计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.210100167

基于地标表示的联合谱嵌入和谱旋转的谱聚类算法

李 鹏 刘力军 黄永东
大连民族大学理学院 辽宁 大连 116600
(lipeng17835@163.com)

摘 要 经典的谱聚类算法包含两个步骤。(1)谱嵌入过程:求解 Laplacian 矩阵的特征值分解,得到分类指示矩阵的连续松弛解。(2)后处理过程:对谱嵌入连续松弛矩阵应用 k-means 或者谱旋转,得到最终的二值指示矩阵。由于有用信息的丢失,这种单独求解步骤不能保证最佳聚类结果。同时,谱聚类算法在处理大规模数据集时,存在聚类精度低、数据相似度矩阵存储开销大和 Laplacian 矩阵特征值分解计算复杂度高的问题。已有的联合谱聚类算法使用标准正交矩阵逼近非标准正交簇指示矩阵,这会导致较大的逼近误差。为了克服这一缺点,提出用一个改进的标准正交簇指示矩阵代替非正交指示矩阵,得到一个新的联合谱嵌入和谱旋转的谱聚类算法。因为两个标准正交矩阵更容易最小化,所以提出的算法可以取得更好的性能。进一步通过地标点方法对原始数据集进行稀疏特征表示,提出一种基于地标表示的联合谱嵌入和谱旋转算法(LJSESR),解决了大规模数据谱聚类的高效求解问题。实验结果表明,提出的 LJSESR 算法具有可行性和有效性。

关键词:谱聚类;谱旋转;谱嵌入;地标表示;联合谱聚类
中图法分类号 TP181

Landmark-based Spectral Clustering by Joint Spectral Embedding and Spectral Rotation

LI Peng, LIU Li-jun and HUANG Yong-dong
School of Science, Dalian Minzu University, Dalian, Liaoning 116600, China

Abstract Classical spectral clustering algorithms consist of two separate stages. One is spectral embedding, computing eigenvalue decomposition of a Laplacian matrix to obtain a relaxed continuous indication matrix. The other is post processing, applying k-means or spectral rotation to round the real matrix into the binary cluster indicator matrix. Such a separate scheme is not guaranteed to achieve jointly optimal result because of the loss of useful information. Meanwhile, there are difficulties of low clustering precision, high storage cost for the similarity matrix and high computational complexity for the eigenvalue decomposition of Laplacian matrix. The existing joint model adopts an orthonormal real matrix to approximate the orthogonal but nonorthonormal cluster indicator matrix. The error of approximating a nonorthonormal matrix is inevitably large. To overcome the drawback, we propose replacing the nonorthonormal cluster indicator matrix with an improved orthonormal cluster indicator matrix. The proposed method is capable of obtaining better performance because it is easy to minimize the difference between two orthonormal matrices. Furthermore, a novel landmark-based joint spectral embedding and spectral rotation algorithm is proposed based on the sparse representation by landmark points, which greatly solves the effective computation of spectral clustering for large scale dataset. Experimental results on benchmark datasets demonstrate the effectiveness of the proposed method.

Keywords Spectral clustering, Spectral rotation, Spectral embedding, Landmark representation, Joint spectral clustering

1 引言

聚类分析作为数据挖掘的一项重要技术,是发现和探索事物内在联系的有效手段,已经成为数据挖掘、模式识别等许多研究领域中的基本问题之一[1]。经典的聚类算法有 k-means[2]、FCM[3]和 PAM[4]等,但这些算法只适合发现球状簇,无法发现非凸形状的簇,且易陷入局部最优而不能发现数据集的真实分布。谱聚类算法是一种基于谱图划分理论的子空间聚类算法[5-6],与传统聚类算法相比,其对聚类样本空间的形状和维度没有特殊要求,且收敛于全局最优解。

谱聚类效果的好坏很大程度上取决于相似度矩阵的好坏。Li 等[7]通过计算样本间的 SimRank 得分得到相似度矩阵。Li 等[8]利用连接距离来测量样本点间的相似程度。考虑到位于同一流形体上的两点间会有许多较短的边连接,而位于不同流形体上的两点需要较长边相连。Tao 等[9]使用低密度分割密度敏感距离构造相似度矩阵,Chen 等[10]提出基于启发式确定类数的 NJW 谱聚类算法。Zhang 等[11]提出融入局部几何特征的流形谱聚类图像分割算法。为了降低谱聚类算法在大规模数据集上的复杂度,Charless 等[12]通过抽取原始数据矩阵的子集,生成原始矩阵的低秩近似方法,提出基于 Nyström 方法的谱聚类算法。Qiu 等[13]针对 Nyström 方法在谱聚类应用中存在聚类效果不稳定、样本代表性较弱的

基金项目:国家自然科学基金(61002039,61572018,11761001)
This work was supported by the National Natural Science Foundation of China(61002039,61572018,11761001).
通信作者:刘力军(manopt@163.com)

问题,提出基于加权集成 Nyström 采样的谱聚类算法。Cai 等[14-15]提出了基于地标的谱聚类(Landmark-based Spectral Clustering,LSC)算法,该方法择取具有代表性的样本点作为地标,通过地标的稀疏线性组合近似构造图相似度矩阵,从而有效降低谱嵌入的计算复杂度,其对大规模数据集表现出了优异的性能。Ye 等[16]将近似奇异值分解应用于基于地标的谱聚类,保证了聚类结果的稳定性和精度。Zhang 等[17]提出一种以增量形式抽样地标的快速谱聚类算法。Chen 等[18]提出通过带有 l_2 正则化项的最小二乘优化模型,求解基于地标的稀疏表示矩阵。Chu 等[19]通过选取数据邻接矩阵中权重最高的节点作为地标,提出了一种加权 PageRank 改进地标表示的自编码谱聚类算法。Zhang 等[20]提出结合度量融合和地标表示的自编码谱聚类算法。

谱聚类算法需要两个独立的阶段,即谱嵌入表示和后处理操作[5],典型的后处理包括 k-means[2] 和谱旋转方法[21-22]。这种独立的求解过程,由于有用信息的损失,不能保证获得联合的最佳聚类结果。为了解决这个问题,Yang 等[23]提出了联合谱嵌入与谱旋转方法,但是这种方法是用标准正交矩阵来逼近非标准正交矩阵,Pang 等[24]提出用正交化的聚类指示矩阵替换非正交聚类指示矩阵,得到了联合谱嵌入和谱旋转的谱聚类算法(Spectral Clustering by Joint Spectral Embedding and Spectral Rotation,JSESR)。Zhu 等[25]提出基于 $l_{2,1}$ 范数的联合谱聚类算法。

为此,本文提出了一种改进的联合谱聚类算法(Improved Joint Spectral Embedding and Spectral Rotation,IJSESR),该方法改进了文献[24]的结果。针对大规模数据集的谱聚类问题,得到了基于地标表示的联合谱聚类算法(Landmark-based Joint Spectral Embedding and Spectral Rotation,LJSESR),所提算法提高了大规模数据集的聚类效率和精度。

2　相关工作

2.1　谱聚类算法

谱聚类的思想来源于谱图划分,将数据聚类问题转化为一个无向图的多路划分问题[26]。该算法首先将样本点 P 定义为一个无向图 $G=(P,A)$ 的顶点,然后利用欧氏距离测度建立图中顶点之间的相似度矩阵 A,并对适当的 Laplacian 矩阵进行谱分解得到数据的谱嵌入表示,最后对低维表示进行 k-means 聚类或谱旋转得到最终的聚类指示矩阵。

下面给出谱聚类算法的一般步骤。

首先,给定样本点集 $P=\{x_1,x_2,\cdots,x_n\}$,$x_i\in\mathbb{R}^d$,n 为样本个数,将其划分为 k 类。可以基于 knn 方法构建相似度矩阵 $A\in\mathbb{R}^{n\times n}$,样本 x_i 和 x_j 的相似度典型定义为:

$$A_{ij}=\begin{cases}\exp\left(-\dfrac{\|x_i-x_j\|^2}{\sigma^2}\right), & i\neq j\\ 0, & i=j\end{cases} \tag{1}$$

其中,$\sigma>0$ 是决定样本点之间相似程度的尺度参数。

度矩阵构造如下:

$$D=\mathrm{diag}\{d_1,d_2,\cdots,d_n\}\in\mathbb{R}^{n\times n} \tag{2}$$

其中:

$$d_i=\sum_{j=1}^{n}A_{ij},i=1,2,\cdots,n \tag{3}$$

图 G 的 Laplacian 矩阵有两种定义方式。

(1)Ratio cut Laplacian[27]:

$L_{\mathrm{cut}}=D-A$;

(2)Normalized cut Laplacian[28-29]:

$L_{\mathrm{sym}}=I-D^{-1/2}AD^{-1/2}$

最后计算 Laplacian 矩阵的前 k 个特征向量,得到原始数据集的谱嵌入表示矩阵 $F\in\mathbb{R}^{n\times k}$,将矩阵 F 的每一行看作 $\mathbb{R}^k$ 中的一个点,对其使用 k-means 聚类或谱旋转得到最终的聚类指示矩阵。

2.2　基于地标的谱聚类算法

传统的谱聚类方法在处理大规模数据集时,数据相似度矩阵存储开销大,Laplacian 矩阵特征值分解计算复杂度高。文献[14,30]中,提出了基于地标的谱聚类算法(Landmarked-based Spectral Clustering,LSC)来加速谱聚类,在给定 n 个样本点的数据集的情况下,选择 $m\ll n$ 个具有代表性的样本点作为地标的特征点,将原始样本点表示为这些地标的线性稀疏组合,然后有效地利用稀疏表示矩阵逼近整个图的相似度矩阵,进而使得可以高效地计算数据的谱嵌入和谱旋转。

地标的生成可以随机选择和基于 k-means 方法生成。由于 k-means 聚类中心比随机选择的数据具有更强的表示能力,通常使用 k-means 来生成地标。假设得到 m 个地标 $W\in\mathbb{R}^{m\times d}$ 后,令 $W_{\langle i\rangle}\in\mathbb{R}^{m\times r}$ 表示 x_i 的 r 个最近的地标集合,可以通过下式得到稀疏表示矩阵 $B\in\mathbb{R}^{n\times m}$。

$$b_{ij}=\frac{K_h(x_i,w_j)}{\sum\limits_{j'\in W_{\langle i\rangle}}K_h(x_i,w_{j'})},j\in W_{\langle i\rangle} \tag{4}$$

其中,$K_h(\cdot)$ 是带宽为 h 的高斯核函数:$K_h(x_i,w_j)=\exp(-\|x_i-w_j\|^2/2h^2)$。

通过稀疏表示矩阵 B,定义对称双随机相似度矩阵 $A\in\mathbb{R}^{n\times n}$ 如下:

$$A=\hat{Z}\hat{Z}^{\mathrm{T}} \tag{5}$$

其中,$\hat{Z}=B\Delta^{-1/2}$,$\Delta\in\mathbb{R}^{m\times m}$ 为对角矩阵,对角元 $\Delta_{jj}=\sum\limits_{i=1}^{n}b_{ij}$。

容易计算,由相似度矩阵 A 得到对应的度矩阵 D 为单位矩阵:

$$D=\mathrm{diag}(A1_n)=\mathrm{I} \tag{6}$$

其中,1_d 表示全为 1 的 $d\times 1$ 向量,$\mathrm{diag}(\cdot)$ 表示由其向量参数构成的对角矩阵。

LSC 算法的谱嵌入表示矩阵可由 $\hat{Z}$ 的奇异值分解得到,在地标数量 $m\ll n$ 时,其时间复杂度 $O(m^3+m^2n)$ 远远低于 $O(n^3)$,这使得 LSC 算法在处理大规模数据集时性能优异。

2.3　联合谱嵌入和谱旋转的谱聚类(JSESR)

文献[23]提出如下融合谱嵌入与谱旋转的联合谱聚类算法目标函数:

$$\min_{F^{\mathrm{T}}F=I,R^{\mathrm{T}}R=I,Y\in\mathrm{Ind}}[\mathrm{tr}(F^{\mathrm{T}}L_{\mathrm{sym}}F)+\alpha\|FR-Y\|_F^2] \tag{7}$$

目标函数中第一项表示谱嵌入,第二项表示谱旋转,谱嵌入表示矩阵 $F\in\mathbb{R}^{n\times k}$,聚类指示矩阵 $Y=[\bar{y}_1,\bar{y}_2,\cdots,\bar{y}_N]^{\mathrm{T}}\in\mathbb{R}^{N\times K}$,$\bar{y}_i\in\{0,1\}$,$R$ 为任意的正交矩阵,$\alpha>0$ 是组合系数。

容易看到 FR 为标准正交矩阵,即 $(FR)^{\mathrm{T}}(FR)=\mathrm{I}$,但 $Y^{\mathrm{T}}Y\neq\mathrm{I}$,因此文献[24]提出如下改进的目标函数:

$$\min_{F^{\mathrm{T}}F=I,R^{\mathrm{T}}R=I,Y\in\mathrm{Ind}}[\mathrm{tr}(F^{\mathrm{T}}L_{\mathrm{sym}}F)+\alpha\|FR-D^{1/2}Y(Y^{\mathrm{T}}DY)^{-1/2}\|_F^2] \tag{8}$$

优化问题(8)不仅包含实值变量 F 和 R,也包含聚类指示矩阵 Y。文献[24]给出了基于坐标下降法的交替优化算法。

3 本文算法

3.1 改进的联合谱聚类算法(IJSESR)

在优化问题(8)中,引入 $Y_s=D^{1/2}Y(Y^{\mathrm{T}}DY)^{-1/2}$ 的目的是使其满足标准正交性,即 $Y_s^{\mathrm{T}}Y_s=\mathrm{I}$。事实上,若令 $\hat{Y}_s=Y(Y^{\mathrm{T}}Y)^{-1/2}$,则易证 $\hat{Y}_s^{\mathrm{T}}\hat{Y}_s=\mathrm{I}$。

因此,本文提出如下联合优化算法:

$$\min_{F^{\mathrm{T}}F=\mathrm{I},R^{\mathrm{T}}R=\mathrm{I},Y\in\mathrm{Ind}}\left[\mathrm{tr}(F^{\mathrm{T}}L_{\mathrm{sym}}F)+\alpha\left\|FR-Y(Y^{\mathrm{T}}Y)^{-1/2}\right\|_F^2\right] \tag{9}$$

其中,$L_{\mathrm{sym}}=\mathrm{I}-D^{-1/2}AD^{-1/2}$。

与(8)式类似,因为 FR 和 $Y(Y^{\mathrm{T}}Y)^{-1/2}$ 都是标准正交矩阵,所以用 FR 逼近 $Y(Y^{\mathrm{T}}Y)^{-1/2}$ 是合理的,然而与(8)式相比,我们的算法式(9)更简洁,运算复杂度更低。更为重要的是,基于地标的联合谱聚类算法的 Laplacian 矩阵 $L_{\mathrm{sym}}=\mathrm{I}-A=L_{\mathrm{cut}}$,即此时的规范化 Laplacian 矩阵转化为非规范化 Laplacian 矩阵,而算法式(9)中谱旋转部分则保持不变,也就是说算法式(9)强调了谱旋转是与 Laplacian 矩阵无关的,这更符合谱聚类的思想。

3.2 优化算法

为方便起见,接下来将 L_{sym} 简记为 L。优化问题(9)不仅包含实变量 F 和 R,还包含聚类指示矩阵 Y,对于这个复杂的问题,很难找到全局最优解。我们采用交替方向法进行求解。具体来说,该算法迭代执行更新 R,Y 和 F。

(1)R-步:固定式(9)中 F 和 Y,省略与 R 无关的项,R 的求解转化为如下优化问题:

$$\min_{R^{\mathrm{T}}R=1}\left\|FR-Y(Y^{\mathrm{T}}Y)^{-1/2}\right\|_F^2 \tag{10}$$

注意到 $F^{\mathrm{T}}F=\mathrm{I}$,则最小化问题(10)等价于:

$$\max_{R^{\mathrm{T}}R=1}\mathrm{tr}(R^{\mathrm{T}}F^{\mathrm{T}}Y(Y^{\mathrm{T}}Y)^{-1/2})=\max_{R^{\mathrm{T}}R=I}\mathrm{tr}(R^{\mathrm{T}}M) \tag{11}$$

其中,$M\overset{\triangle}{=}F^{\mathrm{T}}Y(Y^{\mathrm{T}}Y)^{-1/2}$。

定理 1 给出了式(11)的最优解 R^*,证明见文献[24]。

定理 1 设 M 的奇异值分解为 $M=USV^{\mathrm{T}}$,那么问题式(11)的最优解 R^* 为:

$$R^*=UV^{\mathrm{T}} \tag{12}$$

(2)Y-步:固定式(9)中 R 和 F,省略与 Y 无关的项,Y 的求解转化为如下优化问题:

$$\min_{Y\in\mathrm{Ind}}\left\|FR-Y(Y^{\mathrm{T}}Y)^{-1/2}\right\|_F^2\Rightarrow\min_{Y\in\mathrm{Ind}}\left\|F-Y(Y^{\mathrm{T}}Y)^{-1/2}R^{\mathrm{T}}\right\|_F^2 \tag{13}$$

假设第 k 个簇的元素样本点数正比于该类的度 d_k,则式(13)的最优解 Y^* 是:

$$Y_{ij}^*=\begin{cases}1, & \text{if } j=\arg\min_k\left\|f_i-\dfrac{r_k}{\sqrt{d_k}}\right\|\\ 0, & \text{else}\end{cases} \tag{14}$$

其中,f_i 是矩阵 F 的第 i 行,r_k 是矩阵 R^{T} 的第 k 行。

(3)F-步:固定式(9)中 R 和 Y,省略与 F 无关的项,F 的求解转化为如下优化问题:

$$\begin{aligned}&\min_{F^{\mathrm{T}}F=1}\left[\mathrm{tr}(F^{\mathrm{T}}LF)+\alpha\left\|FR-Y(Y^{\mathrm{T}}Y)^{-1/2}\right\|_F^2\right]\\&\Rightarrow\min_{F^{\mathrm{T}}F=1}\left[\mathrm{tr}(F^{\mathrm{T}}LF)+\alpha(\mathrm{tr}((FR-Y(Y^{\mathrm{T}}Y)^{-1/2})^{\mathrm{T}}(FR-Y(Y^{\mathrm{T}}Y)^{-1/2})))\right]\\&\Rightarrow\min_{F^{\mathrm{T}}F=1}\left[\mathrm{tr}(F^{\mathrm{T}}LF)-2\alpha(\mathrm{tr}(FR)^{\mathrm{T}}Y(Y^{\mathrm{T}}Y)^{-1/2})\right]\\&\Rightarrow\min_{F^{\mathrm{T}}F=1}\left[\mathrm{tr}(F^{\mathrm{T}}LF)-2\alpha(\mathrm{tr}(F^{\mathrm{T}}Y(Y^{\mathrm{T}}Y)^{-1/2}R^{\mathrm{T}}))\right]\\&\Rightarrow\min_{F^{\mathrm{T}}F=1}\left[\mathrm{tr}(F^{\mathrm{T}}LF-2\alpha F^{\mathrm{T}}C)\right]\end{aligned} \tag{15}$$

在式(15)的最后一行,矩阵 C 定义为:

$$C=Y(Y^{\mathrm{T}}Y)^{-1/2}R^{\mathrm{T}} \tag{16}$$

式(15)的问题可以进一步转化为:

$$\max_{F^{\mathrm{T}}F=I}\left[\ \mathrm{tr}\ (F^{\mathrm{T}}\ \widetilde{B}F)+2\alpha(\ \mathrm{tr}\ (F^{\mathrm{T}}C))\right] \tag{17}$$

其中,$\widetilde{B}=\lambda I+D^{-1/2}AD^{-1/2}\in\mathbb{R}^{N\times N}$。$\lambda>0$ 为任意常数,用以保证 $\widetilde{B}$ 为半正定矩阵。

式(17)为紧致 Stiefel 流形上的凸优化问题,可以使用广义幂法求解[31]。具体来说,从任一点 $F^{(t)}$ 开始,设 $E=\widetilde{B}F^{(t)}+\alpha C$,将 E 奇异值分解记为 $E=\widetilde{U}\widetilde{S}\widetilde{V}^{\mathrm{T}}$,其中 $\widetilde{U}\in\mathbb{R}^{n\times k}$,$\widetilde{S}\in\mathbb{R}^{k\times k}$,$\widetilde{V}\in\mathbb{R}^{k\times k}$,更新 $F^{(t+1)}=\widetilde{U}\widetilde{V}^{\mathrm{T}}$,那么式(17)的目标函数单调递增,且迭代算法具有线性收敛性。

具体来说,本文提出了改进的联合谱聚类算法(Improved JSESR,IJSESR)。算法 1 给出了 F-步的详细解决方案。算法 2 给出了 IJSESR 算法流程,其中 R-步、Y-步和 F-步迭代进行。

算法 1 求解问题式(17)的算法

输入:矩阵 F,Y,R,A,D,参数 $\lambda,\alpha>0$,最大迭代次数 T_1

输出:矩阵 F

1. 根据式(16)计算 C 和 $\widetilde{\mathrm{B}}$;
2. while 迭代次数$\leqslant\mathrm{T}_1$ 时 do
3. 更新 $\mathrm{E}=\widetilde{\mathrm{B}}\mathrm{F}+\alpha\mathrm{C}$;
4. 通过奇异值分解 E,计算得 $\widetilde{\mathrm{U}}\widetilde{\mathrm{S}}\widetilde{\mathrm{V}}^{\mathrm{T}}=\mathrm{E}$;
5. 更新 $\mathrm{F}=\widetilde{\mathrm{U}}\widetilde{\mathrm{V}}^{\mathrm{T}}$;
6. end while

算法 2 IJSESR 算法

输入:n 个样本 $\mathrm{P}=[\mathrm{x}_1,\mathrm{x}_2,\cdots,\mathrm{x}_n]^{\mathrm{T}}\in\mathbb{R}^{n\times d}$,聚类数 k,参数 $\alpha>0$,最大迭代次数 T_2

输出:聚类指示矩阵 Y

1. 根据式(1)计算相似度矩阵 A;
2. 随机初始化 F,R 和 Y;
3. while 迭代次数$\leqslant\mathrm{T}_2$ 时 do
4. 固定 Y 和 R,根据算法 1 更新 F;
5. 固定 F 和 Y,根据式(12)计算正交矩阵 R;
6. 固定 F 和 R,根据式(14)更新 Y;
7. end while

3.3 基于地标表示的联合谱聚类算法

为了解决传统的谱聚类方法在处理大规模数据集时数据相似度矩阵存储开销大、计算复杂度高的问题,本节中将地标方法引入 IJSESR 算法中,提出基于地标表示的联合谱聚类算法(Landmark-based JSESR,LJSESR)。即首先通过 k-means 方法生成地标,然后通过式(4)和式(5)构造相似度矩阵 $A=\hat{Z}\hat{Z}^{\mathrm{T}}$,接着根据式(12)、式(14)和式(17),交替迭代更新 F,R,Y,最终得到数据集的聚类指示矩阵 Y。

值得指出的是,在 LJSESR 中 $A\geq0$,$D=I$,因此 $L_{\mathrm{sym}}=\mathrm{I}-A=L_{\mathrm{cut}}$,这样可以令 $\widetilde{B}=\lambda I+A$ 中的 $\lambda=0$,即可保证式(17)的目标函数为凸函数,从而使得交替迭代算法中 F-步得以顺利执行,同时避免了参数 $\lambda>0$ 的选择问题。算法 3 给

出了 LJSESR 算法流程。

算法 3　LJSESR 算法

输入：n 个样本 $P=[x_1,x_2,\cdots,x_n]^T\in\mathbb{R}^{n\times d}$，聚类数 k，参数 $\alpha>0$，最大迭代次数 T_3

输出：聚类指示矩阵 Y

1. 使用 k-means 产生 m 个地标；
2. 根据式(4)计算稀疏表示矩阵 $B\in\mathbb{R}^{n\times m}$；
3. 根据式(5)计算相似度矩阵 A；
4. 随机初始化 F，R 和 Y；
5. while 迭代次数$\leqslant T_3$ 时 do
6.　固定 Y 和 R，根据算法 1 更新 F；
7.　固定 F 和 Y，根据式(12)计算正交矩阵 R；
8.　固定 F 和 R，根据式(14)更新 Y；
9. end while

4　实验与分析

本节将提出的 IJSESR 和 LJSESR 算法，与 k-means[2]，Rcut[27]，Ncut[28-29]，LSC-K[14]，JSESR[24] 进行比较。

4.1　数据集

Circle 数据集是一个人工数据集，$n=1\,000$ 时的数据分布如图 1 所示；其他数据来自 UCI 数据集[32]。表 1 列出了实验中使用的数据集的特征。所有实验均在 Windows 10，CPU 2.4 GHz，8 GB RAM 的计算机上通过 MatlabR2020a 实现。

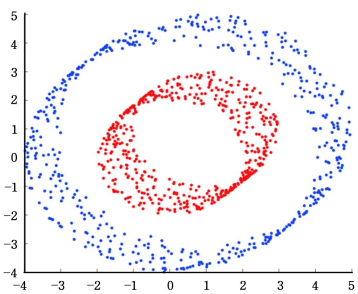


图 1　Circle 数据集

Fig. 1　Circle data set

表 1　11 个数据集的描述

Table 1　Description of 11 data sets

数据集	数据个数	维数	聚类数
Circle	1 000	2	2
Wine	178	13	3
Cars	392	8	3
Iris	150	4	3
Ecoil	336	7	8
Haberman	306	3	2
Cancer	683	9	2
USPS	9 280	256	10
Pendigits	10 992	16	10
LetterRec	20 000	16	26
MNIST	70 000	784	10

4.2　参数设置

为保证算法之间的公平对比，具体参数设置如下：文中列出的全部算法都是基于多次运行 knn 方法，以确定最佳的近邻点数目来构造相似度矩阵，式(1)中的 σ 设置为 1，所有算法有涉及 k-means 迭代的都设为 20 次。利用折中参数 α 平衡了光谱嵌入部分和光谱旋转部分。α 的值从集合 $\{10^{-3},10^{-2},10^{-1},10^{0},10^{1},10^{2},10^{3}\}$ 选取最佳的结果。所有聚类算法都运行了 20 次，并给出了平均结果。设置最大迭代次数 T_1,T_2,T_3 分别为 100，10，10。LSC-K 和 LJSESR 算法针对 11 个数据集的地标数量依次设置为 200，150，120，120，70，160，210，2 000，1 000，1 000，1 000。

4.3　评价指标

通过将每个样本获得的标签与数据集提供的标签进行比较，对聚类结果进行评价。本文利用 3 个常见的评价指标来评价算法的性能，分别是精确度(Accuracy，ACC)、标准化互信息(Normalized Mutual Information，NMI)和纯度(Purity)。

ACC：用于比较聚类结果标签和数据集提供的真实标签。

$$ACC=\frac{1}{n}\sum_{i=1}^{c}|T_i\cap P_i| \tag{18}$$

其中，$T=\{T_1,T_2,\cdots,T_c\}$ 表示原始的 n 个数据包含真实的 c 个类别，$P=\{P_1,P_2,\cdots,P_c\}$ 表示聚类后 n 个数据的 c 个预测类别。T_i 表示第 i 个类别中包含的样本点，$|T_i|$ 表示集合 T_i 包含的点的个数；P_i 表示聚类后第 i 个类别中包含的样本点，$|P_i|$ 表示集合 P_i 中包含的点的个数。

NMI：用来衡量两个聚类结果的相似程度。

$$NMI=\frac{\sum_{i,j=1}^{c}|T_i\cap P_j|\lg\left(\frac{n|T_i\cap P_j|}{|T_i||P_j|}\right)}{\sqrt{\left(\sum_{i=1}^{c}|T_i|\lg|T_i|/n\right)\left(\sum_{i=1}^{c}|P_j|\lg|P_j|/n\right)}} \tag{19}$$

Purity：给每个簇分配一个类别，如某个类别的样本在该簇中出现次数最多，则计算所有 c 个簇中这个次数之和。

$$Purity=\frac{1}{n}\sum_{i=1}^{c}\max_{1\leqslant j\leqslant c}|T_i\cap P_i| \tag{20}$$

4.4　实验结果

表 2－表 4 比较了 11 个数据集在不同算法上的 ACC，NMI 和 Purity 性能，每个数据集的最佳结果用黑体标注。

表 2　各算法的 *ACC* 对比

Table 2　Comparison of *ACC* of each algorithm

(单位：%)

数据集	k-means	Rcut	Ncut	LSC-K	JSESR	IJSESR	LJSESR
Circle	50.90	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**
Wine	70.22	38.76	39.32	70.22	74.43	73.56	**76.71**
Cars	44.89	63.01	44.13	67.09	68.21	67.46	**68.87**
Iris	88.66	90.66	90.66	86.00	87.45	87.56	**96.00**
Ecoil	63.69	58.03	57.44	64.88	56.75	56.76	**71.41**
Haberman	50.00	73.85	73.20	50.32	66.76	65.35	**74.18**
Cancer	96.04	65.15	66.03	96.34	96.74	96.66	**97.07**
USPS	67.49	66.41	72.73	65.37	64.72	64.34	**74.63**
Pendigits	65.17	73.47	66.29	78.29	67.43	66.44	**81.88**
LetterRec	24.71	4.62	8.96	31.91	31.46	30.44	**32.18**
MNIST	54.28	64.23	72.76	69.61	72.43	72.32	**73.42**

表 3　各算法的 *NMI* 对比

Table 3　Comparison of *NMI* of each algorithm

(单位：%)

数据集	k-means	Rcut	Ncut	LSC-K	JSESR	IJSESR	LJSESR
Circle	0.02	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**
Wine	42.87	3.72	7.02	42.87	42.56	42.43	**43.15**
Cars	20.48	4.74	0.33	18.85	18.96	18.47	**20.96**
Iris	74.19	80.57	80.57	73.78	76.65	76.32	**86.41**
Ecoil	60.30	61.82	60.35	59.24	55.35	56.43	**64.25**
Haberman	0.11	3.86	0.89	0.31	5.56	5.39	**8.77**
Cancer	74.78	1.82	5.64	76.02	73.98	74.76	**81.13**
USPS	61.56	**81.86**	78.22	76.75	71.34	71.22	72.91
Pendigits	66.98	79.27	**81.76**	75.83	66.43	65.76	76.06
LetterRec	34.51	5.12	18.04	**44.33**	40.32	38.24	42.76
MNIST	48.40	56.23	66.43	69.07	70.87	70.02	**71.96**

表 4 各算法的 *Purity* 对比

Table 4 Comparison of *Purity* of each algorithm

(单位:%)

数据集	k-means	Rcut	Ncut	LSC-K	JSESR	IJSESR	LJSESR
Circle	50.90	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**
Wine	70.22	39.88	41.01	70.22	72.43	72.34	**74.71**
Cars	65.05	63.01	62.50	68.77	67.43	66.65	**68.87**
Iris	88.66	90.66	90.66	86.00	87.76	87.45	**96.00**
Ecoil	79.16	83.33	79.46	80.14	77.54	78.34	**80.95**
Haberman	73.52	73.85	73.52	73.52	73.67	74.13	**74.18**
Cancer	96.04	65.15	66.03	96.34	94.45	94.66	**97.07**
USPS	73.74	80.75	72.95	79.21	76.35	76.37	**81.02**
Pendigits	70.10	73.69	80.67	78.77	65.64	64.67	**81.88**
LetterRec	26.66	4.72	9.80	**35.66**	32.04	31.56	34.84
MNIST	58.05	65.42	74.12	74.76	75.89	74.32	**76.16**

除 k-means 算法外,其他算法对 Circle 数据集的聚类准确率均能达到 100%,这是因为 k-means 无法正确划分非凸数据集,而谱聚类算法可以实现对任意数据的划分。基于地标表示的 LSC-K 和 LJSESR 对 Cancer 数据集聚类后达到了几乎相同的准确率,在 MNIST 数据集上,LJSESR 表现好于 LSC-K 算法,说明联合聚类算法可以提升聚类准确率。可以发现,对于 ACC,NMI,Purity 3 个指标,除 Circle 数据集外,本文提出的 IJSESR 性能与 JSESR 接近,LJSESR 在多数数据集上优于 k-means,Rcut 和 Ncut。对于 LetterRec 和 MNIST 数据集,LSC-K 和 LJSESR 算法有明显提升,原因是基于地标稀疏表示的有效性[14-15]。在 JSESR 和 IJSESR 算法上有明显提升,是由于联合谱嵌入和谱旋转方法可以避免有用信息的损失。

表 5 列出了 11 个数据集在不同算法上的 CPU 运行时间。可以发现,在多数数据集上,k-means 方法时间性能最好,但当数据规模超过 2000 时,基于地表点表示的 LJSESR 和 LSC-K 的 CPU 占用时间优势明显。

表 5 各算法的 CPU 时间对比

Table 5 Comparison of CPU time of each algorithm

(单位:s)

数据集	k-means	Rcut	Ncut	LSC-K	JSESR	IJSESR	LJSESR
Circle	**0.0050**	0.0260	0.0900	0.0180	0.0430	0.0360	0.0170
Wine	**0.0060**	0.0120	0.0080	0.0080	0.0070	0.0630	0.0130
Cars	**0.0030**	0.0130	0.0139	0.0070	0.0130	0.0130	0.0060
Iris	**0.0019**	0.0139	0.0080	0.0050	0.0140	0.0120	0.0080
Ecoil	**0.0030**	0.0279	0.0160	0.0080	0.0280	0.0240	0.0080
Haberman	**0.0020**	0.0109	0.0090	0.0110	0.0130	0.0120	0.0060
Cancer	**0.0019**	0.0270	0.0179	0.0220	0.0230	0.0210	0.0170
USPS	**0.5910**	7.6630	6.6120	1.3770	6.6310	6.5320	1.1590
Pendigits	**0.0460**	0.5520	1.3420	0.7980	0.8740	0.7450	0.6560
LetterRec	**0.2669**	5.6660	15.5240	2.9230	14.3240	14.2540	2.2790
MNIST	23.2000	45.6334	156.4560	7.1720	125.8340	105.4210	**6.5279**

由图 2 可以看出,随着样本点数的增加,Ncut,JSESR,IJSESR 的 CPU 占用时间迅速增加;Rcut,LSC-K,LJSESR,k-means 算法的 CPU 占用时间少;Rcut 的 CPU 占用时间少于 Ncut 的主要原因是:Rcut 需要求解稀疏矩阵的特征值分解问题,复杂度低于 Ncut 需要求解的广义特征值分解问题。

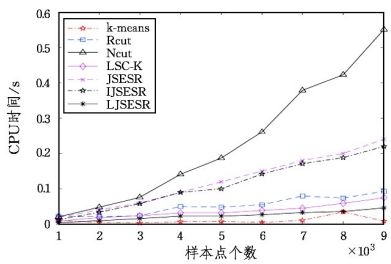


图 2 各算法在 Circle 数据集上的 CPU 时间

Fig. 2 CPU time of each algorithm on the Circle data set

地标个数对 LJSESR 算法的影响如图 3 所示。可以看到,随着地标个数的增加,ACC,NMI,Purity 指标都显著提高。

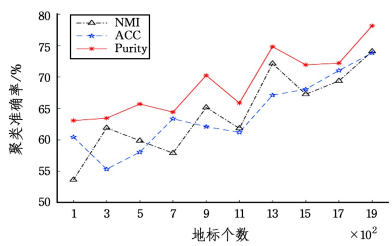


图 3 地标个数对 LJSESR 算法的影响

Fig. 3 Influence of the number of landmarks on LJSESR algorithm

结束语 本文提出了一种改进的联合谱聚类算法。该方法使用规范化的聚类指示矩阵逼近旋转嵌入表示矩阵,交替优化谱嵌入和谱旋转过程,避免了分离优化的精度损失。针对大规模数据集,提出了基于地标表示的联合谱嵌入和谱旋转的交替优化算法,它所构造的数据相似度矩阵是半正定的,避免了谱嵌入计算过程中的参数选择问题。在人工数据集和 UCI 真实数据集上的实验表明,相比于传统的谱聚类算法,基于地标表示的联合谱聚类算法能够以较少的 CPU 占用时间取得较高的聚类精度。

参 考 文 献

[1] JAIN A K, MURTY M N, FLYNN P J. Data clustering: a review[J]. ACM Computing Surveys, 1999, 31(3): 264-323.

[2] MACQUEEN J. Some methods for classification and analysis of multivariate observations[J]. Berkeley Symposium on Mathematical Statistics and Probability, 1967, 1(5): 281-297.

[3] FU G Y. Optimization methods for fuzzy clustering[J]. Fuzzy Sets and Systems, 1998, 93(3): 301-309.

[4] KAUFMAN L, ROUSSEEUW P J. Finding Groups in Data: An Introduction to Cluster Analysis[J]. John Wiley & Sons, New York, USA, 1990.

[5] LUXBURG U V. A tutorial on spectral clustering[J]. Statistics and Computing, 2007, 17(4): 395-416.

[6] JIN J G. Review of Clustering Method[J]. Computer Science, 2014, 41(S2): 288-293.

[7] LI P Q, LI Y D, DENG X L, et al. Spectral Clustering Algorithm

Based on SimRank Score[J]. Computer Science,2018,45(S2):458-461.

[8] LI X Y,GUO L J. Constructing affinity matrix in spectral clustering based on neighbor propagation[J]. Neurocomputing,2012,97:125-130.

[9] TAO X M,WANG R T,CHANG R,et al. Low Density Separation Density Sensitive Distance-based Spectral Clustering Algorithm[J]. Acta Automatica Sinica,2020,46(7):1479-1495.

[10] CHEN J F,ZHANG M,HE Q. Heuristically Determining Cluster Numbers Based NJW Spectral Clustering Algorithm[J]. Computer Science,2018,45(S2):474-479.

[11] ZHANG R G,YAO X L,ZHAO J,et al. Manifold Spectral Clustering Image Segmentation Algorithm Based on Local Geometry Features[J]. Pattern Recognition and Artificial Intelligence,2020,33(4):313-324.

[12] CHARLESS F,SERGE B,FAN C,et al. Spectral grouping using the Nyström method[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence,2004,26(2):214-225.

[13] QIU Y F,LIU C. Spectral Clustering Algorithm Based on Weighted Ensemble Nyström Sampling[J]. Pattern Recognition and Artificial Intelligence,2019,32(5):420-428.

[14] CHEN X L,CAI D. Large scale spectral clustering with landmark-based representation[C]// AAAI Conference on Artificial Intelligence. San Francisco,USA,2011:313-318.

[15] CAI D,CHEN X L. Large scale spectral clustering via landmark-based sparse representation[J]. IEEE Transactions on Cybernetics,2015,45(8):1669-1680.

[16] YE M,LIU W F. Large Scale Spectral Clustering Based on Fast Landmark Sampling[J]. Journal of Electronics & Information Technology,2017,39(2):278-284.

[17] ZHANG X C,ZONG L L,YOU Q Z,et al. Sampling for Nyström extension based spectral clustering: incremental perspective and novel analysis[J]. ACM Transactions on Knowledge Discovery from Data,2016,11(1):1-25.

[18] CHEN X J,NIE F P,HUANG Z X,et al. Scalable normalized cut with improved spectral rotation[C]// Twenty-Sixth International Joint Conference on Artificial Intelligence. 2017:1518-1524.

[19] CHU D R,ZHOU Z P. An autoencoder spectral clustering algorithm for improving landmark representation by weighted PageRank[J]. CAAI Transactions on Intelligent Systems,2020,15(2):302-309.

[20] ZHANG M,ZHOU Z P. An autoencoder-based spectral clustering algorithm combined with metric fusion and landmark representation[J]. CAAI Transactions on Intelligent Systems,2020,15(4):687-696.

[21] YU S X,SHI J B. Multiclass spectral clustering[C]// Proceedings of the 9th IEEE International Conference on Computer Vision(ICCV'03). USA,2003:313-319.

[22] HUANG J,NIE F P,HUANG H. Spectral rotation versus k-means in spectral clustering[C]// Proceedings of the AAAI Confe rence on Artificial Intelligence(AAAI'13). AAAI Press,2013:431-437.

[23] YANG Y,SHEN F M,HUANG Z,et al. A unified framework for discretespectral clustering[C]// International Joint Conference on Artificial Intelligence. 2016:2273-2279.

[24] PANG Y W,XIE J,NIE F P,et al. Spectral clustering by joint spectral embedding and spectral rotation[J]. IEEE Transactions on Cybernetics,2020,50(1):247-258.

[25] ZHU J T,JANG J L,LIU T,et al. Joint spectral clustering based on optimal graph and feature selection[J]. Neural Processing Letters,2020(11):1-17,

[26] ZHOU L,PING X J,XU S,et al. Cluster Ensemble Based on Spectral Clustering[J]. Acta Automatica Sinica,2012,38(8):1335-1342.

[27] STOER M,WAGNER F. A simple min-cut algorithm[J]. Journal of the ACM,1997,44(4):585-591.

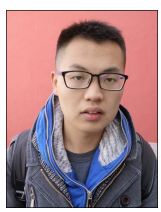
[28] SHI J B,JITENDRA. Normalized cuts and image segmentation [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence,2000,11(9):888-905.

[29] NG A Y,JORDAN M I,WEISS Y. On spectral clustering:Analysis and an algorithm[C]// Proceedings of the 14th International Conference on Neural Information Processing Systems:Natural and Synthetic. Cambridge,MA,USA,2002:849-856.

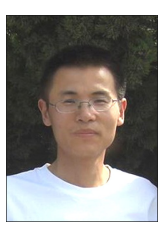
[30] LIU W,HE J F,CHANG S F. Large graph construction for scalable semi-supervised learning[C]// Proceeding,Twenty-Seventh International Conference on Machine Learning. Haifa,Israel,2010:679-686.

[31] JOURNÉE M,NESTEROV Y,RICHTÁRIK P,et al. Generalized power method for sparse principal component analysis[J]. Journal of Machine Learning Research,2010,11(2):517-553.

[32] DHEERU D,TANISKIDOU E K. UCI Machine Learning Repository[EB/OL]. http://archive.ics.uci.edu/ml.



LI Peng, master student. His main research interests include machine learning and optimization method.



LIU Li-jun, Ph. D, associate professor. His main research interests include neural networks, machine learning and optimization method.