2018 年 7 月 第 7 期
Vol. 39 No. 7 2018

# 一种启发式确定聚类数方法

卢建云[1,3],朱庆生[1,2],吴全旺[1]

[1](重庆大学 计算机学院,重庆 400044)
[2](重庆大学 软件理论与技术重庆市重点实验室,重庆 400044)
[3](重庆电子工程职业学院 软件学院,重庆 401331)
E-mail: qszhu@cqu.edu.cn

**摘 要:** 聚类分析是数据挖掘领域中最重要的任务之一,目前许多聚类算法已经被成功应用到图像聚类、文本聚类、信息检索、社交网络等领域.但面对结构复杂,分布不均衡的数据集时,确定数据集的最佳聚类数目显得尤为困难.因此,本文针对结构复杂、分布不均衡的数据集提出了一种启发式最佳聚类数确定的方法.首先,构建随机游走模型对数据集中的点进行重要性排序,通过 k-最近邻距离图谱确定重要数据点的个数,由此排除噪声点和不重要的点对类之间以及类内密度变化的影响.其次,通过设计的启发式规则(k-最近邻链间距和 k-最近邻链最近邻间距)构建决策图确定最佳聚类数目并识别出聚类代表点.最后,通过最近距离传播算法进行聚类.实验表明该方法可以快速准确地找到最佳聚类个数,同时,本文提出的聚类算法与流行的聚类算法相比取得了比较好的聚类结果.

**关 键 词:** 聚类分析;聚类数目;启发式规则;随机游走模型;k-最近邻链
**中图分类号:** TP18 **文献标识码:** A **文 章 编 号:** 1000-1220(2018)07-1381-05

## Heuristic Method of Determining the Number of Clusters

LU Jian-yun[1,3],ZHU Qing-sheng[1,2],WU Quan-wang[1]

[1](School of Computer,Chongqing University,Chongqing 400044,China)
[2](Chongqing Key Laboratory of Software Theory & Technology,Chongqing University,Chongqing 400044,China)
[3](School of Software,Chongqing College of Electronic Engineering,Chongqing 401331,China)

**Abstract:** Cluster analysis is one of the important tasks in data mining. Currently,many clustering algorithms are successfully applied in image clustering,text clustering,information retrieval,social networks,etc. When the dataset is complex with different sizes,shapes and densities,it is difficult to find the best number of clusters. In this paper,we propose a heuristic method of determining the best number of clusters. First,we build a random walk model to sort the data points by their global scores,and then k_dist graph is used to determine the number of important data points in order to reduce the influence of noises and border points. Second,we develop two heuristic rules (the gap of k-nearest neighbors chain and the nearest neighbor gap of k-nearest neighbors chain) to determine the best number of clusters and the representative points of cluster by decision graph. Finally,clustering results are obtained by nearest distance propagation algorithm. Experimental results show that the proposed method can find the correct number of clusters quickly and the proposed clustering algorithm achieves comparable clustering performance with the popular clustering algorithms.

**Key words:** cluster analysis; the number of clusters; heuristic rules; random walk model; k-nearest neighbors chain

## 1 引 言

聚类分析是数据挖掘、模式识别领域的最重要任务之一,具有非常广泛的应用,例如,图像聚类,社交网络,信息检索,文本聚类等.聚类就是将数据集划分成若干个类簇,同一类簇中的数据点具有高度的相似度,不同类簇中的数据点具有极低的相似度.层次聚类可以将数据集表示成树型结构图,根据需求对树型结构图的某一层次进行划分,从而得到相应的聚类.聚类在实际应用中遇到很多的挑战,比如噪声点干扰、类内密度变化、复杂形状、高维数据、不均衡数据等.这些挑战对聚类数目的选择造成了很大的困难,同时聚类结果表现也达不到要求.聚类数目是聚类研究的基础问题之一,大多数聚类算法需要输入聚类数目,在没有更多的先验知识的情况下,确定最佳聚类数目显得尤为困难.

针对复杂数据集确定最佳聚类数目问题,本文提出了一种启发式的最佳聚类数目确定方法.K-最近邻链间距启发规则能够通过半径扩展的方式识别出球凸形状类间的变化,K-最近邻链最近邻间距启发规则能够识别出不规则形状类间的变化.启发式规则能够清楚地识别出类间的变化情况,通过决策图确定出最佳聚类数目和聚类代表点.在数据集上的实验结果表明,我们提出的聚类方法可以有效地找到正确的聚类个数,与流行的聚类算法相比取得了较好的聚类结果.

收稿日期: 2017-05-24 收修改稿日期: 2017-06-28 基金项目: 国家自然科学基金项目(61272194)资助. 作者简介: 卢建云,男,1982 年生,博士,讲师,CCF 会员,研究方向为数据挖掘、机器学习等;朱庆生,男,1956 年生,博士,教授,博士生导师,CCF 会员,研究方向为软件工程、数据挖掘、机器学习;吴全旺,男,1985 年生,博士,讲师,研究方向为云计算、服务计算等.

## 2 相关工作

目前聚类还没有一个统一的定义. 通常把聚类定义为: 同一个类簇内的对象具有很高的相似度,不同类簇间的对象具有很低的相似度; 类簇是一个密度相对较高的空间点的集合,类簇之间被相对密度较低的区域分离[1]. 在聚类分析中,聚类数目往往预先是不知道的,需要对数据集进行分析得到预估的聚类数目. 当面对具有复杂结构的数据集时,设置最佳聚类数目就显得很困难[2].

陈黎飞等人[3]提出了一种基于层次划分的最佳聚类数目确定方法,该方法首先统计数据集的聚类特征值,增量构建不同层次划分聚类的质量曲线,曲线上的极值点所对应的划分即是最佳聚类数目. 周世兵等人[4]认为决定聚类质量的关键是确定最佳聚类数目,并提出了一种基于近邻传播算法的最佳聚类数目确定方法,该方法通过计算样本聚类距离和样本聚类离差距离来确定最佳聚类数目. 刘娜等人[5]针对谱聚类通常缺少聚类数目问题,提出了一种自动确定最佳聚类数目的文档谱聚类方法. 该方法利用形态学对矩阵进行转换、过滤,通过特征间隙确定最佳聚类数目. 针对传统 k-means 算法对初始聚类中心敏感,并且无法事先确定聚类数目,王勇等人[6]提出了一种高效的 k-means 最佳聚类数算法,该方法通过样本数据分层技术得到聚类数目的上界,利用设计的聚类有效性评价指标在聚类数目搜索范围内得到最佳聚类. 同样是选取 k-means 初始聚类中心,何云斌等人[7]提出了通过标准差确定有效密度半径,并从高密度区域中选取具有代表性的样本点作为初始聚类中心,选取距离全局中心最远的点集作为最优的初始中心点集合. 冯柳伟等人[8]提出了最近邻一致性和最远邻相异性的准则,利用最近最远得分评价指标来自动确定类别数的聚类算法.

DBSCAN 算法[9]通过调整参数 $\varepsilon$ 和 *MinPts* 来自适应地学习最佳聚类数目,并且能够找到具有不同形状和大小的类簇. Dp 算法[10]提出了一种搜索密度峰值的聚类算法,它使用临界值核函数或高斯核函数来计算每个数据点的局部密度,基于这些数据点的局部密度使用 δ 距离函数来评估这些数据点之间的距离. Dp 算法选择具有更高局部密度和 δ 距离的数据点作为聚类的中心,根据决策图能够找到正确的聚类数目. 近来,受到 PageRank 算法的启发,Liu 等人提出了基于 PageRank 的聚类算法[11,12]. PageRank 算法的优势体现在快速收敛,与数据集的大小和数据点的维度无关等. 在文献[11]中,K-PRSCAN 算法与数据点的重要性顺序密切相关,并且跨度参数对聚类的输出数目有着很大的影响. 文献[12]中,作者提出了一种影响力聚类算法,简称为 CSIP. CSIP 算法采用升序方式对所有数据点的影响力进行排序,并从具有最低影响力的数据点开始进行聚类. 具有较低影响力的数据点几乎都是边界点,它们描述了每个类簇的轮廓.

## 3 数据点重要性度量

### 3.1 随机游走模型

在数据集上查找重要数据点的过程可以被看作在该数据集构建的有向图上的随机游走过程[13]. 它始发于一个随机数据点,然后随机选择一个能够跳到另一个数据点的外部链接,选中的数据点具有渐近的概率作为数据中心,能够以迭代方式计算出每个数据点作为数据中心的全局值. 全局值较高的数据点能够描述聚类结构和特征. 随机游走模型相关定义如下.

**定义 1**. (数据点 p 的 $k$ 最近邻距离) 在一个数据集 $D$ 中,数据点 p 的 $k$ 最近邻距离表示为 $k\_dist(\mathrm{p})$,它是指数据点 p 和 o 之间的距离 $d(\mathrm{p},\mathrm{o})$,具体性质描述如下:

1) 至少有 $k$ 个对象 $\mathrm{q}\in D/\{\mathrm{p}\}$,那么,$d(\mathrm{p},\mathrm{q}) <= d(\mathrm{p},\mathrm{o})$;

2) 至多有 $(k-1)$ 个对象 $\mathrm{q}\in D/\{\mathrm{p}\}$,那么,$d(\mathrm{p},\mathrm{q}) < d(\mathrm{p},\mathrm{o})$.

**定义 2**. (数据集的 $k$-最近邻有向图 $G$) 给出一个数据集 $D$,数据点 $\mathrm{p},\mathrm{q}\in D,\mathrm{p}\neq\mathrm{q}$. 如果 $d(\mathrm{p},\mathrm{q}) <= k\_dist(\mathrm{p})$,那么,在 $G$ 中有一条有向边从顶点 p 到 q.

**定义 3**. (有向图 $G$ 的邻接矩阵) 给出一个 $n$ 个顶点的有向图 $G$,它的邻接矩阵 $A$ 是一个 $n\times n$ 矩阵. 在该矩阵中,如果从顶点 $i$ 到顶点 $j$ 存在有向边,$A$ 中元素 $a_{ij}$ 的值等于 1; 反之,$a_{ij}=0$.

邻接矩阵用于存储数据集 $k$-最近邻有向图中数据点的链接关系. 在邻接矩阵 $A$ 中,第 $i$ 行元素的总和表示从数据点 $i$ 到其他数据点的内部链接数,第 $j$ 列元素的总和表示从其他数据点到数据点 $j$ 的外部链接数,分别用 $r_i$ 和 $c_j(1<=i,j<=n)$ 表示. 假设 $S$ 为随机游走模型的转移概率矩阵,则 $S$ 可以用公式(1)表示如下. $S$ 中的元素都严格介于 0 和 1 之间,并且每列元素的总和都等于 1.

$$s_{ij}=\begin{cases}a_{ij}/c_j, c_j\neq 0\\ 1/n, c_j=0\end{cases} \tag{1}$$

在实际应用中,$k$-最近邻有向图可能包含完整的子图,这些子图不会链接到任何外部数据点. 这就意味着,游走者可能走进某个子图,并随机漫步在这个封闭的区域内. 因此,它只能发现局部聚类中心,而非全局聚类中心. 为了避免这种现象,我们将具有小概率的阻尼因子 $\alpha$ 添加到随机矩阵 $S$ 中,改进后的 $S$ 如公式(2)所示.

$$s_{ij}=\begin{cases}\alpha a_{ij}/c_j+(1-\alpha)/n, c_j\neq 0\\ 1/n, c_j=0\end{cases} \tag{2}$$

目前 $S$ 是一个列不可约随机矩阵. 根据 Perron-Frobenius 理论[14],$S$ 取 1 为最大特征值,其对应的特征向量取值为 0 和 1 之间的非负数. 数据点的重要性可以转化为计算 $S$ 的特征值 1 所对应的特征向量问题. 描述如下:

$$Sx=x \tag{3}$$

通常查找给定矩阵特征值的特征向量是一个复杂而耗时的问题. 文献[14]给出了几种计算特征向量的方法. 本文选择 Power 方法,其具有存储少、计算时间短和复杂度低等优点. 在 Power 方法中,如果阻尼因子 $\alpha$ 的值太小,则矩阵 $S$ 将很难描述数据点的真正链接关系; 如果阻尼因子 $\alpha$ 的值接近 1,则算法收敛很慢. 在实验中,为了反映数据点的链接关系,我们将 $\alpha$ 值设为 0.95.

### 3.2 重要性排序算法

本节给出了基于随机游走模型迭代计算每个数据点的全局重要性的算法,具体描述如算法 1 所示. 该算法包含三个参数 $k$,$\alpha$ 和 $\varepsilon$. 其中,$k$ 是查找最近邻居个数,$\alpha$ 是防止算法陷入局部最优的阻尼因子,$\varepsilon$ 用来控制算法的收敛速度. 在实验中,设置参数 $k$ 取值为离散区间 [5,30],$\alpha=0.95$,$\varepsilon=10^{-6}$.

**算法 1**. 数据点重要性排序

**输入:** 包含 $n$ 个数据点的数据集 $D$,参数 $k$,$\alpha$ 和 $\varepsilon$.
**输出:** 包含每个数据点的全局重要性的向量 $dpir$.
1. 初始化参数 $k$,$\alpha$ 和 $\varepsilon$.
2. 在数据集 $D$ 中查找每个数据点的 $k$-最近邻.
3. 根据 $k$-最近邻有向图构造邻接矩阵 $A$.
4. 根据公式(2)计算列不可约随机矩阵 $S$.
5. 根据公式(3),使用 Power 方法计算矩阵 $S$ 特征值 1 对应的特征向量.
6. 返回特征向量 $dpir$.

## 4　启发聚类算法

### 4.1　k 最近邻链启发规则

通常聚类中心是数据密集区域或中心区域. 具有较高的重要性的数据点通常位于数据集的密集区或中心区域,而位于稀疏区或边界区的噪声和边界点通常具有较低的重要性. 本节我们提出了基于 $k$ – 最近邻链的启发规则来确定最佳聚类个数,相关定义如下.

**定义 4.** ($k$-最近邻链) 给定一个具有 $n$ 个数据点的数据集 $D$,$k$NN(p) 表示数据点 p 的 $k$-最近邻集合,$t = |k\text{NN}(\text{p})|$,$t \leqslant n-1$. 存在数据点 p 的一个 $k$-最近邻链 $\text{p},q_1,q_2,\cdots,q_i,q_{i+1},\cdots,q_t$,其中 $q_i$ 表示 p 的第 $i$ 个近邻,$q_i \neq q_i+1$,$d(p,q_i) \leqslant d(p,q_{i+1})$,$1 \leqslant i < t-1$.

**启发规则 1.** ($k$-最近邻链间距) 给出数据点 p 的一条 $k$-最近邻链,用 $k$NNC(p) 表示,$t = |k\text{NNC}(\text{p})|$ 表示 $k$NNC(p) 中包含数据点的个数,$k$-最近邻链间距定义为 $\Delta g(q_i) = d(\text{p},q_i) - d(\text{p},q_{i-1})$,$q_i$ 是 p 的第 $i$ 个最近邻,$q_{i-1}$ 是 p 的第 $i-1$ 个最近邻,$1 < i \leqslant t$.

我们注意到,$\Delta g$ 能够描述类簇之间的距离变化. 当 $k$-最近邻链中包含的数据点属于不同的类簇时,绘制出的 $\Delta$g 曲线将包含比较明显的峰值,而不是平滑的曲线. 启发规则 1 对于检测球凸形状的类簇非常有效. 对于具有不规则形状的类簇或一个类簇嵌入到另一个类簇中,我们引入另一种启发规则来找到正确的聚类数. 下面给出 $k$-最近邻链最近邻间距的定义.

**启发规则 2.** ($k$-最近邻链最近邻间距) 给出数据点 p 的一条 $k$-最近邻链 $k$NNC(p),$t = |k\text{NNC}(\text{p})|$ 表示 $k$NNC(p) 中包含数据点的个数. $k$NNC(p) 中第 $i$ 个点 $q_i$ 的最近邻间距定义为 $\Delta nng(q_i) = d(q_i,q_j)$ $1 \leqslant i \leqslant t$,$1 \leqslant j < i$,其中 $q_j$ 是 $q_i$ 的最近邻居.

$k$-最近邻链最近邻间距揭示了类簇内的密度变化. 一旦 $\Delta nng$ 曲线包含比较明显的峰值,我们便认为出现了新的类簇. 如果数据点 $q_i$ 具有较大的 $\Delta$g($q_i$) 值,则 $\Delta nng(q_i)$ 值也较大. 反之,则不一定成立.

### 4.2　算法步骤

聚类过程分两个阶段,第一阶段我们利用 $k\_dist$ 距离图谱来确定截取的重要数据点的个数,然后对截取的数据点构建 $k$-最近邻链,计算链中每个点的 $\Delta g$ 和 $\Delta nng$ 值并绘制二维决策图,得到最佳聚类数目和聚类代表点. 第二阶段利用最近距离传播进行聚类. 我们提出的启发聚类算法如下所示.

**算法 2.** 启发聚类算法(Heuristic Clustering,HC)
**输入:** 数据集 $D$.
**输出:** 聚类标签 $C$.
1. 计算数据集 $k\_dist$ 距离.
2. 确定截取重要数据点个数 $N_{imp}$,形成新的数据集 $D_{imp}$.
3. 为新数据集 $D_{imp}$ 构建欧氏距离矩阵 $E$.
4. 在矩阵 $E$ 中找到具有最大值的数据点 p.
5. 构建一链($N_{imp}-1$) NNC(p) 链.
6. 计算 $\Delta heu(q_i) = \Delta g(q_i) + \Delta nng(q_i)$,$q_i \in (N_{imp}-1)$ NNC(p).
7. 对 $\Delta heu$ 进行降序排序.
8. 从决策图中确定聚类数目 $c$,并从($N_{imp}-1$) NNC(p) 中得到对应的聚类代表点 $q_1,q_2,\cdots,q_c$.
9. 初始化聚类 $C = \{C_1 = \{q_1\},C_2 = \{q_2\},\cdots,C_c = \{q_c\}\}$,$D = D - \{q_1,q_2,\cdots,q_c\}$.
10. for 每个数据点 $p \in D$
11. $dist[p] = min(p,C)$.
12. end
13. $[p_i,C_j] = min(dist)$.
14. $C_j = C_j \cup \{p_i\}$.
15. $D = D - \{p_i\}$.
16. 重复步骤 9-14,直至 $D = \varnothing$.
17. 返回聚类标签 $C$.

## 5　实验与结果

在本节中,在数据集 DataGroup 上对本文提出的确定最佳聚类数和聚类效果进行验证. DataGroup 包含了 10 个 2 维的人工数据集,数据集中包含了噪声和不同形状、不同密度、不同程度连接的类簇,其中前 5 个数据集用来验证最佳聚类数方法,后 5 个数据集用来验证聚类方法,DataGroup 具体描述信息如表 1 所示.

**表 1　DataGroup 的基本信息**
Table 1　Details of DataGroup

| Datasets | Number of Clusters | Number of Samples |
|---|---|---|
| data-c-cc-nu-n | 3 | 289 |
| data-c-cv-nu-n | 3 | 76 |
| data-uc-cc-nu-n | 3 | 191 |
| data-link2 | 3 | 204 |
| dataX | 2 | 202 |
| Aggregation | 7 | 788 |
| Compound | 6 | 399 |
| Flame | 2 | 240 |
| Pathbased | 3 | 300 |
| Spirial | 3 | 312 |

### 5.1　聚类有效性评价指标

我们采用 *Silhouette* 评价指标对最佳聚类数进行验证. *Silhouette* 的定义如下:

$$s(i) = \frac{d_{b(i)} - d_{a(i)}}{\max\{d_{b(i)},d_{a(i)}\}} \tag{4}$$

在公式(4)中,$d_{b(i)}$ 表示数据点 $i$ 与其它类最低平均不相似度,$d_{a(i)}$ 表示数据点 $i$ 与它所在类的平均不相似度,$s(i)$ 的值介于-1 到 1 之间,值越大,聚类效果越好. 数据点 $x = (x_1,x_2,\cdots,x_n)$ 和数据点 $y = (y_1,y_2,\cdots,y_n)$ 之间的距离定义如公式(5)所示:

$$d_{xy} = \sum_{j=1}^{n} |x_j - y_i| \tag{5}$$

我们采用准确率(Accuracy)和标准互信息(Normalization Mutual Information,*NMI*)两个评价指标对聚类结果进行评价. 准确率定义如下:

$$A_{cc} = \frac{TP}{ToTal} \times 100\% \tag{6}$$

在公式(6)中,*TP* 表示正确聚类的样本个数,*Total* 表示总体样本个数. 标准互信息定义如下:

$$NMI = \frac{\sum_i \sum_j s_{i,j} \log \frac{t \times s_{i,j}}{s_i \times s_j}}{\sqrt{\sum_i s_i \log \frac{s_i}{t} \sum_j s_j \log \frac{s_j}{t}}} \quad (7)$$

在公式(7)中,$s_i$ 和 $s_j$ 分别表示第 $i$ 个聚类和第 $j$ 个聚类的样本个数,$s_{i,j}$ 表示第 $i$ 个聚类和第 $j$ 个聚类相同的样本个数,$t$ 表示总体样本个数. *Acc* 和 *NMI* 的值越大,表示聚类的效果越好.

### 5.2 最佳聚类数分析

本节对 DataGroup 前 5 个数据集上确定最佳聚类数进行了实验. 图 1 给出了每个数据集的形态与分布(A1 ~ E1),以及对应的 *k_dist* 距离图谱(A2 ~ E2)和聚类数目决策图(A3 ~ E3).

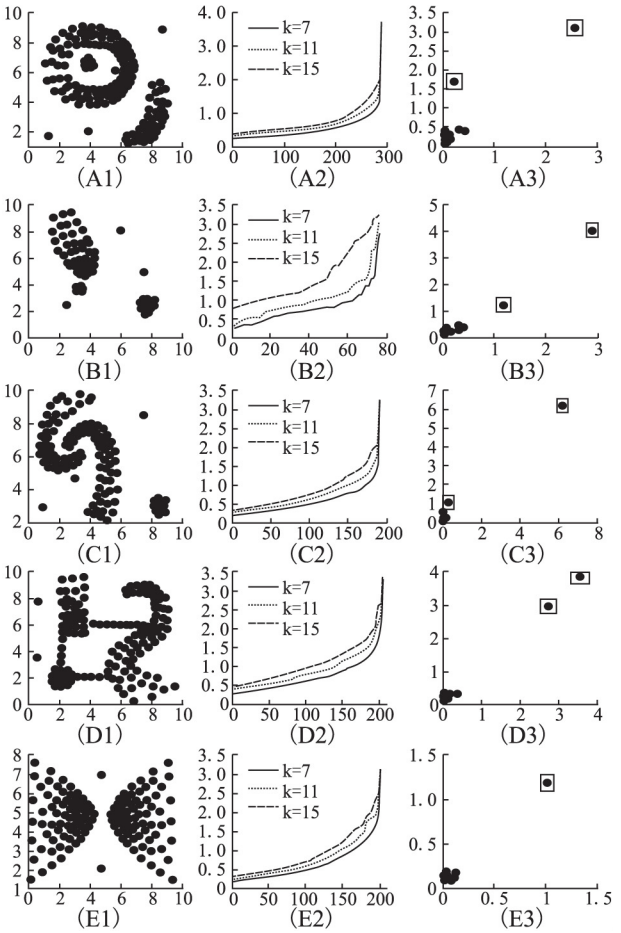


图 1 DataGroup 前 5 个数据集(A1-E1)及其所对应的 *k_dist* 距离图谱(A2-E2)和聚类数目决策图(A3-E3)

Fig. 1 First 5 datasets in DataGroup (A1-E1) and its corresponding *k_dist* graphs (A2-E2) and cluster number decision graphs (A3-E3)

在计算距离图谱时,我们取 $k = 7, 11, 15$. 随着 $k$ 值的增大,*k_dist* 距离也越来越大,距离图谱曲线变的越来越陡峭,曲线当中的拐点也越来越不明显,不利于观察数据集密度的变化情况. 分析和观察距离图谱的拐点,可以选择多个多个数据,最后确定最佳的重要数据点的数目.

聚类数目决策图显示了峰值的分布情况,横坐标表示 $\Delta g$ 值,纵坐标表示 $\Delta nng$ 值,被框起来的点表示选取的峰值. 假设一条 k 最近邻链出现了 n 个峰值,通过峰值进行划分,则 k 最近邻链被划分成了 n + 1 个部分. 因此,峰值数目和聚类数目的关系是: 聚类数目 = 峰值数目 + 1.

对于 B1,D1,E1 数据集,本文方法很好地识别出类簇中的密集区域,并且密集区域间距明显,所以聚类数目决策图 B3,D3 和 E3 中的峰值同时具有较大的 $\Delta g$ 值和 $\Delta nng$ 值. 对于 A1 数据集,存在着一个嵌套的类簇,在聚类数目决策图 A3 中有一个峰值具有较大的 $\Delta nng$ 值,而 $\Delta g$ 值很小. C1 数据集中有两个类簇形成一个环状,并且密集区域离的非常近,所以在 C3 中有一个峰值不是很明显.

图 2 给出了聚类结果对应的轮廓(*Silhouette*)图. 从图中可以看出,数据集 data-c-cv-nu-n、data-link2、dataX 取得了比较好的聚类效果,只有少部分 *Silhouette* 的值是负数. 数据集 data-c-cc-nu-n 中存在一个环状嵌套的类簇,所以轮廓图中第 2 个聚类有一部分 *Silhouette* 的值是负数表明与被嵌套的聚类相似度大于与本类的相似度. 数据集 data-uc-cc-nu-n 中第 1 个聚类的尾部与第 3 个聚类的头部的相似度要大于与本类的相似度,所以第 1 个聚类有一部分 *Silhouette* 值是负数.

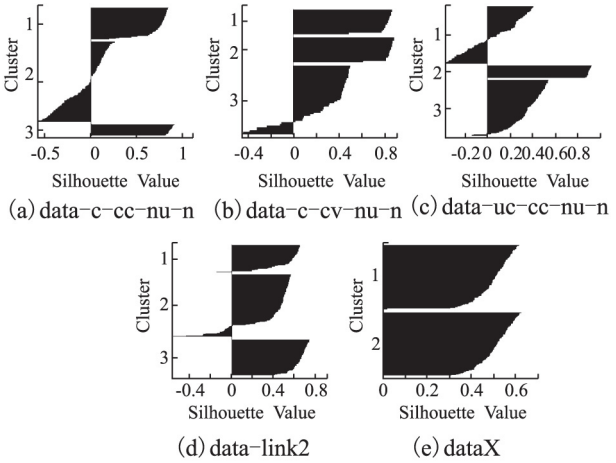


图 2 DataGroup 中前 5 个数据集的聚类轮廓图

Fig. 2 Silhouette results of the first 5 datasets in DataGroup

### 5.3 聚类算法对比分析

在本节中,我们对 DataGroup 后 5 个数据集进行了 2 个实验来验证本文聚类算法的有效性.

表 2 本文聚类方法与其他方法在 DataGroup 后 5 个数据集上的 Acc(%)对比结果

Table 2 Acc (%) results of our method and other popular methods on the last 5 datasets in DataGroup

| Algorithms | Datasets | | | | |
|---|---|---|---|---|---|
| | Aggregation | Compound | Flame | Pathbased | Spirial |
| k-means | 79.31 | 51.88 | 83.75 | 74.33 | 34.62 |
| CSIP | 85.79 | 84.96 | 77.92 | 79.33 | **100** |
| Dp | **99.87** | 63.41 | 78.75 | 73.33 | **100** |
| SC | 97.59 | 69.67 | 86.67 | 63.33 | **100** |
| HC | 95.94 | **87.22** | **100** | **96.00** | **100** |

第 1 个实验,将 HC 算法与流行的 K-means 算法[15]、谱聚类算法 SC[16]、密度峰值 Dp 算法[10]和基于影响的排序 CSIP 算法[12]就聚类准确率和 *NMI* 值进行了比较. 表 2 和表 3 分别给出了 5 个聚类算法在 DataGroup 后 5 个数据集上的 *Acc* 值和 *NMI* 值. 从表 2 和表 3 中可以看出,HC 算法在 Flame、Compound、Pathbased 数据集上取得了最高的准确率和

*NMI* 值. 对于 Flame 和 Spirial 两个数据集,HC 取得了 100% 准确率,同时 NMI 的值为 1. 在 Aggregation 数据集上,HC 算法取得了第 3 好 95.94% 的准确率,NMI 值为 0.95. 这是因为 Aggregation 数据集中有两个类簇之间存在桥接链,桥接链上面的点很密集,距离很近,所以在采用最近距离聚类时会把桥接链上面的点划分在一起,甚至会延伸到另外一个类簇.

表 3　本文聚类方法与其他方法在 DataGroup 后 5 个数据集上的 *NMI* 对比结果

Table 3　*NMI* results of our method and other popular methods on the last 5 datasets in DataGroup

| Algorithms | Datasets | | | | |
|---|---|---|---|---|---|
| | Aggregation | Compound | Flame | Pathbased | Spirial |
| k-means | 0.85 | 0.66 | 0.40 | 0.55 | 0.0006 |
| CSIP | 0.86 | 0.82 | 0.43 | 0.56 | **1** |
| Dp | **0.99** | 0.73 | 0.41 | 0.54 | **1** |
| SC | 0.96 | 0.83 | 0.52 | 0.49 | **1** |
| HC | 0.95 | **0.90** | **1** | **0.88** | **1** |

第 2 个实验,在 Win7 操作系统,6G 内存,2.4GHz 双核 CPU 上,我们将 5 个算法在 DataGroup2 数据集上的运行时间进行了比较,结果如表 4 所示. 从表 4 中可知,K-means 算法在运行时间上取得了最好的性能,Dp 算法仅次于 K-means 算法,两个算法的运行时间在 Aggregation 数据集上均小于 0.03 秒,在其他数据集上更是少于 0.01 秒. HC 算法的运行时间性能优于 CSIP 和 SC 算法. HC 算法在 Aggregation 数据集上运行时间少于 0.2 秒,在其他数据集上的运行时间不足 0.05 秒. SC 算法在每个数据集上的运行时间相对稳定,在 0.3 秒到 0.4 秒之间. 相对其他算法,CSIP 算法在 Aggregation 数据集上运行时间最长为 0.6 秒.

表 4　本文聚类方法与其他方法在 DataGroup 后 5 个数据集上的运行时间(s)对比结果

Table 4　Runtime(s) results of our method and other popular methods on the last 5 datasets in DataGroup

| Algorithms | Datasets | | | | |
|---|---|---|---|---|---|
| | Aggregation | Compound | Flame | Pathbased | Spirial |
| k-means | 0.0173 | 0.0084 | 0.0062 | 0.0067 | 0.0071 |
| CSIP | 0.6124 | 0.1833 | 0.0794 | 0.1536 | 0.1154 |
| Dp | 0.0272 | 0.0092 | 0.0053 | 0.0093 | 0.0096 |
| SC | 0.4096 | 0.3481 | 0.3278 | 0.3513 | 0.3036 |
| HC | 0.1331 | 0.0499 | 0.0276 | 0.0375 | 0.0412 |

## 6　总　结

确定最佳聚类数目是聚类分析研究的一个基础问题. 针对复杂结构数据集,本文提出了一种启发式确定最佳聚类数的方法. 其中,$k$-最近邻链间距曲线能够描述类间的距离变化,$k$-最近邻链最近邻间距曲线能够反映类内的聚类变化. 理论分析和实验结果表明,本文方法对具有密集区域并且被稀疏区域分离的数据,对流行、不规则形状分布的数据比较有效,能够快速准确地找到最佳聚类个数. 与目前流行的聚类算法相比,本文聚类算法取得了较好的结果,在处理速度方面也具有明显的优势.

**References:**


[1] Sun Ji-gui, Liu Jie, Zhao Lian-yu. Clustering algorithms research [J]. Journal of Software, 2008, 19(1): 48-61.

[2] Jin Jian-guo. Review of clustering method [J]. Computer Science, 2014, 11(41): 288-293.

[3] Chen Li-fei, Jiang Qing-shan, Wang Sheng-rui. A hierarchical method for determining the number of clusters [J]. Journal of Software, 2008, 19(1): 62-72.

[4] Zhou Shi-bing, Xu Zhen-yuan, Tang Xu-qing. Method for determining optimal number of clusters based on affinity propagation clustering [J]. Control and Decision, 2011, 26(8): 1147-1152.

[5] Liu Na, Lu Ying, Tang Xiao-jun, et al. Study on automatically determining the optimal number of clusters present in spectral co-clustering documents and words [J]. Journal of Chinese Computer Systems, 2014, 35(3): 610-614.

[6] Wang Yong, Tang Jing, Rao Qin-fei, et al. High efficient K-means algorithm for determining optimal number of clusters [J]. Journal of Computer Applications, 2014, 34(5): 1331-1335.

[7] Feng Liu-wei, Chang Dong-xia, Deng Yong, et al. A clustering evaluation index based on the nearest and the furthest score [J]. CAAI Transactions on Intelligent Systems, 2017, 12(1): 1-8.

[8] He Yun-bin, Liu Xue-jiao, Wang Zhi-qiang, et al. Improved K-means algorithm based on global center and nonuniqueness high-density points [J]. Computer Engineering and Applications, 2016, 52(1): 48-54.

[9] Ester Martin, Kriegel Hans-Peter, Sander Jörg and Xu Xiaowei. A density-based algorithm for discovering clusters in large spatial databases with noise [C]. Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining(KDD-96), 1996: 226-231.

[10] Alex R, Alessandro L. Clustering by fast search and find of density peaks [J]. Science, 2014: 1489-1492.

[11] Liu L, Sun L, Chen S, et al. K-PRSCAN: a clustering method based on PageRank [J]. Neurocomputing, 2016, 175: 65-80.

[12] Liu L, Chen X W, Liu M, et al. An influence power-based clustering approach with PageRank-like model [J]. Applied Soft Computing, 2016, 40(C): 17-32.

[13] Brin S, Page L. The anatomy of a large-scale hypertextual Web search engine [J]. Computer Networks and ISDN Systems, 1998, 30(1-7): 107-117.

[14] Schneider B H. Algebraic perron-frobenius theory [J]. Linear Algebra and Its Applications, 1975, 11(3): 219-233.

[15] John Hartigan, Manchek Wong. A k-means clustering algorithm [J]. Applied Statistics, 1979, 28(1): 100-108.

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


**附中文参考文献:**


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

[2] 金建国. 聚类方法综述 [J]. 计算机科学,2014,11(41):288-293.

[3] 陈黎飞,姜青山,王声瑞. 基于层次划分的最佳聚类数确定方法 [J]. 软件学报,2008,19(1):62-72.

[4] 周世兵,徐振源,唐旭清. 一种基于近邻传播算法的最佳聚类数确定方法 [J]. 控制与决策,2011,26(8):1147-1152.

[5] 刘　娜,路　莹,唐晓君,等. 自动确定单词-文档谱聚类最佳聚类数目的研究 [J]. 小型微型计算机系统,2014,35(3):610-614.

[6] 王　勇,唐　靖,饶勤菲,等. 高效率的 K-means 最佳聚类数确定算法 [J]. 计算机应用,2014,34(5):1331-1335.

[7] 冯柳伟,常冬霞,邓　勇,等. 基于最近最远得分的聚类性能评价指标 [J]. 智能系统学报,2017,12(1):1-8.

[8] 何云斌,刘雪娇,王知强,等. 基于全局中心的高密度不唯一的 K-means 算法研究 [J]. 计算机工程与应用,2016,52(1):48-54.