文章编号: 1001-0920(2017)05-0913-07 DOI: 10.13195/j.kzyjc.2016.0219

一种基于密度和网格的簇心可确定聚类算法

何熊熊†, 管俊轶, 叶宣佐, 詹亦钊

(浙江工业大学 信息工程学院，杭州 310023)

摘 要: 以网格化数据集来减少聚类过程中的计算复杂度,提出一种基于密度和网格的簇心可确定聚类算法. 首先网格化数据集空间,以落在单位网格对象里的数据点数表示该网格对象的密度值,以该网格到更高密度网格对象的最近距离作为该网格的距离值;然后根据簇心网格对象同时拥有较高的密度和较大的距离值的特征,确定簇心网格对象,再通过一种基于密度的划分方式完成聚类;最后,在多个数据集上对所提出算法与一些现有聚类算法进行聚类准确性与执行时间的对比实验,验证了所提出算法具有较高的聚类准确性和较快的执行速度.

关键词: 数据挖掘；数据聚类；网格；密度

中图分类号: TP18 **文献标志码:** A

A density-based and grid-based cluster centers determination clustering algorithm

HE Xiong-xiong†, GUAN Jun-yi, YE Xuan-zuo, ZHAN Yi-zhao

(College of Information Engineering，Zhejiang University of Technology，Hangzhou 310023，China)

Abstract: A density and grid based cluster centers determination clustering algorithm is proposed. The computational complexity of the clustering process is reduced by using the gridding dataset. Firstly, the dataset space is divided into grids with the same size, and the number of data objects that are contained in grid is defined as the value of the grid density. The nearest distance from one grid to another with higher density is defined as the value of grid distance. The cluster center grids can be found since these grids always have high density value and large distance value. Then, a density-based division approach is used to accomplish the task of clustering. Finally, a comprehensive comparison is carried out to examine the clustering accuracy and execution time between the proposed clustering algorithm and some classical algorithms. Experiment results show that the proposed algorithm can lead to a higher accuracy with less execution time.

Keywords: data mining；data clustering；grid；density

0 引 言

数据挖掘(DM)[1]是指从大量数据中发现未知的、有价值的模式或规律等知识的复杂过程. 聚类分析[2]作为数字挖掘技术中一个重要组成部分,它的目的是将物理对象或抽象对象的集合中相似的对象聚在同一个簇中. 通俗地讲,簇就是相似对象的集合,同簇中的对象相似度较高,不同簇之间的对象相似度较低. 聚类分析是一种常见的数据分析工具,在统计学、机器学习、数据挖掘、生物学、市场营销等领域都有着广泛的应用前景[3-4]. 最早的聚类算法是由MacQueen在1967年提出的K-means[5]聚类算法,经过数十年国内外专家学者的深入研究,在聚类算法上已取得了相当丰硕的成果[6]. 如基于划分的聚类算法[7]有K-means、K-medoids[8]、CLARANS[9]等;基于密度的聚类算法有DBSCAN[10]、GDBSCAN[11]、OPTICS[12]等;基于层次的聚类算法有CURE[13]、CHAMELEON[14]、BIR-CH[15]等;基于网格的聚类算法有STING[16]、Wave-Cluster[17]、CLIQUE[18]等. 然而传统的聚类算法都存在着各自的不足,如K-means、K-medoids、Fuzzy K-means、Spectral Clustering[19-20]聚类算法,无法确定簇心个数;K-means、K-medoids、Fuzzy K-means[21]、BIRCH聚类算法不能用来处理任意形状的数据集;尽管DBSCAN、Spectral Clustering聚类算法适用于任意形状的数据集,但它们的聚类质

收稿日期: 2016-02-29；**修回日期:** 2016-07-05.

基金项目: 国家自然科学基金项目(61473262).

作者简介: 何熊熊(1965−), 男, 教授, 博士生导师, 从事重复学习控制、网络控制系统等研究；管俊轶(1991−), 男, 硕士生, 从事数据挖掘的研究.

†通讯作者. E-mail: hxx@zjut.edu.cn

量过于依赖参数的设定;而基于网格的聚类算法如STING、CLIQUE常被用作处理大规模的数据集,可减少计算复杂度,但会降低聚类质量.

Rodriguez等[22]在Science期刊上提出了一种可以处理任何形状数据集的算法.该算法假设聚类中心具有较高的密度ρ,且它与更高密度点具有较大的距离δ.与传统聚类算法相比,该算法具有较高的聚类质量,但仍存在着下列问题:1)数据对象的ρ、δ两值依赖于截断距离参数dc的取值;2)算法需要计算所有数据对象之间的两两距离,计算量过于庞大.

针对上述问题,本文提出一种基于密度和网格的簇心可确定聚类算法(DGCCD).该算法首先网格化数据集空间,形成网格对象集;其次,分别定义网格对象的密度值ρ和距离值δ;再次,将同时具有较高密度值且较大距离值的网格对象选作聚类中心;然后根据一种基于密度的划分方法,完成网格对象集的聚类;最后对每个数据对象的类标进行划分,完成数据集的聚类.

1 DGCCD算法的具体步骤

DGCCD算法需要将数据空间的每一维都均匀划分成相同的段数,记为f_G;形成若干个等大的网格对象,剔除网格对象中数据对象个数为0的网格对象,剩余网格对象为网格对象集,记为G;网格对象集中网格对象数记为N_G.实验结果表明,网格对象集中网格对象数量N_G在大于或等于数据集中数据量n的$1/6$的情况下,算法具有较好的聚类质量.DGCCD算法的具体步骤如下.

Step 1:将含有n个数据对象的数据集空间中每一维均匀划分成相同段数,记为f(赋予f初始值2),形成网格;

Step 2:剔除网格对象中数据对象数量为0的网格,剩余非空网格对象数记为N;

Step 3:如果$N < n/6$,则令$f = f+1$,返回Step 1;

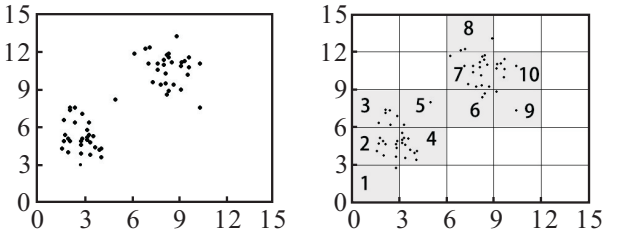
Step 4:确定网格对象集G,划分段数f_G,网格对象数N_G.

本文将以上过程称为DGCCD算法的网格化过程.

针对样本数据集D_1,其二维分布如图1(a)所示.将数据集D_1通过上述DGCCD算法网格化后的网格对象集分布如图1(b)所示.

网格对象集一旦确定,本文接下来需通过计算求得网格对象的密度值ρ和距离值δ,定义分别如下.

定义1 以落在网格对象i中数据对象的数量作为网格对象密度值,记为ρ_i.



(a) D_1的二维分布 (b) D_1的网格对象集分布

图1 D_1数据集的二维分布及其网格对象集的二维分布

定义2 以网格对象i到更高密度网格对象j的最近距离作为网格对象的距离值,记为δ_i,定义如下:

$$\delta_i = \min_{j:\rho_j>\rho_i}(d_{ij}), \tag{1}$$

其中d_{ij}为网格对象i中心位置到网格对象j中心位置之间的欧氏距离.

令网格对象集中最高密度的网格对象为i_m,其距离值$\delta_{i_m} = \max(\delta_j)$,其中$j$为除$i_m$以外的所有网格对象.

在Rodriguez和Laio在Science期刊上提出的聚类中心具有较高的密度ρ且与更高密度点具有较大的距离δ的相同假设下,DGCCD算法网格化后的网格对象集中,处在簇心位置的网格对象会同时具有较高的密度ρ和较大距离值δ.如图2所示,将编号为“4”、“7”的网格对象作为簇心网格对象,它们都同时具有较高的密度ρ和较大的距离δ.在确定网格对象的密度值ρ和距离值δ之后,作出网格对象集对应的ρ和δ分布图.然后根据处在簇心位置的网格对象会同时具有较高的密度ρ和较大距离值δ的特征,可通过人为监督的方式选取出簇心网格对象.

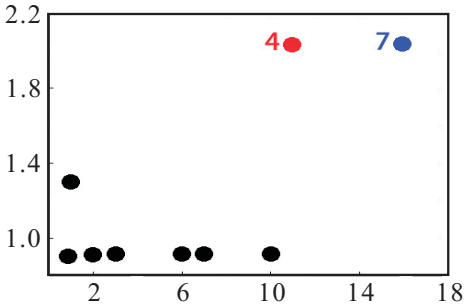


图2 D_1的网格对象集的ρ和δ分布

在簇心网格对象确定之后,首先给每个簇心网格对象赋予不同的类标,然后采用一种基于密度的划分方式,即每个网格对象的类标跟随它的最近且密度高于它的网格对象类标,对剩余网格对象进行划分,完成网格对象集的聚类.网格对象集聚类完成后,每个数据对象类标只需跟随其落入的网格对象类标进行划分,从而完成整个数据集的聚类.

图3为网格对象划分的图示,网格中的数字表示网格对象的密度值.设网格“8”类标为A,对于网格

“6”而言,网格“8”是与它最近且密度值大于它的网格对象,故网格“6”的类标则跟随网格“8”变为A. 对于网格“2”和网格“5”,网格“6”是距离它们最近且密度值大于它们的网格对象,故类标跟随网格“6”变为A.

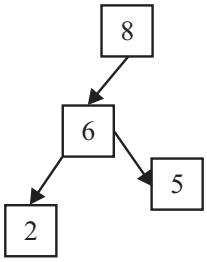
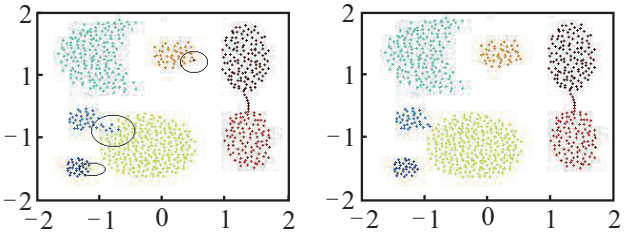


图 3　网格对象的划分规则

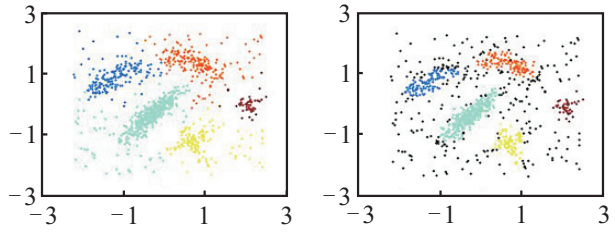
算法完成划分后,首先选取边缘的网格对象,计算边缘网格对象中的每个数据点与所有非边缘数据点彼此间的距离,以距离该数据点最近的非边缘数据点的类标作为该边缘数据点的类标,从而完成边缘数据点的划分. DataSet1经算法边缘处理前后的对比如图4所示. 图4(a)由于网格化,在类与类之间的边缘划分较为粗糙,经过本文算法的边缘处理,边缘粗糙的问题基本得到解决,如图4(b)所示.



(a) 边缘处理前　(b) 边缘处理后

图 4　边缘处理前后DataSet 1的聚类结果

噪声点的处理不需要人为设定噪声点阈值截断,而是通过先寻找出类与类之间的边界网格对象,然后找出边界网格对象中密度最高的网格对象的密度作为阈值,记为ρ_b,只需保留类中大于或等于ρ_b的网格对象,而低于ρ_b的网格对象即为噪声点. 图5为样本数据集未去噪和去噪后的图示,其中黑点为噪声点.



(a) 去噪前　(b) 去噪后

图 5　噪声处理前后样本数据集的聚类结果

2　仿真实验与性能分析

实验所用操作系统为Windows 7, 仿真软件Matlab 7.12.0(R2011a)(64位), 处理器为Intel(R)Core(TM)i5, 安装内存4.00 GB.

2.1　聚类性能对比

测试DGCCD算法聚类性能的实验数据集为DataSet1、DataSet2、DataSet3、DataSet4、DataSet5、DataSet6和DataSet7(Iris数据集),数据集的具体信息如表1所示. 采用的聚类准确率r为正确分类数据量占总数据量的百分比. 实验中将每个算法在每一个数据集上进行10次聚类运算,求得平均聚类准确度r_a和平均算法执行时间t_a,提取各算法10次测试中最高聚类准确度$r_{\max}$、最低聚类准确度$r_{\min}$和10次聚类准确度的方差σ^2,用作分析聚类结果的稳定性.

表 1　7个数据集的基本属性

聚类数据集	数据集属性		
	数据集维数	类属性数	数据量
DataSet 1	2	7	788
DataSet 2	2	31	3 100
DataSet 3	2	20	1 000
DataSet 4	2	15	600
DataSet 5	2	5	2 000
DataSet 6	2	2	1 829
DataSet 7	4	3	150

DataSet1～DataSet6数据集通过DGCCD算法聚类后的二维分布如图6所示. DataSet1～DataSet6数据集网格化后网格对象集对应的ρ和δ分布如图7所示,DataSet7数据集网格化后网格对象对应的ρ和δ分布如图8所示,实验中选取密度值和距离值都较大的网格对象作为簇心网格对象. 7个数据集网格化后网格对象集对应的N和f_G,如表2所示.

表 2　7个数据集网格化后的基本属性

聚类数据集	网格对象集属性	
	最终划分段数f_G	网格对象数N
DataSet 1	14	134
DataSet 2	31	540
DataSet 3	19	171
DataSet 4	22	106
DataSet 5	27	345
DataSet 6	23	324
DataSet 7	11	116

本文以K-means、Fuzzy K-means、DBSCA-N、Spectral Clustering (SC)、Rodriguez等在Science提出的算法(下文用Rodriguez-Clustering表示)和DGCCD

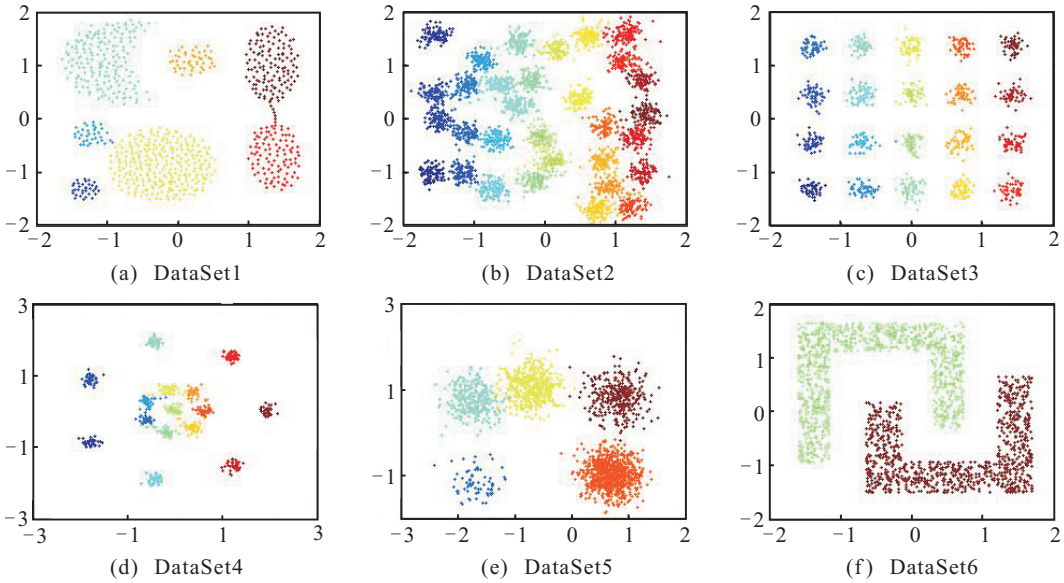


图 6 6个数据集的二维分布以及DGCCD算法的聚类结果

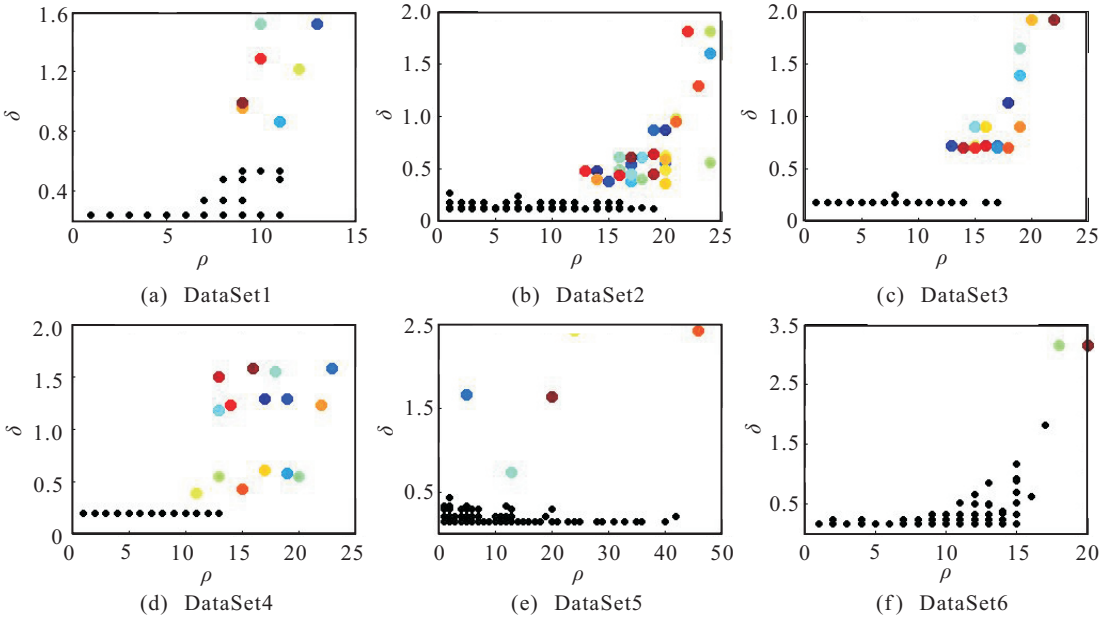


图 7 6个数据集网络化后的网格对象对应的ρ和δ分布

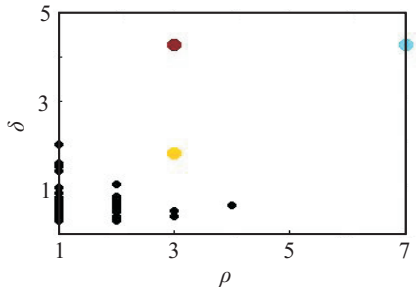


图 8 DataSet的网格对象集的ρ和δ分布

算法,在以上给出的7个数据集上的聚类性能进行对比实验,实验结果分别如表3~表9所示.

2.2 与Rodriguez-Clustering的算法执行时间对比

由表3~表9的实验结果可知,DGCCD算法与Rodriguez-Clustering算法的聚类准确度较为接近,且相比其他算法有着较高的聚类准确性,并都具有绝对的聚类稳定性. 表10为DGCCD算法与Rodriguez-Clustering算法在7个数据集上的平均执行时间的对比表. 由此可见,DGCCD算法在7个数据集上的执行时间都小于Rodriguez-Clustering算法,其中DataSet2、 DataSet5和DataSet6由于数据量相对较大,Rodriguez-Clustering算法的执行时间大幅度上升,而DGCCD算法的执行时间小于Rodriguez-Clustering算法. 由此可得,DGCCD算法的执行速度快于Rodriguez-Clustering算法.

表 3　6种算法在DataSet1数据集上的实验结果

聚类算法	DataSet1				
	平均聚类准确度r_a	最高准确度$r_{\max}$	最低准确度$r_{\min}$	聚类准确度方差σ^2	平均执行时间t_a/s
K-means	0.784 4	0.868 0	0.725 9	0.003 3	0.016 0
DBSCAN(Eps = 0.134 8)	0.824 9	0.824 9	0.824 9	0	0.058 3
Fuzzy K-means	0.723 6	0.799 5	0.588 8	0.006 8	0.214 5
SC(a = 1)	0.917 5	0.996 2	0.680 2	0.011 9	0.400 0
Rodriguez-Clustering	0.998 7	0.998 7	0.998 7	0	3.507 9
DGCCD	0.986 0	0.986 0	0.986 0	0	0.169 4

表 4　6种算法在DataSet2数据集上的实验结果

聚类算法	DataSet2				
	平均聚类准确度r_a	最高准确度$r_{\max}$	最低准确度$r_{\min}$	聚类准确度方差σ^2	平均执行时间t_a/s
K-means	0.809 7	0.889 7	0.718 1	0.002 5	0.086 6
DBSCAN(Eps = 0.076 0)	0.464 5	0.464 5	0.464 5	0	0.461 4
Fuzzy K-means	0.850 3	0.908 7	0.726 5	0.002 9	1.413 3
SC(a = 1)	0.762 9	0.819 4	0.684 5	0.002 0	9.722 4
Rodriguez-Clustering	0.967 4	0.967 4	0.967 4	0	179.584 4
DGCCD	0.941 3	0.941 3	0.941 3	0	2.278 3

表 5　6种算法在DataSet3数据集上的实验结果

聚类算法	DataSet3				
	平均聚类准确度r_a	最高准确度$r_{\max}$	最低准确度$r_{\min}$	聚类准确度方差σ^2	平均执行时间t_a/s
K-means	0.808 4	0.913 0	0.689 0	0.006 0	0.023 9
DBSCAN(Eps = 0.119 1)	0.990 0	0.990 0	0.990 0	0	0.101 9
Fuzzy K-means	0.890 4	0.926 0	0.787 0	0.002 3	0.227 4
SC(a = 1)	0.687 5	0.810 0	0.528 0	0.006 8	0.716 7
Rodriguez-Clustering	1.000 0	1.000 0	1.000 0	0	6.339 9
DGCCD	0.999 0	0.999 0	0.999 0	0	0.237 3

表 6　6种算法在DataSet4数据集上的实验结果

聚类算法	DataSet4				
	平均聚类准确度r_a	最高准确度$r_{\max}$	最底准确度$r_{\min}$	聚类准确度方差σ^2	平均执行时间t_a/s
K-means	0.815 3	0.966 7	0.700 0	0.006 8	0.015 5
DBSCAN(Eps = 0.097 2)	0.945 0	0.945 0	0.945 0	0	0.053 0
Fuzzy K-means	0.996 7	0.996 7	0.996 7	0	0.140 4
SC(a = 1)	0.671 3	0.826 7	0.521 7	0.007 4	0.272 0
Rodriguez-Clustering	0.996 7	0.996 7	0.996 7	0	1.327 0
DGCCD	0.966 7	0.966 7	0.966 7	0	0.176 0

表 7　6种算法在DataSet5数据集上的实验结果

聚类算法	DataSet5				
	平均聚类准确度r_a	最高准确度$r_{\max}$	最低准确度$r_{\min}$	聚类准确度方差σ^2	平均执行时间t_a/s
K-means	0.752 8	0.995 0	0.630 5	0.022 8	0.021 0
DBSCAN(Eps = 0.100 2)	0.797 0	0.797 0	0.797 0	0	0.227 9
Fuzzy K-means	0.756 0	0.756 0	0.756 0	0	0.278 7
SC(a = 1)	0.890 2	0.990 5	0.690 5	0.010 8	3.172 1
Rodriguez-Clustering	0.993 0	0.993 0	0.993 0	0	79.441 5
DGCCD	0.992 0	0.992 0	0.992 0	0	1.225 9

表 8 6种算法在DataSet6数据集上的实验结果

聚类算法	DataSet6				
	平均聚类准确度r_a	最高准确度$r_{\max}$	最低准确度$r_{\min}$	聚类准确度方差σ^2	平均执行时间t_a/s
K-means	0.888 5	0.888 5	0.888 5	0	0.012 6
DBSCAN(Eps = 0.119 7)	1.000 0	1.000 0	1.000 0	0	0.184 3
Fuzzy K-means	0.890 7	0.890 7	0.890 7	0	0.089 7
SC($a=1$)	1.000 0	1.000 0	1.000 0	0	1.514 7
Rodriguez-Clustering	1.000 0	1.000 0	1.000 0	0	38.305 1
DGCCD	1.0000	1.000 0	1.000 0	0	0.209 2

表 9 6种算法在DataSet7数据集上的实验结果

聚类算法	DataSet7				
	平均聚类准确度r_a	最高准确度$r_{\max}$	最低准确度$r_{\min}$	聚类准确度方差σ^2	平均执行时间t_a/s
K-means	0.781 1	0.825 1	0.564 3	0.002 4	0.018 9
DBSCAN(Eps = 1.081 4)	0.770 0	0.770 0	0.770 0	0	0.073 8
Fuzzy K-means	0.811 0	0.900 0	0.690 0	0.005 8	0.048 4
SC($a=1$)	0.843 3	0.990 0	0.550 0	0.021 4	0.430 4
Rodriguez-Clustering	0.910 0	0.910 0	0.910 0	0	0.115 7
DGCCD	0.8530	0.852 0	0.853 0	0	0.097 9

表 10 Rodriguez-Clustering算法与DGCCD算法在7个数据集上的执行时间对比表果

数据集	平均执行时间/s		
	数据量	Rodriguez-Clustering	DGCCD
DataSet1	788	3.507 9	0.169 4
DataSet2	310 0	179.584 4	2.278 3
DataSet3	100 0	6.339 9	0.237 3
DataSet4	600	1.327 0	0.176 0
DataSet5	200 0	79.441 5	1.225 9
DataSet6	182 9	38.305 1	0.209 2
DataSet7	150	0.115 71	0.097 9

2.3 算法复杂度分析

假设聚类数据集有n个m维数据集,首先,算法网格化并获取网格对象的密度,该过程的计算代价为$O(mnf_G^2)$;然后计算每个网格对象的距离需要的计算代价为$O((N^2-N)/2)$,算法进行一次划分完成聚类,其计算代价为$O(N\log(N)+N/2)$;最后边缘处理的过程的计算代价为$O(bn)$,其中b表示边缘点的个数. 因此DGCCD算法的算法时间复杂度为$O((mnf_G^2)+(N^2-N)/2+N\log(N)+N/2+bn))$.

表11中列出了本文算法和对比算法的算法复杂度. 由表11可以分析得到:相比Rodriguez-Clustering和SC,本文算法的时间复杂度较低,所以相比前两个算法,本文算法有着较快的执行速度;而相比DBSCAN,K-means,Fuzzy K-means算法,本文算法的时间复杂度较高,其中主要消耗在获取每个网格对象距离值的过程中. 但本文算法的优点在于能确定簇心位置,以及在确定簇心网格后基于密度的划分方式,使得本文算法对于任意形状分布的数据集都有着较好的聚类结果,因此有着较高的聚类准确性.

表 11 算法时间复杂度对比表

聚类算法	时间复杂度
K-means	$O(tkn)$
DBSCAN	$O(n\log(n))$
Fuzzy$K$-means	$O(ndk^2t)$
SC	$O(n^3)$
Rodriguez-Clustering	$O(n^2+(n^2-n)/2+n\log(n)+n/2)$
DGCCD	$O((mnf_G^2)+(N^2-N)/2+N\log(N)+N/2+bn))$

2.4 实验结果分析

综上7个数据集的实验结果,可得Rodriguez-Clustering算法聚类准确度最高,但在处理稍大的数据集时(数据量在2 000以上),算法执行时间过长;K-means、Fuzzy K-means、DBSCAN、SC算法的聚类准确性都不如DGCCD算法.

实验结果表明:由以上6种算法针对7个数据集的测试结果可知,DGCCD算法与K-means、Fuzzy K-means、DBSCAN和SC算法相比,具有着较高的聚类准确度;与K-means、Fuzzy K-means、SC算法相比,具有较高的聚类稳定性;与Rodriguez-Clustering算法相

比,具有较快的执行速度.

3 结 论

本文提出了一种基于密度和网格的簇心可确定聚类算法. 该算法对数据集进行网格化,减少了聚类过程中的计算量, 加快了算法执行速度. 在算法网格化的过程中,通过对众多数据集的实验测试学习,设定了最终网格对象总数$N_G \geqslant n/6$. 在实验过程中,笔者发现对于处理一些数据类属性数较少且数据集形状复杂度不高的大规模数据集时,网格对象数量N_G在小于$n/6$的情况下也不会影响聚类准确性,能进一步减少算法的执行时间. 即在不影响聚类准确性的前提下, f_G值越小,算法执行速度越快. 因此,如何给定一个最为合适的f_G来优化数据集的网格化,提升算法的执行速度,将是笔者下一步的研究重点之一.

参考文献(References)

[1] Culler D, Estrin D, Srivastava M. Guest editors' introduction: Overview of sensor networks[J]. Computer, 2004, 37(8): 41-49.

[2] Wood A D, Stankovic J A. Denial of service in sensor networks[J]. Computer, 2002, 35(10): 54-62.

[3] Yang H. Data mining: Concepts and techniques[J]. San Francisco, 2001, 29(S1): 1-18.

[4] 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): 321-0328.
(Wang J, Wang S T, Deng Z H. Survey on challenges in clustering analysis research[J]. Control and Decision, 2012, 27(3): 321-328.)

[5] L loyd S. Least squares quantization in PCM[J]. IEEE Trans on Information, 1982, 28(2): 129-137.

[6] 覃艳, 王洪, 周全华. 数据挖掘中聚类算法的研究[J]. 网络安全技术与应用, 2014(1): 65-66.
(Qin Y, Wang H, Zhou Q H. The research of clustering algorithm in data mining[J]. Network Security Technology & Application, 2014(1): 65-66.)

[7] 吴杨, 王韬, 李进东. 基于密度的划分式聚类过程参数选择算法[J]. 控制与决策, 2016, 31(1): 21-29.
(Wu Y, Wang T, Li J D. Clustering parameters selection algorithm based on density for divisional clustering process[J]. Control and Decision, 2016, 31(1): 21-29.)

[8] 夏宁霞, 苏一丹, 覃希. 一种高效的K-medoids聚类算法[J]. 计算机应用研究, 2010, 27(12): 4517-4519.
(Xia N X, Su Y D, Qin X. Efficient K-medoids clustering algorithm[J]. Application Research of Computers, 2010, 27(12): 4517-4519.)

[9] Ng R T, Han J. CLARANS: A method for clustering objects for spatial data mining[J]. IEEE Trans on Knowledge & Data Engineering, 2002, 14(5): 1003-1016.

[10] Ester B M, Kriegel H P, Sander J, et al. A density based algorithm for discovering clusters in large spatial databases[C]. Proc of Int Conf on Knowledge Discovery and Data Mining. Protland, 1996.

[11] Sander J, Ester M, Kriegel H P, et al. Density-based clustering in spatial databases: The algorithm GDBSCAN and its applications[J]. Data Mining & Knowledge Discovery, 1998, 2(2): 169-194.

[12] Ankerst M, Breunig M M, Kriegel H P, et al. OPTICS: Ordering points to identify the clustering structure[J]. Stanford Research Inst Memo Stanford University, 1999, 28(2): 49-60.

[13] Guha S, Rastogi R, Shim K. CURE: An efficient clustering algorithm for large databases[C]. Proc of the ACM SIGMOD Int Conf on Management of Data. New York: ACM Press, 1998: 73-84.

[14] Karypis G, Han E H, Kumar V. CHAMELEON: A hierarchical clustering algorithm using dynamic Modeling[J]. IEEE Computer, 1999, 32(8): 68-75.

[15] Zhang T, Ramakrishnan R, Livny M. BIRCH: An efficient data clustering method for very large databases[J]. Acm Sigmod Record, 1996, 25(2): 103-114.

[16] Wang W, Yang J, Muntz R R. STING: A statistical information grid approach to spatial data mining[C]. Proc of the 23rd Int Conf on Very Large Data Bases. Morgan Kaufmann Publishers Inc, 1997: 186-195.

[17] Sheikholeslami G, Chatterjee S, Zhang A. WaveCluster: A multi-resolution clustering approach for very large spatial databases[C]. Int Conf on Very Large Data Bases. NewYork: Morgan Kaufmann Publishers Inc, 1998: 428-439.

[18] Agrawal R, Gehrke J E, Gunopulos D, et al. Automatic subspace clustering of high dimensional data for data mining applications[C]. Acm Sigmod Int Conf on Management of Data. Seattle: ACM Press, 1998: 95-105.

[19] Ng A Y, Jordan M I, Weiss Y. On spectral clustering: Analysis and an algorithm[J]. Proc of Advances in Neural Information Processing Systems, 2001, 14: 849-856.

[20] 周林, 平西建, 徐森, 等. 基于谱聚类的聚类集成算法[J]. 自动化学报, 2012, 38(8): 1335-1342.
(Zhou L, Ping X J, Xu S, et al. Cluster ensemble based on spectral clustering[J]. Acta Automatica Sinica, 2012, 38(38): 1335-1342.)

[21] Gasch A P, Eisen M B. Exploring the conditional coregulation of yeast gene expression through fuzzy K-means clustering[J]. Genome Biology, 2002, 3(11): 129-137.

[22] Rodriguez A, Laio A. Clustering by fast search and find of density peaks[J]. Science, 2014, 344(6191): 1492-1496.

(责任编辑: 孙艺红)