基于影响空间的稳健密度峰值聚类算法

陈春涛 陈优广

(华东师范大学计算机科学与软件工程学院 上海 200062)

摘 要 DP(Clustering by Fast Search and Find of Density Peaks)是一种新提出的基于局部密度和距离的聚类算法,具有能够发现任意形状的类簇、易于理解并且可以高效划分数据的优点。但是该算法无法处理单个类簇中同时存在多个密度峰值的情况,并且数据划分不稳定,容易导致连锁错误划分;当类簇间的密度差异较大时,其无法准确识别稀疏的类簇。为弥补以上不足,提出一种基于影响空间的稳健密度峰值聚类算法。该改进算法通过邻近数据计算局部密度,增强对小规模类簇的识别能力。为了提高数据划分的稳定性,引入了影响空间,并定义了一种新的对称关系,提出了一种新的分配策略。其通过计算目标数据与邻近数据的局部密度比值,并对影响空间进行加权,使算法能够处理具有多密度分布特征的数据。基于人工合成数据集和 UCI 数据集的模拟对比实验表明,提出的改进策略增强了算法对稀疏类簇的识别能力,提高了数据划分的稳定性,在 NMI 和 Acc 评价指标方面取得了较优的结果。

关键词 局部密度,聚类算法,影响空间,分配策略,稳定性

中图法分类号 TP391 **文献标识码** A **DOI** 10.11896/jsjkx.181001846

Influence Space Based Robust Fast Search and Density Peak Clustering Algorithm

CHEN Chun-tao CHEN You-guang

(College of Computer Science and Software Engineering, East China Normal University, Shanghai 200062, China)

Abstract DP (Clustering by Fast Search and Find of Density Peaks) is a novel clustering algorithm based on local density and distance between data, which has the advantages of being able to discover the clusters of arbitrary shapes, easy to accept logic principle and efficiently partitioning data into data sets. However, this algorithm has some disadvantages, for instance, it can't deal with the case when multiple density peaks co-exist in a single cluster, and it is characterized with the unstable class label allocation. At the same time, it can't accurately identify the clusters of sparse data when the density difference between clusters is big enough. In order to overcome the above deficiencies, this paper proposed a robust density peak clustering algorithm based on influence space. The proposed algorithm is capable of calculating local density by neighboring data and enhances the ability to recognize small-scale clusters. In order to improve the robustness of data partition, the data influence space is introduced, a new symmetric relationship is defined, and a new allocation strategy is proposed. By calculating the weighted density ratio of the target data to the adjacent data, the algorithm is able to process data sets with multiple density distribution features. The local density is obtained by calculating the density ratio of the target data to the neighbor data, and the ratio is weighted through using the size of the influence space. In order to verify the availability of the proposed algorithm, simulation experiments were conducted on synthetic data sets and UCI data sets, and NMI and Acc indexs were used to evaluate clustering algorithms. Simulation experiment results show that the proposed algorithm can reduce the influence of the cutoff distance on the algorithm and classify the data more stably. Compared with other algorithms, the proposed algorithm achieves better experimental results on both NMI and Acc indexs.

Keywords Local density, Clustering algorithm, Influence space, Allocation strategy, Robustness

1 引言

聚类是无监督机器学习的一种,其不需要先验知识,利用数据间的相似度将数据划分为不同的集合,称为类簇。同一个类簇内的数据拥有较高的相似度,不同类簇的数据拥有较低的相似度。聚类算法可以用于发现数据的隐含信息和识别噪声。经典的聚类算法主要分为 4 类:基于划分的算法、基于层次的算法、基于密度的算法和基于网格的算法,代表算法分别为 K-means[1],CHAMELEON[2],DBSCAN[3],STING[4]。对经典聚类算法的诸多改进方法主要集中在 3 个方面:1)提

到稿日期:2018-10-06 返修日期:2019-02-13

陈春涛(1993—),男,硕士生,主要研究方向为数据挖掘、模式识别;**陈优广**(1971—),男,博士,高级工程师,硕士生导师,主要研究方向为图像处理与模式识别,E-mail:ygchen@cc.ecnu.edu.cn(通信作者)。

高算法的运行速度[5-7]；2）减少超参数对算法的影响[8-9]；3）提高算法处理特殊数据的能力[10-11]。

文献[12]提出了一种新的基于密度与距离的 DP 聚类算法。该算法同时考虑了数据的局部密度与距离，通过局部密度与距离的关系确定类簇中心并高效划分数据。该算法可以发现任意形状的类簇并对数据进行高效的划分，适用于大规模数据集。DP 算法由于具有以上优点，得到了广泛的研究与应用。文献[13]利用该算法进行多文档摘要，选取文章中的重要语句。文献[14-15]提出针对数据流的实时环境的应用。文献[16]将该算法用于光谱数据的处理。但是该算法存在以下不足：1）无法处理具有多密度分布特征的数据；2）数据划分策略不稳定，DP 算法将数据样本划分到离它最近且密度比它大的数据所在的集合，该方式对数据分配准确性的要求高，容易导致连锁错误分配；3）人工选择类簇中心和截断距离 d_c，易受主观因素影响；4）无法处理同一个类簇中存在多个密度峰值的情况。针对 DP 算法的不足，学者们提出了大量的改进方法。Li 等提出基于相对距离和邻近数据的 CDP 算法[17]。该算法通过邻近数据计算局部密度，以减小 DP 算法对截断距离 d_c 的敏感度。并且提出相对距离，将样本数据与距离它最近的高密度和低密度数据的距离差值作为距离参数，以提高类簇中心选择的准确性。Ding 等提出基于切比雪夫不等式的类簇中心数据选择方法[18]。高诗莹等提出基于密度比例的 R-CFSFDP[19]算法，把不同参数对应的局部密度的比值作为样本数据的最终局部密度，以提高算法对具有多密度分布特征的数据集的处理能力。Wang 等通过计算信息熵来选择合适的截断距离 d_c[20]。Liu 等提出基于 K 邻域的 ADPC-KNN 改进算法[21]，通过位于邻域的数据计算截断距离 d_c。

现有的改进算法虽然提高了 DP 算法的性能，但是存在以下不足：1）需要额外输入超参数，降低了 DP 算法的可用性；2）数据划分的稳定性仍有待提高。针对以上不足，本文提出了一种基于影响空间的稳健密度峰值聚类算法 I-DP（Influence Space Based Robust Fast Search and Density Peak Clustering）。该算法在不需要增加额外参数的情况下，着重提高算法处理多密度数据的能力和数据划分的稳定性。本文算法的主要创新点包括：1）通过邻近数据计算局部加权密度比，并将其作为最终的局部密度，以提高算法处理具有多密度分布特征的数据集的能力；2）根据影响空间定义了一种新的对称数据关系，并提出了一种新的分配策略，提高了数据划分的稳定性，同时也在一定程度上降低了单个类簇中存在的多个密度峰值数据的影响。将本文提出的改进算法与经典聚类算法、DP 算法及其改进算法分别在人工合成数据集与 UCI 数据集上进行对比实验，结果证明了本文提出的改进思路的正确性。

2　DP 算法分析

快速搜索与密度峰值聚类算法 DP 可以发现任意形状的类簇，并且可以快速、高效地划分数据、分配类簇标签，适用于大规模数据集。DP 算法有两个重要的条件假设：1）类簇中心数据的局部密度比该类簇中其他数据的局部密度大；2）局部密度大的数据之间有较远的距离。根据以上假设，算法对每一个数据计算两个重要的参数：局部密度 ρ 和距离 δ。局部密度 ρ 有两种计算方式，对于数据集 $S=\{x_i\}_{i=1}^{n}$，$I_s=\{1,2,\cdots,n\}$ 为对应的下标。当数据为离散数据或者数据集规模较小时，计算公式为 Cut-off kernel：

$$\rho_i=\sum_{j\in I_s\setminus\{i\}}\chi(d_{ij}-d_c) \tag{1}$$

其中：

$$\chi(x)=\begin{cases}1, & x<0\\ 0, & x\geqslant 0\end{cases} \tag{2}$$

当数据是连续数据或者数据集规模较大时，计算公式为 Gaussian kernel：

$$\rho_i=\sum_{j\in I_S\setminus\{i\}}\mathrm{e}^{-\left(\frac{d_{ij}}{d_c}\right)^2} \tag{3}$$

其中，$d_{ij}=dist(x_i,x_j)$ 表示数据 x_i，x_j 之间的距离；$d_c>0$ 为截断距离，需要人工指定。距离 δ_i 的计算公式为：

$$\delta_i=\min(d_{ij}) \tag{4}$$

其中，j 表示距离 i 最近且局部密度比其均较的数据。对于局部密度最大的数据 g，δ_g 的定义为：

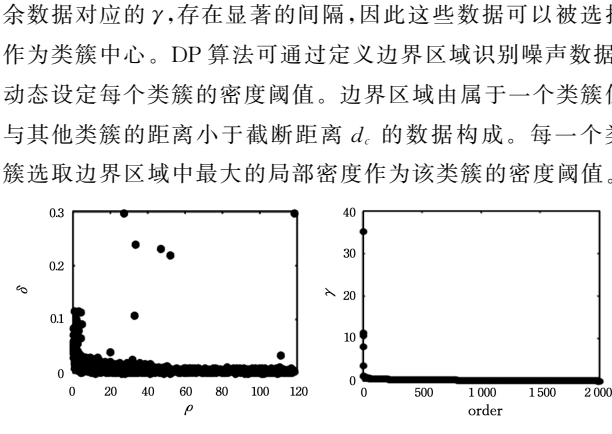
$$\delta_g=\max(d_{ij}) \tag{5}$$

由式(4)、式(5)可知，类簇中心拥有比簇内其他数据大的距离。利用距离 δ 和局部密度 ρ 构建决策图，人工选取 δ 和 ρ 均较大的数据作为类簇中心。对于剩余数据，采取高效的一步划分策略，将其划分到局部密度比它大且距离最近的数据所在的类簇。针对决策图在数据较少时难以确定类簇中心的情况，DP 算法引入参数 γ，采取一种启发式的类簇中心选择方式，γ 的计算公式为：

$$\gamma_i=\rho_i\delta_i,i\in I_s \tag{6}$$

其中，γ 越大，对应的数据越有可能是类簇中心。将 γ 按降序排列，选取 γ 最高的数据作为类簇中心。

图 1(a)是由文献[12]提供的样本数据生成的决策图，横坐标是局部密度，纵坐标是距离。由图 1(a)可知，离散分布于坐标轴上方位的数据同时拥有较大的局部密度 ρ 和距离 δ，可以作为中心数据。图 1(b)给出参数 γ 的分布情况，对其按降序排列，位于 y 轴坐标上的数据对应的参数 γ 明显大于其余数据对应的 γ，存在显著的间隔，因此这些数据可以被选择作为类簇中心。DP 算法可通过定义边界区域识别噪声数据，动态设定每个类簇的密度阈值。边界区域由属于一个类簇但与其他类簇的距离小于截断距离 d_c 的数据构成。每一个类簇选取边界区域中最大的局部密度作为该类簇的密度阈值。



(a)决策图　　(b)参数 γ 的分布

图 1　DP 算法的聚类过程

Fig. 1　Clustering process of DP algorithm

DP 算法的具体实现如算法 1 所示。

算法 1

输入:数据集 S,邻域规模 K

输出:聚类结果 R

Step1 根据式(1)－式(5)对每个数据计算局部密度 ρ 与距离 δ;

Step2 由 ρ 和 δ 生成决策图,手动选择类簇中心集合 C,并为其分配类簇标签;

Step3 按密度降序排列数据,将未分配数据划分到局部密度比其大且距离最近的数据所在的集合,并分配类簇标签;

Step4 生成各个类簇的边界区域,判定噪声数据。

3 I-DP 算法

3.1 影响空间

DP 算法高效的数据划分策略存在数据划分不稳定的问题,单个数据的错误划分会导致连锁错误分配。本文引入影响空间 IS(Influence Space),定义一种对称的数据关系,制定新的数据划分策略,以提高类簇标签分配的稳定性,同时减小单个类簇内存在的多个密度峰值数据的影响。把目标数据与邻近数据的加权密度比作为目标数据的最终局部密度,提高了算法处理具有多密度分布特征数据集的能力。影响空间 IS 由文献[22]首次提出,被广泛应以噪声检测领域,以寻找局部稀疏数据。局部稀疏数据指位于高密度区域附近的稀疏数据。不同于全局稀疏数据,局部稀疏数据由于距离密集区域近,在给定的半径范围内能够包含较多的邻近数据,同时也具有较小的 K-dist 距离。当数据分布呈现多密度分布特征时,位于高密度区域附近的局部稀疏数据往往拥有比稀疏区域的数据更大的局部密度和更小的 K-dist 距离,容易导致数据的错误评估。IS 同时考虑了最近邻域(NN_k)和反向最近邻域(RNN_k),能够准确评估数据的局部特征[22]。下面给出 IS 的相关定义。

定义 1(反向最近邻域) 反向最近邻域 RNN_k 是 K 最近邻域 NN_k 的反向邻域。对于数据 $p\in S$,$RNN_k(p)$的定义为:

$$RNN_k(p)=\{q\in S,p\in NN_k(q)\}$$

对于数据 p,NN_k 邻域代表数据集中与该数据距离最近的 K 个数据构成的固定规模的集合,但 RNN_k 反向邻域能够返回规模不固定的数据集合。局部稀疏数据虽然会有规模为 K 的邻域集合,但由于与邻近密集区域的数据有较大的密度差异,不会被包含在高密度数据的邻域中,因此其反向邻域返回规模较小的集合。相反,由于周围数据分布密集,位于高密集区域的数据会被包含在较多数据的邻域中,因此能够返回规模较大的 RNN_k 集合。

定义 2(影响空间) 设数据 $p\in S$,$IS(p)$的定义为:

$$IS(p)=NN_k(p)\cap RNN_k(p)$$

IS 定义了对称的双向邻域关系,只包含相互存于对方邻域的数据。结合 RNN_k 与 NN_k 集合,聚类算法能够识别稀疏数据,准确地反映数据的局部特征。图 2 给出了 NN_k,RNN_k 与 IS_k 的相关示例。设最近邻域 $K=3$,对于中心数据 p_4,其邻域为 $NN_k(p_4)=\{p_1,p_3,p_6\}$,对应反向最近邻域为 $RNN_k(p_4)=\{p_1,p_2,p_3,p_6,p_7\}$。由于 p_4 周围的数据分布密集,p_4 被包含于多个数据的邻域范围内,有较大的反向最近邻域集合,因此对应的影响空间为 $IS_k(p_4)=\{p_1,p_3,p_6\}$。边缘数据 p_6 的邻域为 $NN_k(p_6)=\{p_3,p_4,p_7\}$。$p_6$ 虽然位于由数据 p_1,p_2,p_3,p_4 构成的密集区域附近,但只被包含于 p_4 的邻域中,因此其反向最近邻域为 $RNN_k(p_6)=\{p_4,p_7\}$,对应的影响空间为 $IS_k(p_6)=\{p_4,p_7\}$。同理,对于噪声数据 p_7,其邻域为 $NN_k(p_7)=\{p_3,p_4,p_6\}$,反向最近邻域为 $RNN_k(p_7)=\emptyset$,影响空间为 $IS_k(p_7)=\emptyset$。

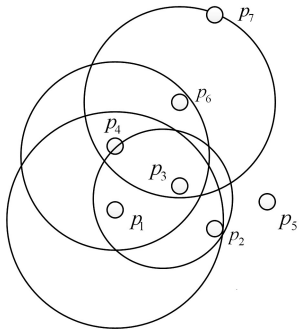


图 2 RNN 和影响空间

Fig. 2 RNN and influence space

定义 3(直接空间可达) 设数据 $q,p\in S$,若满足 $q\in IS(p)$,则称 q 是从 p 的直接空间可达。

定义 4(空间可达) 设数据 $p_1,p_2,p_3,\cdots,p_m\in S$。其中,$m\geqslant 4$。若满足 p_{i+1} 是从 p_i 直接空间可达,则称 p_{i+1} 是从 p_i 空间可达。

定义 5(空间相连) 设数据 $p_1,p_2,p_3\in S$,若 p_1 和 p_2 均是关于 p_3 的空间可达,则称 p_1 和 p_2 是空间相连。

定义 1－定义 5 定义了一种双向的数据关系,保证了数据间空间关系的稳定。与通过 NN_k 生成的邻域关系相比,IS 只包含互相存在邻域关系的数据,严格限制了邻域的规模。在使用 NN_k 的相关算法中,为识别核心数据和避免邻域过度扩张,需要引入两个超参数:K 和半径 ε,这在一定程度上降低了算法的可用性。影响空间 IS 在不需要引入额外超参数的情况下,减少了邻域过度扩张导致的数据划分错误。图 2 中,p_7 虽然包含于 p_6 的领域中,却没有被包含于 p_6 的影响空间中,因此避免了邻域的过度扩张。

定义 6(核心数据) 计算每个数据的局部密度,将局部密度超过平均值的数据称为核心数据,其定义为:

$$Core=\{o\in S|\rho_o>\frac{1}{n}\sum_{1}^{n}\rho_i\} \tag{7}$$

由局部密度 ρ 和距离 δ 生成决策图,选择在图中明显突出的数据样本作为类簇中心。以类簇中心为起点,采取本文提出的分配策略,把与类簇中心空间可达的核心数据划分到相同的集合,分配类簇标签。被划分的核心数据构成了该类簇的核心区域,位于核心区域的数据对类簇的形成有重要的影响,这些数据的错误划分会导致大量边缘数据被错误分配。新的分配策略不仅能够提高数据划分的稳定性,且当同一个类簇中的多个密度峰值数据存在空间可达关系时,能够将其划分到同一个集合中,避免了类簇分裂,减小了同一个类簇内同时存在的多个密度峰值数据的影响。对于剩下未被分配的数据,采取 DP 算法高效的一步划分策略,将剩余数据划分到比其局部密度高且距离最近的数据所在的集合,分配相同的类簇标签。I-DP 算法通过对核心数据采取新的分配策略,来减小连锁错误分配的影响,提高了数据划分的稳定性。

距离 δ 采取文献[17]提出的相对距离计算方式,将数据

p 与最近高密度数据和低密度数据的距离差值作为该数据的距离 δ。本文提出的新的分配策略和 I-DP 算法的详细步骤如算法 2 和算法 3 所示。

算法 1　新的分配策略

输入：类簇中心集合 C，影响空间 IS

输出：核心数据划分结果

Step1　在类簇中心集合 C 中选出一个未被访问的类簇中心 C_i，并将其标记为已访问；

Step2　将 C_i 影响空间 IS(c_i)中的核心数据划分到 C_i 所在的集合，初始化队列 V，将 IS(c_i)中的核心数据加入队列；

Step3　取队列 V 的头数据 q，将集合 IS(q)中的未被分配的核心数据划分到 q 所属的类簇，并将数据加入到队列 V 尾部；

Step4　若队列 V 不为空，则转 Step3；

Step5　若 C 中还有未被访问的类簇中心，则转 Step1，否则结束策略。

算法 2　I-DP 算法流程

输入：数据集 S，邻域规模 K

输出：聚类结果 R

Step1　计算每个数据的 ρ，δ；

Step2　由 ρ 和 δ 生成决策图，选出类簇中心集合 C；

Step3　根据本文提出的新的分配策略对类簇中心数据与核心数据进行分配；

Step4　将剩余数据划分到局部密度比其大且距离最近的数据所在的集合；

Step5　判定边界区域，去除噪声数据。

3.2　局部密度

DP 算法定义了两种局部密度公式，其中式(3)基于高斯核的密度计算公式由于可以减小局部密度发生冲突的概率，因此能够被应用于大规模数据集。但该公式需要输入参数 d_c，参数的设置会对聚类产生较大的影响[17,20]。根据式(3)，局部密度 ρ 对 d_c 的导数为：

$$\frac{\partial \rho_i}{\partial d_c}=2\sum_{j\in I_s\setminus\{i\}} e^{-\frac{d(i,j)^2}{d_c^2}}\frac{d(i,j)^2}{d_c^3} \tag{8}$$

式(8)表明局部密度 ρ 同样受到距离 d_{ij} 的影响。当数据分布不规则，且各类簇包含的数据分布差异大时，通过数据集全局计算局部密度会对规模较小、数据分布稀疏的类簇造成不利影响。缩小局部密度计算范围，通过 K 邻近数据计算密度更能反映数据的局部特征。本文采用 K 邻近数据计算局部密度 ρ。对于数据 i，局部密度 ρ_i 的计算公式定义为：

$$\rho_i=\sum_{j\in KNN(i)} e^{-(\frac{d_{ij}}{d_c})^2} \tag{9}$$

其中，$KNN(i)$ 是距离数据 i 最近的样本数据构成的集合。当数据分布不规则，类簇间的密度差异大，呈现多密度分布特征时，稀疏类簇的识别度较低。通过邻近数据计算的绝对局部密度不能够准确识别稀疏类簇的中心数据，从而导致稀疏数据的划分错误。多密度分布数据集如图 3 所示，该数据集存在密度差异较大的两个类簇。p_1 表示左侧类簇 1 的边缘数据，p_3 表示左侧类簇 1 的中心数据，p_2 表示右侧类簇 2 的中心数据。虽然 p_2 是稀疏类簇的中心，相对其邻近数据有较大的局部密度，但是由于 p_2 所在类簇稀疏，导致 p_2 的绝对局部密度 ρ 较小。由于边缘数据 p_1 所在的类簇数据分布密集，因此 p_1 的局部密度大于 p_2 的局部密度。

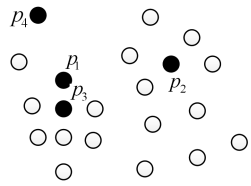


图 3　多密度分布数据集

Fig. 3　Multi-density distribution data set

I-DP 算法采用相对密度作为局部密度，计算目标数据与邻近数据密度的比值，从而提高算法识别稀疏数据的能力。虽然 p_2 的绝对局部密度小，但是相对邻域数据仍具有较高的相对局部密度，相反，p_1 具有较小的相对局部密度。对于数据 i，相对局部密度 ρ' 的计算公式定义为：

$$\rho_i'=\sum_{j\in KNN(i)}\frac{\rho_i}{\rho_j}|IS(i)| \tag{10}$$

其中，$|IS_i|$ 是数据 i 的影响空间规模。需要注意的是，当异常数据 p_4 被包含于 p_1 的邻域中时，由于 p_4 的局部密度远小于 p_1，因此 p_1 得到大的密度比值，将影响中心数据的选择。针对这一问题，引入参数 $|IS|$，其相当于权重系数，对影响空间大的数据赋予大的权重。由于 p_1 位于类簇边缘，邻域稀疏，影响空间的规模相对较小。

3.3　算法分析

算法空间复杂度主要来自计算局部密度 ρ 所需的距离矩阵。对于规模为 n 的数据集 S，存储距离矩阵的空间复杂度为 $O(n^2)$。相比 DP 算法，I-DP 算法通过 K 邻近数据计算相对局部密度，增加了存储每个数据 K 邻域所需的空间，增加的空间复杂度为 $O(nK)$。由于 K 远小于数据集的规模 n，因此 I-DP 算法与 DP 算法有相同级别的空间复杂度。时间复杂度主要来源于 4 个部分：1)计算距离矩阵，计算每一个数据与其余所有数据的距离，时间复杂度为 $O(n^2)$；2)计算局部密度 ρ，式(3)的时间复杂度为 $O(n^2)$，I-DP 根据 K 邻近数据计算相对局部密度，计算时间比 DP 算法短，但由于搜索 K 邻近数据的时间复杂度是 $O(n^2)$，因此计算局部密度 ρ 的时间复杂度都为 $O(n^2)$；3)计算样本距离，I-DP 算法分别计算每一个样本数据到密度比它高和低的最近数据的距离，时间复杂度都为 $O(n^2)$，总的时间复杂度为 $2O(n^2)$；4)数据划分，I-DP 算法增加了新的分配策略，依次采取宽度优先的方式对每个类簇中心影响空间中的核心数据进行搜索，增加的时间复杂度为 $O(tn_c)$，n_c 是核心数据的规模，t 是类簇中心的数量，且 t 远小于数据集与核心数据的规模。

4　实验分析

为验证本文提出的改进方法的可行性，在人工合成数据集和 UCI 真实数据集上对 I-DP 算法进行模拟实验。将 I-DP 算法与 DP、K-mediods、K-means、DBSCAN、文献[17]提出的 CDP 算法以及文献[21]提出的 ADPC-KNN 算法进行比较。实验环境如下：操作系统为 Windows10，CPU 主频为 2.50 GHz，内存 8 GB，仿真软件 Matlab2014b。表 1 列出了实验所用数据集的具体信息，其中 Fullmoon 和 Corner 数据集为人工合成数据集。本文选取准确率(Acc)与标准互信息(NMI)作为聚类结果的评价指标。

表 1 测试数据集

Table 1 Test datasets

Datasets	Instance	Dimensions	Clusters
Fullmoon	5000	2	2
Corner	600	2	4
Seed	210	7	3
Cmc	1473	9	3
Iris	150	4	4
Abalone	4177	8	3
Optdigits	5620	64	10

NMI 指标度量真实划分与聚类划分的相近程度，相关定义如下：设数据集合 $U=\{u_1,\cdots,u_n\}$，$V=\{v_1,\cdots,v_n\}$ 分别表示真实数据分类与聚类结果，则集合对应的信息熵分别表示为：

$$H(U)=-\sum_{i=1}^{|U|}P(i)\log(P(i))$$

$$H(V)=-\sum_{i=1}^{|V|}P'(i)\log(P'(i))$$

其中，$P(i)=\frac{|u_i|}{N}$，$P'(j)=\frac{|v_j|}{N}$，集合 U,V 之间的互信息 MI (Mutual Information)定义为：

$$MI(U,V)=\sum_{i=1}^{|U|}\sum_{j=1}^{|V|}P(i,j)\log\left(\frac{P(i,j)}{P(i)P'(j)}\right)$$

其中，$p(i,j)=\frac{|U_i\bigcap V_j|}{N}$，则标准互信息定义为：

$$NMI(U,V)=\frac{2MI(U,V)}{H(U)+V(Y)}$$

为了消除属性的量纲影响，对数据进行标准化处理。截断距离 d_c 由 K 决定，Rodriguez 在文献[12]中建议 DP 算法中的数据邻域应包含占数据集 1%～2%的数据。实验中，K 取小于或等于 2%的数据集规模。当数据 i 是噪声时，对应的影响空间 $|IS_i|$ 可能为 0。为更准确地比较局部密度，将每个数据的影响空间规模增加 1。

表 2 列出了 I-DP 算法与对比算法在 Acc，NMI 指标上的对比结果，可见 I-DP 算法在人工合成数据集与 UCI 真实数据集上都获得了较好的结果。

表 2 实验结果

Table 2 Experimental results

(单位：%)

Datasets		k-mediods	K-means	DBSCAN	DP	ADPC-KNN	CDP	I-DP
Fullmoon	Acc	0.51	0.51	0.75	1	0.79	0.81	0.85
	NMI	0.001	0.001	0.005	1	0.37	0.38	0.43
Corner	Acc	1	0.50	1	1	0.54	0.70	1
	NMI	1	0.50	1	1	0.55	0.65	1
Seed	Acc	0.84	0.82	0.85	0.35	0.91	0.87	0.89
	NMI	0.54	0.54	0.75	0.01	0.71	0.62	0.64
Cmc	Acc	0.37	0.37	0.43	0.43	0.43	0.42	0.44
	NMI	0.01	0.01	0.001	0.002	0.01	0.02	0.01
Iris	Acc	0.97	0.97	0.68	0.83	0.96	0.97	0.98
	NMI	0.90	0.90	0.59	0.67	0.89	0.89	0.92
Abalone	Acc	0.46	0.46	0.37	0.37	0.47	0.46	0.48
	NMI	0.10	0.10	0.001	0.001	0.09	0.10	0.11
Optdigits	Acc	0.78	0.70	0.78	0.60	0.88	0.93	0.95
	NMI	0.71	0.70	0.71	0.68	0.76	0.90	0.93

图 4 给出了本文提出的新的分配策略在 Corner 数集上的分配结果。

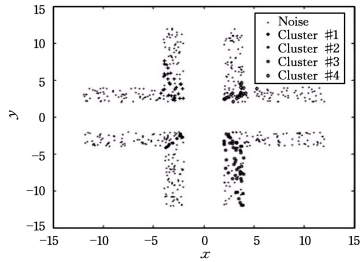


图 4 Corner 数据集

Fig. 4 Corner dataset

由图 4 可看出，由于类簇 3、类簇 4 的数据分布密集，与类簇中心空间可达的核心数据较多，因此影响空间得到了比较广泛的延伸。虽然类簇 1、类簇 2 的数据分布相对稀疏，但通过空间可达关系，类簇中心的影响空间也得到延伸。其中，类簇 2－类簇 4 的空间可达区域覆盖了多个密度峰值数据，将其划分到同一个集合能减小单个类簇内存在的多个密度峰值数据对算法的影响。需要注意的是，在类簇中心影响空间的延伸区域中存在未被划分的局部稀疏数据。这些稀疏数据虽然距离密集区域近，但由于其本身的密度低，因此不在核心数据的影响空间范围内。当类簇之间的距离较小时，这部分稀疏数据会构成类簇的边界区域。同时，这部分稀疏数据由于距离高密度数据近，所以可以通过 DP 算法高效的分配策略进行准确划分。通过对核心数据使用新的分配策略，I-DP 数据划分的稳定性得到了提高，减小了连锁错误分配的影响，进而提高了算法的整体稳定性。

表 3 列出了 I-DP 算法与 DP 及其改进算法的时间比较结果。

表 3 运行时间

Table 3 Running time

(单位：s)

Dataset	DP	ADPC-KNN	CDP	I-DP
Fullmoon	2.77	22.058	22.531	21.545
Corner	0.102	0.203	0.224	0.209
Seed	0.032	0.076	0.070	0.077
Cmc	0.338	0.857	0.948	1.220
Iris	0.020	0.048	0.057	0.056
Abalone	1.843	4.968	5.645	5.908
Optdigits	35.310	43.207	42.593	47.111

由于 DP 算法及其改进算法需要人工手动选择类簇中

心，为降低人工因素的影响，实验自动选取 γ 最大的 N 个数据作为类簇中心，其中 N 为真实类簇的数量。可见，I-DP 算法在提高聚类效果的同时与对比改进算法有相近的运行时间。

结束语　针对 DP 算法数据划分稳定性差、容易导致连锁错误分配，且无法处理复杂数据集的缺点，本文提出基于影响空间的稳健 I-DP 改进算法。I-DP 算法不仅使用了新的局部密度计算方式，同时也定义了一种新的数据关系以提高数据分配的稳定性。该算法计算目标数据与邻近数据的密度比值，并利用影响空间进行加权得到最终的局部密度，从而提高算法对复杂数据集的处理能力。本文引入影响空间 IS，定义了一种新的对称邻域关系，并制定了一种新的数据划分策略。对类簇中心与核心数据采用新的数据划分策略，在提高数据划分稳定性的同时，降低了单个类簇内同时存在的多个密度峰值数据的影响。在基于人工数据集和 UCI 真实数据集的对比实验中，I-DP 改进算法均取得较优的结果。人工选择的方式不仅存在主观因素，同时效率低，影响了算法的实用性。因此下一步将重点研究在复杂数据集上类簇中心自动选择的方法。

参 考 文 献

[1] MACQUEEN J. Some Methods for Classification and Analysis of MultiVariate Observations[C]// Berkeley Symposium on Mathematical Statistics and Probability. Berkeley: University of California Press, 1966: 281-297.

[2] KARYPIS G, HAN E H, KUMAR V. CHAMELEON. A hierarchical clustering algorithm using dynamic modeling[J]. Computer, 1999, 32(8): 68-75.

[3] ESTER M, KRIEGEL H P, XU X. A Density-based Algorithm for Discovering Clusters in Large Spatial Databases with Noise[C]// International Conference on Knowledge Discovery and Data Mining. Palo Alto: AAAI Press, 1996: 226-231.

[4] WANG W, YANG J, MUNTZ R R. STING: A Statis-tical Information Grid Approach to Spatial Data Mining[C]// International Conference onVery Large Data Bases. San Mateo: Morgan Kaufmann. 1997: 186-195.

[5] HU Q, WU J, BAI L, et al. Fast K-means for Large Scale Clustering[C]// Conference on Information and Knowledge Management. New York: ACM, 2017: 2099-2102.

[6] PARK H S, JUN C H. A Simple and Fast Algo-rithm for K-medoids Clustering[J]. Expert Systems with Applications, 2009, 36(2): 3336-3341.

[7] HE Y, TAN H, LUO W, et al. MR-DBSCAN: An Efficient Parallel Density-Based Clustering Algorithm Using MapReduce[C]// International Conference on Parallel and Distributed Systems. New York: IEEE, 2012: 473-480.

[8] JAHIRABADKAR S, KULKARNI P. Algorithm to Determine ℓ-distance Parameter in Density based Clustering[J]. Expert Systems with Ap-plications, 2014, 41(6): 2939-2946.

[9] NANDA S J, PANDA G. Design of Computationally Efficient Density-based Clustering Algorithms[J]. Data & Knowledge Engineering, 2015, 95: 23-38.

[10] BAI X, YANG P, SHI X. An Overlapping Community Detection Algorithm Based on Density Peaks[J]. Neurocomputing, 2016, 226(22): 7-15.

[11] WANG H Y, ZHANG T F, MA F M. Rough K-means Algorithm with Self-adaptive Weights Measurement Based on Space Distance[J]. Computer Science, 2018, 45(7): 190-196. (in Chinese)
王慧研，张腾飞，马福民. 基于空间距离自适应权重度量的粗糙 K-means 算法[J]. 计算机科学，2018，45(7)：190-196.

[12] RODRIGUEZ A, LAIO A. Machine Learning. Clustering by Fast Search and Find of Density Peaks[J]. Science, 2014, 344(6191): 1492-1496.

[13] ZHANG Y, XIA Y, LIU Y, et al. Clustering Sentences with Density Peaks for Multidocument Summarization[C]// Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 2015: 1262-1267.

[14] ZHANG Y, CHEN S. Efficient distributed density peaks for clustering large data sets in mapreduce[J]. IEEE Transactions on Knowledge and Data Engineering, 2016, 28(12): 3218-3230.

[15] XU J, WANG G, LI T, et al. Fat node leading tree for data stream clustering with density peaks[J]. Knowledge-Based Systems, 2017, 120: 99-117.

[16] CHEN Y, MA S, CHEN X, et al. Hyperspectral data clustering based on density analysis-ensemble[J]. Remote Sensing Letters, 2017, 8(2): 194-203.

[17] LI Z J, TANG Y C. Comparative density Peaks Clustering[J]. Expert Systemswith Applications, 2018, 95: 236-247.

[18] DING J, CHEN Z, HE X, et al. Clustering by Finding Density Peaks Based on Chebyshev's inequality[C]// Control Conference. New York: IEEE, 2016: 7169-7172.

[19] GAO S Y, ZHOU X F, LI S. Clustering by Fast Search and Find of Density Peaks Based on Density Ratio[J]. Computer Engineering and Applications, 2017, 53(16): 10-17. (in Chinese)
高诗莹，周晓锋，李帅. 基于密度比例的密度峰值聚类算法[J]. 计算机工程与应用，2017，53(16)：10-17.

[20] WANG S L, WANG D K, LI C Y, et al. Clustering by Fast Search and Find of Density Peaks with Data Field[J]. Chinese Journal of Electronics, 2016, 25(3): 397-402.

[21] LIU Y, MA Z, YU F. Adaptive Density Peak Clustering Based on K-nearest Neighbors with Aggregating Strategy[J]. Knowledge Based Systems, 2017, 133: 208-220.

[22] JIN W, TUNG A K H, HAN J, et al. Ranking Outliers Using Symmetric Neighborhood Relationship[C]// Pacific-Asia Conference on Knowledge Discovery and Data Mining. Berlin Heidelberg: Springer, 2006: 577-593.