文章编号:1005-0523(2025)04-0120-07

基于自然和加权共享最近邻的密度峰值聚类算法

王 森,陈 翔,詹小秦,徐 璐,吴启正

(华东交通大学理学院,江西 南昌 330013)

摘要:密度峰值聚类(DPC)作为一种高效且不需要迭代的聚类算法得到广泛应用。研究发现,该算法使用密度不均匀数据集上时,DPC很难选择正确的簇中心,且该算法受参数的影响较大。为了解决DPC算法在密度分布不均匀的数据集上效果不佳的问题,提出了一种基于自然和加权共享最近邻的密度峰值聚类算法。该算法首先引入自然最近邻计算加权值,再根据一阶和二阶共享最近邻的定义重新计算数据对象之间的相似度,然后通过融合共享最近邻相似度的定义和自然最近邻权重值计算相对密度和相对距离,最后还设计了新的分类型簇中心扩散分配策略。

关键词:聚类算法;密度峰值聚类;自然最近邻;共享最近邻;簇中心扩散

中图分类号:TP181 **文献标志码:**A

本文引用格式:王森,陈翔,詹小秦,等. 基于自然和加权共享最近邻的密度峰值聚类算法[J]. 华东交通大学学报,2025,42(4):120-126.

Density Peak Clustering Algorithm Based on Natural and Weighted Shared Nearest Neighbors

Wang Sen, Chen Xiang, Zhan Xiaoqin, Xu Lu, Wu Qizheng

(School of Science, East China Jiaotong University, Nanchang 330013, China)

Abstract: Density peak clustering (DPC) has been widely used as an efficient and non-iterative clustering algorithm. However, studies have found that DPC struggles to select correct cluster centers, especially in datasets with non-spherical clusters and non-uniform density. Moreover, DPC is heavily influenced by the truncation distance parameter. In order to address the issue of poor performance of DPC on datasets with uneven density distributions, a density peak clustering algorithm based on natural and weighted shared nearest neighbors is proposed. It first introduced natural nearest neighbor computations to calculate weights. Then, it redefined the similarity between data objects based on the definitions of first-order and second-order shared nearest neighbors. Subsequently, by fusing the definitions of shared nearest neighbor similarity and natural nearest neighbor weights, relative density and relative distance were calculated. Finally, a novel strategy for distributing cluster centers was designed.

Key words: clustering algorithm; density peak clustering; natural nearest neighbor; shared nearest neighbor; cluster center diffusion

Citation format: WANG S, CEHN X, ZHAN X, et al. Density peak clustering algorithm based on natural and

收稿日期:2024-03-19

基金项目:江西省自然科学基金项目(20224BAB211005)

weighted shared nearest neighbors[J]. Journal of East China Jiaotong University, 2025, 42(4): 120−126.

聚类是机器学习中的一种分类方法，旨在根据数据集的某些特征将其划分为多个潜在的簇。该方法致力于确保簇内的数据对象具有较高相似性，而簇间的相似性则尽可能低[1]。2014年，Rodriguez等[2]提出了密度峰值聚类（density peak clustering，DPC）算法。由于DPC算法具有较强的可解释性、无需迭代、简单且能够识别具有形状的数据集，因此受到了广泛关注。然而，传统的DPC算法存在以下不足之处：人为设置的截断参数值对聚类效果产生较大影响；对于密度分布不均匀的数据集，容易忽略稀疏簇的中心点；DPC算法进行非中心点标签分配时，一个点的分配错误会导致后续点也分配错误的链式反应[3]。

Liu等[4]提出的ADPC-KNN算法的核心思想是在计算截断距离时通过KNN自动计算参数值从而自适应地找到了合适的截断距离，该算法虽然自适应地找到了合适的截断距离值，但是却增加了新的参数K；Zhang等[5]借助自然界的衰减现象的规律提出了DGDPC算法，该算法先构造密度衰减图，根据此图的互相关系建造关系图来识别潜在的初始聚类中心，最后根据自然界衰减规律的统一性完成最后聚类任务；Guo等[6]提出了一种基于K近邻集的NDPC算法，其核心思想是簇中心点被低密度的数据对象包围，所以也被以该点为邻居的所有数据对象包围，计算所有视该点为邻居的数据对象的个数即可得到该点的相对密度，该方法在密度不均匀的数据集上使用效果较好；Cheng等[7]提出了一种基于局部核心点的DLORE-DPC算法，该算法结合了自然最近邻的思想先寻找微簇的核心点作为代表点，计算代表点间的相似度，再对微簇进行合并。Liu等[8]提出了一种基于共享最近邻的SNN-DPC算法，该算法根据共享最近邻的概念重新定义了数据对象间的相似度，计算相对距离时提出了基于K近邻的补偿值，最后在分配非中心点时则使用两步分配策略。

针对DPC算法的不足之处，本文提出一种基于自然和共享最近邻的密度峰值聚类算法（DPC-NN-WSNN）。

1　DPC算法

DPC算法其核心思想基于两个假设：首先，每个簇的中心密度相对较高且被低密度区域所包围；其次，每个簇中心之间的距离会相对较远[9]。为了对应这两个假设引入了相对密度ρ_i和相对距离δ_i这两个定义，数据对象i的相对密度记作ρ_i，定义为

$$\rho_i=\sum_{i\neq j}\chi\left(\mathrm{dist}(i,j)-d_c\right) \tag{1}$$

式中：i和j分别为数据集中的两个数据对象；dist(·)为欧式距离，d_c为截断距离；$\chi(\cdot)$表示以d_c为半径的领域内的数据点数量即为该点的相对密度，该公式也被称为截断核密度公式。除了截断核密度公式之外，还有高斯核密度公式，定义为

$$\rho_i=\sum_{i\neq j}\exp\left(-\frac{\mathrm{dist}(i,j)^2}{d_c^2}\right) \tag{2}$$

式中：exp(·)表示底数为e的幂函数。当数据集规模较大时一般使用截断核相对密度公式，数据集规模较小时则使用高斯核相对密度公式。

数据集中每个数据对象到其密度更高的且最近的数据对象的相对距离δ_i定义为

$$\delta_i=\min_{j:\rho_j>\rho_i}\left(\mathrm{dist}(i,j)\right) \tag{3}$$

DPC算法认为当相对密度ρ_i和相对距离δ_i的值都较大时，对应的数据对象被选取为簇中心，最后将剩余非中心点分配至距离最近且密度比其大的簇。

2　优化算法DPC-NN-WSNN

2.1　基于自然和共享最近邻的加权相对密度

Zhu等[10]提出了自然最近邻的概念，该定义自适应地根据数据对象的特点寻找自身的自然最近邻，无须设置参数K就能自适应找到每个数据对象的自然最近邻，可以降低参数值对聚类结果产生的影响。基于自然最近邻的思想，将提出自然最近邻权重值的定义。

定义1　数据对象i的自然近邻权重值记作

$W_n(i)$，定义为

$$W_n(i)=\frac{\sum_{m\in \mathrm{NN}(i)}\mathrm{dist}(i,m)}{\left|\mathrm{NN}(i)\right|} \tag{4}$$

式中：$\mathrm{NN}(i)$为自然近邻集函数；m为自然近邻集的任意一点。

在计算孤立点的相对密度时，由于孤立点远离簇中心，在有些情况下找不到其一阶共享最近邻也难以计算其一阶共享最近邻相似度，并且一阶共享最近邻的信息在计算过程中是不够精确的，二阶邻居的信息对计算结果也能产生较大的影响。针对一阶共享最近邻的缺陷，在此基础上提出二阶共享最近邻的定义，并且提出基于二阶共享最近邻的相似度定义。

定义2 数据对象i和j的二阶共享最近邻记作$\mathrm{SSNN}(i,j)$，函数定义为

$$\mathrm{SSNN}(i,j)=\mathrm{SKNN}(i)\cap\mathrm{SKNN}(j) \tag{5}$$

式中：$\mathrm{SKNN}(i)$为数据对象x_i的二阶邻居集合函数。

定义3 数据对象i和j的二阶共享最近邻相似度记作$\mathrm{SSim}(i,j)$，函数定义为

$$\mathrm{SSim}(i,j)=\frac{\left|\mathrm{SSNN}(i,j)\right|^2}{\sum_{p\in \mathrm{SSNN}(i,j)}\left(\mathrm{dist}(p,i)+\mathrm{dist}(p,j)\right)} \tag{6}$$

式中：$\left|\mathrm{SSim}(i,j)\right|$为点$x_i$和点$x_j$的二阶共享最近邻数量函数；$p$为二阶共享最近邻集的任意一点。特别地，当数据对象$i$和$j$没有二阶共享最近邻时，二阶共享最近邻相似度设置为0。

定义4 数据对象i和j的融合相似度记作$\mathrm{FSim}(i,j)$，函数定义为

$$\mathrm{FSim}(i,j)=\alpha\mathrm{Sim}(i,j)+(1-\alpha)\mathrm{SSim}(i,j) \tag{7}$$

式中：α为可以调整的平衡值，考虑到一阶共享最近邻相似度对数据对象的相似度贡献值较大，后续实验将α的值设置为0.8。该式计算了数据对象的融合相似度，其中包括了多阶的融合共享最近邻信息，对于数据集中的边界点或者孤立点，有效地提升了其相似度的计算精度[11]。

定义5 数据对象i的相对密度记作$\rho_{\mathrm{r}}(i)$，定义为

$$\rho_{\mathrm{r}}(i)=\sum_{j\in L(i)}\mathrm{FSim}(i,j)W_n(i) \tag{8}$$

式中：$L(i)$为与数据对象i的融合相似度最高的K个点的集合。该式计算所有点数据对象i的融合相似度，找到最高的K个点然后相加，就能得到融合相似度之和，再乘上该点的自然最近邻权重值$W_n(i)$，即为该数据对象的相对密度。

本文提出的相对密度ρ_{r}能够自适应地根据簇的稀疏程度通过自然近邻权重值调整数据对象的相对密度值，对于处于稀疏簇的数据对象，其相对密度值会因此放大，得到更为真实的相对密度值[12]。

2.2 基于自然最近邻的加权相对距离

在DPC算法中，相对距离δ_i的计算方式是数据对象i到密度比它大的数据对象的最短距离，但是这样计算忽略了数据对象的周围环境信息，对于密度稀疏的点没有权重补正[13]，因此本小节提出基于自然最近邻的加权相对距离表达式。

定义6 数据对象i的加权相对距离表达式记作$\delta_r(i)$，定义为

$$\delta_r(i)=\left[\min_{j:\rho_j>\rho_i}\mathrm{dist}(i,j)\right]\left(W_n(i)+W_n(j)\right) \tag{9}$$

该式对于稀疏簇来说，依然会有效放大其相对距离值，有助于识别正确地簇中心[14]。

2.3 分类型簇中心扩散策略

传统DPC算法的分配过程只要有一个标签分配错误，后面就会出现链式错误。针对这个问题，提出了分类型簇中心扩散策略[15]，该策略基于图1对从属点的解释图，对稀疏和密集簇中心点进行区分，设计了不同的识别从属点阈值。

定义7 必然从属点：假设数据对象i的标签已经被分配，当且仅当i满足如下条件时

$$\left|\mathrm{SNN}(i,j)\right|\geqslant\frac{k}{n} \tag{10}$$

j是i的一个必然从属点，如图1所示。

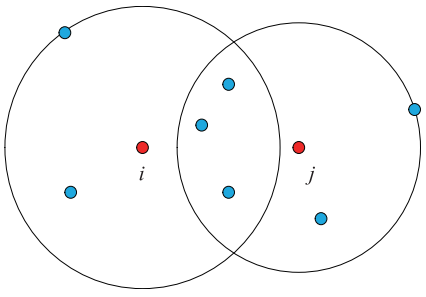

图1 必然从属点

Fig. 1 Inevitable Subordinate Point

定义8 可能从属点：假设数据对象 i 的标签已经被分配，当且仅当 j 满足如下条件时

$$0<\left|\mathrm{SNN}(i,j)\right|<\frac{k}{n} \tag{11}$$

j 是 i 的一个可能从属点，如图2所示。

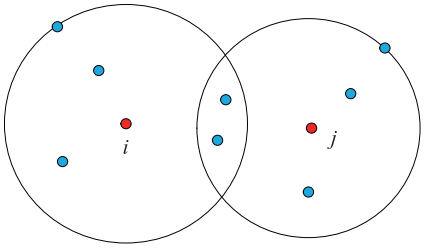

图2　可能从属点
Fig. 2　Possible subordinate point

定义9 不可能从属点：假设数据对象 i 的标签已经被分配，当且仅当 j 满足如下条件时

$$\left|\mathrm{SNN}(i,j)\right|=0 \tag{12}$$

j 是 i 的一个不可能从属点，如图3所示。

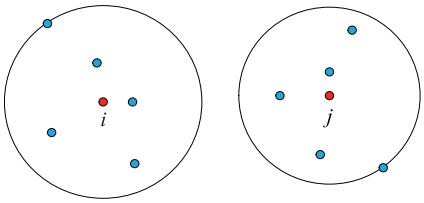

图3　不可能从属点
Fig. 3　Impossible subordinate point

式(10)~式(12)中：k 为最近邻的数量；n 为任意正整数(后续实验取值为2或者3)；如果数据对象 i 和 j 共享最近邻数量高于阈值 $\frac{k}{n}$，那么 j 是 i 的必然从属点，j 和 i 的标签将会保持一致，反之 j 是 i 的可能从属点。特别地，共享最近邻数量为0时，j 是 i 的不可能从属点。

对于密度较高的簇，数据对象分布更密集，会识别出更多的共享最近邻，所以在进行簇中心扩散分配策略时，处于密度较高簇的中心点更容易识别出更多的从属点，从而影响标签分配的准确率。针对这个问题，提出两个定义对两种密度差别较大的簇中心进行分类，然后分类使用不同的从属点的判断阈值以解决这一问题[16]。

定义10 相对高密度簇中心：假设数据对象 i 是被决策图选取出的簇中心，当且仅当 i 满足如下条件时

$$0\leqslant W_n(i)<\frac{\sum\limits_{p\in L(c_i)}W_n(p)}{\left|L(c_i)\right|} \tag{13}$$

x_i 为相对高密度簇中心。

定义11 相对低密度簇中心：假设数据对象 i 是被决策图选取出的簇中心，当且仅当 i 满足如下条件时

$$W_n(i)\geqslant\frac{\sum\limits_{p\in L(c_i)}W_n(p)}{\left|L(c_i)\right|} \tag{14}$$

i 为相对低密度簇中心。

式(13)，式(14)中：$L(c_i)$ 是所有被识别出的簇中心集合；$\left|L(c_i)\right|$ 簇中心的个数是右式计算的是簇中心的自然最近邻权重的平均值。因为自然最近邻权重越大，密度相对越低，所以所有簇中心里自然最近邻权重值大于平均值的都视为相对低密度簇中心，反之为相对高密度簇中心。对于密度较低的簇中心，设置更大的 n 值，从而减小阈值，保证低密度簇在进行标签分配时也能识别更多的必然从属点，从而可以进行更精确的分配[17]。分类型簇中心扩散策略如下。

输入：数据集 X 的初始簇中心 $C(X)=(c_1,c_2,\cdots,c_m)$。

Step 1：初始化创建队列 Q，将 $C(X)$ 里的簇中心索引加入队列中；

Step 2：检查所有簇中心的自然最近邻权重值，其权重值小于 $\frac{\sum\limits_{p\in L(c_i)}W_n(p)}{\left|L(c_i)\right|}$ 的簇中心设置为相对高密度簇中心，反之则设置为低密度簇中心；

Step 3：对于队列中的相对高密度簇中心 n 设定为2，低密度簇中心 n 设定为3，并找到其邻居点 i 并检查其是否已经被分配过标签且和其一阶共享最近邻的数量是否大于 $\frac{k}{n}$，如果都满足则将 i 分配给该簇，并且将 i 加入队列 Q，并且重复迭代扩展队列 Q；

Step 4：初始化创建数组 A，将未被分配的点的索引加入数组中，检查剩下未被分配的点的邻居点的分配情况，查询每个簇在邻居点里的数量，找到数量最多的簇，将其分配到数量最多的簇里；

Step 5：更新数组 A，去除已经分配的点，如果仍然有未被分配的点，重复Step 4；

Step 6：如果依然有未被分配的点，则 $k+1$。

输出：聚类结果 $\phi=(C_1,C_2,\cdots,C_m)$，$m$ 为簇的个数。

3 对比实验

该算法在Jain、Pathbased、Compound-2、Aggregation数据集上进行实验。表1列出了数据集的具体信息，本实验采用调整后的互信息（AMI）、调整后的兰德系数（ARI）和Fowlkes-Mallows指标（FMI）3个常用的聚类算法评价指标进行算法性能评估。

3.1 实验结果

本节展示算法DPC-NN-WSNN在4个数据集上相较于DBSCAN，DPC，SNN-DPC算法的优势，图4至图7分别可视化了在4种数据集上聚类算法的结果图，表2列出了4组聚类评价指标的对比。

表1 实验数据集

Tab.1 Experimental datasets

数据集	容量/个	簇/个
Jain	373	2
Pathbased	300	3
Compound-2	349	5
Aggregation	788	7

3.2 分析与评估

根据聚类结果图和聚类评价指标的表格，优化算法DPC-NN-WSNN在4个数据集上有着相等甚至更好的聚类效果，而在簇间密度差异较大的

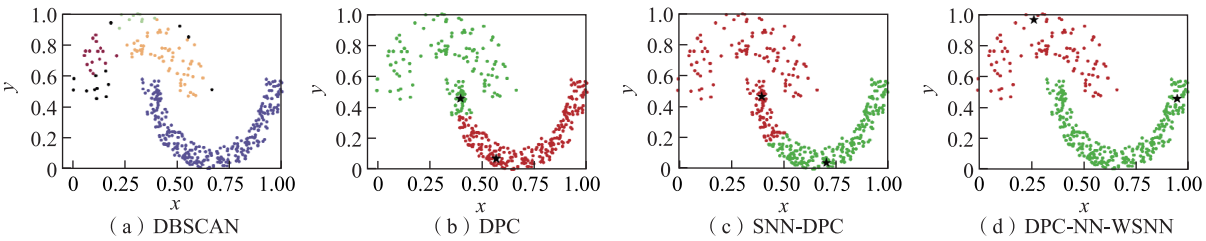

图4 4种算法在Jain的聚类结果对比

Fig. 4 Comparison of clustering results of 4 algorithms in Jain

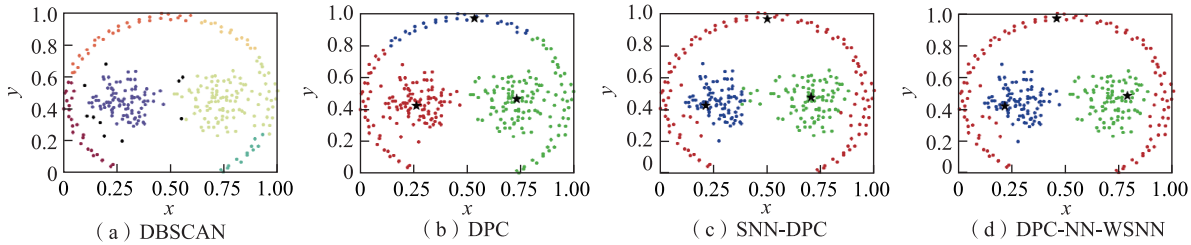

图5 4种算法在Pathbased的聚类结果对比

Fig. 5 Comparison of clustering results of 4 algorithms in Pathbased

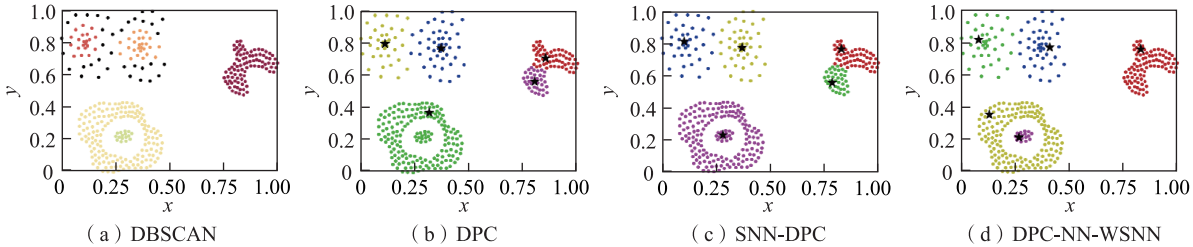

图6 4种算法在Compound-2的聚类结果对比

Fig. 6 Comparison of clustering results of 4 algorithms in Compound-2

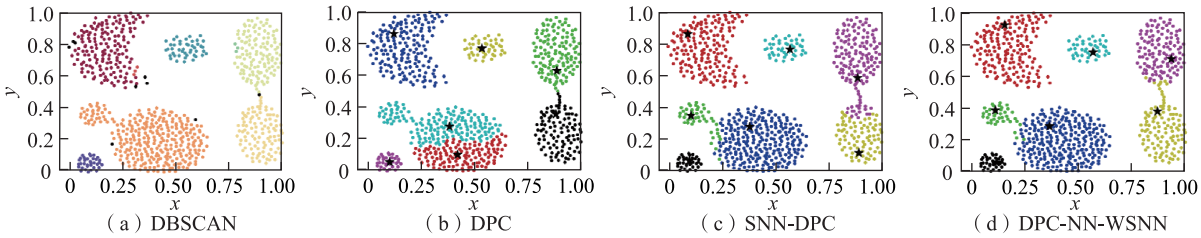

图7　4种算法在Aggregation的聚类结果对比

Fig. 7　Comparison of clustering results of 4 algorithms in Aggregation

表2　4种算法在4个数据集的评价指标对比

Tab.2　Comparison of evaluation indicators of 4 algorithms in 4 datasets

Datasets	Algorithms	AMI	ARI	FMI
Jain	DBSCAN	0.927 6	0.975 8	0.990 6
	DPC	0.576 1	0.618 3	0.838 6
	SNN-DPC	0.429 9	0.385 6	0.725 9
	DPC-NN-WSNN	1.000 0	1.000 0	1.000 0
Pathbased	DBSCAN	0.690 5	0.644 4	0.754 2
	DPC	0.536 0	0.453 0	0.658 5
	SNN-DPC	0.897 9	0.907 7	0.910 3
	DPC-NN-WSNN	0.902 4	0.929 4	0.953 3
Compound-2	DBSCAN	0.941 6	0.973 8	0.981 7
	DPC	0.860 7	0.818 8	0.874 6
	SNN-DPC	0.877 5	0.824 2	0.878 3
	DPC-NN-WSNN	1.000 0	1.000 0	1.000 0
Aggregation	DBSCAN	0.934 4	0.941 7	0.943 6
	DPC	0.892 1	0.755 0	0.807 4
	SNN-DPC	0.931 4	0.927 2	0.942 8
	DPC-NN-WSNN	0.954 8	0.959 4	0.968 1

Compound-2和Jain数据集上聚类准确性有着显著提升。故相比于其余3个聚类算法，本文提出的基于自然和加权共享最近邻的密度峰值聚类算法不仅能发现并利用数据集潜在的结构和形状信息，避免了稀疏簇的遗漏，同时也能更为准确地识别簇中心并且完成聚类后续标签的分配。

4　结论

采用在不同数据集上进行对比实验的方法，针对DPC各种不足之处进行了相关改进，得出以下结论。

1）为了使DPC算法能在识别不同形状和不同密度的簇时有更好的表现，在传统DPC算法的基础上提出了一种基于自然和加权共享最近邻的密度峰值聚类算法，在多个数据集上的实验结果表明，DPC-NN-WSNN算法在应对复杂形状和簇间稀疏程度差异大的数据上具有明显的优势，聚类精度更高。

2）该算法加入二阶共享最近邻导致时间复杂度较高，如何对该算法进行降维以适应高维数据集可以作为未来继续探索研究的方向。

3）该算法在处理包含噪声点和识别簇间有连接部分的数据集时聚类效果有所下降，如何去除噪声点的影响和对连接部分的数据点进行准确的分配可以作为未来继续探索研究的方向。

参考文献：

[1] 王森, 邢帅杰, 刘琛. 密度峰值聚类算法研究综述[J]. 华东交通大学学报, 2023, 40(1): 106-116.
WANG S, XING S J, LIU C. Survey of density peak clustering algorithm[J]. Journal of East China Jiaotong University, 2023, 40(1): 106-116.

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

[3] XING S, SU Q M, XIONG Y J, et al. PDCSN: a partition density clustering with self- adaptive neighborhoods[J]. Expert Systems with Applications, 2023, 227: 120195.

[4] LIU Y H, MA Z M, YU F. Adaptive density peak clustering based on K-nearest neighbors with aggregating strategy[J]. Knowledge-Based Systems, 2017, 133: 208-220.

[5] ZHANG Z Y, ZHU Q S, ZHU F, et al. Density decay graph-based density peak clustering[J]. Knowledge-Based Systems, 2021, 224: 107075.

[6] GUO W J, WANG W H, ZHAO S P, et al. Density peak clustering with connectivity estimation[J]. Knowledge-Based Systems, 2022, 243: 108501.

[7] CHENG D D, ZHANG S L, HUANG J L. Dense members of local cores- based density peaks clustering algorithm[J]. Knowledge-Based Systems, 2020, 193: 105454.

[8] LIU R, WANG H, YU X M. Shared- nearest- neighbor-

based clustering by fast search and find of density peaks[J]. Information Sciences, 2018, 450: 200-226.

[9] 王芙银, 张德生, 肖燕婷. 基于加权共享近邻与累加序列的密度峰值算法[J]. 计算机工程, 2022, 48(4): 61-69.
WANG F Y, ZHANG D S, XIAO Y T. Density peak algorithm based on weighted shared nearest neighbor and accumulated sequence[J]. Computer Engineering, 2022, 48 (4): 61-69.

[10] ZHU Q S, FENG J, HUANG J L. Natural neighbor: a self-adaptive neighborhood method without parameter K[J]. Pattern Recognition Letters, 2016, 80: 30-36.

[11] ZHAO J, WANG G, PAN J S, et al. Density peaks clustering algorithm based on fuzzy and weighted shared neighbor for uneven density datasets[J]. Pattern Recognition, 2023, 139: 109406.

[12] GUAN J Y, LI S, ZHU J H, et al. Fast main density peak clustering within relevant regions via a robust decision graph[J]. Pattern Recognition, 2024, 152: 110458.

[13] 徐晓, 丁世飞, 丁玲. 密度峰值聚类算法研究进展[J]. 软件学报, 2022, 33(5): 1800-1816.
XU X, DING S F, DING L. Survey on density peaks clustering algorithm[J]. Journal of Software, 2022, 33(5): 1800-1816.

[14] 孙林, 秦小营, 徐久成, 等. 基于K近邻和优化分配策略的密度峰值聚类算法[J]. 软件学报, 2022, 33(4): 1390-1411.
SUN L, QIN X Y, XU J C, et al. Density peak clustering algorithm based on *K*-nearest neighbors and optimized allocation strategy[J]. Journal of Software, 2022, 33(4): 1390-1411.

[15] 吕莉, 朱梅子, 康平, 等. 面向密度分布不均数据的混合近邻密度峰值聚类算法[J]. 控制理论与应用, 2024, 41 (10): 1821-1830.
LYU LI, ZHU M Z, KANG P, et al. Multiplex neighbor density peaks clustering for uneven density data sets[J]. Control Theory & Applications, 2024, 41(10): 1821-1830.

[16] 丁志成, 葛洪伟. 优化分配策略的密度峰值聚类算法[J]. 计算机科学与探索, 2020, 14(5): 792-802.
DING Z C, GE H W. Density peaks clustering with optimized allocation strategy[J]. Journal of Frontiers of Computer Science and Technology, 2020, 14(5): 792-802.

[17] 赵嘉, 姚占峰, 吕莉, 等. 基于相互邻近度的密度峰值聚类算法[J]. 控制与决策, 2021, 36(3): 543-552.
ZHAO J, YAO Z F, LYU L, et al. Density peaks clustering based on mutual neighbor degree[J]. Control and Decision, 2021, 36(3): 543-552.

第一作者:王森(1969—),男,教授,硕士生导师,研究方向计算机算法与应用。E-mail: 0012@ecjtu.edu.cn。

通信作者:陈翔(1998—),男,硕士研究生,研究方向为聚类分析与数据挖掘。E-mail: 2022088085410004@ecjtu.edu.cn。

(责任编辑:姜红贵)