbf{U}|\mathbf{U}_k, \mathbf{V}_k) = f(\mathbf{U}_k, \mathbf{V}_k) + \text{tr}\{(\mathbf{U} - \mathbf{U}_k)^T \nabla_{\mathbf{U}} f(\mathbf{U}_k, \mathbf{V}_k)\} + \frac{1}{2} \text{vec}(\mathbf{U} - \mathbf{U}_k)^T \bar{\mathbf{H}}_{\mathbf{U}_k} \text{vec}(\mathbf{U} - \mathbf{U}_k) \quad (21)$$

式中,  $\text{vec}(\cdot)$  表示列向量化运算符, 目标函数  $f(\mathbf{U}, \mathbf{V}_k)$  中精确的 Hessian 矩阵  $(\mathbf{H}_{\mathbf{U}_k})$  用  $md \times md$  阶的对角矩阵  $\bar{\mathbf{H}}_{\mathbf{U}_k}$  近似代替. 当  $\lambda > 0$  时, 矩阵

$\bar{\mathbf{H}}_{\mathbf{U}_k}$  和  $\bar{\mathbf{H}}_{\mathbf{V}_k}$  是正定矩阵, 因此可逆, 可求得目标函数的局部最小值.

$$\bar{\mathbf{H}}_{\mathbf{U}_k} = \mathbf{I}_m \otimes \tilde{\mathbf{H}}_{\mathbf{U}_k} \quad (22)$$

$\otimes$  表示克罗内克积运算;  $\tilde{\mathbf{H}}_{\mathbf{U}_k}$  的定义如下式.

$$\tilde{\mathbf{H}}_{\mathbf{U}_k} = \mathbf{V}_k^T \mathbf{V}_k + \lambda \mathbf{D}_{(\mathbf{U}_k, \mathbf{V}_k)} \quad (23)$$

式中,

$$\mathbf{D}_{(\mathbf{U}, \mathbf{V})} = p \text{diag}((\|u_1\|_2^2 + \|v_1\|_2^2 + \eta^2)^{p/2-1}, (\|u_2\|_2^2 + \|v_2\|_2^2 + \eta^2)^{p/2-1}, \dots, (\|u_d\|_2^2 + \|v_d\|_2^2 + \eta^2)^{p/2-1}) \quad (24)$$

**步骤 2** 在点  $(\mathbf{U}_{k+1}, \mathbf{V}_k)$  对  $f(\mathbf{U}_{k+1}, \mathbf{V})$  近似二阶泰勒展开, 定义为上界函数  $l(\mathbf{V}|\mathbf{U}_{k+1}, \mathbf{V})$ , 求其最小化.

$$\mathbf{V}_{k+1} = \arg \min_{\mathbf{V}} l(\mathbf{V}|\mathbf{U}_{k+1}, \mathbf{V}_k) \quad (25)$$

其中,

$$l(\mathbf{V}|\mathbf{U}_{k+1}, \mathbf{V}_k) = f(\mathbf{U}_{k+1}, \mathbf{V}_k) + \text{tr}\{(\mathbf{V} - \mathbf{V}_k)^T \nabla_{\mathbf{V}} f(\mathbf{U}_{k+1}, \mathbf{V}_k)\} + \frac{1}{2} \text{vec}(\mathbf{V} - \mathbf{V}_k)^T \bar{\mathbf{H}}_{\mathbf{V}_k} \text{vec}(\mathbf{V} - \mathbf{V}_k) \quad (26)$$

同理,  $\bar{\mathbf{H}}_{\mathbf{V}_k}$  是  $md \times md$  阶的正定对角矩阵,  $\tilde{\mathbf{H}}_{\mathbf{V}_k}$  的定义如下.

$$\tilde{\mathbf{H}}_{\mathbf{V}_k} = \mathbf{U}_{k+1}^T \mathbf{U}_{k+1} + \lambda \mathbf{D}_{(\mathbf{U}_{k+1}, \mathbf{V}_k)} \quad (27)$$

交替迭代后, 得到最优解  $\hat{\mathbf{U}}$  和  $\hat{\mathbf{V}}$ , 从而求出数据矩阵补全的最优低秩数据矩阵  $\hat{\mathbf{X}}$ , 如式(7)用  $\hat{\mathbf{X}}$  补全数据矩阵  $\mathbf{Y}$ . 主要步骤如算法 1.

#### 算法 1 对称加权算法(SW)

输入:

- 1) 不完全矩阵  $\mathbf{Y}$
- 2) 低秩正则化参数  $\lambda > 0$

初始化:

- 3)  $k=0$ ,  $\mathbf{U}_0, \mathbf{V}_0, \mathbf{D}_{(\mathbf{U}_0, \mathbf{V}_0)}$

迭代计算:

- 4)  $\mathbf{U}_{k+1} = \mathbf{U}_k - (\mathcal{P}_\Omega(\mathbf{U}_k \mathbf{V}_k^T - \mathbf{Y}) \mathbf{V}_k + \lambda \mathbf{U}_k \mathbf{D}_{(\mathbf{U}_k, \mathbf{V}_k)})(\mathbf{V}_k^T \mathbf{V}_k + \lambda \mathbf{D}_{(\mathbf{U}_k, \mathbf{V}_k)})^{-1}$
  - 5)  $\mathbf{V}_{k+1} = \mathbf{V}_k - (\mathcal{P}_\Omega(\mathbf{V}_k \mathbf{U}_{k+1}^T - \mathbf{Y}^T) \mathbf{U}_{k+1} + \lambda \mathbf{V}_k \mathbf{D}_{(\mathbf{U}_{k+1}, \mathbf{V}_k)})(\mathbf{U}_{k+1}^T \mathbf{U}_{k+1} + \lambda \mathbf{D}_{(\mathbf{U}_{k+1}, \mathbf{V}_k)})^{-1}$
  - 6)  $k=k+1$
  - 7) 判断是否收敛, 收敛时输出, 否则转步骤 4)
- 输出:
- 8)  $\hat{\mathbf{U}} = \mathbf{U}_{k+1}, \hat{\mathbf{V}} = \mathbf{V}_{k+1}$



算法 1 中, 步骤 1)~2) 输入需要补全的矩阵  $\mathbf{Y}$  和正则化参数  $\lambda$ , 根据式 (2) 和式 (3) 对低秩矩阵采样分解.

步骤 3) 初始化迭代次数  $k=0$ , 则分解后的低秩矩阵因子为  $\mathbf{U}_0$  和  $\mathbf{V}_0$ . 再由式 (24) 运算得到  $\mathbf{D}_{(\mathbf{U}_0, \mathbf{V}_0)}$ , 代入式 (23) 和 (27) 得到近似的 Hessian 矩阵.

步骤 4)~5) 根据式 (20) 和式 (25) 交替计算局部目标函数的最小值, 两次连续迭代间的重构数据相对减少时算法收敛, 得到最优的低秩矩阵因子.

步骤 8) 输出算法收敛时的低秩矩阵因子  $\hat{\mathbf{U}}$  和  $\hat{\mathbf{V}}$ , 算法结束.

SW 算法用 Hessian 矩阵求最优解, 并用拟牛顿 (Quasi-Newton) 更新公式构造 Hessian 矩阵的近似值, 收敛速度明显提高<sup>[15]</sup>. 最终用  $d \times d$  阶的对角矩阵  $\tilde{\mathbf{H}}_{\mathbf{U}_k}$  和  $\tilde{\mathbf{H}}_{\mathbf{V}_k}$  使计算复杂度降到了  $O(d^3)$ , 更新矩阵  $\mathbf{U}$  和  $\mathbf{V}$  的计算成本更低.

### 3.2 SW 算法收敛性分析

SW 算法每次迭代的最小化函数  $h(\mathbf{U}|\mathbf{U}_k, \mathbf{V}_k)$  和  $l(\mathbf{V}|\mathbf{U}_{k+1}, \mathbf{V})$  是函数  $f(\mathbf{U}, \mathbf{V}_k)$  和  $f(\mathbf{U}_{k+1}, \mathbf{V})$  的严格上界, 因此, 局部替代函数是实际目标函数的上界. 如命题 1, 算法每次迭代后的目标函数单调递减.

**命题 1** 通过 SW 算法产生的顺序集  $\{\mathbf{U}_k, \mathbf{V}_k\}$  的目标函数是单调递减的, 如式 (28),

$$f(\mathbf{U}_{k+1}, \mathbf{V}_{k+1}) \leq f(\mathbf{U}_{k+1}, \mathbf{V}_k) \leq f(\mathbf{U}_k, \mathbf{V}_k) \quad (28)$$

证明: 已知  $h(\mathbf{U}|\mathbf{U}_k, \mathbf{V}_k) \geq f(\mathbf{U}, \mathbf{V}_k)$ , 结合式 (20) 有,

$$h(\mathbf{U}_{k+1}|\mathbf{U}_k, \mathbf{V}_k) \leq h(\mathbf{U}_k|\mathbf{U}_k, \mathbf{V}_k) = f(\mathbf{U}_k, \mathbf{V}_k) \quad (29)$$

由已知条件可得  $h(\mathbf{U}_{k+1}|\mathbf{U}_k, \mathbf{V}_k) \geq f(\mathbf{U}_{k+1}, \mathbf{V}_k)$ , 所以,

$$f(\mathbf{U}_{k+1}, \mathbf{V}_k) \leq f(\mathbf{U}_k, \mathbf{V}_k) \quad (30)$$

同理可证,

$$f(\mathbf{U}_{k+1}, \mathbf{V}_{k+1}) \leq f(\mathbf{U}_{k+1}, \mathbf{V}_k) \quad (31)$$

结合式 (30) 和式 (31), 可得到命题 1.

**命题 2** 在  $k \rightarrow \infty$  时, 目标函数  $f(\mathbf{U}_k, \mathbf{V}_k)$  收敛, 有  $f^\infty \geq 0$ .

证明 通过命题 1 可知 SW 算法的目标函数单调递减, 并且下界是 0, 因此, 当  $k \rightarrow \infty$  时,  $f^\infty \geq 0$ . 通过该结论可以分析算法的收敛点和收敛速度.

1) 收敛到平稳点.

SW 算法通过单调递减的目标函数更新矩阵

$(\mathbf{U}_k, \mathbf{V}_k)$ , 可以推出目标函数到平稳点的收敛速度. 给出任何一对  $(\mathbf{U}, \mathbf{V})$ , 通过下面的最小化问题定义矩阵  $\mathbf{U}_*$  和  $\mathbf{V}_*$ .

$$\mathbf{U}_* = \arg \min_{\mathbf{U}^+} h(\mathbf{U}^+|\mathbf{U}, \mathbf{V}) \quad (32)$$

$$\mathbf{V}_* = \arg \min_{\mathbf{V}^+} l(\mathbf{V}^+|\mathbf{U}_*, \mathbf{V}) \quad (33)$$

用  $\Delta((\mathbf{U}, \mathbf{V}), (\mathbf{U}_*, \mathbf{V}_*))$  表示  $(\mathbf{U}, \mathbf{V})$  和  $(\mathbf{U}_*, \mathbf{V}_*)$  之间的相似度测量.

$$\begin{aligned} \Delta((\mathbf{U}, \mathbf{V}), (\mathbf{U}_*, \mathbf{V}_*)) &= \frac{1}{2} (\|\mathbf{V}(\mathbf{U} - \mathbf{U}_*)^T\|_F^2 + \|\mathbf{U}_*(\mathbf{V} - \mathbf{V}_*)^T\|_F^2 + \\ &\frac{\lambda}{2} (\|D_{(\mathbf{U}, \mathbf{V})}^{\frac{1}{2}}(\mathbf{U} - \mathbf{U}_*)^T\|_F^2 + \\ &\|D_{(\mathbf{U}_*, \mathbf{V}_*)}^{\frac{1}{2}}(\mathbf{V} - \mathbf{V}_*)^T\|_F^2)) \end{aligned} \quad (34)$$

推论: 算法中目标函数  $f(\mathbf{U}, \mathbf{V})$  的连续目标值的差值有下界, 结果如式 (35),

$$f(\mathbf{U}_k, \mathbf{V}_k) - f(\mathbf{U}_{k+1}, \mathbf{V}_{k+1}) \geq \Delta((\mathbf{U}_k, \mathbf{V}_k), (\mathbf{U}_{k+1}, \mathbf{V}_{k+1})) \quad (35)$$

**命题 3** 当且仅当  $(\mathbf{U}, \mathbf{V})$  是 SW 算法得到的平稳点时,  $\Delta((\mathbf{U}, \mathbf{V}), (\mathbf{U}_*, \mathbf{V}_*)) = 0$ .

证明: 假设  $(\mathbf{U}, \mathbf{V})$  是一个平稳点,  $\mathbf{U} = \mathbf{U}_*$ ,  $\mathbf{V} = \mathbf{V}_*$ , 很容易证明  $\Delta((\mathbf{U}, \mathbf{V}), (\mathbf{U}_*, \mathbf{V}_*)) = 0$ . 其次, 由式 (35) 可知  $\Delta((\mathbf{U}, \mathbf{V}), (\mathbf{U}_*, \mathbf{V}_*))$  非负, 若  $\Delta((\mathbf{U}, \mathbf{V}), (\mathbf{U}_*, \mathbf{V}_*)) = 0$ , 则有,

$$h(\mathbf{U}|\mathbf{U}, \mathbf{V}) - h(\mathbf{U}_*|\mathbf{U}, \mathbf{V}) = 0 \quad (36)$$

$$l(\mathbf{V}|\mathbf{U}_*, \mathbf{V}) - l(\mathbf{V}_*|\mathbf{U}_*, \mathbf{V}) = 0 \quad (37)$$

$h(\mathbf{U}|\mathbf{U}, \mathbf{V})$  和  $l(\mathbf{V}|\mathbf{U}_*, \mathbf{V})$  是严格凸函数,  $(\mathbf{U}_*, \mathbf{V}_*)$  是唯一极值解, 因此, 只有当  $(\mathbf{U}, \mathbf{V}) = (\mathbf{U}_*, \mathbf{V}_*)$  时, 结论成立. 所以  $(\mathbf{U}_*, \mathbf{V}_*)$  也是最小化目标函数的一个平稳点.

如上所述,  $\Delta((\mathbf{U}_k, \mathbf{V}_k), (\mathbf{U}_{k+1}, \mathbf{V}_{k+1}))$  用来量化算法连续迭代生成的  $(\mathbf{U}_k, \mathbf{V}_k)$  和  $(\mathbf{U}_{k+1}, \mathbf{V}_{k+1})$  之间的距离, 当  $\Delta = 0$  时, 算法收敛到平稳点  $(\mathbf{U}_{k+1}, \mathbf{V}_{k+1})$ .

2) 收敛速度.

当  $\lambda > 0$  时, 由算法产生的序列  $\{\mathbf{U}_k, \mathbf{V}_k\}$  的极值点是目标函数  $f(\mathbf{U}, \mathbf{V})$  的一个平稳点. 定义一个值  $\xi_k$ ,  $\xi_k = \Delta((\mathbf{U}_k, \mathbf{V}_k), (\mathbf{U}_{k+1}, \mathbf{V}_{k+1}))$ , 假设一个  $K \rightarrow \infty$ .

$$\begin{aligned} \sum_{k=1}^K \xi_k &\leq f(\mathbf{U}_1, \mathbf{V}_1) - f(\mathbf{U}_K, \mathbf{V}_K) \leq \\ &f(\mathbf{U}_1, \mathbf{V}_1) - f^\infty < \infty \end{aligned} \quad (38)$$

所有序列的  $\xi_k$  值非负, 且收敛到 0. 通过命题 3 可知  $\xi_k$  的零极值点, 从而得到 SW 算法的收敛点. 并

且  $K \min_{1 \leq k \leq K} \xi_k \leq \sum_{k=1}^K \xi_k$ , 因此算法到平稳点的收敛速度表达式, 如下式.

$$\min_{1 \leq k \leq K} \xi_k \leq \frac{f(\mathbf{U}_1, \mathbf{V}_1) - f^{\infty}}{K} \tag{39}$$

## 4 仿真分析

### 4.1 数据集

与类似算法仿真一致, 仿真分析采用数据稀疏性较强的通用数据源 MovieLens 数据集, 该数据集包含用户信息、电影信息以及多个用户对多部电影的评分, 其评分数据矩阵非常稀疏, 因此, 可用该数据集来研究数据矩阵的缺失填补算法, 对潜在用户的评分进行预测<sup>[17]</sup> (MovieLens 开源数据集网站 <https://grouplens.org/datasets/movielens/>). 其中, MovieLens100 K 是经过充分研究的大型数据集, 包含在不同时段收集的用户评分, 用整数 1 到 5 表示.

表 1 MovieLens100 K 的数据集信息

Tab. 1 MovieLens100 K data set information

数据集	电影数	用户数	评分数	缺失数据 (百分比)
100 K	1 682	943	$1.0 \times 10^5$	93.7%

如表 1 所示, 在 MovieLens100 K 数据集缺失 93.7% 的评分数据, 稀疏性很强. 我们分别对数据随机采样 65%, 50% 和 35% 作为训练集, 25%, 35% 和 45% 作为验证集, 其余作为测试集. 建立一个  $943 \times 1682$  的电影评分矩阵, 假设不同用户的评分之间存在较高的相关性, 找出低秩结构数据矩阵, 进行数据补全.

为了与最新的 APALM, IRSVF 和 IRNN 算法比较, 分两种不同条件选择低秩正则化参数  $\lambda$ . 条件 A: 选择参数  $\lambda$  使算法收敛时, 数据集的误差最小, 来评估算法的补全精度; 条件 B: 选择参数  $\lambda$ , 使所有算法得到的低秩估计矩阵的秩都为 10. 在条件 A 中 APALM 算法中低秩估计矩阵的秩与初始阶段相同,  $d = 100$ , 其余算法得到  $\hat{\mathbf{U}}$  和  $\hat{\mathbf{V}}$  矩阵的秩为 6.

### 4.2 归一化平均绝对误差分析

为了评估算法的补全精度, 对归一化平均绝对误差 (Normalized Mean Absolute Value Error, NMAE) 进行分析<sup>[5-6]</sup>.  $\text{card}(\cdot)$  表示集合的基数, NMAE 定义为

$$\text{NMAE} = \frac{1}{4\text{card}(\Omega)} \sum_{(i,j) \in \Omega} | [\mathbf{U} \mathbf{V}^T]_{i,j} - [\mathbf{Y}]_{i,j} |.$$

表 2 在两种条件下算法对随机采样数据的 NMAE

Tab. 2 The NMAE of the algorithm on randomly sampled data under two conditions

	数据集 采样比例/%	APALM	IRSVF	IRNN	SW
条件 A	35	0.2081	0.1797	0.2159	0.1637
	50	0.1986	0.1702	0.2097	0.1583
	65	0.1936	0.1667	0.2002	0.1557
条件 B	35	0.2532	0.2110	0.2183	0.1785
	50	0.2501	0.2047	0.2092	0.1702
	65	0.2483	0.2010	0.2036	0.1675

从表 2 中可以看到, 随机采样 65% 的训练数据集时, 算法在达到收敛时的 NMAE 值最小, 且有适当的验证数据调整超参数, 优化模型. 因此, 在保证验证集和测试集的比例上, 尽可能多采样训练数据得到的模型泛化能力更好. 针对最优的训练模型, 分别得出 4 种算法在不同迭代时间下的补全误差, 分析 NMAE 值.

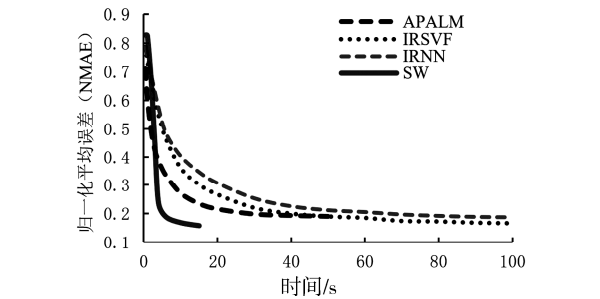


图 3 条件 A 中采样 MovieLens100K 数据集的精度分析

Fig. 3 Accuracy analysis of the sampled MovieLens 100 K dataset in condition A

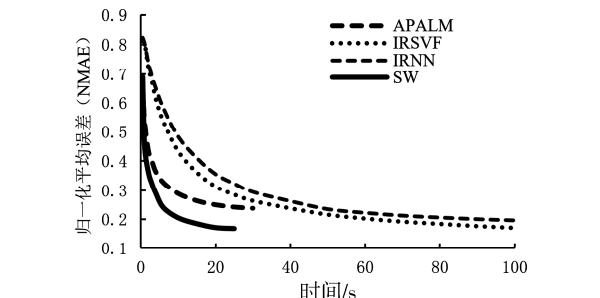


图 4 条件 B 中采样 MovieLens100 K 数据集的精度分析

Fig. 4 Accuracy analysis of the sampled MovieLens100K dataset in condition B

从图 3 和图 4 可以看出, 在 A, B 两种条件下, 算法的 NMAE 值随迭代次数的增加都逐渐减小,

最终趋于平稳点. APALM 和 IRNN 算法收敛时的 NMAE 值稳定在 0.2~0.3, IRSVF 和 SW 算法收敛时的 NAME 值稳定在 0.1~0.2. IRNN 算法迭代的时间最久, APALM 算法虽减少了迭代次数,但精度不够高, IRSVF 算法的计算量仍然较大,而 SW 算法误差最低,提高了补全精度,并且在条件 A 中该算法的时间复杂度明显降低.

4.3 收敛性分析

通过迭代产生的矩阵因子序列之间的相似度测量,计算两个连续目标函数之间的差值,可以得到算法收敛时的情况. 针对 A,B 两种条件,在第一种采样数据训练的模型中分析 APALM, IRSVF, IRNN 和 SW 算法的收敛性.

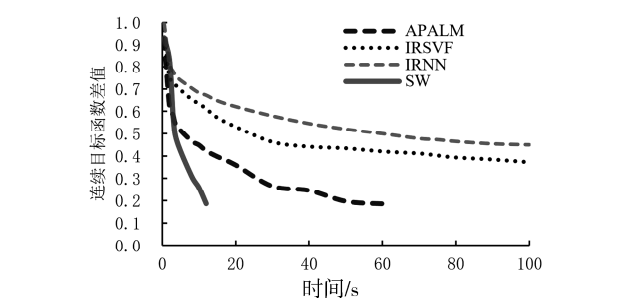


图 5 条件 A 中连续目标函数差值随时间的演变  
Fig. 5 The evolution of the difference of the continuous objective function with time in condition A

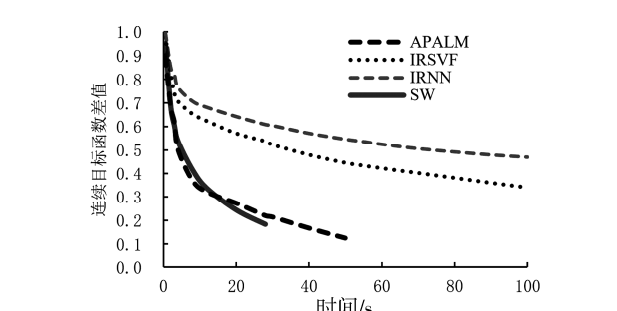


图 6 条件 B 中连续目标函数差值随时间的演变  
Fig. 6 The evolution of the difference of the continuous objective function with time in condition B

从图 5 和图 6 可以看出四种算法在收敛时的情况. IRSVF 和 IRNN 算法在两种条件下的平均每次迭代时间复杂度很高,并且收敛效果不好. 其他两种算法的收敛效果差不多,但在条件 A 中明显看出 APALM 算法收敛时的迭代时间更长. SW 算法在两种条件下的收敛效果都较好且平均每次迭代的时间复杂度最低,特别是在条件 A 中收敛时的时间不到其他算法的 20%.

通过 NMAE 和收敛性分析表明,SW 算法对比于其他三种算法可以更精确的补全数据矩阵,并

且降低时间计算复杂度并不会影响该算法的补全性能. 其次,SW 算法的列修剪优化在很大程度上减少了计算负担,使仿真数据集收敛时的迭代次数更少,收敛速度明显比现有算法快,可更高效的进行补全.

5 结 论

用矩阵表示数据集,可以高效解决数据分析中的缺失元素问题,高维数据矩阵也可以用低维特征表示或重构. 根据 LRMF 的补全模型,利用正则化矩阵联合列稀疏性,进行列修剪降低计算复杂度. 然后基于加权稀疏矩阵的思想,对矩阵因子对称加权得出正则化加权矩阵补全模型及目标函数. 最后,结合 BCD 和 ALS 方法,使 SW 算法迭代得出最优低秩补全矩阵. 仿真结果表明,SW 算法与现有算法相比具有更高的精度,适用于处理大量和高维的数据. 其次,SW 算法的收敛速度明显提高,更高效地完成数据矩阵补全. 下步将根据各类数据矩阵的稀疏程度和特性,继续深入研究这种稀疏度对算法性能的影响.

参考文献:

[1] Zeng X, Shen L, Xu Y, *et al.* Matrix completion via minimizing an approximate rank [J]. Anal Appl, 2019, 17: 689.

[2] 陈蕾, 陈松灿. 矩阵补全模型及其算法研究综述 [J]. 软件学报, 2017, 28: 1547.

[3] 李可欣, 徐彬, 高克宁. 异常值自识别的低矩阵补全方法 [J]. 计算机科学与探索, 2019, 13: 1272.

[4] Li Q, Zhu Z, Tang G. The non-convex geometry of low-rank matrix optimization [J]. Inform Infer, 2018, 8: 51.

[5] Fan J, Zhao M, Chow T W S. Matrix completion via sparse factorization solved by accelerated proximal alternating linearized minimization [J]. IEEE T Big Data, 2020, 6: 119.

[6] Zhang H, Gong C, Qian J, *et al.* Efficient recovery of low-rank matrix via double nonconvex nonsmooth rank minimization [J]. IEEE T Neur Net Lear, 2019, 30: 2916.

[7] Lu C, Tang J, Yan S, *et al.* Nonconvex nonsmooth low rank minimization via iteratively reweighted nuclear norm [J]. IEEE T Image Process, 2016, 25: 2.

[8] Shi Q, Sun H, Lu S, *et al.* Inexact block coordinate descent methods for symmetric nonnegative

matrix factorization[J]. IEEE T Signal Process, 2017, 65: 5995.

[9] 董立岩, 王宇, 任怡, 等. 基于矩阵分解和聚类的协同过滤算法[J]. 吉林大学学报: 理学版, 2019, 57: 105.

[10] Zhu Z, Li Q, Tang G, *et al.* Global optimality in low-rank matrix optimization[J]. IEEE T Signal Process, 2018, 66: 3614.

[11] 吴渝, 彭茂玲, 钱仁飞, 等. 一种采用遗传规划优化的基于非负矩阵分解的入侵检测模型[J]. 西南师范大学学报: 自然科学版, 2020, 45: 139.

[12] 胡思才, 孙界平, 琚生根, 等. 基于深度神经网络和概率矩阵分解的混合推荐算法[J]. 四川大学学报: 自然科学版, 2019, 56: 1033.

[13] Giménez-Febrer P, Pagès-Zamora A, Giannakis G B. Matrix completion and extrapolation via kernel regression [J]. IEEE Sign Proc, 2019, 67: 5004.

[14] Wu Q, Zhang F, Wang H, *et al.* Generalized low-rank matrix completion via nonconvex Schatten-p-norm minimization [C]//Proceedings of the 2018 IEEE 88th Vehicular Technology Conference (VTC-Fall). [S.l.]: IEEE, 2018.

[15] 陈元媛, 高岩, 刘志敏, 等. 一类特殊优化问题的光滑梯度法[J]. 运筹学学报, 2017, 21: 119.

[16] Kristensen P K, Martínez-Pañeda E. Phase field fracture modelling using quasi-Newton methods and a new adaptive step scheme[J]. Theor Appl Fract Mec, 2020, 107: 1.

[17] Kurdija A S, Silic M, Vladimir K, *et al.* Efficient global correlation measures for a collaborative filtering dataset [J]. Knowl-Based Syst, 2018, 147: 36.

引用本文格式:

中 文: 刘云, 郑文凤, 张轶. 对称加权算法对数据矩阵补全的优化研究[J]. 四川大学学报: 自然科学版, 2021, 58: 043001.

英 文: Liu Y, Zheng W F, Zhang Y. Optimization of data matrix completion by symmetric weighting algorithm [J]. J Sichuan Univ: Nat Sci Ed, 2021, 58: 043001.