软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2022,33(5):1800−1816 [doi: 10.13328/j.cnki.jos.006122]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

密度峰值聚类算法研究进展*

徐 晓[1], 丁世飞[1,2], 丁 玲[1]

[1](中国矿业大学 计算机科学与技术学院, 江苏 徐州 221116)

[2](矿山数字化教育部工程研究中心, 江苏 徐州 221116)

通信作者: 丁世飞, E-mail: dingsf@cumt.edu.cn

摘 要: 密度峰值聚类 (density peaks clustering, DPC) 算法是聚类分析中基于密度的一种新兴算法, 该算法考虑局部密度和相对距离绘制决策图, 快速识别簇中心, 完成聚类. DPC 具有唯一的输入参数, 且无需先验知识, 也无需迭代. 自 2014 年提出以来, DPC 引起了学者们的极大兴趣, 并得到了快速发展. 首先阐述 DPC 的基本理论, 并通过与经典聚类算法比较, 分析了 DPC 的特点; 其次, 分别从聚类精度和计算复杂度两个角度分析了 DPC 的弊端及其优化方法, 包括局部密度优化、分配策略优化、多密度峰优化以及计算复杂度优化, 并介绍了每个类别的主要代表算法; 最后介绍了 DPC 在不同领域中的相关应用研究. 对 DPC 的优缺点提供了全面的理论分析, 并对 DPC 的优化以及应用进行了全面阐述. 还试图找出进一步的挑战来促进 DPC 研究发展.

关键词: 密度峰值聚类; 聚类精度; 计算复杂度; 应用

中图法分类号: TP311

中文引用格式: 徐晓, 丁世飞, 丁玲. 密度峰值聚类算法研究进展. 软件学报, 2022, 33(5): 1800–1816. http://www.jos.org.cn/1000-9825/6122.htm

英文引用格式: Xu X, Ding SF, Ding L. Survey on Density Peaks Clustering Algorithm. Ruan Jian Xue Bao/Journal of Software, 2022, 33(5): 1800–1816 (in Chinese). http://www.jos.org.cn/1000-9825/6122.htm

Survey on Density Peaks Clustering Algorithm

XU Xiao[1], DING Shi-Fei[1,2], DING Ling[1]

[1](School of Computer Science and Technology, China University of Mining and Technology, Xuzhou 221116, China)

[2](Mine Digitization Engineering Research Center of the Ministry of Education, Xuzhou 221116, China)

Abstract: Density peaks clustering (DPC) algorithm is an emerging algorithm in density-based clustering analysis which draws a decision-graph based on the calculation of local-density and relative-distance to obtain the cluster centers fast. DPC is known as only one input parameter without prior knowledge and no iteration. Since DPC was introduced in 2014, it has attracted great interests and developments in recent years. This survey first analyzes the theory of DPC and the satisfactory behaviors of DPC by comparing it with classical clustering algorithms. Secondly, DPC survey is described in terms of clustering accuracy and computational complexity, including local-density optimization, allocation-strategy optimization, multi-density peaks optimization, and computational complexity optimization, to provide a clear organization. The main representative algorithms of each category are presented simultaneously. Finally, it introduces the related application research of DPC in different fields. This overview offers a comprehensive analysis for the advantages and disadvantages of DPC, and gives a comprehensive description for the improvements and applications of DPC. It is also attempted to find out some further challenges to promote DPC research.

Key words: density peaks clustering (DPC); clustering accuracy; computational complexity; application

随着互联网的高速发展, 生成数据的方式越来越多. 面对各种各样的数据, 有效且高效地挖掘大规模复杂数据成为技术改革的标志, 对于促进社会发展和创造产业价值变得越来越重要[1,2]. 聚类是一种重要的数据挖掘技术,

* 基金项目: 国家自然科学基金 (61976216, 61672522)
收稿时间: 2019-11-17; 修改时间: 2019-04-19, 2020-06-17; 采用时间: 2020-07-23; jos 在线出版时间: 2020-09-10

旨在识别隐藏在数据中的潜在模式[3]. 聚类主要应用于模式识别中的语音识别和字符识别、机器学习中的图像分割和机器视觉[4,5]. 另外, 聚类在统计学、生物学、心理学、考古学、地质学和地理学中也起着重要作用[6−9].

聚类是一种典型的无监督学习, 不需要任何的先验知识[10]. 簇是通过某种相似性度量得到的数据对象的集合[11]. 同一簇中的数据对象尽可能相似, 但要不同于其他簇中的对象[12]. 目前, 典型的聚类算法包括基于划分的 K-means[13]、基于层次的 CHAMELEON[14]、基于网格的 CLIQUE[15]、基于密度的 DBSCAN[16]以及基于图论的 Spectral Clustering(SC)[17]等. K-means 使用迭代方式将数据对象划分为簇, 最大程度地减少每个数据对象及其对应簇中心之间的差异之和. 然而, 它需要先验知识来预设不同的适当数量的簇, 并且不能获得非凸簇[18]. CHAMELEON 试图在不同级别划分数据对象以形成树状图聚类结构, 它不需要预先确定簇数, 并且可以找到非球形簇, 但是受参数设置的影响, 时间复杂度很高[19]. CLIQUE 从组织分布信息在块上的矩形块划分模式中获取围绕空间的网格结构的值以实现聚类, 算法对于大型高维空间数据的聚类是高效的, 但是聚类准确性不高[20]. SC 将数据集转换为图结构, 然后通过图划分的方式找到最佳子图以完成聚类. 然而, 传统的 SC 算法构造相似度矩阵和特征分解需要消耗大量资源, 并且它需要先验知识来预设不同的适当数量的簇[21]. 事实上, 大多数传统聚类算法, 在面对任意形状和密度的簇时都无法获得令人满意的聚类结果[22,23]. 然而, DBSCAN 作为一种基于密度的聚类算法, 它引入了密度可达的概念来定义核心对象, 当正确选择了半径参数 *Eps* 和最小样本数参数 *Minpts* 时, DBSCAN 可以在嘈杂的空间中找到形状各异的簇[24]. 但是, DBSCAN 在重叠密度方面表现不佳, 并且缺乏参数选择的理论基础[25].

Rodrigues 等人[26]提出了一种新颖的基于密度的聚类算法, 称为密度峰值聚类 (density peaks clustering, DPC) 算法, 该算法已证明其在解决非球形聚类问题上的能力, 并且较 DBSCAN 更易确定唯一输入参数. 自 2014 年发布以来, DPC 备受关注, 聚类准确性和计算效率均得到了提高, 优化的 DPC 算法可以刺激理论研究和实际应用[27,28]. 因此, 分析 DPC 的最新发展并研究其特性非常有意义. 首先, 我们对 DPC 进行了理论上的综合分析; 其次, 本研究以一种新的方式进行阐述, 根据 DPC 的缺点, 将 DPC 的优化算法分为面向聚类精度和面向计算效率的两大类, 面向聚类精度的优化又分为局部密度优化的 DPC 算法、分配策略优化的 DPC 算法、多密度峰优化的 DPC 算法. 具体来说: 首先, 局部密度计算是 DPC 的关键步骤, 需要根据数据的分布结构设计新的度量方式, 从而消除参数的敏感性并统一在不同规模数据集上的计算方法; 其次, 需要为非中心点分配更鲁棒的标记提高 DPC 的聚类精度; 第三, 当数据集中一个簇涉及多个密度峰时, 需要通过优化 DPC 识别准确的簇中心; 第四, 由于存储设备的限制, 需要降低 DPC 的计算复杂度以适应大规模数据集; 此外, 我们列举了相关算法的实验结果以进一步解释说明; 最后, 我们总结了 DPC 算法在实际推广中的应用以及未来的挑战. 本研究整体架构如图 1 所示.

1 DPC 聚类算法

密度峰值聚类是 2014 年在《Science》上发表的一种新颖的基于密度的聚类算法[29]. 为了找出 DPC 与传统聚类算法的区别, 本节介绍了 DPC 的理论基础和算法细节, 并对其性能和特点进行了综合分析.

1.1 DPC 算法原理

DPC 算法的关键是根据簇中心的特征绘制决策图, 以快速识别准确的簇中心[30]. 簇中心具有两大特征: 一, 簇中心被密度不超过它的邻居点包围, 因此簇中心的局部密度相对较大; 二, 簇中心的位置相对远离具有较高局部密度的任何其他对象, 即两个簇中心之间的距离相对较远[31]. 在 DPC 中, 搜索簇中心需要做两个准备工作.

1) 构造相似度矩阵[32]

探索合适的相似性度量方法是聚类执行的重要过程. 假设一个数据集 $X=\{x_1, x_2, \ldots, x_n\}$, n 是数据集的规模. 在 DPC 中, 欧几里得距离用于计算数据对象之间的相似度:

$$d_{ij} = \|x_i - x_j\|_2 \tag{1}$$

DPC 的相似度矩阵是通过计算所有数据对象之间的距离构建的:

$$D = [d_1, d_2, \ldots, d_n]^{\mathrm{T}} \in R_{n\times n} \tag{2}$$

其中, $d_i=[d_{i1}, d_{i2}, \ldots, d_{in}]$. D 是一个对称矩阵.

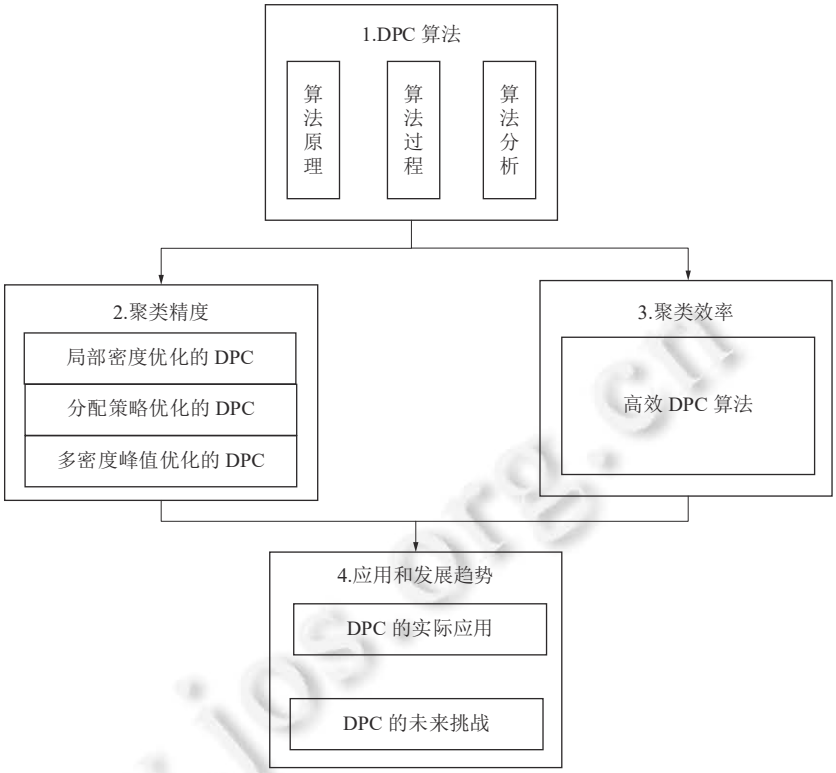


图 1 本文的整体架构

2) 计算局部密度和相对距离[33]

为了评估局部密度和相对距离, 为每个数据对象定义了相应的度量. 局部密度定义为:

$$\begin{cases}\rho_i=\sum_j \chi(d_{ij}-d_c)\\ \chi(x)=\begin{cases}1, & x<0\\ 0, & x\geqslant 0\end{cases}\end{cases} \tag{3}$$

其中, 参数 d_c 称为截止距离, 是唯一的输入参数. 针对不同的数据集, 需要根据经验设置不同的 d_c, 通常选择数据集总体相似性 2% 位置处的值. 另外, 当数据集规模较小时, ρ_i 可以通过引入高斯核函数来计算:

$$\rho_i=\sum_j \exp\left(-\frac{d_{ij}^2}{d_c^2}\right) \tag{4}$$

从公式 (3) 和公式 (4) 看出, ρ_i 表示距离 x_i 不超过 d_c 的所有数据对象的集合. 与此同时, 相对距离 δ_i 表示局部密度大于 x_i 且距离其最近点的距离, 定义为:

$$\delta_i=\begin{cases}\min\limits_{j:\rho_j>\rho_i}(d_{ij}), & \text{if } \exists j \text{ s.t. } \rho_j>\rho_i\\ \max\limits_j(d_{ij}), & \text{otherwise}\end{cases} \tag{5}$$

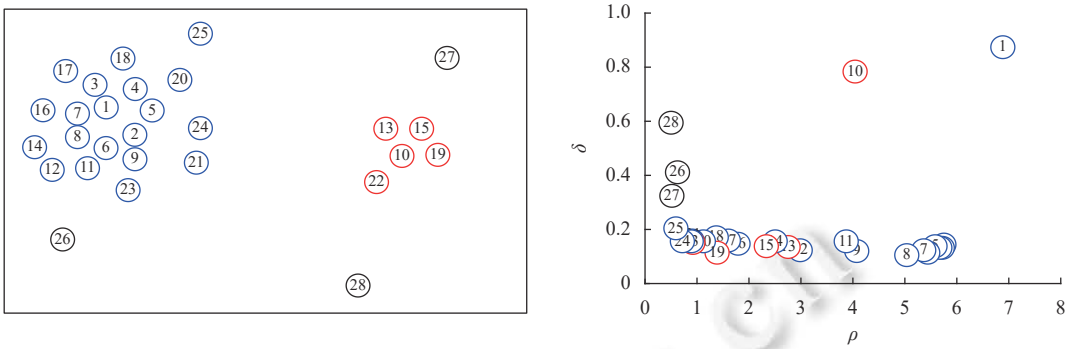
1.2 DPC 聚类过程

为数据集构建相似度矩阵并计算所有对象的局部密度和相对距离属性后, DPC 基于两个假设完成聚类: 其一, 簇中心是密度峰值; 其二, 非中心点与其最近的高密度点的簇相同[34]. 因此, DPC 执行两个关键步骤.

1) 簇中心识别[35]

根据计算的属性值, DCP 以 ρ 为横坐标、δ 为纵坐标绘制二维决策图, 并将所有数据对象映射到图中. 然后,

将具有较高 ρ_i 和 δ_i 的数据对象标识为簇中心. 例如, 图 2(a) 和图 2(b) 分别代表 28 个点的数据集分布和根据这些点绘制的决策图. 从图 2(b) 中可以看出, 点 1 和点 10 具有较高的 ρ_i 和 δ_i, 所以被选为簇中心.



(a) 数据集分布 (b) 决策图

图 2 DPC 绘制决策图

2) 非中心点聚类[36]

确定了簇中心, 其他非中心点被划分到距离其最近的高密度邻居的相同簇中. 此外, 从图 2(a) 中可以看出, 点 26~点 28 远离其他点, 是异常值. 为了将离群点分离, 定义属于该簇但是距离其他簇不超过 d_c 的点的集合为边界区域, 将边界区域中密度最高的点定义为 ρ_b. 簇中密度等于或小于 ρ_b 的对象被视为离群值.

DPC 的主要过程总结为算法 1.

算法 1. DPC 算法.

输入: 数据集 $X=\{x_1, x_2,\ldots, x_n\}$;

输出: 类别标签 Y.

步骤:

Step 1. 根据公式 (1)、公式 (2) 计算相似度矩阵 D;

Step 2. 根据公式 (3)–公式 (5) 计算局部密度 ρ_i 和相对距离 δ_i;

Step 3. 根据 ρ_i 和 δ_i 绘制二维决策图, 并标识具有高 ρ_i 和 δ_i 的点为簇中心;

Step 4. 分配非中心点到最近高密度点簇;

Step 5. 筛选局部密度不超过 ρ_b 的离群点;

Step 6. 输出数据划分结果.

1.3 DPC 优缺点

DPC 算法基于局部密度和相对距离识别簇中心, 因此可以处理非球形数据集, 并且利用局部密度特点, 可以很好地筛选离群点. DPC 根据决策图确定簇中心, 无需迭代, 无需先验知识, 且只有一个输入参数[37]. 为了进一步说明 DPC 的有效性, 我们参考文献 [26], 在 4 个非球形、簇间有重叠的人工数据集上比较了 DPC, SC 和 DBSCAN 这 3 种算法, 聚类结果如图 3–图 6 所示.

从图 3–图 6 可以看出:

(1) 在这 4 个数据集上, SC 不能获得令人满意的聚类结果. 因为该类数据集簇间有重叠, 并且 SC 在流形数据集上无法获得令人满意的聚类效果. 另外, SC 算法需要预先给定簇数.

(2) 同时, 由于数据集重叠, DBSCAN 在 Flame, Aggregation 和 S 上的性能不佳. 尽管在 Spiral 数据集上 DBSCAN 获得了满意的结果, 但难以调整其参数.

(3) DPC 算法在这 4 个数据集上获得了令人满意的聚类结果, 无需先验知识即可很好地处理非球形簇以及重叠簇, 并且具有良好的噪声处理能力.

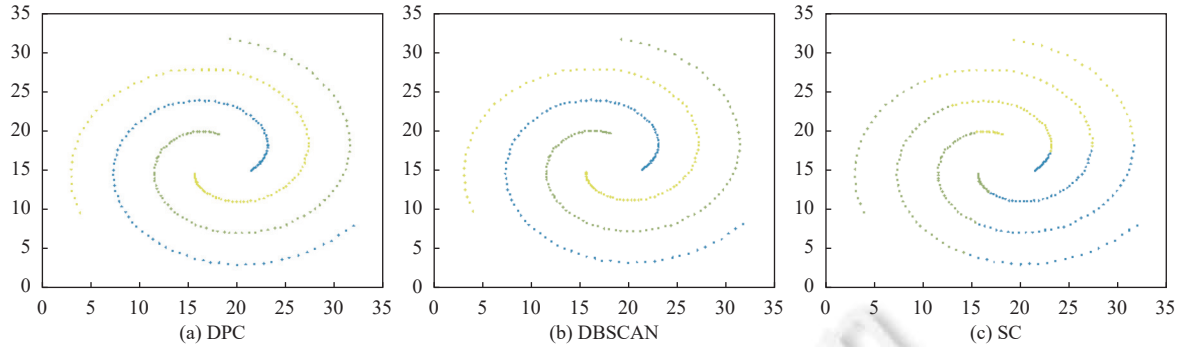


图 3　3 种算法在 Spiral 上的聚类结果

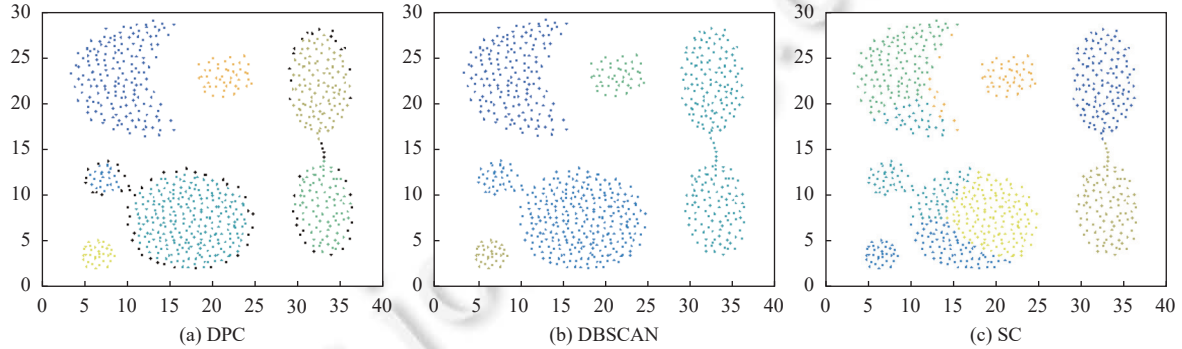


图 4　3 种算法在 Aggregation 上的聚类结果

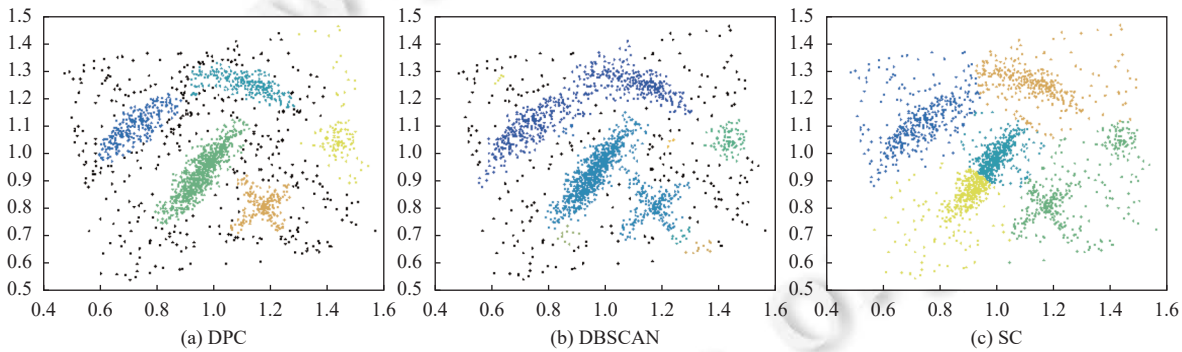


图 5　3 种算法在 S 上的聚类结果

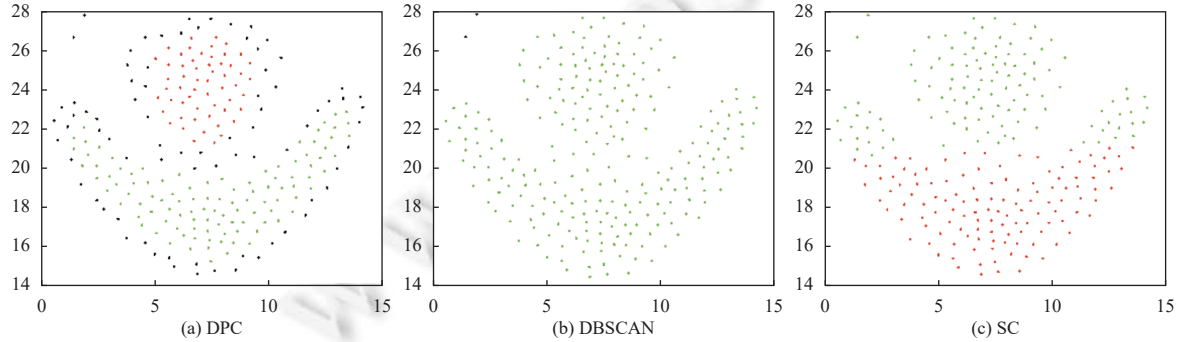


图 6　3 种算法在 Flame 上的聚类结果

理想情况下, 令人满意的算法应该是鲁棒、高效且有效的[38,39]. 尽管 DPC 算法简单有效, 在大部分数据集上表现出良好的聚类性能, 但是仍然存在许多问题. 我们主要从两个方面进行分析: 聚类精度和计算复杂度.

(1) 聚类精度主要考虑局部密度、多密度峰和分配策略 3 个方面.

第一, 根据公式 (3) 和公式 (4) 看出: 局部密度的估计没有统一的计算方法, 需要根据数据集的不同规模选择不同的函数, 获得不同的聚类结果[40]. 另外, 局部密度的计算依赖截止距离参数, 在小规模复杂数据集上, d_c 的变化会导致聚类结果的明显波动[41];

第二, DPC 采用一种单步分配策略, 该策略的容错性较差, 一旦一个对象被错误地聚类, 将导致更多的错误[42];

第三, 当数据集的一个簇中有多个密度峰时, DPC 无法获得准确的簇中心, 聚类的效果不理想[43].

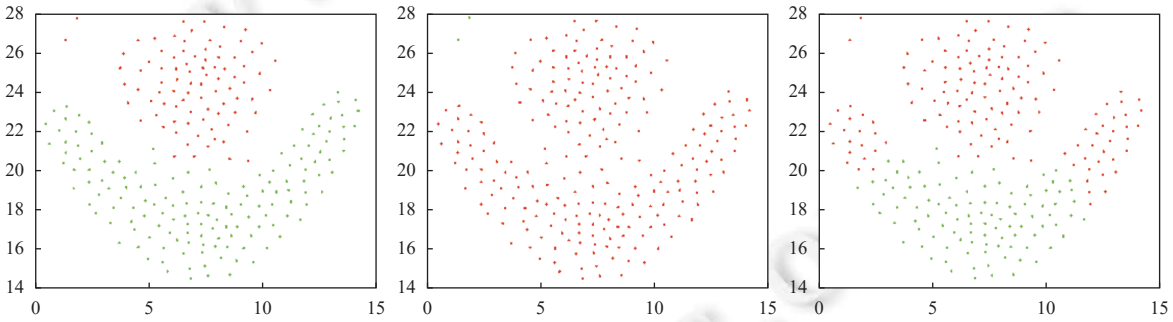
(2) 计算复杂度方面, 包括时间复杂度和空间复杂度. DPC 需要度量并存储所有数据对象之间的相似度, 以确定每个对象的局部密度和相对距离属性, 而这需要很高的计算成本, 限制了 DPC 在大规模数据集上的应用[44].

2 鲁棒的 DPC 聚类算法

从提高 DPC 聚类精度的角度来看, 相关研究主要可以分为 3 个方面: 局部密度度量的优化、合理的分配策略、簇中心的准确识别.

2.1 局部密度优化的 DPC

局部密度度量是 DPC 算法的关键步骤, 但是根据公式 (3) 和公式 (4) 发现: 局部密度的度量方式没有统一的计算方式, 需要根据数据集的大小选择不同的函数, 而数据集大小的判断没有具体的标准[45]. 另外, 局部密度的计算依赖于 d_c 的选择, 尤其是在小规模数据集上. 由于 d_c 的选取只考虑了数据的全局分布, 没有考虑数据集的局部性质, 因此聚类结果对 d_c 是敏感的[46]. 因此, DPC 在处理交叉缠绕或密度不均匀的数据集上效果不理想. 例如: 图 7 是一个小规模的数据集 Flame, 包含 2 个簇. 图 7(a) 通过公式 (4) 计算局部密度, 设置 d_c=4%. 图 7(b) 中, d_c 的取值与图 7(a) 相同, 但是局部密度通过公式 (3) 获取. 图 7(c) 以与图 7(a) 相同的方式计算局部密度, 但是 d_c 取 2%. 比较图 7(a)–图 7(c), 我们发现: 仅在图 7(a) 中, DPC 可以将数据集划分为两个令人满意的簇.



(a) Eq. (4) is used with d_c=4% (b) Eq. (3) is used with d_c=4% (c) Eq. (4) is used with d_c=2%

图 7 不同方式在 Flame 上的 DPC 聚类结果

图 7 再次证明: 不同的局部密度计算方法和不同的 d_c 会影响聚类结果, 尤其是在小规模数据集上. 为了消除聚类结果对 d_c 的敏感性并统一局部密度的计算方式, 最常见的方法是考虑数据集的局部结构, 并设计一种度量局部密度的新方法来代替 DPC 中的度量方法.

Mehmood 等人[47]提出了一种 CFSFDP-HD 算法优化 DPC 算法, 该算法通过一种非参数密度估计方法来估计局部密度: 首先, 通过优化的 Sheather-Jones 算法获得核密度估计的最佳带宽; 其次, 通过热扩散方程设计一种新的核密度估计形式. 所提出的算法不仅考虑了截止距离的选择, 而且考虑了核密度估计. 最终, 它根据 DPC 完成聚类. 与 DPC 相比, 该算法具有更高的鲁棒性和有效性. Wang 等人[48]提出了一种 STClu 算法, 采用基于密度指标的外部统计测试来识别簇中心: 首先, 它基于 K 密度定义一个新的度量, 以评估每个数据对象的密度和中心性; 其次, 它通过外部统计测试自动识别中心度指标大的对象为簇中心; 最后, 它将每个非中心点分配到具有较高密度的最

近邻居的簇. 所提出的算法不仅表现出比 DPC 更好的性能, 而且对参数不敏感. Liu 等人[49]提出了一种 ADPC-KNN 算法, 该算法基于 K 近邻的概念重新定义了局部密度的计算方法, 并自动识别初始簇中心, 以获得满意的结果: 首先, 基于高斯核和 KNN 的组合定义统一的新局部密度方法, 扩大了每个核心对象的密度与边界中其他对象的差异; 其次, 设计了一种新的簇中心初始化方法, 并引入了 DBSCAN 和 OPTICS 的思想来聚合多余的簇. 所提出的算法仅需要一个参数, 并且比 DPC 算法更鲁棒. Seyed 等人[50]提出了一种 DPC-DLP 算法, 该算法基于 KNN 重新定义了局部密度: 首先, 通过统一的局部密度计算方法确定簇中心; 其次, 将每个簇中心与其相邻簇中心结合起来, 构建一个 KNN 图, 并构建一个聚类主干; 第三, 将剩余数据对象通过新的基于图的标签传播方法分配标签. 优化的算法更适用于图像聚类和基因表达. Liu 等人[51]提出了一种 CCFDP 算法, 该算法以很少的约束信息即可准确执行: 首先, 基于 KNN 度量局部密度; 其次, 分析多个决策图以提取多个簇中心; 第三, 根据约束条件获得初始簇; 最终, 它基于 SVM 的思想获得最终的簇. 该算法有效地提出了半监督约束、层次聚类和密度聚类的组合. Du 等人[52]介绍了一种 DPC-KNN 算法, 该算法结合了 KNN 和 PCA 的概念优化 DPC: 首先, 该算法使用 KNN 的概念重新定义了局部密度, 以考虑数据集的局部分布; 其次, 提出了 DPC-KNN-PCA 算法将 PCA 引入 DPC-KNN 中, 以预处理高维数据集. 所提出的算法在处理密度不均匀数据集比 DPC 更鲁棒. Xie 等人[53]提出了一种 FKNN-DPC 算法, 它通过一种新颖的局部密度度量方式和一种新颖的分配策略来优化 DPC: 首先, 基于 KNN 设计了一种新的局部密度度量方法; 然后, 基于两种新策略分配其余非中心点. FKNN-DPC 中, 局部密度的度量标准独立于数据集的规模, 并且与截止距离无关. 因此, 这种局部密度的计算是 DPC 的优化但基于 DPC. 所提出的算法在识别任意形状和规模的簇方面比 DPC 更鲁棒. Liu 等人[54]提出了一种 SNN-DPC 算法, 该算法基于 SNN, 提高了 DPC 在多尺度、交叉缠绕和变化密度数据集上的优越性能: 首先, 提出了一种基于 SNN 的相似度计算新方法; 然后, 基于共享邻居提出了局部密度和相对距离度量; 其次, 引入了两种基于共享邻居信息的分配策略, 以提高非中心点分配的准确性. 局部密度和相对距离的新定义可以更客观地适应局部环境. 我们发现: 这类优化算法从根本上均是引入 KNN 重新定义了新的局部密度度量方式, 其中, SNN 也是增强的 KNN.

引入 KNN 重新定义局部密度, 不再仅考虑数据的全局结构, 而是引入数据集分布的局部信息, 可以有效统一局部密度的计算方式并消除参数 d_c 的敏感性, 有效提高 DPC 处理交叉数据集和密度不均匀数据集的聚类精度. 但是, 这些优化算法都引入了邻居数 k, 需要预先指定代表最近邻居数的参数. 尽管比起 DPC 依旧只有一个参数, 但是如何选择合适的 k 还需要进一步研究.

2.2 分配策略优化的 DPC

完成簇中心的识别任务后, DPC 采用单步分配策略, 即一次分配所有非簇中心. 该策略的容错性较差, 容易产生“多米诺效应”, 一旦一个数据对象被错误地聚类, 将导致更多与之相连的对象的错误划分[55]. 在 DPC 算法中, 将非中心点标识为具有较高密度且距离其最近的数据对象一簇. 假设存在一点 i, 点 i 必须被局部密度不超过它的邻居包围, 所以从该点到真实分布位于同一簇中的具有较高局部密度的点的最近距离较大. 如果此距离大于 i 到其他具有更高局部密度的其他簇的点的距离, 则会导致 i 的错误分配. 而由于 i 的邻居分配基于 i 的聚类信息, 所以一旦 i 的分配存在错误, 将影响其他相关样本的分配. 如图 8 所示: DPC 在选取簇中心后, 其他的非中心点分配到离其最近的高密度点一簇, 所以点 a 会被分配到点 b 一簇, 虽然点 a 应该与点 c 分配给相同的簇, 但是点 b 和点 c 的局部密度都高于点 a, 并且从点 a 到点 b 的距离小于从点 a 到点 c 的距离, 因此点 a 将被错误地分配到与点 b 同簇, 并且和点 a 相邻的点也将错误地分配. 因此, 此分配策略不可靠.

为了克服单步分配策略的差容错性, 通常采用多步分配策略来逐步分配非中心点, 边分配边调错, 提高分配的准确率. Seyed 等人[50]提出了一种基于图的动态标签分配策略来分配非中心点: 首先, 根据基于 K 近邻的数据集构造加权图; 其次, 使用半监督技术, 通过遍历图来分配未分配数据对象的标签. DPC-DLP 中提出的多步分配方法可以更新误分类对象的标签, 并完成对位于边界或重叠区域中的数据对象的正确分配. Xie 等人[53]提出了两种分配方法来研究数据集的结构: 在第 1 种方法中, 非异常值是通过 KNN 的方式从簇中心开始的广度优先搜索来分配的; 在第 2 种方法中, 未分配数据是通过模糊加权 K 最近邻居的方式分配的. 通过第 1 种方法标记的数据对象包括核心点, 而通过第 2 种方法标记的其余对象为噪声点. 这两种策略保证了所提出的 FKNN-DPC 算法的最高准确

性. Liu 等人[54]介绍了一种两步分配方法, 根据共享邻居的信息, 快速准确地找到满意的聚类分配. 将数据对象分为不可避免的从属点和可能的从属点. 首先, 为每对数据对象计算共享邻居信息, 以标识和分配确实属于该簇的对象; 其次, 它将剩余数据对象分配给包含更多邻居的簇. SNN-DPC 中, 基于共享邻居的分配方法可以避免 DPC 分配方法产生的问题.

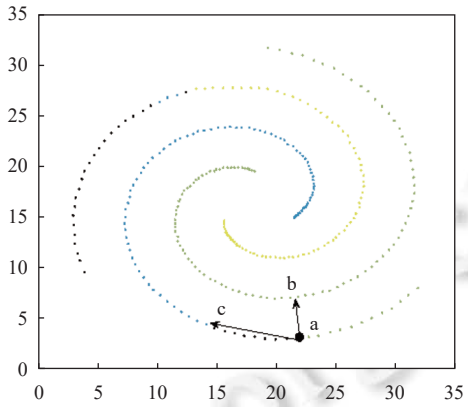


图 8 单步分配策略

为了进一步说明这些优化算法的有效性, 我们将上述 3 种算法与 DPC 在 5 个 UCI 数据集上进行了比较, 结果以众所周知的 FMI 索引显示[56], 见表 1.

表 1 5 个 UCI 数据集上 4 种算法的 FMI 值

Datasets	DPC-DLP	FKNN-DPC	**SNN-DPC**	DPC
Iris	0.947 8	0.935 5	**0.947 9**	0.923 3
Parkinson	0.800 5	0.658 2	**0.803 2**	0.618 7
Ecoli	0.800 8	0.691 9	**0.824 3**	0.577 5
WDBC	0.844 1	0.765 8	**0.930 5**	0.725 7
Seeds	0.849 5	0.827 6	**0.858 9**	0.844 4

从表 1 可以看出: 采用多步分配策略, 会比 DPC 获得更好的聚类精度. 这些提出的算法将非中心点分配与错误校正同时进行, 从而避免一步分配策略中出现的一步错步步错的现象. 其中, SNN-DPC 算法比其他的优化算法具有更令人满意的结果. 这不仅得益于其分配策略, 还因为 SNN 的概念是对 KNN 的增强. 另外, 这些分配策略均是基于准确识别簇中心的前提, 因此, 优化局部密度的度量方式仍然是提高 DPC 聚类精度的关键研究点.

2.3 多密度峰优化的 DPC

DPC 有一个严格的隐性要求, 即数据集中的每个簇有且只能有一个密度峰[57]. 因此, 当数据集中某个簇出现多个密度峰时, DPC 无法识别准确的簇中心, 聚类效果不理想. 例如, 在图 9(a) 中: 点 *a* 和点 *c* 分别属于两个簇, 而点 *a* 和点 *b* 属于同一个簇. 但是, 点 *a* 和点 *b* 是两个具有相同局部密度和相对距离的数据对象, 因此我们可以在一个簇中找到两个具有较高局部密度和相对距离的“异常大”的数据对象, 点 *a* 和点 *b* 会同时被识别为两个簇中心, 并将其他数据对象划分到这两个簇, 即点 *c* 划分到点 *a* 一簇.

由于不必要的伪簇中心, DPC 算法的性能会降低. DPC 基于有且只有一个密度峰的假设, 而图 9 中的 4 个数据集均是一个簇中包含有多个密度峰, 因此, DPC 在图 9 的 4 个数据集上聚类结果不佳. 为了克服多密度峰的问题, 通用方法是分而治之: 首先, 通过 DPC 尽可能多地初始化簇; 然后, 以递归的方式合并子簇获得最终的聚类结果. Liang 等人[58]提出了一种 3DC 算法, 该算法识别具有较高局部密度和相对距离的簇中心, 与 DPC 相同, 但又与 DPC 不同: 3DC 算法以递归方式选择簇中心, 而不是一次选择所有可能的簇中心. 首先, 3DC 算法根据原始数据集构造一个决策图, 仅选择两个具有较高局部密度和相对距离的最大点作为簇中心; 然后, 通过分配其余非中心点形成两个子簇; 之后, 递归处理两个子簇, 分别再聚成两个子簇. 该算法直到子簇满足 DBSCAN 中密度可达到的

要求, 终止该过程. Bie 等人[59]提出了一种模糊 CFSFDP 算法, 该算法首先找到所有簇, 然后自适应地合并多余的子簇: 首先, 它将每个密度峰标识为单个簇; 其次, 它将其余对象分配给本地簇, 不包括噪声和簇中心; 第三, 当子簇的峰值接近并且子簇在共享边界位置具有平均密度时, 合并这些子簇. Gao 等人[60]提出了一种称为 ICFS 的自动算法, 包括预聚类、合并和拆分阶段. 因此, ICFS 可以自动修改簇并增强 DPC 算法: 首先, 它利用优化的 DPC 算法获得初始簇并完成预聚类, 与 DPC 不同, 此增强算法使用新方法重新定义并自动选择簇中心, 此外, 其余对象将根据新策略分配; 其次, 使用合并和拆分方法来自动调整簇的结构. Zhang 等人[61]提出了一种 E_CFSFDP 算法, 该算法首先使用 DPC 生成粗糙簇, 然后根据基于层次的聚类将合适的子簇合并以形成正确的聚类划分. 它通过 DPC 尽可能多地初始化簇, 然后提出了 KNN 图的变体, 以优化 CHAMELEON 合并子簇. Chen 等人[62]提出了一种 CLUB 算法, 该算法将中心替换为密度主干以找到簇: 首先, 通过相互邻近的 K 个邻居找到初始簇; 然后, 以初始簇为输入, 通过 K 近邻来识别密度主干, 在此步骤中, CLUB 可以划分初始簇; 第三, 它将每个未标记的对象分配给包含其最近的较高密度邻居的簇; 最后, 过滤掉每个簇中的离群值. 所提出的算法有效、鲁棒, 并且可以找到具有各种密度和任意形状的簇. Xu 等人[63]提出了一种 FDPC 算法, 该算法利用 DPC 和 SVM 分割并合并簇, 以获得准确的数据划分: 首先, 它通过 DPC 尽可能多地初始化簇; 其次, 根据支持向量计算每两个簇的反馈值; 最后, 通过反馈值递归地合并子簇, 获得令人满意的聚类结果.

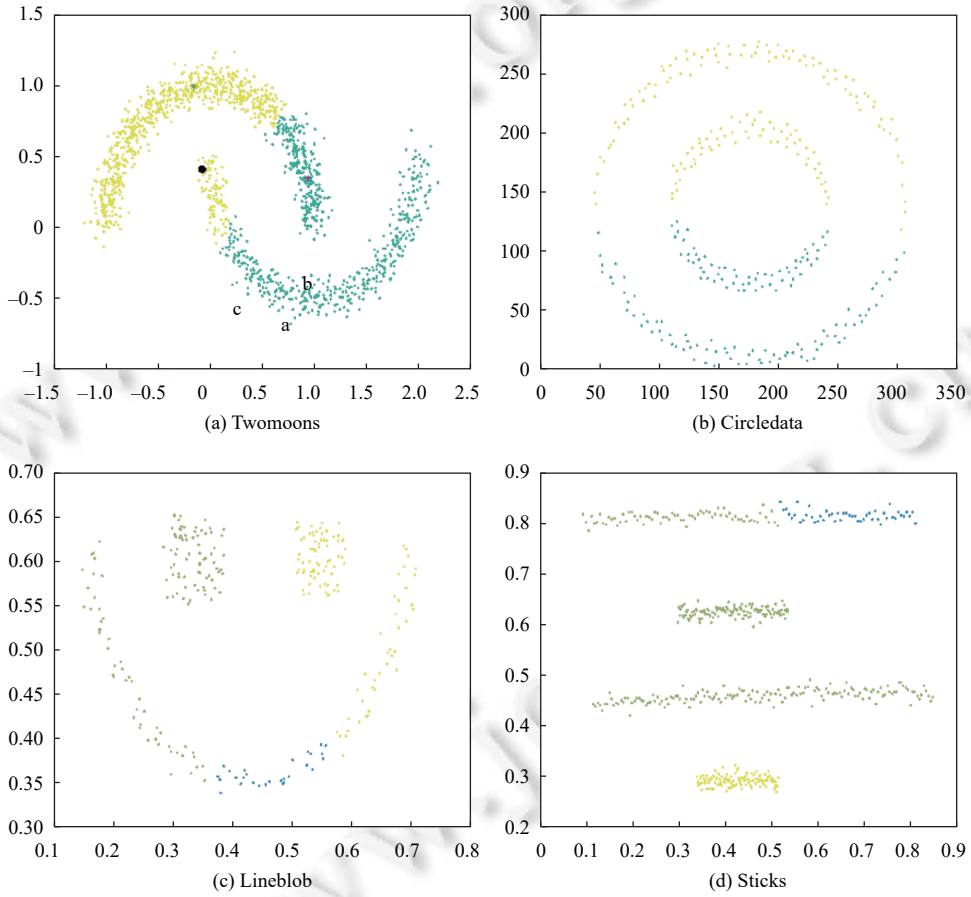


图 9 DPC 在 4 个合成数据集上的聚类结果

基于分治策略优化的 DPC 算法可以处理同一簇中出现多个密度峰的数据集. 结合拆分和合并的过程获得合适的簇数, 提高多密度峰数据集的聚类精度. 如图 10 所示, 在这些数据集上均可以获得令人满意的聚类结果. 但是, 这些算法将引入其他参数来确定是否合并和拆分子簇, 无多余参数的多密度峰优化的算法需进一步探索.

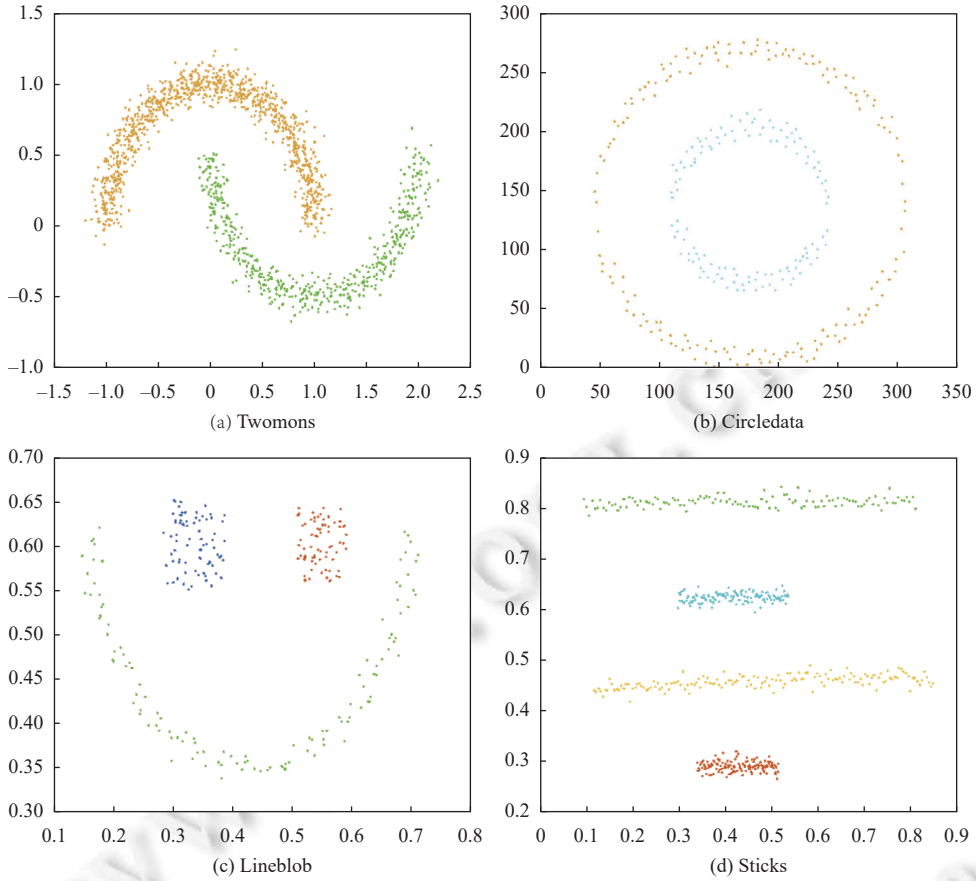


图 10 在 4 个合成数据集上的聚类结果

3 高效的 DPC 聚类算法

DPC 需要度量任意一对数据对象之间的距离构建相似度矩阵, 从而确定每个数据对象的局部密度和相对距离属性来识别簇中心, 而这需要很高的计算成本[64]. 假设有一个数据集 $X=\{x_1, x_2, \ldots, x_n\}$, n 是数据集的规模. 计算局部密度和相对距离的时间复杂度为 $O(n^2)$. 分配策略会产生 $O(n)$ 的时间复杂度. 总的来说, DPC 需要消耗时间复杂度 $O(n^2)$. 与时间复杂度相同, DPC 空间复杂度 $O(n^2)$ 主要来源是存储相似度矩阵. 随着数据集规模的扩大, DPC 的应用受到很大限制. 为了克服这个问题, 许多学者进行了减少相似度计算的研究.

Xu 等人[65]提出了一种 DPCG 算法, 它引入了非空网格的思想, 用网格对象代替对应的数据对象, 以减少距离计算: 首先, 它将数据集分配到相应的网格空间中; 其次, 它计算每个网格单元的信息; 第三, 每个非空网格单元看作 DPC 聚类的数据对象执行 DPC, 以获得令人满意的结果. 假设 $m(<<N)$ 表示非空网格单元的规模, 因为 DPCG 只对非空网格单元执行聚类操作, 所以仅需要 $O(m^2)$ 空间复杂度和时间复杂度. DPCG 与 DPC 具有相同的优势, 但是它不需要计算所有数据对象之间距离, 可以大大降低原始 DPC 算法的计算复杂度.

Gong 等人[66]提出了一种基于分布式计算的 EDDPC 算法, 该算法通过 Voronoi 分割、数据复制和数据过滤的方式避免了大规模的距离计算和数据传输耗时. 它首先使用储层采样选择 Voronoi 的种子对象和合适的 d_c; 其次, 它创建一个 MapReduce 作业以计算 ρ 及其复制模型; 第三, 执行两个 MapReduce 作业以进行复制、过滤和计算 δ. 假设 α, β 分别是计算 ρ 和 δ 的复制因子, 并且 N 表示 Voronoi 分组的大小. 因此, 用于计算 ρ 需要的距离成本为 $O(\alpha \cdot n^2/N)$, 用于计算 δ 需要的距离成本为 $O(\beta \cdot n^2/N)$, 与 DPC 相比大大减少了距离成本的计算量. 同时, 采用

分布式处理也将大大降低空间复杂度.

Bai 等人[67]提出了 CFSFDP+A, CFSFDP+DE 算法, 两个算法依靠 K-means 的优势来提高 DPC 的聚类效率. 首先, 利用近似概念设计 CFSFDP+A 算法, 它搜索近似空间以进行计算并通过距离进行估计; 另外, 由于它需要很少的计算时间来计算更少的距离, 因此可以与 DPC 快速获得相近的聚类结果; 其次, 利用示例聚类的概念设计 CFSFDP+DE 算法, 进一步提高了 DPC 的可扩展性. 它可以使用 DPC 快速获得相似的聚类结果. 假设 k_m(<<N) 是随机选择的初始簇中心, 并且 t 表示迭代次数. $n1$ (<<N), $n2$(<<N) 是 CFSFDP+A 中用于计算 ρ 和 δ 的平均距离数. 因此, 在 CFSFDP+A 中需要距离计算 O(Nk_mt+$Nn1$+$Nn2$), CFSFDP+DE 中需要 O($Nk_mt+k_m^2$). 对于大规模据集, CFSFDP+DE 比 CFSFDP+A 更可扩展. CFSFDP+A 和 CFSFDP+DE 需要空间复杂度 O(Nk_m). 所以, CFSFDP+A 和 CFSFDP+DE 算法比 DPC 算法更适用于处理大规模数据集.

Xu 等人[68]提出了两种策略, 分别称为 GDPC 和 CDPC, 它们通过保留部分有效数据对象代替所有数据对象识别簇中心完成聚类. 首先, GDPC 和 CDPC 分别使用网格和圆对数据集进行划分预处理: GDPC 算法利用网格划分筛选去除小密度网格单元中的数据对象, CDPC 使用圆划分法筛选去除小密度圆中的数据对象, 网格划分和圆划分都基于数据集的分布. GDPC 和 CDPC 筛选去除的数据对象不满足 DPC 中的高局部密度特性, 因此可以在保证聚类精度的情况下提高算法的效率. 假设 m(<<N) 表示筛选后保留数据对象的规模, 则 GDPC 的空间复杂度为 O(m^2). 假设数据集被划分为 k 个簇, GDPC 的时间复杂度由 O(N)+O(Nlog$N$)+O(m^2)+O(k(N−m)) 这 4 个部分组成. 增加 O(N) 的空间复杂度来判断 CDPC 算法中数据对象的稀疏性. 假设 g 代表相应的圆个数. CDPC 的时间复杂度将略小于 O(N(g+1))+O(Nlog$N$)+O(m^2)+O(k(N−m)). 与 GDPC 相比, 尽管 CDPC 具有更高的计算复杂度, 但对于大规模数据集而言更为精确. GDPC 和 CDPC 均提高了 DPC 处理大规模数据集的性能.

从计算复杂度的分析来看, 减少距离的计算可以有效增强 DPC 处理大规模数据集的效率. 但是, 计算复杂度的降低会影响一定的聚类精度, 而精度的保证也会影响计算复杂度. 提高聚类精度的同时降低计算复杂度的 DPC 算法还需要进一步探索.

4 DPC 的应用

DPC 算法依靠决策图快速识别簇中心, 简单有效, 能很好地处理噪声孤立点, 可以得到任意形状的簇; 同时, 不需要提前指定簇的数量, 并且需要用户指定的参数较少, 因此, 近几年得到了很多学者的关注, 在聚类中起着重要作用[69−72]. DPC 已被扩展优化并广泛应用在各个领域, 例如社区检测、机器故障诊断、患者分层以及室内定位等.

推荐系统作为一种高效便捷的自动化信息筛选工具而受到了广泛的关注和使用. 不同于传统的搜索引擎, 推荐系统能主动地为用户提供准确并且个性化的推荐服务[73]. 最近, 各种社交媒体网站, 如 Twitter、腾讯和新浪微博, 已成为提供流行服务的重要信息平台, 采用推荐系统为用户推荐好友以及内容[74]. 推荐算法作为推荐系统的中枢, 其选取以及优化程度的不同都将直接影响推荐系统的性能. 尽管最近的策略产生了积极的影响, 但是大多数推荐系统仍然需要进一步提高[75]. Zheng 等人[76]提出了一种基于 HIOC 检测方法的个性化推荐模型, 该模型可以为用户有效地推荐相关的和多样的兴趣: 首先, 它设计了一个多粒度相似度来描述用户之间的相似关系; 然后, 基于 DPC 算法, 根据节点的兴趣密度发展了 HIOC 检测方法; 最后, 实现了一种新颖的推荐模型, 为用户提供多粒度语义相关的主题. 与传统的方法 LPA, CPM 以及 LFM 等相比, HIOC 检测方法在精度、召回率和 $F1$ 度量方面具有更好的性能. 这是由于基于密度峰值的核心方法可以选择稳定的社区结构.

社交网络分析已逐渐成为学术界和工业界的热点问题, 由于社区结构通常代表具有相似属性、爱好或亲密关系的特定组织用户群体, 是社交网络的重要属性, 因此社区发现对于了解复杂网络的特征、发现潜在的拓扑结构、预测网络的发展等至关重要[77]. 社区发现的目的是找到一群具有相似想法、信念、动机或其他共同特征的用户, 以便更好地理解社交网络[78]. 然而, 现有的社区发现方法大多数仅考虑结构特征, 可能忽略许多与社区相关联的信息[79]. Wang 等人[80]提出了一种基于 DPC 算法的社区发现优化算法, 该算法利用了用户的个人资料和拓扑信息: 首先, 基于 in-link Salton 和 out-link Salton 进行用户的拓扑结构构建; 其次, 介绍了一种新颖的密度估计并增加社交圈整合步骤, 以优化 DPC 算法. 与传统的方法 COPRA, CONCLUDE, DCM 以及 CLUTO 等相比, 所提出的方

法面对复杂的重叠社交网络, 在 BER 和 $F1$ 分数方面具有更好的结果. 通过使用基于密度峰值的优化聚类方法, 可以对社交圈进行整合, 快速、准确地检测出重叠的社交圈.

机器在现代工业过程中起着重要作用, 因此, 保证机器运行对整个工业过程至关重要. 传统方法通常需要专业技能和丰富经验[81]. 为了能够自动、可靠、快速处理大量的工业数据并自动检测有用的故障特征, 在智能故障诊断领域进行了许多关于聚类或分类的研究[82]. 然而, 机械设备的故障样本实际上非常少, 所以强烈需要构建更加可靠、智能和自动的过程以进行机器状态监测和诊断[83]. Wang 等人[84]提出了一种 3 阶段方法来诊断 3 个特定工业案例的智能故障: 首先, 针对自适应聚类故障类别, 提出了一种基于 DPC 的优化 ADPS 算法; 其次, 通过基于 VMD 的特征趋势提取以及自加权算法提高了算法性能. 提出的故障诊断方法无论是在轴承故障诊断、齿轮故障诊断还是从运行到失败测试的轴承诊断案例中, 准确率均大于 99%; 并且与传统的 DPC, AP, K-medoids 等聚类方法相比无需先验知识, 这种优化的算法可以满足工业在线应用的需求.

患者分层是现代医学保证药物开发成功直至临床验证的主要挑战, 在实现高效个性化医疗方面起着重要作用[85]. 患者分层的一项重要任务是发现有效治疗的疾病亚型. 患者分层的目标是开发有效的计算方法, 为亚型进行精确的临床决策[86]. Li 等人[87]在 DPC 算法的基础上提出了一种新的多目标算法, 用于患者分层: 首先, 它选择特征并通过对象的相异权重和相关的截止点自动计算聚类密度; 其次, 使用 5 个合适的聚类索引来度量固有聚类; 最后, 采用 MOEA/D-DE 来优化此目标功能. 无论是与传统的 KM, AL, SL, CL, DBSCAN, LCEASRS, ECC, DPC 等聚类方法, 还是与一些最新的 NSGAII, GrEA, HyEA, SPEA2, MOPSO 等多目标进化算法相比, 提出的方法在真实人类转录调控网络的合成数据集、真实患者分层数据集以及真实世界医学数据集中, 都体现了多目标框架的竞争优势.

随着 WiFi 技术的发展变得越来越成熟, WiFi 热点的覆盖范围也在增加. 如果我们能够充分利用分布在室内环境中的 WiFi 热点, 将为我们的生活带来极大的便利[88]. 由于 WiFi 的信号强度不需要使用特殊的硬件设备, 因此将 RSSI 用于室内定位突出了其明显的优势[89]. 但是在室内环境中, 由于接收器信号弱、环境噪声、多径干扰和非视线传播, 传统定位算法存在许多问题[90]. Meng 等人[91]提出了一种无线室内定位算法: 首先, BP 和 RBF 用于构建室内定位模型; 然后, 使用加权中值-高斯滤波方法预处理 RSSI, 并建立位置指纹数据库; 其次, 利用优化的 DPC 算法和 LM 算法设计了新的 RBF 模型. 使用优化的 DPC 方法来确定 RBF 神经网络的结构, 无论是路由器对称分布还是路由器均匀分布的情况下, 都比基于 LM 算法的 BP 配置以及 RBF 神经网络获得的定位误差要小. 这主要是因为网络结构和参数是通过优化的 DPC 确定的, 可以实现快速拟合, 更适用于室内定位.

5 总结与展望

DPC 作为当前最热门的基于密度的聚类算法之一, 已经吸引了越来越多学术界和工业界学者对其不断的研究与发展. 本文从理论和方法的角度介绍了 DPC 算法的发展概况: 从理论的角度, 我们综合分析了 DPC 的原理和 DPC 的优缺点; 从方法的角度来看, 我们尽力提供清晰的分类, 即, 根据 DPC 缺点将所有优化的 DPC 算法分为面向聚类精度优化和面向计算效率优化的算法. 其中: 面向聚类精度的优化主要包括基于局部密度的优化、基于分配策略的优化以及基于多密度峰的优化, 面向计算效率的优化同时降低了时间复杂度和空间复杂度. 对于每个类别, 我们列出了一些典型的算法进行总结和分析, 并给出了一些实验结果对比证明. 最后, 本文介绍了 DPC 在实际推广中的重要应用. 本研究可以促进 DPC 理论方法和实际应用的进一步扩展.

虽然 DPC 的主要弊端已经得到改进, 且具有广泛的应用场景, 但是该算法的开发时间不长, 仍然存在很多问题. 通过以上分析, 大多数优化算法引入了多余参数以提高 DPC 的性能.DPC 是典型的基于密度的聚类算法, 其困难在于如何保持聚类精度的同时追求算法的高效性. 同时, 如何不引入多余参数自动确定簇的数量并识别簇中心, 实现真正意义上的自动 DPC, 依旧有待研究. 另外, DPC 处理高维稀疏数据效果不令人满意. 如何利用深度学习有效提取数据特征, 探索 DPC 在不同领域的更多应用, 有待进一步研究.

1) 高效 DPC 算法

随着 Internet 的发展, 有效并高效地挖掘大规模数据变得越来越重要. DPC 作为基于密度的聚类算法, 面临复杂度较高的问题, 在确保其准确性的情况下追求高效性具有一定的挑战性. 虽然目前从计算复杂度的优化来看已

经提出一些有效降低复杂度的方法, 但是计算复杂度的降低影响了一定的聚类精度, 没有真正实现提高聚类精度的同时降低计算复杂度.

DPC 聚类算法的主要步骤是, 根据簇中心的特征构造决策图来搜索簇中心. 然而在计算每个数据对象的特征属性时, 高度依赖两两数据对象之间的相似度, 这也是 DPC 复杂度的主要构成部分. 减少数据对象之间的距离计算, 可以有效提高 DPC 处理大规模数据集的效率. 如何保证计算局部密度与相对距离需要的距离都被度量是关键. 根据簇中心两大特征的定义可以发现: 局部密度表示距离“当前点”不超过 d_c 的数据对象的集合, 相对距离表示局部密度大于“当前点”的最近距离. 因此, DPC 的局部密度和相对距离的度量仅取决于“当前点”与“最近邻点”的相似度. 然而计算近邻点会引入近邻参数, 因此可以考虑判断数据对象的不相似性. 如何在不增加计算复杂的情况下进行不相似度的判断, 需要进一步研究. 同时, 由于 DPC 中每个数据对象的局部密度与相对距离的计算均是独立的, 研究分布式处理方法对数据进行聚类有一定价值, 也可以利用 GPU 并行化处理 DPC.

2) 自动 DPC 算法

DPC 聚类算法可以通过绘制决策图快速识别簇中心, 但簇中心的准确识别仅限于处理的数据集中每个簇是均匀分布的, 并且人为地选择簇中心会有一定的主观性. 因此, 自动选择簇中心是必然趋势. 目前虽有作者提出一定的方法解决簇中心的自动选择问题, 但是主要集中于两个方法: 预先固定簇数目、增加参数判断决策图中的“拐点”. 此两种方法均增加了额外的参数, 并且人为地规定簇数目从本质上并不属于真正意义上的“自动”.

聚类分析是一种无监督学习, 评估聚类质量有两种标准: 内部质量评价指标和外部评价指标. 而内部评价指标是利用数据集本身的性质评价算法的聚类好坏, 通过计算簇间以及簇内平均相似度评价聚类质量. 而通常这类指标结合的是内聚度和分离度两种因素, 无需引入多余的参数, 因此可以根据聚类结果自动判断最优的簇数目. 但是现有的内部质量评价指标并不能很好地应用于 DPC, 而且复杂度较高. 因此, 根据 DPC 的局部密度和相对距离的特点设计一种符合 DPC 的聚类指标, 并且不增加复杂度, 可以实现真正的自动聚类.

3) 深度 DPC 算法

DPC 算法计算局部密度和相对距离绘制决策图完成聚类, 主要是通过将高维数据映射到 2 维空间进行簇中心的选择. 然而高维数据往往是稀疏的, 映射到 2 维空间进行簇中心的选择常常无法获得理想表现. 目前, 面对高维数据, DPC 首先通过 PCA 对数据进行降维预处理, 但在高维, 特别是图像数据上表现依旧不够理想.

随着深度学习的快速发展, 深度聚类也逐渐引起了广大学者的关注. 深度聚类学习算法在数据表征的基础上进一步平衡数据表征之间的维度, 完成高维稀疏大数据到标签的映射, 并进行优化预测. 但是, 目前的深度聚类模型需要先验地设置聚类数, 并且在高维稀疏数据上的聚类效果还需进一步提高. 卷积自编码器利用了传统自编码器的无监督的学习方式, 结合了卷积神经网络的卷积和池化操作, 从而对图像可以很好地保留空间信息, 实现特征提取. 因此, 基于卷积自编码器的特征学习能力构建深度 DPC 聚类模型, 使构造的聚类模型摆脱先验知识的限制, 并高效处理高维稀疏大数据, 是一个重要的研究方向.

References:

[1] Xie DY, Gao QX, Wang QQ, Zhang XD, Gao XB. Adaptive latent similarity learning for multi-view clustering. Neural Networks, 2020, 121: 409–418. [doi: 10.1016/j.neunet.2019.09.013]

[2] Tsakiris M, Vidal R. Theoretical analysis of sparse subspace clustering with missing entries. In: Proc. of the 35th Int'l Conf. on Machine Learning. Stockholm: PMLR, 2018. 4975−4984.

[3] Deng TQ, Ye DS, Ma R, Fujita H, Xiong LN. Low-rank local tangent space embedding for subspace clustering. Information Sciences, 2020, 508: 1−21. [DOI: 10.1016/j.ins.2019.08.060] [doi: 10.1016/j.ins.2019.08.060]

[4] Wang CD, Lai JH, Zhu JY. Graph-based multiprototype competitive learning and its applications. IEEE Trans. on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 2012, 42(6): 934–946. [doi: 10.1109/TSMCC.2011.2174633]

[5] Yan XQ, Ye YD, Qiu XY, Yu H. Synergetic information bottleneck for joint multi-view and ensemble clustering. Information Fusion, 2020, 56: 15–27. [doi: 10.1016/j.inffus.2019.10.006]

[6] Pothula KR, Smyrnova D, Schröder GF. Clustering cryo-EM images of helical protein polymers for helical reconstructions. Ultramicroscopy, 2019, 203: 132–138. [doi: 10.1016/j.ultramic.2018.12.009]

[7] Llobell F, Vigneau E, Qannari EM. Clustering datasets by means of CLUSTATIS with identification of atypical datasets. Application to sensometrics. Food Quality and Preference, 2019, 75: 97–104. [doi: 10.1016/j.foodqual.2019.02.017]

[8] Hu SB, Chen ZT, Nia VP, Chan LW, Geng YH. Causal inference and mechanism clustering of a mixture of additive noise models. In: Proc. of the 32nd Int'l Conf. on Neural Information Processing Systems. Montreal: Curran Associates Inc. , 2018. 5212−5222. [doi: 10.5555/3327345.3327427]

[9] Baranwal M, Salapaka S. Clustering and supervisory voltage control in power systems. Int'l Journal of Electrical Power & Energy Systems, 2019, 109: 641–651. [doi: 10.1016/j.ijepes.2019.02.025]

[10] Zhang YP, Zhou J, Deng ZH, Zhong FL, Jiang YZ, Hang WL, Wang ST. Multi-view fuzzy clustering approach based on medoid invariant constraint. Ruan Jian Xue Bao/Journal of Software, 2019, 30(2): 282−301 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/5625.htm [doi: 10.13328/j.cnki.jos.005625]

[11] Wang CD, Lai JH, Suen CY, Zhu JY. Multi-exemplar affinity propagation. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2013, 35(9): 2223–2237. [doi: 10.1109/TPAMI.2013.28]

[12] Wang CD, Lai JH, Huang D, Zeng WS. SVStream: A support vector-based algorithm for clustering data streams. IEEE Trans. on Knowledge and Data Engineering, 2013, 25(6): 1410–1424. [doi: 10.1109/TKDE.2011.263]

[13] Zhao WL, Deng CH, Ngo CW. *k*-Means: A revisit. Neurocomputing, 2018, 291: 195–206. [doi: 10.1016/j.neucom.2018.02.072]

[14] He SN, Ji B, Chan SHG. Chameleon: Survey-free updating of a fingerprint database for indoor localization. IEEE Pervasive Computing, 2016, 15(4): 66–75. [doi: 10.1109/MPRV.2016.69]

[15] Segundo PS, Rodriguez-Losada D. Robust global feature based data association with a sparse bit optimized maximum clique algorithm. IEEE Trans. on Robotics, 2013, 29(5): 1332–1339. [doi: 10.1109/TRO.2013.2264869]

[16] Ester M, Kriegel HP, Sander J, Xu XW. A density-based algorithm for discovering clusters in large spatial databases with noise. In: Proc. of the 2nd Int'l Conf. on Knowledge Discovery and Data Mining. Portland: AAAI Press, 1996. 226−231. [doi: 10.5555/3001460.3001507]

[17] Zhang YL, Rohe K. Understanding regularized spectral clustering via graph conductance. In: Proc. of the 32nd Int'l Conf. on Neural Information Processing Systems. Montreal: Curran Associates Inc. , 2018. 10654−10663. [doi: 10.5555/3327546.3327723]

[18] Ismkhan H. I-K-Means−+: An iterative clustering algorithm based on an enhanced version of the *k*-means. Pattern Recognition, 2018, 79: 402–413. [doi: 10.1016/j.patcog.2018.02.015]

[19] Jiang YT, Ma XY, Shao XJ, Wang MY, Jiang Y, Miao P. Chameleon silver nanoclusters for ratiometric sensing of miRNA. Sensors and Actuators B: Chemical, 2019, 297: 126788. [doi: 10.1016/j.snb.2019.126788]

[20] Bohn B, Garcke J, Griebel M. A sparse grid based method for generative dimensionality reduction of high-dimensional data. Journal of Computational Physics, 2016, 309: 1–17. [doi: 10.1016/j.jcp.2015.12.033]

[21] Huang J, Nie FP, Huang H. Spectral rotation versus *k*-means in spectral clustering. In: Proc. of the 27th AAAI Conf. on Artificial Intelligence. Bellevue: AAAI Press, 2013. 431−437.

[22] Lv YH, Ma TH, Tang ML, Cao J, Tian Y, Al-Dhelaan A, Al-Rodhaan M. An efficient and scalable density-based clustering algorithm for datasets with complex structures. Neurocomputing, 2016, 171: 9−22. [doi: 10.1016/j.neucom.2015.05.109]

[23] Zhu Y, Ting KM, Carman MJ. Density-ratio based clustering for discovering clusters with varying densities. Pattern Recognition, 2016, 60: 983–997. [doi: 10.1016/j.patcog.2016.07.007]

[24] Chen YW, Tang SY, Bouguila N, Wang C, Du JX, Li HL. A fast clustering algorithm based on pruning unnecessary distance computations in DBSCAN for high-dimensional data. Pattern Recognition, 2018, 83: 375–387. [doi: 10.1016/j.patcog.2018.05.030]

[25] Bryant A, Cios K. RNN-DBSCAN: A density-based clustering algorithm using reverse nearest neighbor density estimates. IEEE Trans. on Knowledge and Data Engineering, 2018, 30(6): 1109–1121. [doi: 10.1109/TKDE.2017.2787640]

[26] Rodriguez A, Laio A. Clustering by fast search and find of density peaks. Science, 2014, 344(6191): 1492–1496. [doi: 10.1126/science.1242072]

[27] Chen JY, He HH. A fast density-based data stream clustering algorithm with cluster centers self-determined for mixed data. Information Sciences, 2016, 345: 271–293. [doi: 10.1016/j.ins.2016.01.071]

[28] Li ZJ, Tang YC. Comparative density peaks clustering. Expert Systems with Applications, 2018, 95: 236–247. [doi: 10.1016/j.eswa.2017.11.020]

[29] Ding JJ, Chen ZT, He XX, Zhan YZ. Clustering by finding density peaks based on Chebyshev's inequality. In: Proc. of the 35th Chinese Control Conf. Chengdu: IEEE, 2016. 7169−7172. [doi: 10.1109/ChiCC.2016.7554490]

[30] Wang M, Min F, Zhang ZH, Wu YX. Active learning through density clustering. Expert Systems with Applications, 2017, 85: 305–317. [doi: 10.1016/j.eswa.2017.05.046]

[31] Du MJ, Ding SF, Xu X, Xue Y. Density peaks clustering using geodesic distances. Int'l Journal of Machine Learning and Cybernetics,

2018, 9(8): 1335–1349. [doi: 10.1007/s13042-017-0648-x]

[32] Xu J, Wang GY, Li TR, Deng WH, Gou GL. Fat node leading tree for data stream clustering with density peaks. Knowledge-based Systems, 2017, 120: 99–117. [doi: 10.1016/j.knosys.2016.12.025]

[33] Xu XH, Ju YS, Liang YL, He P. Manifold density peaks clustering algorithm. In: Proc. of the 3rd Int'l Conf. on Advanced Cloud and Big Data. Yangzhou: IEEE, 2015. 311−318. [doi: 10.1109/CBD.2015.57]

[34] Xie JY, Gao HC, Xie WX. *K*-nearest neighbors optimized clustering algorithm by fast search and finding the density peaks of a dataset. Scientia Sinica Informationis, 2016, 46(2): 258–280 (in Chinese with English abstract). [doi: 10.1360/N112015-00135]

[35] Jiang JH, Chen YJ, Hao DH, Li KQ. DPC-LG: Density peaks clustering based on logistic distribution and gravitation. Physica A: Statistical Mechanics and its Applications, 2019, 514: 25–35. [doi: 10.1016/j.physa.2018.09.002]

[36] Chen JY, Lin X, Zheng HB, Bao XT. A novel cluster center fast determination clustering algorithm. Applied Soft Computing, 2017, 57: 539–555. [doi: 10.1016/j.asoc.2017.04.031]

[37] Zhang WK. Research on density-based hierarchical clustering algorithm [MS. Thesis]. Hefei: University of Science and Technology of China, 2015 (in Chinese with English abstract).

[38] Ye XL, Zhao JY, Zhang L, Guo LJ. A nonparametric deep generative model for multimanifold clustering. IEEE Trans. on Cybernetics, 2019, 49(7): 2664–2677. [doi: 10.1109/TCYB.2018.2832171]

[39] Zhang T, Ji P, Harandi M, Huang WB, Li HD. Neural collaborative subspace clustering. In: Proc. of the 36th Int'l Conf. on Machine Learning. Long Beach: PMLR, 2019. 7384−7393.

[40] Xu J, Wang GY, Deng WH. DenPEHC: Density peak based efficient hierarchical clustering. Information Sciences, 2016, 373: 200–218. [doi: 10.1016/j.ins.2016.08.086]

[41] Yan HQ, Wang L, Lu YG. Identifying cluster centroids from decision graph automatically using a statistical outlier detection method. Neurocomputing, 2019, 329: 348–358. [doi: 10.1016/j.neucom.2018.10.067]

[42] Xu X, Ding SF, Sun TF, Liao HM. Large-scale density peaks clustering algorithm based on grid screening. Journal of Computer Research and Development, 2018, 55(11): 2419–2429 (in Chinese with English abstract). [doi: 10.7544/issn1000-1239.2018.20170227]

[43] Wang XF, Xu YF. Fast clustering using adaptive density peak detection. Statistical Methods in Medical Research, 2017, 26(6): 2800–2811. [doi: 10.1177/0962280215609948]

[44] Wu B, Wilamowski BM. A fast density and grid based clustering method for data with arbitrary shapes and noise. IEEE Trans. on Industrial Informatics, 2017, 13(4): 1620–1628. [doi: 10.1109/TII.2016.2628747]

[45] Hou J, Cui HX. Experimental evaluation of a density kernel in clustering. In: Proc. of the 7th Int'l Conf. on Intelligent Control and Information Processing. Siem Reap: IEEE, 2016. 55−59. [doi: 10.1109/ICICIP.2016.7885876]

[46] Yang XH, Zhu QP, Huang YJ, Xiao J, Wang L, Tong FC. Parameter-free Laplacian centrality peaks clustering. Pattern Recognition Letters, 2017, 100: 167–173. [doi: 10.1016/j.patrec.2017.10.025]

[47] Mehmood R, Zhang GZ, Bie RF, Dawood H, Ahmad H. Clustering by fast search and find of density peaks via heat diffusion. Neurocomputing, 2016, 208: 210–217. [doi: 10.1016/j.neucom.2016.01.102]

[48] Wang GT, Song QB. Automatic clustering via outward statistical testing on density metrics. IEEE Trans. on Knowledge and Data Engineering, 2016, 28(8): 1971–1985. [doi: 10.1109/TKDE.2016.2535209]

[49] Liu YH, Ma ZM, Yu F. Adaptive density peak clustering based on *K*-nearest neighbors with aggregating strategy. Knowledge-based Systems, 2017, 133: 208–220. [doi: 10.1016/j.knosys.2017.07.010]

[50] Seyedi SA, Lotfi A, Moradi P, Qader NN. Dynamic graph-based label propagation for density peaks clustering. Expert Systems with Applications, 2019, 115: 314–328. [doi: 10.1016/j.eswa.2018.07.075]

[51] Liu RH, Huang WP, Fei ZS, Wang K, Liang J. Constraint-based clustering by fast search and find of density peaks. Neurocomputing, 2019, 330: 223–237. [doi: 10.1016/j.neucom.2018.06.058]

[52] Du MJ, Ding SF, Jia HJ. Study on density peaks clustering based on *k*-nearest neighbors and principal component analysis. Knowledge-based Systems, 2016, 99: 135–145. [doi: 10.1016/j.knosys.2016.02.001]

[53] Xie JY, Gao HC, Xie WX, Liu XH, Grant PW. Robust clustering by detecting density peaks and assigning points based on fuzzy weighted *K*-nearest neighbors. Information Sciences, 2016, 354: 19–40. [doi: 10.1016/j.ins.2016.03.011]

[54] Liu R, Wang H, Yu XM. Shared-nearest-neighbor-based clustering by fast search and find of density peaks. Information Sciences, 2018, 450: 200–226. [doi: 10.1016/j.ins.2018.03.031]

[55] Zhao L, Chen ZK, Yang Y, Zou L, Wang ZJ. ICFS clustering with multiple representatives for large data. IEEE Trans. on Neural Networks and Learning Systems, 2019, 30(3): 728–738. [doi: 10.1109/TNNLS.2018.2851979]

[56] Milligan GW, Schilling DA. Asymptotic and finite sample characteristics of four external criterion measures. Multivariate Behavioral Research, 1985, 20(1): 97–109. [doi: 10.1207/s15327906mbr2001_6]

[57] Masud A, Huang JZ, Wei CH, Wang JK, Khan I, Zhong M. I-nice: A new approach for identifying the number of clusters and initial cluster centres. Information Sciences, 2018, 466: 129–151. [doi: 10.1016/j.ins.2018.07.034]

[58] Liang Z, Chen P. Delta-density based clustering with a divide-and-conquer strategy: 3DC clustering. Pattern Recognition Letters, 2016, 73: 52–59. [doi: 10.1016/j.patrec.2016.01.009]

[59] Bie RF, Mehmood R, Ruan S, Sun YC, Dawood H. Adaptive fuzzy clustering by fast search and find of density peaks. Personal and Ubiquitous Computing, 2016, 20(5): 785–793. [doi: 10.1007/s00779-016-0954-4]

[60] Gao J, Zhao L, Chen ZK, Li P, Xu H, Hu YM. ICFS: An improved fast search and find of density peaks clustering algorithm. In: Proc. of the 14th IEEE Int'l Conf. on Dependable, Autonomic and Secure Computing, 14th Int'l Conf. on Pervasive Intelligence and Computing, 2nd Int'l Conf. on Big Data Intelligence and Computing and Cyber Science and Technology Congress. Auckland: IEEE, 2016. 537−543. [doi: 10.1109/DASC-PICom-DataCom-CyberSciTec.2016.103]

[61] Zhang WK, Li J. Extended fast search clustering algorithm: Widely density clusters, no density peaks. Computer Science, 2015, 5(7): 1–17. [doi: 10.5121/csit.2015.50701]

[62] Chen M, Li LJ, Wang B, Cheng JJ, Pan LN, Chen XY. Effectively clustering by finding density backbone based-on *k*NN. Pattern Recognition, 2016, 60: 486–498. [doi: 10.1016/j.patcog.2016.04.018]

[63] Xu X, Ding SF, Xu H, Liao HM, Xue Y. A feasible density peaks clustering algorithm with a merging strategy. Soft Computing, 2019, 23(13): 5171–5183. [doi: 10.1007/s00500-018-3183-0]

[64] Wang J, Zhu C, Zhou Y, Zhu XQ, Wang YL, Zhang WM. From partition-based clustering to density-based clustering: Fast find clusters with diverse shapes and densities in spatial databases. IEEE Access, 2018, 6: 1718–1729. [doi: 10.1109/ACCESS.2017.2780109]

[65] Xu X, Ding SF, Du MJ, Xue Y. DPCG: An efficient density peaks clustering algorithm based on grid. Int'l Journal of Machine Learning and Cybernetics, 2018, 9(5): 743–754. [doi: 10.1007/s13042-016-0603-2]

[66] Gong SF, Zhang YF. EDDPC: An efficient distributed density peaks clustering algorithm. Journal of Computer Research and Development, 2016, 53(6): 1400–1409 (in Chinese with English abstract). [doi: 10.7544/issn1000-1239.2016.20150616]

[67] Bai L, Cheng XQ, Liang JY, Shen HW, Guo YK. Fast density clustering strategies based on the *k*-means algorithm. Pattern Recognition, 2017, 71: 375–386. [doi: 10.1016/j.patcog.2017.06.023]

[68] Xu X, Ding SF, Shi ZZ. An improved density peaks clustering algorithm with fast finding cluster centers. Knowledge-based Systems, 2018, 158: 65–74. [doi: 10.1016/j.knosys.2018.05.034]

[69] Ding SF, Du MJ, Sun TF, Xu X, Xue Y. An entropy-based density peaks clustering algorithm for mixed type data employing fuzzy neighborhood. Knowledge-based Systems, 2017, 133: 294–313. [doi: 10.1016/j.knosys.2017.07.027]

[70] Zeng XH, Chen AZ, Zhou M. Color perception algorithm of medical images using density peak based hierarchical clustering. Biomedical Signal Processing and Control, 2019, 48: 69–79. [doi: 10.1016/j.bspc.2018.09.013]

[71] Qiu L, Fang F, Yuan SF. Improved density peak clustering-based adaptive Gaussian mixture model for damage monitoring in aircraft structures under time-varying conditions. Mechanical Systems and Signal Processing, 2019, 126: 281–304. [doi: 10.1016/j.ymssp.2019.01.034]

[72] Tu B, Zhang XF, Kang XD, Zhang GY, Li ST. Density peak-based noisy label detection for hyperspectral image classification. IEEE Trans. on Geoscience and Remote Sensing, 2019, 57(3): 1573–1584. [doi: 10.1109/TGRS.2018.2867444]

[73] Bai L, Cheng XQ, Liang JY, Guo YK. Fast graph clustering with a new description model for community detection. Information Sciences, 2017, 388−389: 37−47. [doi: 10.1016/j.ins.2017.01.026]

[74] Bai X, Cambazoglu BB, Gullo F, Mantrach A, Silvestri F. Exploiting search history of users for news personalization. Information Sciences, 2017, 385−386: 125−137. [doi: 10.1016/j.ins.2016.12.038]

[75] Bernabé-Moreno J, Tejeda-Lorente A, Porcel C, Fujita H, Herrera-Viedma E. Quantifying the emotional impact of events on locations with social media. Knowledge-based Systems, 2018, 146: 44–57. [doi: 10.1016/j.knosys.2018.01.029]

[76] Zheng JX, Wang SG, Li DY, Zhang BF. Personalized recommendation based on hierarchical interest overlapping community. Information Sciences, 2019, 479: 55–75. [doi: 10.1016/j.ins.2018.11.054]

[77] Gong WH, Chen YQ, Pei XB, Yang LH. Community detection of multi-dimensional relationships in location-based social networks. Ruan Jian Xue Bao/Journal of Software, 2018, 29(4): 1163−1176 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/5269.htm [doi: 10.13328/j.cnki.jos.005269]

[78] Schall D. Who to follow recommendation in large-scale online development communities. Information and Software Technology, 2014, 56(12): 1543–1555. [doi: 10.1016/j.infsof.2013.12.003]

[79] Wang WY, Jiang YC. Community-Aware task allocation for social networked multiagent systems. IEEE Trans. on Cybernetics, 2014, 44(9): 1529–1543. [doi: 10.1109/TCYB.2013.2289327]

[80] Wang MM, Zuo WL, Wang Y. An improved density peaks-based clustering method for social circle discovery in social networks.

Neurocomputing, 2016, 179: 219–227. [doi: 10.1016/j.neucom.2015.11.091]
[81] Wang YX, Xiang JW, Markert R, Liang M. Spectral kurtosis for fault detection, diagnosis and prognostics of rotating machines: A review with applications. Mechanical Systems and Signal Processing, 2016, 66−67: 679−698. [doi: 10.1016/j.ymssp.2015.04.039]
[82] Zhong JJ, Tse PW, Wei YH. An intelligent and improved density and distance-based clustering approach for industrial survey data classification. Expert Systems with Applications, 2017, 68: 21–28. [doi: 10.1016/j.eswa.2016.10.005]
[83] Li WH, Zhang SH, Rakheja S. Feature denoising and nearest-farthest distance preserving projection for machine fault diagnosis. IEEE Trans. on Industrial Informatics, 2016, 12(1): 393–404. [doi: 10.1109/TII.2015.2475219]
[84] Wang YX, Wei ZX, Yang JW. Feature trend extraction and adaptive density peaks search for intelligent fault diagnosis of machines. IEEE Trans. on Industrial Informatics, 2019, 15(1): 105–115. [doi: 10.1109/TII.2018.2810226]
[85] Khakabimamaghani S, Ester M. Bayesian biclustering for patient stratification. In: Proc. of the Pacific Symp. on Biocomputing 2016. Kohala Coast: PSB, 2016. 345−356.
[86] Wu J, Cui Y, Sun XL, Cao GH, Li BL, Ikeda DM, Kurian AW, Li RJ. Unsupervised clustering of quantitative image phenotypes reveals breast cancer subtypes with distinct prognoses and molecular pathways. Clinical Cancer Research, 2017, 23(13): 3334–3342. [doi: 10.1158/1078-0432.CCR-16-2415]
[87] Li XT, Wong KC. Evolutionary multiobjective clustering and its applications to patient stratification. IEEE Trans. on Cybernetics, 2019, 49(5): 1680–1693. [doi: 10.1109/TCYB.2018.2817480]
[88] Zhang W, Liu K, Zhang WD, Zhang YM, Gu J. Deep neural networks for wireless localization in indoor and outdoor environments. Neurocomputing, 2016, 194: 279–287. [doi: 10.1016/j.neucom.2016.02.055]
[89] Wang XY, Gao LJ, Mao SW, Pandey S. CSI-based fingerprinting for indoor localization: A deep learning approach. IEEE Trans. on Vehicular Technology, 2017, 66(1): 763–776. [doi: 10.1109/TVT.2016.2545523]
[90] Xiao J, Zhou ZM, Yi YW, Ni LM. A survey on wireless indoor localization from the device perspective. ACM Computing Surveys, 2016, 49(2): 25. [doi: 10.1145/2933232]
[91] Meng H, Yuan F, Yan TH, Zeng MF. Indoor positioning of RBF neural network based on improved fast clustering algorithm combined with LM algorithm. IEEE Access, 2019, 7: 5932–5945. [doi: 10.1109/ACCESS.2018.2888616]

附中文参考文献:

[10] 张远鹏, 周洁, 邓赵红, 钟富礼, 蒋亦樟, 杭文龙, 王士同. 代表点一致性约束的多视角模糊聚类算法. 软件学报, 2019, 30(2): 282−301. http://www.jos.org.cn/1000-9825/5625.htm [doi: 10.13328/j.cnki.jos.005625]
[34] 谢娟英, 高红超, 谢维信. *K*近邻优化的密度峰值快速搜索聚类算法. 中国科学: 信息科学, 2016, 46(2): 258–280. [doi: 10.1360/N112015-00135]
[37] 张文开. 基于密度的层次聚类算法研究 [硕士学位论文]. 合肥: 中国科学技术大学, 2015.
[42] 徐晓, 丁世飞, 孙统风, 廖红梅. 基于网格筛选的大规模密度峰值聚类算法. 计算机研究与发展, 2018, 55(11): 2419–2429. [doi: 10.7544/issn1000-1239.2018.20170227]
[66] 巩树凤, 张岩峰. EDDPC: 一种高效的分布式密度中心聚类算法. 计算机研究与发展, 2016, 53(6): 1400–1409. [doi: 10.7544/issn1000-1239.2016.20150616]
[77] 龚卫华, 陈彦强, 裴小兵, 杨良怀. LBSN中融合多维关系的社区发现方法. 软件学报, 2018, 29(4): 1163−1176. http://www.jos.org.cn/1000-9825/5269.htm [doi: 10.13328/j.cnki.jos.005269]

徐晓(1992－), 女, 博士, 讲师, 主要研究领域为机器学习, 聚类分析.

丁玲(1994－), 女, 博士生, 主要研究领域为机器学习, 聚类分析.

丁世飞(1963－), 男, 博士, 教授, 博士生导师, 主要研究领域为人工智能与模式识别, 机器学习, 数据挖掘.