大规模数据集下谱聚类算法的求解

史卫亚[1,2] 郭跃飞[3]

(粮食信息处理与控制教育部重点实验室 郑州 450001)[1]

(河南工业大学信息科学与工程学院 郑州 450001)[2] (复旦大学计算机科学与技术系 上海 200433)[3]

摘 要 谱聚类算法是一种流行的数据聚类方法,该算法使用特征分解技术计算邻接矩阵的特征解,但是在大规模数据集的情况下,因储存和计算的问题而无法进行求解。基于线性代数中对称矩阵的性质,提出使用邻接矩阵的每一列作为迭代算法的输入样本,通过迭代计算出邻接矩阵的特征解。所提算法的空间复杂度只有 $O(m)$,时间复杂度也降低为 $O(pkm)$。实验结果验证了算法的有效性。

关键词 谱方法,邻接矩阵,大数据集,特征分解

Computation of Spectral Clustering Algorithm for Large-scale Data Set

SHI Wei-ya[1,2] GUO Yue-fei[3]

(Key Laboratory of Grain Information Processing and Control of Ministry of Education, Zhengzhou 450001, China)[1]

(School of Information Science and Engineering, Henan University of Technology, Zhengzhou 450001, China)[2]

(Department of Computer Science and Technology, Fudan University, Shanghai 200433, China)[3]

Abstract Spectral clustering algorithm is a popular data clustering method. It uses eigen-decomposition technique to extract the eigenvectors of the affinity matrix. But the method is infeasible for large-scale data set because of the store and computational problem. Motivated by the property of symmetric matrix, in this paper each column of the affinity matrix was used as the input sample for the iterative algorithm. The eigenvectors of the affinity matrix could be iteratively computed. The space complexity of proposed method was only $O(m)$, the time complexity was reduced to $O(pkm)$. The effectiveness of proposed method was validated from experimental results.

Keywords Spectral algorithm, Affinity matrix, Large-scale data set, Eigen-decomposition

1 引言

在统计机器学习、模式识别以及数据挖掘等过程中,聚类分析常用来发现数据分布规律和其中的隐含模式[1]。所谓聚类就是把数据集合分成若干类(或簇),使得同一类中的数据之间具有较高的相似度,而不同类之间的数据差别较大。常用的方法有 K-means 算法、EM 算法等。谱聚类算法[2]是近些年出现的一类新型聚类算法,其思想来源于谱图划分理论[3]。与传统的聚类算法相比,它对不规则的误差数据不是太敏感,计算复杂度较小,而且性能也要好一些,近些年成功应用于机器视觉、图像分割等领域[4,5]。

谱聚类算法将聚类问题转化为图的最优划分问题,计算过程中首先构造图并用邻接矩阵(或亲合矩阵)的形式表示出来,然后对该矩阵进行特征分解求得其特征值和特征向量,进而选择合适的特征向量对这些数据点进行聚类。该算法的空间复杂度和时间复杂度分别是 $O(m^2)$ 和 $O(m^3)$,其中 m 是数据集样本的数量。尽管算法很简单,但是随着样本数量的增加,在大规模数据集的情况下,由于存储空间的限制,普通计算机上实现谱聚类算法变得不可行。

为实现大规模数据集情况下谱聚类算法的求解,许多作者提出了解决方法。文献[6-8]提出分割原数据集,即通过分割来减少构造谱图的节点数目。在分割过程中,一般先对大规模数据集进行预处理,常用的方法是先利用 K-means 方法聚类数据集,然后取每个聚类的中心作为该聚类的代表点,最后将全部聚类中心的代表点作为谱图的节点,这样可以较大程度地减少邻接矩阵的大小。文献[9,10]提出使用线性代数中秩近似的思想,即从数据集中选取部分数据点,利用这些数据点形成邻接矩阵,特征分解该矩阵可得到其特征向量,进而使用 Nyström 近似可以求解原矩阵的特征解。

我们曾提出一种求解大规模数据集的核主成分分析方法[11]。本文中扩展该方法来实现大规模数据集情况下的谱聚类算法的求解。主要思想是利用线性代数中对称矩阵的性质,首先使用初始的邻接矩阵创建一个新的矩阵。因为新构成的矩阵和原先的矩阵具有相同的特征向量,所以我们可以计算新矩阵的特征向量,通过分析协方差矩阵的性质,可以把邻接矩阵的每一列看成迭代算法的输入样本。这样经过若干

本文受国家自然基金项目(60875003),河南省教育厅自然科学研究计划项目(2010B520005),河南工业大学博士基金项目(2009BS013),河南省科技厅重点科技攻关项目(112102210190),郑州市科技发展计划项目(2010SFXM470)资助。

史卫亚(1973-),男,博士,副教授,主要研究方向为数据挖掘、模式识别,E-mail:wyshi@fudan.edu.cn;郭跃飞(1964-),男,博士,副教授,主要研究方向为机器学习、模式识别。

次迭代后，可以求出邻接矩阵的特征值和特征向量。所提出的方法不需要事先存储邻接矩阵，空间复杂度从 $O(m^2)$ 减少到 $O(m)$。更为重要的是在处理大规模数据集情况下，传统的特征分解技术无法使用，文中提出的方法仍然可以较好实现数据聚类。在人工合成的数据集以及真实的数据上进行的实验，充分验证了该算法的有效性。

2 基于迭代的谱聚类方法

2.1 谱聚类算法回顾

谱聚类算法的实现基本过程可以大致概括如下：

1)根据样本数据集构造一个图，图的每一个节点对应一个数据点，点和点之间用边连接，并且边的权重用于表示数据之间的相似度。这个图可以用邻接矩阵的形式表示出来，记为 W，其每一个元素为边的权重。

2)把矩阵 W 的每一列元素加起来得到 m 个数，把它们分别放在 $m\times m$ 的矩阵的对角线上，其他位置元素都是零，记该矩阵为 D。并令 $L=D-W$。

3)利用特征分解方法求出矩阵 L 的前 k 个特征值 $\{\lambda\}_{i=1}^{k}$ 以及对应的特征向量 $\{\nu\}_{i=1}^{k}$。

4)这 k 个特征向量组成一个 $m\times k$ 的矩阵，其中每一行可看作 k 维空间中的一个向量，最后使用 K-means 算法对这 m 个向量进行聚类。

很多作者提出了不同的谱聚类算法，其主要区别在第 3 步，例如文献[4]取第二小的特征值和对应的特征向量，文献[12]按照特征值的大小从大到小的顺序取前 k 个，…等等。本文利用文献[12]提出的 NJW 方法实现数据的聚类，其它谱聚类算法也可以利用文中方法类似实现。

2.2 提出的方法

本文所用方法的主要思想来源于线性代数的一个基本定理。下面首先给出该定理的简要说明。

定理 1 矩阵 H 和 H^2 具有不同的特征值和相同的特征向量。

证明：假定 λ 和 ω 分别是矩阵 H 所对应的特征值和特征向量，有

$$H\omega=\lambda\omega \tag{1}$$

$$H^2\omega=HH\omega=\lambda H\omega=\lambda^2\omega \tag{2}$$

从式(1)和式(2)可知，矩阵 H^2 所对应的特征值和特征向量分别为 λ^2 和 ω，即矩阵 H 和 H^2 具有不同的特征值，和相同的特征向量。

因为邻接矩阵 L 是半正定的，具有 $L=L^T$ 的对称性质，所以首先构造一个新的矩阵，该新矩阵定义为 $G=L*L^T=L^2$。根据前面所介绍线性代数的定理，新构造的矩阵 G 与原邻接矩阵 L 具有不同的特征值 λ_G 和 λ_L($\lambda_G=(\lambda_L)^2/m$)和相同的特征向量 $\{U_G\}=\{U_L\}$。

进一步推导 $G=L*L^T=L^2$ 可得：

$$\begin{aligned}G=LL^T&=\begin{pmatrix}l_{11}&\cdots&l_{1m}\\ \vdots&\ddots&\vdots\\ l_{m1}&\cdots&l_{mm}\end{pmatrix}\begin{pmatrix}l_{11}&\cdots&l_{1m}\\ \vdots&\ddots&\vdots\\ l_{m1}&\cdots&l_{mm}\end{pmatrix}^T\\&=(L(x_1),\cdots,L(x_m))(L(x_1),\cdots,L(x_m))^T\\&=\sum_{i=1}^{m}L(x_i)L(x_i)^T\end{aligned} \tag{3}$$

式中，$L(x_i)=(l_{i1},l_{i2},\cdots,l_{mi})^T$。分析 G 的组成，可以把该矩阵看成是 m 个数据 $L(x_i)$ 组成的协方差矩阵。由于 $L(x_i)$ 是由某一个数据 x_i 与其它 m 个数据之间的相似度构成，这样在实际计算中，只需事先把 m 个数据储存起来，对于每一个样本数据 x_i，计算它与这 m 个数据之间的相似度进而得到 $L(x_i)$。然后把矩阵 L 的每一列 $L(x_i)$ 作为迭代算法的输入样本，经过若干次迭代后可以得到矩阵 G 的特征解。

2.3 迭代算法分析

由于传统的特征分解方法在计算过程中事先需要存储特征矩阵，其计算空间复杂度和时间复杂度分别是 $O(m^2)$ 和 $O(m^3)$，其中 m 是数据集样本的数量。当样本数量大的时候需要非常大的内存。一些迭代方法(GHA[13]，APEX[14])先后被提出，用来解决这个问题，这些方法还可以应用到实时数据处理中。但是这些计算方法收敛速度较慢。文献[15]提出了一种增量的协方差无关的方法(Candid covariance-free incremental principal component analysis CCIPCA)，该方法利用统计学上的效能估计概念，相比其它迭代方法，其收敛速度更快，计算复杂度更低。因此我们选用 CCIPCA 作为文中求解过程中的迭代算法。

2.4 实现过程

下面具体给出使用迭代方法 CCIPCA 求解邻接矩阵 L 特征解的详细算法过程。新构造的特征矩阵 G 的特征值 λ_G 和特征向量 $\{U_G\}$ 符合下面的公式：

$$\omega(n)=\lambda_G U_G=GU_G \tag{4}$$

式中，$\omega(n)$ 是在第 n 时刻估计的特征向量，根据式(3)可以把 $\{L(x_1),\cdots,L(x_m)\}$ 视为新的“输入样本”，这样依次将样本 $L(x_i)$ 输入到迭代算法 CCIPCA 中，因此在第 n 时刻所估计的第 i 阶特征解 $\omega_i(n)$ 可以推导如下：

$$\begin{aligned}\omega_i(n)&=GU_G\\&=\frac{1}{n}\sum_{t=1}^{n}L_i(x_t)L_i^T(x_t)\frac{\omega_i(t-1)}{\|\omega_i(t-1)\|}\\&=\frac{1}{n}\sum_{t=1}^{n-1}L_i(x_t)L_i^T(x_t)\frac{\omega_i(t-1)}{\|\omega_i(t-1)\|}+\frac{1}{n}L_i(x_n)L_i^T(x_n)\frac{\omega_i(n-1)}{\|\omega_i(n-1)\|}\\&=\frac{n-1}{n}\omega_i(n-1)+\frac{1}{n}L_i(x_n)L_i^T(x_n)\frac{\omega_i(n-1)}{\|\omega_i(n-1)\|}\end{aligned} \tag{5}$$

式中，$L_i(x_t)$ 是在 t 时刻计算第 i 阶特征解时的输入样本。其它对应的高阶特征解可以使用残留的数据样本向量计算得到(残留的数据样本向量是使用数据样本向量减去其在低阶特征向量上的投影($L_1(x_n)=L(x_n)$)：

$$L_{i+1}(x_n)=L_i(x_n)-L_i(x_n)^T\frac{\omega_i(n)}{\|\omega_i(n)\|}\frac{\omega_i(n)}{\|\omega_i(n)\|} \tag{6}$$

这样通过迭代计算就可以得到各阶特征向量和特征值。在得到特征向量 $\omega(n)$ 后，容易计算出矩阵 G 的特征值 $\lambda_G=\|\omega\|$ 和特征向量 $U_G=\omega/\|\omega\|$。

迭代过程的算法概括如下，其它步骤和谱聚类算法的具体实现过程相同：

1)使用邻接矩阵的前 k 列{L(x_1),…,L(x_k)}初始化需要计算的前 k 阶特征向量

2)for Iteration=1:p

3) for t=1:m

3.1) $L_1(x_t)=L(x_t)$

3.2) For i=1,2,……,min{k,m}

a) If i=m,$\omega_i(t)=L_i(x_t)$

(b) 使用式(5)和式(6)计算前 k 阶主成分

4)输出特征值 λ_G 和特征向量$\{U_G\}$

2.5 算法的空间和时间复杂度分析

在文中给出的算法的计算过程中,不需要使用传统方法对矩阵 L 特征分解。而是在每一步处理样本 $L(x_i)$,其空间复杂度只有 $O(m)$。因为文中算法使用 CCIPCA 算法进行一些迭代,总的时间复杂度为 $O(pkm)$,其中 p,k,m 依次表示迭代次数、特征向量数目和总的样本数。在大规模数据集的情况下,迭代次数和所提取特征向量远小于样本的数目,因此文中算法的时间复杂度也大大降低。

3 实验结果和讨论

为验证文中算法的有效性,我们使用标准的 NJW 算法和提出的方法分别在模拟数据和真实的大规模数据集上完成实验,以验证算法的可行性。

3.1 模拟数据

首先我们随机产生一个 100×20 的矩阵,分别利用特征分解方法和提出的方法计算其特征向量,并利用其前 10 个特征向量重组原矩阵,重组误差公式为:$\| X-W^TWX \|_F$,其中 $\| \cdot \|_F$ 表示 Frobenius 模,X 表示模拟产生的随机矩阵,W 为一个 10×100 特征向量矩阵,其每行为计算所得的特征向量。表 1 给出了其重组误差:从结果可以看出,所提出的方法可以得到与特征分解方法几乎接近的效果。

表 1 使用前 10 个特征向量计算的重组误差

	特征分解方法	提出的方法
重组误差	18.1358	18.1588

接着随机产生 3 类数据,其中心分别为[-1,0],[2,5]和[5,1],每个类有 700 个样本,并且符合均值为 0、方差为 1 的高斯分布。利用 NJW 算法和提出的方法进行计算,得到的分类结果见图 1。从图中可以观察到,经过若干迭代后,所提出的方法可以得到与采用标准 NJW 算法完全相似的聚类结果,这也表明所提出的方法得到的特征向量较好地收敛到标准 NJW 算法产生的特征向量。

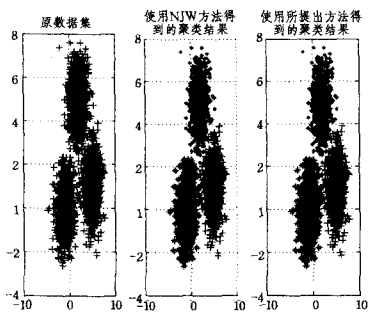

图 1 使用特征分解方法和所提方法得到的聚类结果

3.2 真实数据

我们使用 3 类大规模数据集来验证所提算法的有效性,这些数据集来源于 UCI 机器学习库,数据集的基本特征概述见表 2。由于这些数据集的样本数量都很大,因此所得的邻接矩阵也非常大,在一般的计算机上无法完成矩阵的特征分解。我们使用提出的方法完成谱聚类。为了说明所提算法的有效性,我们也采用 K-means 聚类和基于 Nyström 的谱聚类算法在相同数据集上完成实验,进行对比。

表 2 实验中使用的数据集

数据集	样本数量	特征数	类别数量
penDigits	10,992	16	10
mGamma	19,020	10	2
Connect-4	67,557	42	3

所有数据实验前都进行预处理,确保每个样本特征的均值为 0,方差为 1。实验中邻接矩阵计算过程中带宽的确定使用交叉验证的方法,搜索范围为[0,200],步长为 0.1。聚类的精确性通过计算正确分类数量进行衡量。所有的实验都进行 50 次,最后给出的是实验的平均结果。表 3 给出了使用 K-means 聚类、基于 Nyström 的谱聚类和所提方法在 3 类大规模数据集上分别进行实验所得到的分类率。从结果可以看出,基于 Nyström 的谱聚类和所提方法的聚类准确度都优于 K-means 聚类,而且我们提出的方法在 3 种方法中性能有较好的改善。

表 3 使用不同方法得到的分类结果(%)

数据集	K-means	Nyström	提出的方法
penDigits	49.45	53.18	55.31
mGamma	60.63	68.45	70.16
Connect-4	59.27	62.84	64.56

结束语 本文提出了大规模数据集的情况下谱聚类算法求解的一种有效计算方法。其基于线性代数中对称矩阵的性质,不像传统方法那样特征分解原邻接矩阵,而是用邻接矩阵的每一列作为迭代算法的输入样本,通过迭代计算其特征解。这样可以有效解决传统特征分解方法在大规模数据集的情况下无法计算的问题。文中方法的空间复杂度降低为 $O(m)$,其相应的时间复杂度也降低为 $O(pkm)$。而且当样本数目非常大时,文中方法仍然可以完成对数据的聚类。

参 考 文 献

[1] Xu R, Wunsch D. Survey of Clustering Algorithms[J]. IEEE Transaction on Neural Networks, 2005, 16(3): 645-678

[2] 蔡晓妍,戴冠中,杨黎斌. 谱聚类算法综述[J]. 计算机科学, 2008, 351(17): 14-18

[3] Fiedler M. Algebraic connectivity of graphs[J]. Czech, Math. J, 1973, 23: 298-305

[4] Malik J, Belongie S, Leung T, et al. Contour and texture analysis for image segmentation[J]. International Journal of Computer Vision, 2001, 43(1): 7-27

[5] Shi J, Malik J. Normalized cuts and image segmentation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2000, 22(8): 888-905

[6] Karypis G, Kumar V. A fast and high quality multilevel scheme for partitioning irregular graphs[J]. SIAM Journal on Scientific Computing, 1999, 20: 359-392

[7] Mitra P, Murthy C A, Pal S K. Density-based multiscale data condenation[J]. IEEE Transaction on Pattern analysis and Machine Intelligence, 2002, 24(6): 1-14

[8] Yan D, Huang L, Jordan M I. Fast approximate spectral clustering[C]// 15th ACM Conference on Knowledge Discovery and Data Mining(SIGKDD). Paris, France, 2009

[9] Drineas P, Mahoney M W. On the Nyström method for approximating a Gram matrix for improved kernel-based learning[C]// Proceedings of COLT. 2005: 323-337

(下转第 330 页)

为 n。

在 MATLAB 命令窗口输入如下的程序代码：

```
x=[6 20 40 71 103]
y=[30 61 93 124 156]
hold on
[p2,s2]=polyfit(x,y,2)
p2=-0.0069 2.0024 20.9692
s2=R:[3x3 double]
df:2
normr:6.8271
y2=polyval(p2,x);
[p4,s4]=polyfit(x,y,4)
p4 =0.0000 0.0000 -0.0218 2.7563 14.2381
s4=R:[5x5 double]
df:0
normr:6.0292e-014
y4=polyval(p4,x);
plot(x,y,'ro')
plot(x,y2,'g-')
plot(x,y4,'m--')
xlabel('x')
ylabel('y')
legend('原始数据','2次拟合','4次多项式拟合')[13];
```

图 3 所示为拟合的多项式的曲线图像，'o'为原模型，'—'为 2 次多项式模型，'- - -'为 4 次多项式函数模型。s4 的均方误差为 normr:6.0292e-014，s2 的均方误差为 normr:6.8271，s4 的均方误差小，说明提高多项式的次数可以提高拟合精度，图 3 中 4 次多项式曲线拟合最优[14]。

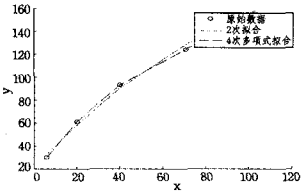

图 3 "论文被引频次"的多项式曲线拟合图

我们再在 MATLAB 中用图形用户界面进行曲线拟合。操作如下：

①在命令窗口中输入要拟合的数据，用 Plot 画图；

②在 Figure 窗口选中 Tools 菜单的 Basic Fitting 选项；

③在 Plot fits 复选框中选择 linear、quadratic、4th degree polynomial 选项，进行线性、2 次和 4 次多项式的拟合，窗口右部分为图像和均方误差信息。

命令行拟合和图形界面拟合的结果如图 4 所示。

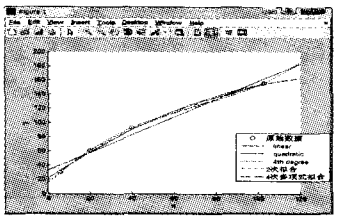

图 4 对原始数据进行命令行拟合和图形界面拟合的结果

结束语 本次研究说明针对被引频次这一网络信息计量指标，布拉德福定律仍然适用，证明了此研究是可行的。由于网络结构的复杂性和易变性，需要将数据进行深加工，找出其特有的共性，构造出适用于网络计量学本身的、通用的模型[15]，相关研究在国内尚不成熟，还有很长的路要走。

参 考 文 献

[1] 王召兵，陈燕. EndNote 在网络信息计量分析中的应用[J]. 情报探索，2011(1)：95-96

[2] 王召兵，王标，孔繁超，等. 网络信息计量在高校图书馆绩效评估中的应用[J]. 山东图书馆学刊，2011(4)：68-70

[3] 张洋，张淑玲. 中美医学院网络信息计量指标的比较分析[J]. 图书情报工作，2011(4)：26-29

[4] 殷之明，冷熠. 网络信息计量实证研究——中国社会科学院研究所网站评价[J]. 科技情报开发与经济，2009(19)：106-108

[5] 万锦望，花平寰，孙秀坤. 期刊论文被引用及其 Web 全文下载的文献计量分析[J]. 现代图书情报技术，2005(4)：58-62

[6] 张洋，弋云. 应用网络信息计量指标测定我国图书情报学核心网站的实证研究[J]. 图书情报知识，2011(1)：82-87

[7] 张洋. 期刊 Web 下载总频次的布拉德福分布研究[J]. 图书情报知识，2006(6)：38-42

[8] 沙勇忠，阎劲松. 网络著者分布规律实证研究：以 Python. cn 论坛为例[J]. 图书·情报·知识，2006，114(6)：17-21

[9] 崔旭，邵力军. 揭开布鲁克斯公式 K、N 关系之奥秘[J]. 情报杂志，2003(09)：42-43

[10] 赵隽. 基于布拉德福定律区域法的学术论文分布研究[J]. 现代情报，2007(05)：26-28

[11] 申红莲. Matlab 中曲线拟合的方法[J]. 福建电脑，2010(7)

[12] 马卫东，李幼平，周明天. 万维网无尺度特征与主动服务网格[J]. 计算机科学，2005(9)：31-34

[13] 赵丹群. 试论引文分析方法的网络化发展与应用[J]. 图书情报工作，2009(8)：40-43

[14] 马晓佳. 网络引文分析与传统引文分析的比较[D]. 南京：南京大学，2011

[15] Wallace D P. The relationship between journal productivity & obsolescence[J]. Journal of the American society forinformation science，1986，37：135-136

[16] Goffman W，Morris T G. Bradford's law and library acquisitions [J]. Nature，1970，226：922-923

(上接第 314 页)

[10] Fowlkes C，Belongie S，Chung F，et al. Spectral grouping using the Nyström method[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence，2004，26(2)：214-225

[11] 史卫亚，郭跃飞，薛向阳. 一种解决大规模数据集问题的核主成分分析算法[J]. 软件学报，2009，20(8)：2153-2159

[12] Ng A Y，Jordan M I，Weiss Y. On spectral clustering：Analysis and an algorithm[C]// Advances in Neural Information Processing Systems. Cambridge，MA，MIT Press，2002，14：849-856

[13] Sander T D. Optimal unsupervised learning in a single-layer linear feedforward neural network[J]. Neural Network，1989，12：459-473

[14] Kung S Y，Diamantaras K I. A neural network learning algorithm for adaptive principal component extraction(apex)[C]// Proc. of IEEE Conf. on Acoustics，Speech，and Signal. Albuquerque，1990，2：861-864

[15] Weng J，Zhang Y，Huang W S. Candid covariance-free incremental principal component analysis[J]. IEEE Trans Pattern Analysis. Machine. Intelligence，2003，25(8)：1034-1040