一种基于特征间隙的检测簇数的谱聚类算法

胡海峰[1,2],刘萍萍[1]
(1. 南京邮电大学 通信与信息工程学院 宽带无线通信与传感网
技术教育部重点实验室,江苏 南京 210003;
2. 东南大学 移动通信国家重点实验室,江苏 南京 210096)

摘 要:数据挖掘中如何根据数据之间的相似度确定簇(Cluster)数一直是聚类算法中需要解决的难题。文中在经典谱聚(Spectral Clustering)算法的基础上提出了一种基于特征间隙检测簇数的谱聚类算法(Spectral Clustering with Identifying Clustering Number based on Eigengap,SC-ICNE)。通过构建规范的拉普拉斯矩阵,顺序求解其特征值和相应特征向量,并得到矩阵相邻特征值的间隙,通过判断特征间隙的位置来确定簇数 k。最后,通过对前 k 个特征向量的 k-means 算法实现数据集的聚类。文中通过仿真分析了高斯相似度函数对 SC-ICNE 聚类性能的影响,在非凸球形数据集和 UCI 数据集上进行了性能仿真,并和 k-means 聚类算法进行了对比,在检测簇数和聚类准确性方面,验证了 SC-ICNE 算法的有效性。

关键词:谱聚类;簇数;特征间隙;高斯相似度
中图分类号:TP301.6 **文献标识码**:A **文章编号**:1673-629X(2015)09-0037-06
doi:10.3969/j.issn.1673-629X.2015.09.008

A Spectral Clustering Algorithm with Identifying Clustering Number Based on Eigengap

HU Hai-feng[1,2],LIU Ping-ping[1]
(1. Key Lab of Broadband Wireless Communication and Sensor Network Technology of Ministry of Education,
College of Communication and Information Engineering,Nanjing University of Posts and Telecommunications,
Nanjing 210003,China;
2. National Mobile Communications Research Laboratory,Southeast University,Nanjing 210096,China)

Abstract:Choosing the number k of clusters based on the degree of correlation is a general problem for all clustering algorithms. Based on the classical spectral clustering algorithm,propose a Spectral Clustering with Identifying Clustering Number based on Eigengap (SC-ICNE) algorithm. The SC-ICNE algorithm computes eigenvalues and corresponding eigenvectors of normalized graph Laplacians sequentially. Furthermore,the number of cluster k can be identified via the eigengap between the adjacent eigenvalues. Finally,the data can be clustered using the first k eigenvectors with the k-means algorithm. In the simulation,the effect of the Gaussian similarity function on the cluster performance of SC-ICNE is discussed,and compare the cluster performance of SC-ICNE with the k-means algorithm in non-spherical convex data set and the UCI data set. Simulation results show that the SC-ICNE algorithm achieves high performance in terms of clustering accuracy and identifying the cluster number.

Key words:spectral clustering;clustering number;eigengap;Gaussian similarity

0 引 言

聚类[1-2]是一种常用的数据分析方法,它将数据集分成不同的簇,使簇内数据之间尽可能相似,而不同簇的数据之间尽可能不同。聚类研究是近年来机器学习领域的研究热点[3],它被广泛应用于统计、计算机科学、生物和社会科学等领域中[4]。聚类算法多种多样,一般分为以下几类:层次化聚类算法[5]、划分式聚类算法、基于密度和网格的聚类算法[6-7]等。如经典的 k-

收稿日期:2014-11-02 **修回日期**:2015-02-06 **网络出版时间**:2015-08-26
基金项目:国家自然科学基金资助项目(61071092);东南大学移动通信国家重点实验室开放研究基金资助课题(2011D04);中国博士后科学基金项目(2012M511309);江苏省博士后科研资助计划(1101125C);华为公司创新研究计划(YB2014010003)
作者简介:胡海峰(1973-),男,博士,副教授,研究方向为无线通信与网络信号处理、大数据背景下基于数据挖掘的多标记智能搜索等;刘萍萍(1989-),女,硕士生,研究方向为数据挖掘的谱聚类算法研究。
网络出版地址:http://www.cnki.net/kcms/detail/61.1450.TP.20150826.1558.048.html

means 聚类算法[8]就属于划分式聚类算法。以上经典的聚类算法都存在着各种缺点[9],如大多数聚类算法需要预先给出参数,层次化聚类算法聚合或分裂过程的有效终止条件不易明确,k -means 等划分式聚类算法不同的初始条件会导致不同的聚类结果等。

近年来,谱聚类算法[10-11]是聚类算法中非常热门的一个研究方向。谱聚类算法是一种基于两点间相似关系的聚类方法[12],基本思想是对数据集构造邻接矩阵,转换成拉普拉斯矩阵,通过求解拉普拉斯矩阵的特征值[13]来对邻接矩阵进行数据降维[14],再用 k -means 或其他的传统聚类方法对特征向量聚类[15]。相比于传统聚类算法(如 k -means 聚类和 EM 算法[16]),谱聚类算法不关心数据集的形状,对非凸球形数据集也能有效聚类,而且实现简单,更适用于大型数据集。谱聚类算法的难点在于邻接矩阵的构造和簇数的检测[17]。

文献[18]提到,当数据簇数为 k 时,该数据集对应的拉普拉斯矩阵特征求解后,前 k 个特征值数值接近于 1,而第 $k+1$ 个特征值相对于第 k 个特征值将迅速减小。这为簇数的探测提供了理论基础,但是在数据集合比较大的情况下[19],如何快速有效地探测出特征间隔,并在此基础上如何快速进行谱聚类算法,还存在很大挑战。

针对上述谱聚类算法存在的问题,文中提出一种基于特征间隙的检测簇数的谱聚类算法 SC-ICNE(Spectral Clustering with Identifying Clustering Number based on Eigengap),通过构建基于邻接矩阵的规范的拉普拉斯矩阵,对拉普拉斯矩阵特征求解,并按照特征值从大到小的顺序依次进行,获得矩阵相邻特征值的间隙,通过判断特征间隙的位置来确定簇数 k,所以只需求解出前 $k+2$ 个特征值和对应的特征向量就能确定 k 值,而不是求出矩阵的全部特征值,因此减少了特征求解的计算量。最后,通过前 k 个特征向量的 k -means 算法实现数据集的聚类。

1 谱聚类算法模型

谱聚类是建立在图论中的谱图(spectrum diagram)理论基础上,其本质就是将聚类问题转化为图的最优划分问题[2]。将一组数据集 $\bar{\boldsymbol{x}}=(\boldsymbol{x}_1,\cdots,\boldsymbol{x}_i,\cdots,\boldsymbol{x}_n,\boldsymbol{x}_i\in\mathbb{R}^d)$ 看作一个图 $G(\boldsymbol{V},\boldsymbol{E})$,$n$ 为数据集中的数据项的数量,其中每一个样本数据 $\boldsymbol{x}_i$ 对应图中的点 $\boldsymbol{v}_i\in\boldsymbol{V}$,$e_{ij}\in\boldsymbol{E}$ 为点 $\boldsymbol{v}_i$ 和 $\boldsymbol{v}_j$ 间的连接边,w_{ij} 为 e_{ij} 上的权重值即相似度,图 $G(\boldsymbol{V},\boldsymbol{E})$ 的邻接矩阵 $\boldsymbol{W}$ 定义为

$$W_{ij}=\begin{cases}w_{ij}, & \text{如果 } \boldsymbol{v}_i \text{ 和 } \boldsymbol{v}_j \text{ 之间存在边 } e_{ij}\\ 0, & \text{其他}\end{cases} \tag{1}$$

两点间的相似度可以有很多种表示方法,最常用的就是高斯相似度,表示为

$$\begin{cases}w_{ij}=\exp(-|\boldsymbol{x}_i-\boldsymbol{x}_j|^2/2\sigma^2), & i\neq j\\ w_{ij}=0, & i=j\end{cases} \tag{2}$$

其中,$\boldsymbol{x}_i$ 和 $\boldsymbol{x}_j$ 代表数据集中的任意不同数据项;w_{ij} 代表 $\boldsymbol{x}_i$ 和 $\boldsymbol{x}_j$ 两点间的相似度;σ 是一个尺度参数,对谱聚类算法性能影响很大。

谱聚类问题就可以看成是图的最优划分问题[5],即按照一定的划分准则将图划分为最优的几个子图,从而使子图内部相似度最大,子图间的相似度最小。谱聚类算法首先需要构建对角矩阵 $\boldsymbol{D}$ 和非规范的拉普拉斯矩阵。

$$d_{ii}=\sum_k w_{ik} \tag{3}$$

$$\boldsymbol{L}=\boldsymbol{D}-\boldsymbol{W} \tag{4}$$

在此基础上,构建规范的拉普拉斯矩阵

$$\boldsymbol{L}_{\text{sym}}=\boldsymbol{D}^{-\frac{1}{2}}\boldsymbol{L}\boldsymbol{D}^{-\frac{1}{2}}=\boldsymbol{I}-\boldsymbol{D}^{-\frac{1}{2}}\boldsymbol{W}\boldsymbol{D}^{-\frac{1}{2}} \tag{5}$$

为了方便计算,设

$$\boldsymbol{H}_{\text{sym}}=\boldsymbol{D}^{-\frac{1}{2}}\boldsymbol{W}\boldsymbol{D}^{-\frac{1}{2}} \tag{6}$$

为规范的拉普拉斯矩阵另一种表达形式,显然有下面性质:

$$\lambda_i(\boldsymbol{L}_{\text{sym}})=1-\lambda_i(\boldsymbol{H}_{\text{sym}}) \tag{7}$$

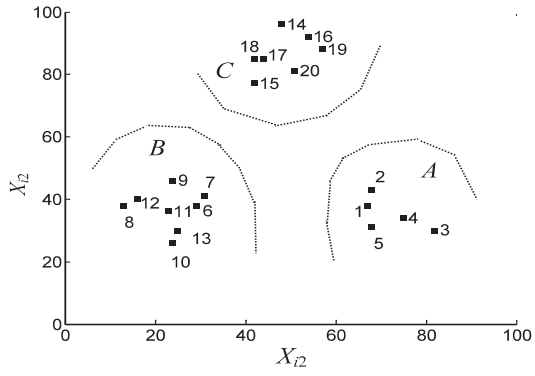
其中,$\lambda_i(\boldsymbol{L}_{\text{sym}})$ 和 $\lambda_i(\boldsymbol{H}_{\text{sym}})$ 分别是 $\boldsymbol{L}_{\text{sym}}$ 和 $\boldsymbol{H}_{\text{sym}}$ 的第 i 个特征值。且 $\boldsymbol{H}_{\text{sym}}$ 的特征值满足[10]:

$$1=\lambda_0(\boldsymbol{H}_{\text{sym}})\geqslant\lambda_1(\boldsymbol{H}_{\text{sym}})\geqslant\cdots\geqslant\lambda_{n-1}(\boldsymbol{H}_{\text{sym}}) \tag{8}$$

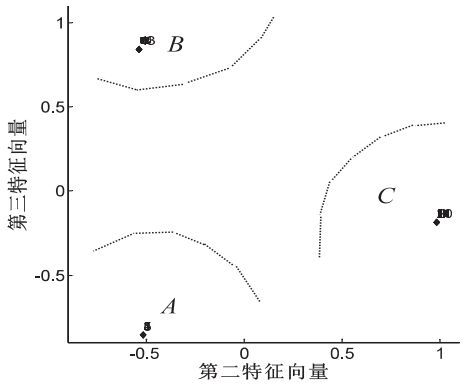
并且,对应的特征向量分别为 $f_0(\boldsymbol{H}_{\text{sym}})$,$f_1(\boldsymbol{H}_{\text{sym}})$,$\cdots$,$f_{n-1}(\boldsymbol{H}_{\text{sym}})$。

谱聚类算法对 $\boldsymbol{H}_{\text{sym}}$ 进行特征求解。如果数据集的簇数是 k,就对 $f_1(\boldsymbol{H}_{\text{sym}})$ 到 $f_{k-1}(\boldsymbol{H}_{\text{sym}})$ 特征向量组成 $n\times(k-1)$ 矩阵 $\boldsymbol{U}$(注意:$f_0(\boldsymbol{H}_{\text{sym}})$ 不包含聚类信息),将这 $\boldsymbol{U}$ 的行向量 $\boldsymbol{u}_i(i\in n)$ 分别作为数据点,从而得到原数据集 $\bar{\boldsymbol{x}}=(\boldsymbol{x}_1,\cdots,\boldsymbol{x}_i,\cdots,\boldsymbol{x}_n,\boldsymbol{x}_i\in\mathbb{R}^d)$ 到特征数据集 $\bar{\boldsymbol{u}}=(\boldsymbol{u}_1,\cdots,\boldsymbol{u}_i,\cdots,\boldsymbol{u}_n,\boldsymbol{u}_i\in\mathbb{R}^{k-1})$ 的转换,并在 $\bar{\boldsymbol{u}}$ 上进行 k -means 聚类运算。比如一组由 15 个二维数据点构成的点集如图 1(a)所示,对其构建规范的拉普拉斯矩阵并进行特征求解,因为已知数据集的簇数为 $k=3$,故取前 2 个特征向量组成的 $n\times 2$ 矩阵 $\boldsymbol{U}$,特征数据集如图 1(b)所示。

可以清楚的看出,两个特征向量构成的特征数据集相对于原数据集,更加明显地聚成三个簇:Cluster A:{1,2,3,4,5},Cluster B:{6,7,8,9,10,11,12},Cluster C:{13,14,15,16,17,18,19,20}。因此,对变换后的特征数据集使用经典的聚类算法,聚类会更加准确,收敛的速度也会加快。



(a)原数据集



(b)特征数据集

图 1　数据集谱聚前后示意图

2　基于特征间隙的检测簇数的谱聚类算法

在谱聚类算法中,簇数 k 是作为输入参数提前给定的。但实际上,并不能根据数据本身就提前获得其簇数 k。因此谱聚类算法中 k 值的自动确定,即自动谱聚算法,是谱聚算法的一个关键问题。

理想情况下,聚类过程的簇定义为全连接子集,簇间没有连接,规范的拉普拉斯矩阵 $\boldsymbol{H}_{\text{sym}}$ 的 $\lambda_0(\boldsymbol{H}_{\text{sym}})=1$ 对应的特征向量是包含聚类信息的指示向量(indicator vectors)[10],并且 $\lambda_0(\boldsymbol{H}_{\text{sym}})=1$ 对应的特征向量个数和簇数 k 相同。根据摄动理论,在实际非全连接聚类情况下,簇数为 k 时,$\boldsymbol{H}_{\text{sym}}$ 特征值按从大到小排列,则前 k 个特征值接近于 1,第 $k+1$ 个特征值会明显变小,即可以利用相邻特征值之间的特征间隙(eigengap)来检测簇的个数 k。文中特征间隙定义如下:

$$\eta(i)=\lambda_{i+1}(\boldsymbol{H}_{\text{sym}})/\lambda_i(\boldsymbol{H}_{\text{sym}}),\ i\in 0,1,\cdots,n-2, \tag{9}$$

并且,根据式(8),当数据集的簇数为 k 时,满足下列条件:

$$\begin{cases}\eta(i)\leqslant 1\ \cap\ \eta(i)\to 1 \\ \qquad \text{if}(\ i=0,1,\cdots,k-2)\ \cap\ (k>1) \\ \eta(k-1)\ll 1 \\ \eta(i)\leqslant 1\ \cap\ \eta(i)\to 1\ \text{if}\ i=k\leqslant n-2\end{cases} \tag{10}$$

从上式可知,如果能依次求出 $\eta(i)$ ($i\in 0,1,\cdots,n-2$),当 $\eta(i)$ 出现第一个极小值点,$i+1$ 就等于簇数 k。而且,当算法求出数据集的簇数 k,矩阵的特征求解便可结束。因为簇数 k 远小于数据点的个数 n,从而只需要依次求出 $\lambda_i(\boldsymbol{H}_{\text{sym}})(i=0,1,\cdots,k+1\ll n)$,而不需要计算出 $\boldsymbol{H}_{\text{sym}}$ 所有的特征值及其对应特征向量,这样可以大大减少特征求解的计算量。

$\boldsymbol{H}_{\text{sym}}$ 特征值顺序求解法,不是文中论述的重点,具体可参看文献[20],基本思想简述如下:$\boldsymbol{H}_{\text{sym}}$ 经过 Householder 变换转换为三对角矩阵 $\boldsymbol{H}_{\text{sym}}$,根据 Gershorin 定理,可以求出 $\boldsymbol{H}_{\text{sym}}$ 的特征值 $\lambda_i(\boldsymbol{H}_{\text{sym}})$ 所分布的区间;然后,利用 Sturm sequence 性质可求出 $a(\tau)$,即小于 τ 的 $\lambda_i(\boldsymbol{H}_{\text{sym}})(i=0,1,\cdots,n-1)$ 的个数;最后,利用二分法逐次逼近,依次得出 $\lambda_i(\boldsymbol{H}_{\text{sym}})$ 的近似值,并根据 Inverse Power 算法求出对应的特征向量。

文中提出的算法 SC-ICNE 的步骤如下:

输入:数据集 $(x_1,\cdots,x_i,\cdots,x_n,x_i\in\mathbb{R}^d)$

Step1:构建对应的图 $G(\boldsymbol{V},\boldsymbol{E})$ 高斯相似度邻接矩阵 $\boldsymbol{W}$;

Step2:根据公式(6),构建规范的拉普拉斯矩阵 $\boldsymbol{H}_{\text{sym}}=\boldsymbol{D}^{-\frac{1}{2}}\boldsymbol{W}\boldsymbol{D}^{-\frac{1}{2}}$;

Step3:经过 Householder 变换为三对角矩阵 $\hat{\boldsymbol{H}}_{\text{sym}}$;根据 Gershorin 定理,可以求出 $\boldsymbol{H}_{\text{sym}}$ 的特征值 $\lambda_i(\boldsymbol{H}_{\text{sym}})$ 所分布的区间;

Step4:利用 Sturm sequence 性质可求出 $a(\tau)$,即小于 τ 的 $\lambda_i(\boldsymbol{H}_{\text{sym}})(i=0,1,\cdots,n-1)$ 的个数;

Step5:设 $i=0$;

Step6:利用二分法逐次逼近,依次得出 $\lambda_{i+1}(\boldsymbol{H}_{\text{sym}})$ 和 $\lambda_{i+2}(\boldsymbol{H}_{\text{sym}})$ 的近似值,并根据 Inverse Power 算法求出 $f_{i+1}(\boldsymbol{H}_{\text{sym}})$ 和 $f_{i+2}(\boldsymbol{H}_{\text{sym}})$;

Step7:计算 $\eta(i)=\lambda_{i+1}(\boldsymbol{H}_{\text{sym}})/\lambda_i(\boldsymbol{H}_{\text{sym}})$ 和 $\eta(i+1)=\lambda_{i+2}(\boldsymbol{H}_{\text{sym}})/\lambda_{i+1}(\boldsymbol{H}_{\text{sym}})$;

Step8:判断是否满足式(10),如果不满足,$i=i+1$,转到 Step6,否则,转到 Step9;

Step9:分簇数 $k=i+1$,$f_1(\boldsymbol{H}_{\text{sym}})$ 到 $f_{k-1}(\boldsymbol{H}_{\text{sym}})$ 特征向量组成 $n\times(k-1)$ 矩阵 $\boldsymbol{U}$;

Step10:将这 $\boldsymbol{U}$ 的行向量 $\boldsymbol{u}_i(i\in n)$ 作为特征数据集 $\bar{\boldsymbol{u}}=(\boldsymbol{u}_1,\cdots,\boldsymbol{u}_i,\cdots,\boldsymbol{u}_n,\boldsymbol{u}_i\in\mathbb{R}^{k-1})$;

Step11:并在 $\bar{\boldsymbol{u}}$ 上进行 k-means 聚类运算,得到簇 $C_1,C_2,\cdots,C_k$;

Step12:得到原始数据集所分得的簇:$A_1,A_2,\cdots,A_k$,其中 $A_i=\{j\mid u_j\in C_i\}$。

SC-ICNE 算法最大的计算开销在于 Householder 变换,计算复杂度为 $O(n^3)$,但是考虑到 SC-ICNE 只需要按顺序计算出 $k+1\ll n$ 个特征值和特征向量,而

且对变换后的特征数据集 $\bar{\boldsymbol{u}}$ 使用聚类算法,聚类会更加准确,收敛的速度也会加快。

对如图 2 所示的 125 个数据的数据集使用 SC-ICNE 算法进行簇数估计,从图上可明显看出簇数 k 应该等于 6。为了验证检测簇数的有效性,使用表 1 对 SC-ICNE 算法过程进行记录。

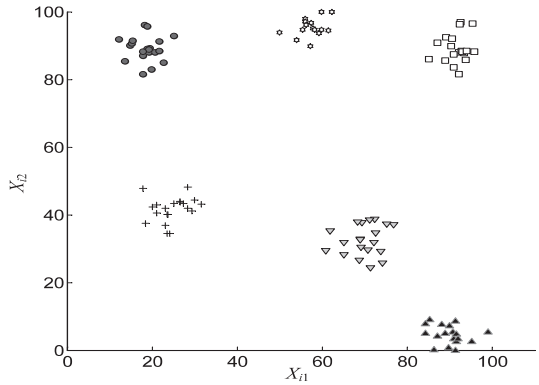


图 2 验证性的数据集的分布情况

表 1 SC-ICNE 运算过程

i	$\lambda_i(\boldsymbol{H}_{\text{sym}})$	$\eta(i-1)$	式(10)条件是否满足
0	1.000 0		
1	0.999 1	0.999 1	否,算法继续
2	0.993 9	0.994 8	否,算法继续
3	0.983 4	0.989 4	否,算法继续
4	0.938 1	0.954 0	否,算法继续
5	0.924 2	0.985 1	否,算法继续
6	0.07	0.075 2	否,算法继续
7	0.045 5	0.653 0	是,算法结束,$k=6$

SC-ICNE 算法在计算到 $\lambda_7(\boldsymbol{H}_{\text{sym}})$ 时,满足了式(10)的条件,即当 $i=0,1,\cdots,4$ 时,$\eta(i)$ 相差不大,都略小于 1;当 $i=5$ 时,$\eta(i)=0.075\ll 1$;当 $i=6$ 时,$\eta(i)=0.653$;说明当 $i=5$ 时,特征间隙 $\eta(i)$ 出现了第一个极小值点,根据 SC-ICNE 算法,簇数 k 等于 6,从而验证了本算法在簇数探测方面的有效性,并且可以看出,SC-ICNE 算法只需要按照顺序计算前 7 个特征值及对应的特征向量,而不是求出所有的 125 个特征值和特征向量,大大减少了特征计算量。

3 仿真及分析

为了进一步验证 SC-ICNE 算法的有效性,分析其聚类性能。首先,使用高斯相似度函数来模拟不同数据环境,并讨论不同参数对 SC-ICNE 算法性能的影响;然后,使用典型的同心圆环数据集和标准的 UCI 数据库中的 4 组数据集,分别对比了 SC-ICNE 算法和 k-means 算法的聚类性能。文中所有的仿真都在随机的数据环境和标准的数据集基础上,使用 Matlab 工具进行了性能对比和分析。

(1)SC-ICNE 算法在不同高斯相似度函数下的检测簇数性能分析。

聚类仿真中,一般用高斯相似度函数形成邻接矩阵,来模拟不同的相关度的数据集:

$$\begin{cases} w_{ij}=\exp(-|\boldsymbol{x}_i-\boldsymbol{x}_j|^2/2\sigma^2), & i\neq j \\ w_{ij}=0, & i=j \end{cases} \tag{11}$$

其中,$\boldsymbol{x}_i$ 和 $\boldsymbol{x}_j$ 代表数据集中的任意不同数据项;w_{ij} 代表 $\boldsymbol{x}_i$ 和 $\boldsymbol{x}_j$ 两点间的相似度;σ 是一个可调节的尺度参数,对谱聚类算法性能影响很大。

下面讨论不同 σ 下的高斯相似度对 SC-ICNE 算法的影响。

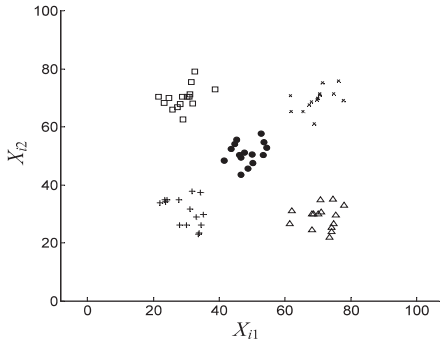
由相似度定义可知,当 σ 较小时,w_{ij} 随 $|\boldsymbol{x}_i-\boldsymbol{x}_j|$ 变化陡峭,对数据点之间的距离比较敏感;相反,当 σ 较大时,w_{ij} 随 $|\boldsymbol{x}_i-\boldsymbol{x}_j|$ 变化缓慢,分簇效果不明显。在仿真过程中,对于给定的数据集,可以通过设置不同的 σ 值来得到所期望的簇数。

通过控制各簇之间的重叠范围大小,可以生成不同聚类效果的数据集:当各簇之间重叠范围小时,数据集聚类效果明显;当各簇之间重叠范围大,则聚类效果不明显。据此,在 [0,100] × [0,100] 的区域内,生成两组 $k=5$ 但聚类效果不同的二维数据集。接着对这两组已知簇数 k 的数据集在不同 σ 取值时,对 SC-ICNE 算法探测簇数性能进行了仿真。如图 3 所示,其中左列为待测试的数据点集,右列是不同 σ 时,特征间隙 $\eta(i)$ 随 i 的变化情况,图中不同折线代表 σ 的不同取值,由上到下 σ 依次增大。

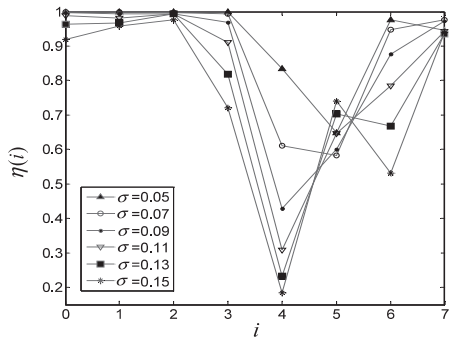
由图 3(a)可知,待测数据集的聚类效果比较明显。从图 3(b)可以看出,当 σ 小于 0.09 时,按照式(10)的定义,$\eta(i)$ 取得第一个极小值时,i 大于 4,此时簇数 $k>5$,有新的簇分裂出现;而当 σ 大于等于 0.09 时,$\eta(i)$ 在 $i=4$ 处取得第一个极小值,簇数 $k=5$。由图 3(c)可知,待测数据点的分布比较均匀,聚类效果不明显。从图 3(d)可以看出,当 σ 大于 0.09 时,根据式(10),$\eta(i)$ 在 $i=0$ 处取得第一个极小值点,即簇数 $k=1$,说明此时整个数据点都合为一个簇;当 σ 小于等于 0.09 时,$\eta(i)$ 在 $i=4$ 处取得第一个极小值点,簇数 $k=5$,因此,特征间隙 $\eta(i)$ 对 σ 值的变化比较敏感。

(2)SC-ICNE 算法和 k-means 在同心圆环数据点集上的仿真性能对比。

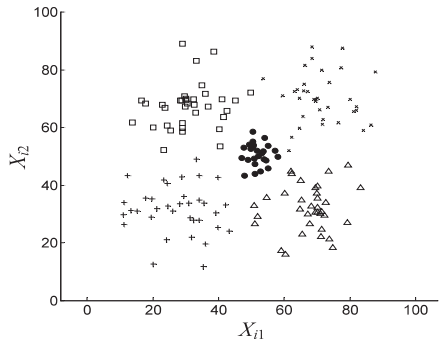
SC-ICNE 算法相对于 k-means 算法的优势之一就是对非凸球形点集同样能准确聚类。选取了一组同心圆环数据集,数据集共有 3 个簇,如图 4(a)所示。分别用 k-means 算法和 SC-ICNE 算法对其聚类,不同簇用颜色深浅不同的点区分。



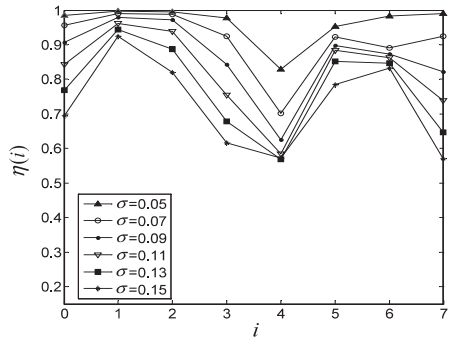
(a)聚类效果明显时，数据点集分布情况



(b)聚类效果明显时，特征间隙 $\eta(i)$ 随 i 的变化（不同 σ）

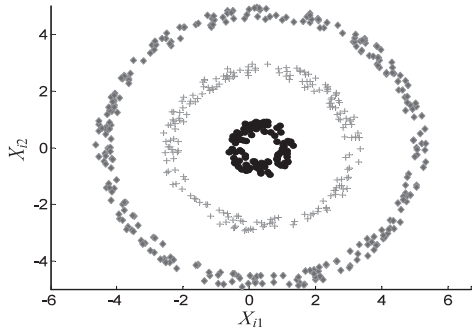


(c)聚类效果不明显时，数据点集分布情况

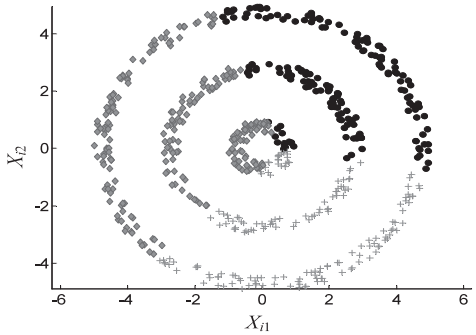


(d)聚类效果不明显时，特征间隙 $\eta(i)$ 随 i 的变化（不同 σ）

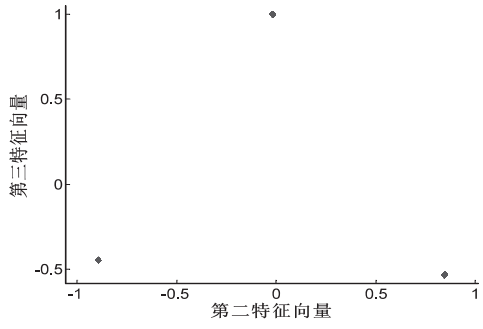
图 3　数据点聚类效果不明显时，σ 值对 SC-ICNE 算法的影响



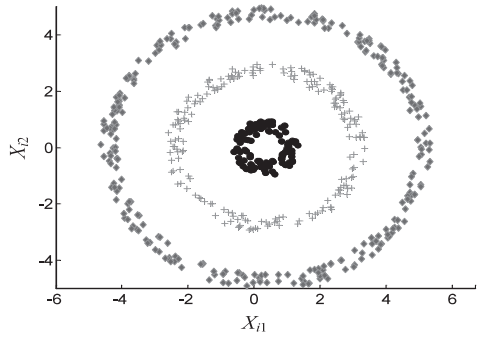
(a)同心圆环原始数据点集



(b) k-means 聚类结果



(c)SC-ICNE 特征数据点集



(d)SC-ICNE 算法聚类结果

图 4　同心圆环 k-means 和 SC-ICNE 算法聚类效果比较

对图 4(a)所示的数据集用 k-means 算法聚类，得到的分簇效果如图 4(b)所示。很明显，得到的分簇从整体上看是非常不合理的，实际上，k-means 算法不适用于这种类型的数据集。而 SC-ICNE 算法对原数据集构建规范拉普拉斯矩阵后进行特征分解，按顺序计算出前 5 个特征值对应的特征向量，算法根据式(10)判断出簇数 $k=3$。选取包含聚类信息的第二特征向量和第三特征向量。将这两个特征向量组成矩阵的行向量分别作横、纵坐标，形成新的特征数据集，也就是将原数据集映射为图 4(c)中的特征数据点集，对特征数据集进行聚类可获得三个同心圆的分簇效果，如图 4(d)所示。

(3)SC-ICNE算法和k-means在UCI数据集的仿真性能对比。

UCI数据库是美国加州大学欧文分校所提出的,专门用于机器学习测试的一个数据库。在其中挑选了四组数据集:Seeds,Iris,Wine,Breast Cancer Wisconsin。并对数据进行了归一化处理。以这四组数据集为对象,比较了SC-ICNE算法和k-means算法的聚类准确度。计算公式为:

$$r = 1 - \frac{n_e}{n} \tag{12}$$

其中,n为数据集中的数据项的数量;n_e为聚类结果中分簇错误的数据项数量。

计算聚类准确度时,进行100次仿真再取其平均值,两种算法的比较结果如表2所示。

表2 UCI数据集仿真结果

数据集	样本点数	样本数据维度	簇数k	k-means聚类准确度/%	SC-ICNE聚类准确度/%	簇数判断
Seeds	210	7	3	85.95	89.29	正确
Iris	150	4	3	81.80	96.00	正确
Wine	178	13	3	78.09	94.26	正确
Breast Cancer Wisconsin	683	9	2	80.38	90.63	正确

从表2中可以看出,对于这四组UCI数据集,SC-ICNE算法可以准确判断出数据集的簇数,并在聚类准确度方面相对于k-means算法有了显著的提高。

综上所述,SC-ICNE算法以适当增加计算复杂度的情况下,实现了对数据集簇数的准确判断,扩展了聚类算法的适用数据范围,并在聚类准确度方面有了显著提高。

4 结束语

文中提出了一种基于特征间隙检测簇数的谱聚类算法-SC-ICNE。通过构建规范的拉普拉斯矩阵,顺序求解其特征值和相应特征向量,并得到矩阵相邻特征值的间隙,通过判断特征间隙的位置来确定簇数k。最后,通过对前k个特征向量的k-means算法实现数据集的聚类。SC-ICNE算法判断出数据集的簇数后,停止对剩余特征值及相应特征向量的特征求解,从而减少了特征求解的计算量。仿真结果表明,SC-ICNE算法能准确检测出不同相关数据集的簇数,相对于k-means算法,可适用于多种类型的数据集,并提高了聚类准确度。

参考文献:

[1] Jain A,Murty M,Flynn P. Data clustering:a review[J]. ACM Computing Surveys,1999,31(3):264-323.

[2] Xu Rui,Donald Wunsch II. Survey of clustering algorithms[J]. IEEE Trans on Neural Networks,2005,16(3):645-678.

[3] 贺 玲,吴玲达,蔡益朝. 数据挖掘中的聚类算法综述[J]. 计算机应用研究,2007,24(1):10-13.

[4] Schultz T,Kindlmann G L. Open-box spectral clustering:applications to medical image analysis[J]. IEEE Trans on Visualization and Computer Graphics,2013,19(12):2100-2108.

[5] Maqbool O,Babri H A. Hierarchical clustering for software architecture recovery[J]. IEEE Trans on Software Engineering,2007,33(11):759-780.

[6] Amini A,Wah T Y,Saybani M R. A study of density-grid based clustering algorithms on data streams[C]//Proc of 8th international conference on fuzzy systems and knowledge discovery. Shanghai:IEEE,2011.

[7] 孙吉贵,刘 杰,赵连宇. 聚类算法研究[J]. 软件学报,2008,19(1):48-61.

[8] 周爱武,于亚飞. K-Means聚类算法的研究[J]. 计算机技术与发展,2011,21(2):62-65.

[9] Gelbard R,Goldman O,Spiegler I. Investigating diversity of clustering methods:an empirical comparison[J]. Data & Knowledge Engineering,2007,63(1):155-166.

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

[11] Fu Chuanyi,Xing Jieqing. Spectral clustering and its research progress[C]//Proc of 7th international conference on computational intelligence and security. [s.l.]:[s.n.],2011.

[12] Bach F R,Jordan M I. Learning spectral clustering[C]//Proc of 17th annual conference on neural information processing systems. Canada:[s.n.],2003.

[13] Chung F. Spectral graph theory[M]. American:American Mathematical Society,1997.

[14] Fiedler M. A property of eigenvectors of non-negative symmetric matrices and its application to graph theory[J]. Czech Math J,1975,25(4):619-633.

[15] 田 铮,李小斌,句彦伟. 谱聚类的扰动分析[J]. 中国科学:E辑,2007,37(4):527-543.

[16] Mitchell T. Machine learning[M]. New York:McGraw-Hill,1997.

[17] 蔡晓妍,戴冠中,杨黎斌. 谱聚类算法综述[J]. 计算机科学,2008,35(7):14-18.

[18] Liu Ning. Spectral clustering for graphs and Markov chains[D]. Raleigh:North Carolina State University,2010.

[19] Qian Pengjiang,Chung Fu. Fast graph-based relaxed clustering for large data sets using minimal enclosing ball[J]. IEEE Trans on Systems,Man,and Cybernetics,Part B:Cybernetics,2012,42(3):672-687.

[20] Golub G H,van Loan C F. Matrix computations[M]. 3rd ed. [s.l.]:John Hopkins,1996.

一种基于特征间隙的检测簇数的谱聚类算法

作者： 胡海峰， 刘萍萍， HU Hai-feng， LIU Ping-ping

作者单位： 胡海峰,HU Hai-feng(南京邮电大学 通信与信息工程学院 宽带无线通信与传感网技术教育部重点实验室，江苏 南京 210003；东南大学 移动通信国家重点实验室，江苏 南京 210096)， 刘萍萍,LIU Ping-ping(南京邮电大学 通信与信息工程学院 宽带无线通信与传感网技术教育部重点实验室,江苏 南京,210003)

刊名： 计算机技术与发展 ISTIC

英文刊名： Computer Technology and Development

年，卷(期)： 2015(9)