文章编号: 1001-0920(2017)08-1421-06 DOI: 10.13195/j.kzyjc.2016.0984

维度概率摘要模型及其层次聚类算法

刘世华[1,2], 黄德才[1†]

(1. 浙江工业大学 计算机科学与技术学院，杭州 310023；2. 温州职业技术学院 信息技术系，浙江 温州 325035)

摘 要: 提出一种维度概率摘要模型,将聚类产生的簇摘要信息采用各维度的概率分布来表示;定义点簇相似度、簇簇相似度等相似性度量方法;提出一种基于维度概率摘要模型的凝聚层次聚类算法. 实验分析发现,所提模型和算法能够产生高质量的聚类,能够避免噪声点的影响并发现离群点,能够自动发现聚类,算法稳定可靠且对高维数据集聚类效果很好.

关键词: 维度概率分布；维度概率摘要模型； 点簇相似度； 簇簇相似度； 层次聚类

中图分类号: TP273 **文献标志码:** A

Hierarchical clustering algorithm with dimensions probability summary model

LIU Shi-hua[1,2], *HUANG De-cai*[1†]

(1. College of Computer Science & Technology, Zhejiang University of Technology，Hangzhou 310023，China；
2. Department of Information Technology，Wenzhou Vocational & Technical College，Wenzhou 325035，China)

Abstract: A dimensions probability summary model is proposed, which uses dimensions probability distributions to represent the cluster summary. Similarities from point to cluster and from cluster to cluster are defined as the similarity measurement. A hierarchical clustering algorithm based on the dimensions probability summary model is proposed. Experimental analysis shows that, the model and algorithm can produce high quality of the clustering, avoid the noise and find the outliers, and automatically determine the cluster number. The proposed algorithm is stable and reliable, and is suitable for high dimensional data clustering.

Keywords: dimensions probability distributions；dimensions probability summary model；similarity from point to cluster； similarity from cluster to cluster；hierarchical clustering

0 引 言

聚类分析一直是数据挖掘和机器学习领域的研究热点之一,在模式识别、统计学、生物学、市场营销等各个领域都有广泛的应用[1-3]. 聚类分析的目标是发现数据对象集中有某种意义的“自然”分组,即所谓的“簇”[2-4]. 聚类研究是研究某种数据分类方法,使同一簇内数据对象尽量紧凑或相似,而不同簇之间的对象尽量不同[5].

K-means算法出现的几十年来,聚类分析的各种算法层出不穷[1-3],包括改进距离度量方式、结合生物计算或智能算法进行优化等[6-8]. 但聚类分析研究的几大主要问题却仍未得到很好地解决,其主要包括:如何合理计算数据点之间的相异或相似程度,如何消除噪声或离群点数据的影响,如何聚类高维数据,如何自动确定数据集包含的簇数目等[3].

以K-means为代表的基于划分的聚类算法以其简单高效的优势得到了广泛的应用,但其仍存在很多问题,如:对初始簇中心点的选取敏感,易于陷入局部最优解;需要事先指定簇数目;对噪声点或离群点敏感等[3]. 此外,基于划分的聚类算法基本上都是用一个或多个中心点(如K-means中的均值和K-medoids中的中心代表点)来代表整个簇,这将导致簇中的部分信息丢失. 例如一个分布紧凑的簇和一个相对分散的簇有可能具有一样的中心代表点,但其簇内的紧密程度是不一样的. 为此,王玲等[6]提出了聚类中心和聚类半径的概念,罗印升等[7]定义了一个包含簇中数据点数目、中心点数目和半径的三元组来描述一个簇,都对簇的表示进行了有效的尝试,不过这些表

收稿日期: 2016-08-01；**修回日期:** 2016-10-24.

基金项目: 水利部公益性行业科研专项基金项目(201401044)；国家科技支撑计划子课题项目(2012BAD10B0101).

作者简介: 刘世华(1978–), 男, 高级工程师, 博士生, 从事数据挖掘、聚类分析的研究； 黄德才(1958–), 男, 教授, 博士生导师, 从事数据仓库与数据挖掘等研究.

†通讯作者. E-mail: hdc@zjut.edu.cn

示方式都以欧氏距离为基础.

本文基于文献[9]中的维度距离思想提出一种全新的维度概率摘要模型,将聚类产生的簇分维度采用概率分布来表示其摘要信息,并定义一个数据点到簇的维度概率摘要的点簇相似度以及簇之间的簇簇相似度作为相似性度量,以此为基础给出一个全新的基于维度概率摘要模型的层次聚类算法. 该模型及其算法为聚类分析提供了一种全新的视角,采用人工和真实数据集进行实验分析发现,该算法能够有效地发现高质量的簇,通过改进层次聚类设置合适的聚类合并阈值,能够自动确定簇的数目,且能够发现离群点.

1 维度概率摘要模型

1.1 维度距离

维度距离思想认为,在高维情况下,两个对象在每一维上都接近比在少数维度上接近更有意义[9]. 因此,它在每个维度设定一个相似度阈值ε,如果两个数据点在某个维度上的距离小于该阈值,则认为这两个点在该维度上相等,两个数据点是否接近则采用相等维度的数目多少来判断. 如图1所示,以二维坐标系的两个点$A(3,3)$和$B(4,0)$为例,如果采用欧氏距离来描述的话,则$|OB|<|OA|$,故点B比点A更接近于原点O. 但如果采用面向维度的距离描述的话,设阈值$\varepsilon=3.1$,点A在每一维上与点O的差都小于ε,因此相等维度数为2,而点B在x轴上与点O的距离大于ε,故相等维度数为1,因此点A比点B更接近于点O.

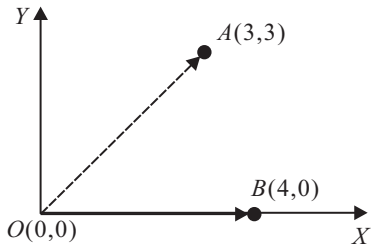


图1 维度距离与传统欧氏距离比较示例 ($\varepsilon=3.1$)

上述维度距离只是根据设定的维度阈值ε来确定维度相等或不相等(即只有{1,0}判断),可将此维度距离扩展定义一个维度相似度度量,使其值处于[0,1]区间,这样对每一维度上的数据对象的相似性可作更细致的度量和区分,具体见下节定义.

1.2 相关定义

设n个数据点构成的d维数据集S的一种聚类划分$C=\{C_1,C_2,\cdots,C_k\}$,称$C_i(i=1,2,\cdots,k)$为聚类C上的一个簇,k为聚类C的簇数. 维度概率摘要模型认为,同一簇中数据点同一维度上的数值应服从一定的概率分布,本文采用最常见的正态分布表示,因此可定义相关模型和相似性度量.

定义1 簇的维度概率分布. 设簇C_i中包含m个数据点,定义一个正态分布$N(\mu_j,\sigma_j)$来表示簇中所有信息点在第j维的数值信息摘要,称为簇的维度概率分布,记为$\mathrm{DN}_j(\mu_j,\sigma_j)$. 其中:$j=1,2,\cdots,d$,$d$表示数据的维度,$\mu_j$表示簇$C_i$中所有数据点在第$j$维上的值的均值,$\sigma_j$表示该簇中所有数据点在第$j$维上的标准差.

定义2 簇的维度概率摘要. 对于某个簇C_i,定义$\mathrm{DS}_i=\{\mathrm{DN}_1,\mathrm{DN}_2,\cdots,\mathrm{DN}_d\}$表示该簇的摘要信息,称为簇$C_i$的维度概率摘要,$\mathrm{DN}_j$为簇$C_i$在第$j$维的维度概率分布. 维度概率摘要$\mathrm{DS}_i$用于代替传统的中心点坐标来代表一个簇的信息. 其在每一维度都保留了对应该维的均值和标准差信息,与采用中心点代表一个簇的方式相比,减少了信息的丢失.

定义3 点簇维度概率相似度. 维度概率分布采用正态分布表示,对于如何判断一个数据点与一个已有簇在对应维上的相似度,可定义一个取值范围在[0,1]上的分段函数来表示点簇维度概率相似度,记为$\mathrm{SimPN}(\mathrm{PV}_j,\mathrm{DN}_j)$,简记为SimPN.

根据正态分布的3σ法则,属于某正态分布的数据点中有99.73%是落在正态分布的均值附近$[\mu-3\sigma,\mu+3\sigma]$范围以内,因此如果某个数据点超出这个范围,则认为它不属于该分布,并认为该数据点与该分布的相似度为0. 定义分段函数如图2所示.

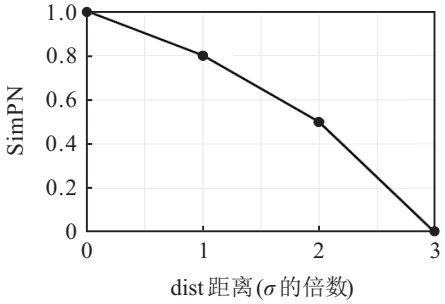


图2 点簇维度概率相似度分段函数图

具体分段函数定义如下式所示:

$$\mathrm{SimPN}=\begin{cases}1-\dfrac{0.2\times\mathrm{dist}_j}{\sigma_j},\ \mathrm{dist}_j\leqslant\sigma_j;\\ 1.1-\dfrac{0.3\times\mathrm{dist}_j}{\sigma_j},\ \sigma_j<\mathrm{dist}_j\leqslant2\sigma_j;\\ 1.5-\dfrac{0.5\times\mathrm{dist}_j}{\sigma_j},\ 2\sigma_j<\mathrm{dist}_j\leqslant3\sigma_j;\\ 0,\ \mathrm{dist}_j>3\sigma_j.\end{cases}\tag{1}$$

其中:$\mathrm{dist}_j=|\mathrm{PV}_j-\mu_j|$表示数据点在第$j$维上的值$\mathrm{PV}_j$与簇$C_i$第$j$维的维度概率分布均值$\mu_j$之间的距离,采用二者差的绝对值表示. 如果二者距离相差不超过该维度概率分布的一倍标准差(即$\mathrm{dist}_j\leqslant\sigma_j$),则认为该数据点与该维度概率分布有80%以上的相似度,且dist_j越小,相似度越接近1. 当$\mathrm{dist}_j=0$(即

$\mathrm{PV}_j = \mu_j$)时, SimPN = 1; 当$\mathrm{dist}_j = \sigma_j$时, SimPN = 0.8; 当$\mathrm{dist}_j = 2\sigma_j$时, SimPN = 0.5; 当$\mathrm{dist}_j \geqslant 3\sigma_j$时, SimPN = 0.

定义4　点簇相似度. 通过计算每一维的点簇维度概率相似度,可以定义数据点和簇的维度概率摘要之间的相似度为数据点到簇的所有维度上的平均点簇维度概率相似度,简称为点簇相似度,记为SimD(PV,DS_i). 其中: PV是数据点的原始值, DS_i是簇C_i的维度概率摘要. 点簇相似度简记为SimD,计算公式如下:

$$\mathrm{SimD} = \frac{\sum_{j=1}^{d} \mathrm{SimPN}_j}{d}. \tag{2}$$

其中: d为数据点的维度数, SimPN_j为第j维上的点簇维度概率相似度. 显然, SimD的取值范围也是[0,1].

定义5　簇簇维度概率相似度. 设两个簇的概率摘要模型在某维度上的维度概率分布为$\mathrm{DN}_a(\mu_a, \sigma_a)$和$\mathrm{DN}_b(\mu_b, \sigma_b)$,这两个簇的维度概率相似度可定义为SimNN($\mathrm{DN}_a, \mathrm{DN}_b$),简记为SimNN,如下式所示:

$$\mathrm{SimNN} = \frac{\mathrm{SimPN}(\mu_a, \mathrm{DN}_b) + \mathrm{SimPN}(\mu_b, \mathrm{DN}_a)}{2}. \tag{3}$$

两个簇的维度概率相似度可以用簇a的均值到簇b的点簇维度概率相似度和簇b的均值到簇a的点簇维度概率相似度的平均值来度量. 同样的, SimNN的取值范围为[0,1].

定义6　簇簇相似度. 将两个簇所有维度上的簇簇维度概率相似度的平均值定义为簇簇相似度,记为SimN(C_a, C_b),简记为SimN,如下式所示:

$$\mathrm{SimN} = \frac{\sum_{i=1}^{d} \mathrm{SimNN}_i}{d}. \tag{4}$$

其中: d为数据点的维度数, SimNN_i为两个簇在第i维上的簇簇维度概率相似度. 显然, SimN的取值范围也在[0,1]区间.

2　基于维度概率摘要模型的层次聚类

2.1　聚类思路

基于上节维度概率摘要和各相似度的定义,可以采用Squeezer算法[10]思想定义一个面向维度概率摘要模型的一趟聚类算法DPC(dimensions probability modeled clustering). 算法首先以第一数据点为基础构建出第一个簇的维度概率摘要,然后依次输入其余数据点,根据点簇相似度阈值来确定该数据点是加入已有簇还是新建一个簇,后续数据点在相似度大于给定阈值的情况下,以点簇相似度最大的那个簇作为自己所属簇,当有新的数据点加入簇时,可在一定条件下更新该簇的维度概率摘要信息. 如此对数据集S进行一趟扫描,即可将所有数据点划分到其所属簇中,此过程称为DPC聚类. 在DPC聚类结果的基础上,基于簇簇相似性度量,定义一个簇簇合并阈值t,采用层次凝聚算法将相似度大于t的簇进行合并. 最终,若某簇中数据点数量较少,则可根据应用需要认定其为离群点.

2.2　DPC算法流程

DPC算法的输入为数据集对象X和点簇相似度聚类阈值ε,输出为簇编号和各簇的维度概率摘要模型.

算法的具体流程如下.

Step 1: 选取第一个点x_1,并构建第一个簇的维度概率摘要DS_1.

Step 2: 从数据集中读取下一个数据点x,采用式(2)计算该点到各簇的点簇相似度SimD_i,如果$\mathrm{SimD}_i > \varepsilon$,则表示该点可能属于簇$C_i$,转入Step 3;否则,转入Step 4.

Step 3: 在所有符合条件的簇$\mathrm{SimD}_i (i = 1, 2, \cdots, k)$中,取相似度最大的一个簇$C_j$作为该点所属簇,这里$j = \mathrm{argmax}(\mathrm{SimD}_i)$. 将该点加入簇$C_j$中,同时根据指定条件更新$C_j$的簇维度摘要$\mathrm{DS}_j$. 如果数据集中还有数据,则转入Step 2;否则,输出聚类结果,算法结束.

Step 4: 以当前数据点x为中心构建一个新的簇的维度概率摘要DS_{k+1},并生成一个新的簇C_{k+1},将其加入到聚类划分C中.

Step 5: 重复Step 2 ~ Step 4,直至所有数据点都被处理,输出聚类结果.

点簇相似度阈值ε代表了DPC算法聚类后的聚簇内聚度. ε越大,生成的初始聚簇数越多. 当$\varepsilon = 1$时,相当于将每个点都作为一个新簇,后期层次聚类等同于传统层次聚类算法;当$\varepsilon = 0$时,算法会将所有点归为一类. 有研究人员发现,一个有n个数据点的数据集的聚簇数最大不超过$\sqrt{n}$个,又有人提出聚簇数取$\sqrt{n/2}$左右为宜. 从实验中可以得知,当ε取0.5 ~ 0.6时,DPC算法的初次聚类聚簇数在$\sqrt{n}$左右,最小不小于$\sqrt{n/2}$,为简单起见,本文取$\varepsilon = 0.5$.

2.3　DPC算法关键问题

2.3.1　单数据点簇的维度概率摘要

DPC算法需要解决的一个问题是新建一个簇时维度概率摘要的构建,当簇中只有一个数据点时,按常规来说每个维度的概率分布中的标准差为$\sigma = 0$,

这将导致点簇维度概率相似度无法计算.

DPC算法基于以下假设: 当一个数据集中的n个数据点被分为k个簇时, 聚类后的数据集在某一维度上的取值也被分为k个服从某种正态分布的子集$\{(\mu_1,\sigma_1),(\mu_2,\sigma_2),\cdots,(\mu_k,\sigma_k)\}$, 子集的标准差值$\sigma_i(i=1,\cdots,k)$应该小于或等于所有数据点在该维度上的标准差$\sigma$, 即$\sigma_i<\sigma$. 因此, 在簇$C_i$只有一个数据点时, 以该数据点维度的值作为该簇的维度概率分布的均值μ_i, 以整个数据集在该维的标准差的$1/m$作为该簇的维度概率分布的标准差$\sigma_i=\sigma/m$, 其中m可取值为1到$\sqrt{n}$($\sqrt{n}$为一个有n个数据点的数据集有可能含有的最大簇数).

从式(1)中可知, m取值越大, 每个初始簇的σ_i值越小, 在距离dist_j相同的情况下, SimPN就越小, 此时就越容易产生新的簇, 即聚类算法所产生的簇越多. 根据实验测试, 取$m=\sqrt{n/2}$左右时效果较佳, 产生的紧凑的小簇经过后续的层次聚类进行凝聚后可达到较好的聚类结果.

2.3.2 簇的维度概率摘要的更新

在上述算法的Step 3中, 需要更新簇C_j维度概率摘要$\text{DS}_j=\{\text{DN}_1,\text{DN}_2,\cdots,\text{DN}_d\}$, 可以将簇中的所有数据点按维度计算其均值和标准差作为该簇的维度概率分布.

摘要更新可以采用即时更新和批量更新两种. 为了优化算法性能, 可以采用批量更新的方式, 即设定一个更新触发阈值σ_{tj}(第j维的更新阈值, $j=1,2,\cdots,d$), 当一个新的数据点加入时, 如果该数据点与欲加入的簇的维度概率分布均值的距离大于该阈值(即$\text{dist}_j>\sigma_{tj}$), 则运行更新; 否则, 直接将数据点加入而不更新. 该阈值应保证在新增数据点时, 所计算出来的新的维度概率分布的标准差不大于原有簇的维度概率分布的标准差. 根据概率计算的推导, 设某簇的第j维的维度分布为(μ_j,σ_j), 该簇中原有M个数据点, 则当新增一个数据点与该簇维度距离满足以下式所示条件时, 可保证新增数据点后形成的新簇的维度标准差σ不大于原来的维度标准差σ_j.

$$\text{dist}_j\leqslant\sqrt{\frac{M+1}{M}}\sigma_j. \tag{5}$$

由此可知, σ_{tj}可设置为式(5)所示条件右侧的值作为批量更新的触发条件. 由于该值与M有关且动态变化, M最小值取1, 此时$\sigma_{tj}\geqslant 1.414\times\sigma_j$. 为简化处理, 设定该阈值为一个固定值$\sigma_{tj}=1.5\times\sigma_j$.

2.4 DPC的凝聚层次聚类

DPC算法的聚类结果为一些数据点较少的小簇, 且由于m的取值较大将产生比预期多的簇, 因此可以采用AGNES(AGglomerative NESting)凝聚层次聚类算法[11]思想将DPC算法中产生的簇进行合并, 综合算法称为DPCAH算法. 该算法首先采用DPC算法进行一次聚类, 然后使用上述定义的簇簇相似度作为相似性度量, 一步一步地合并相似度较大的簇, 直到达到预期目标.

DPCAH算法的具体流程如下:

Step1 : 采用DPC算法对数据集进行初次聚类, 生成第一阶段的候选簇;

Step 2: 基于凝聚层次聚类的思想, 采用定义6中定义的簇簇相似度, 计算各候选簇的相似度, 找出两个相似度最大且符合合并条件的簇进行合并, 并更新相应维度概率摘要;

Step 3: 在合并后的簇集重复Step 2, 直至得到符合需求的结果, 如获得预期的簇数或者所有簇的簇簇相似度均小于簇合并的阈值t.

3 实验分析

为了验证本文方法的有效性, 采用2个人工数据集和UCI的IRIS数据集进行实验分析. 人工数据集采用来自Alexander Strehl在文献[12]中用到的8个维度5个簇的X8d5k和用于高维聚类研究的Dim064数据集. 数据集的基本情况如表1所示.

表 1 实验数据集概况

数据集	维度数	数据点数	簇数目	数据集特征
X8d5k	8	1 000	5	8维度5类高斯分布数据, 每类200个点
IRIS	4	150	3	UCI的真实数据集
Dim064	64	1 024	16	用于高维聚类研究的著名数据集

实验环境: 硬件采用Intel core i5-5200u型号CPU和1.52 G内存; 软件为virtualbox虚拟机和Windows XP系统, 在Matlab 2012a中实现算法并运行比较.

3.1 有效性实验

将K-means和AGNES层次聚类算法与本文所提出的DPCAH算法进行比较. 在上述数据集上分别运行上述算法进行聚类分析, 并分别计算错误率和2个常见内部有效性指标Davies-Bouldin(DB)、Calinski-Harabasz(CH)以及4个外部有效性指标Adjusted Rand index(AI)、Rand index(RI)、Mirkin index(MI)、Hubert index(HI)以验证算法的有效性. 由于K-means算法稳定性不高, 采用运行100遍取平均值的方式计算错误率等指标, DPCAH和AGNES算法每次结果是确定的, 所以只需运行一次. DPCAH的初始参数为$m=\sqrt{n/2}$, 合并阈值$t=0$. 三种算法在3个数据集上的实验结果如表2所示.

表 2　有效性实验结果

数据集	算法	错误率/%	AI	RI	MI	HI	DB	CH
X8d5k	**DPCAH**	**1.00**	**0.985**	**0.995 2**	**0.004 8**	**0.990 5**	0.691 4	**993.325 6**
	K-means	12.22	0.802 8	0.928 7	0.071 3	0.857 3	1.020 8	917.691 1
	AGNES	19.90	0.781 6	0.919 9	0.080 1	0.839 8	**0.610 7**	680.962 2
IRIS	**DPCAH**	**12.00**	**0.706**	**0.867 9**	**0.132 1**	**0.735 8**	0.603 7	480.720 1
	K-means	17.27	0.669 1	0.847 3	0.152 7	0.694 6	0.613 9	**506.164 3**
	AGNES	32.00	0.563 8	0.776 6	0.223 4	0.553 3	**0.430 4**	277.492 7
Dim064	**DPCAH**	**0.00**	**1**	**1**	**0**	**1**	**0.048 2**	**544 51**
	K-means	77.57	0.758	0.963 8	0.036 2	0.927 5	0.751	299.225 3
	AGNES	0.00	1	1	0	1	0.049 1	544 51

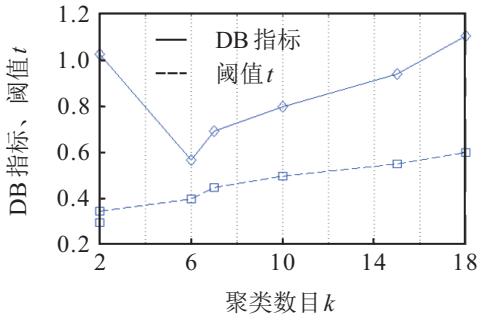
从表2的结果可以看出: 对于所有外部有效性指标, 本文提出的DPCAH算法均优于另外两种比较算法, 聚类结果的错误率远低于其他两种算法; 在DB指标表现上, 均优于K-means算法; CH指标在IRIS数据集上比K-means略差, 在2个人工数据集上表现较好, 特别在高维数据集Dim064上表现非常好. 由此可以验证本文算法的有效性.

3.2　自动确定簇数目实验

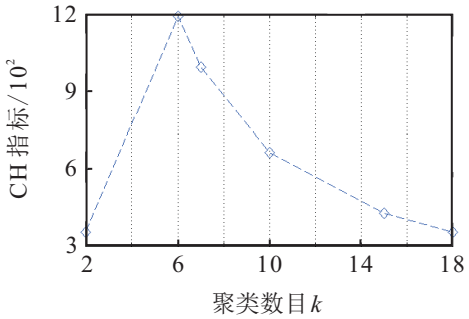
为了自动确定簇数目k, 在很多传统算法上, 经常使用内部有效性指标(如DB和CH指标)进行指导. Squeezer算法可以通过调节相似度参数来自动确定聚类数目. 层次聚类算法中, 在簇簇合并时, 每次均找出最相似的两个簇进行合并, 但这里的最相似只是相对而言的, 如果通过相似度度量, 最相似的两个簇真实距离也很大, 则应考虑停止继续合并簇, 此时便可自动获得簇数目. 由于DPCAH算法定义了簇簇相似度, 可以考虑设定一个簇合并的阈值t, 当最相似的两个簇的距离超过t时, 认为其距离太大, 不适合合并到一个簇中, 此时可停止继续聚类而直接输出结果.

为了验证t值对算法聚类结果的影响, 采用DB和CH指标进行指导, 在X8d5k数据集上进行实验分析. t取值从0.6到0, 每次递减0.05. 阈值t的取值、相应的聚簇数k、DB和CH值的相互关系如图3所示.

根据上述实验结果, 在层次聚类过程中, 如果在设定簇簇合并阈值$t = 0.4$时, 采用DPCAH算法进行聚类, 则用户指定的聚类数目在小于等于6的情况下, 最终聚类结果都会变成6个簇. 此时的DB和CH指标数都达到最优, 但与X8d5k实际含有5个簇每个簇200个数据点的预设不符. 进一步分析发现, 采用DPCAH算法聚类的6个簇所含数据点数分别为: 198、201、200、201、199、1. 也就是说, 6个簇中有一个簇中只有1个数据点(第23个数据点), 通过簇簇相似度计算出来其与其他5个类的相似度都小于0.4, 由此可将该数据点当做离群点来处理, 这表明DPCAH算法能够用来进行离群点分析.



(a) DB指标、阈值t与簇数目k的关系



(b) CH指标与簇数目k的关系

图 3　自动确定簇数目k实验结果

3.3　m值参数调节实验

在DPC算法运行采用单数据点构建新的簇的维度概率摘要时, m的取值将影响初始簇的个数和最终的聚类效果, 为研究m值对算法的影响, 同样采用DB和CH指标作为指导, 在X8d5k数据集上进行实验分析. 为了方便比较效果, 簇合并阈值t取0.4.

X8d5k数据集有数据点个数$n = 1\,000$, 实验时, m取值从2到$\sqrt{n}$, 每次递增1, 结果如图4所示.

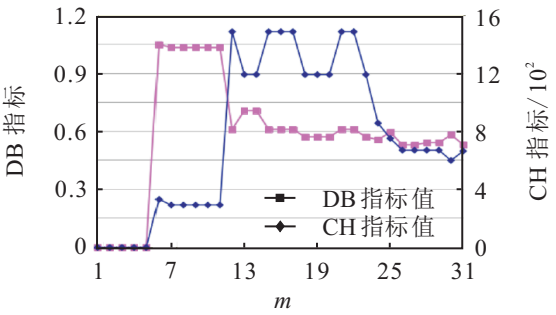


图 4　m值与聚类指标关系

从图4可以看出,两个指标对m值变化的影响有所不同;从CH指标来看,m在[12, 23]范围时,CH值在1 189.35及以上,效果较佳,在$m = 12$、15、16、17、21、22时达到最大值1 484.61;从DB指标看,m取12和大于等于15时,DB值在0.608 2及以下. 从以上实验结果可以看到,针对本数据集,m取值在[15, 23]区间中的任何一个值,且采用簇簇合并阈值$t = 0.4$时,其聚类效果都要优于另外两类比较算法.

3.4 算法效率分析

从时间复杂度来看,DPC算法属于一趟聚类算法,其主要计算量来自于点簇相似度的计算,算法复杂度介于$O(n \times d)$和$O(n \times d \times k)$之间. 其中:n是数据点个数;d是数据维度数;k是一趟聚类的簇数,它随着循环的进行逐渐增加,其中每趟中还有每个簇的维度概率摘要更新的计算. 在凝聚层次聚类阶段,DPCAH算法的主要计算量来自于相似度矩阵的计算,设一阶段基础簇数为k,则计算k个簇之间的相似度需要$2 \times k \times (k-1)$次运算量,由于k随着循环逐渐减小,无法准确估计其计算复杂度. 为了进一步精确地比较DPCAH算法与K-means和AGNES算法的计算复杂度,将3种算法在X8d5k数据集各运行100次统计其运行时间,结果为:DPCAH运行31.512 4 s,K-means运行0.810 6 s,AGNES运行5.308 4 s.

从运行时间来看,DPCAH算法的时间复杂度偏高,这是因为算法需要进行按维度的概率信息计算,同时,Matlab实现的算法也没有进行过优化. 总体来说,其算法时间是传统层次聚类的6倍左右且没有指数级的复杂度,说明算法还是可行和适用的.

4 结 论

本文提出了一种全新的维度概率摘要模型及其层次聚类算法,对算法的阈值ε、单个数据点新建簇时的标准差确定参数m和DPCAH算法中的合并阈值t等进行了初步的数学分析和实验研究. 同时采用UCI的实际数据集进行聚类实验,验证了所提模型和算法的有效性,与传统的K-means算法和层次聚类算法相比,聚类正确率更高,且能够自动确定聚簇数目和发现离群点.

当然,算法中的参数调优、单数据点构建簇的维度概率分布的处理方式,以及如何将算法应用于高维数据聚类及其他数据挖掘的应用场合都有待进一步的研究探讨.

参考文献(References)

[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.
(Sun J G, Liu J, Zhao L Y. Clustering algorithms research[J]. J of Software, 2008, 19(1): 48-61.)

[2] Jain A K. Data clustering: 50 years beyond K-means[J]. Pattern Recognition Letters, 2010, 31(8): 651-666.

[3] 王骏, 王士同, 邓赵红. 聚类分析研究中的若干问题[J]. 控制与决策, 2012, 27(3): 321-328.
(Wang J, Wang S T, Deng Z H. Survey on challenges in clustering analysis research[J]. Control and Decision, 2012, 27(3): 321-328.)

[4] Tan P N, Steinbach M, Kumar V. Introduction to data mining[M]. Beijing: China Machine Press, 2010: 487-490.

[5] Han J W, Kamber M, Pei J. Data mining: Concepts and techniques[M]. Morgan Kaufmann, 2006: 444-445.

[6] 王玲, 孙华, 基于自适应学习的演化聚类算法[J]. 控制与决策, 2016, 31(3): 423-428.
(Wang L, Sun H. Evolving clustering method based on self-adaptive learning[J]. Control and Decision, 2016, 31(3): 423-428.)

[7] 罗印升, 李人厚, 张维玺. 一种基于克隆选择的聚类算法[J]. 控制与决策, 2005, 20 (11): 1261-1264.
(Luo Y S, Li R H, Zhang W X. Cluster ing A lgor ithm Based on Clone Selection Theory[J]. Control and Decision, 2005, 20(11): 1261-1264.)

[8] 夏卓群, 欧慧, 李平, 等. 基于改进流形距离和人工蜂群的二阶段聚类算法[J]. 控制与决策, 2016, 31(3): 410-416.
(Xia Z Q, Ou H, Li P, et al. Two-phase clustering algorithm based on the improved manifold distance and the artificial bee colony algorithm[J]. Control and Decision, 2016, 31(3): 410-416.)

[9] Woo K G, Lee J H. FINDIT: A fast and intelligent subspace clustering algorithm using dimension voting[J]. Information and Software Technology, 2004, 46(4): 255-271.

[10] He Z Y, Xu X F, Deng S C. Squeezer: An efficient algorithm for clustering categorical data[J]. J of Computer Science & Technology, 2002, 17(5): 611-624.

[11] 黄德才. 数据仓库与数据挖掘教程[M]. 北京: 清华大学出版社, 2016: 307-309.
(Huang D C. Data warehouse and data mining[M]. Beijing: Tsinghua University Press, 2016: 307-309.)

[12] Alexander S, Joydeep G. Cluster ensembles — A knowledge reuse framework for combining multiple partitions[J]. J on Machine Learning Research, 2002, 3: 583-617.

(责任编辑:齐 霁)