

面向流形数据的自然近邻图优化和微簇合并 密度峰值聚类算法

赵 嘉^{1,2}, 何超凡^{1,2}, 肖人彬³, 樊棠怀^{1,2}, 潘正祥⁴

(1. 江西水利电力大学 信息工程学院, 南昌 330099; 2. 江西省水利大数据智能处理与预警技术工程研究中心, 南昌 330099; 3. 华中科技大学 人工智能与自动化学院, 武汉 430074; 4. 南京信息工程大学 人工智能学院, 南京 210044)

摘 要: 针对密度峰值聚类算法难以发现流形数据的密度峰值; 分配策略易将远离密度峰值的样本错误分配的问题, 提出一种面向流形数据的自然近邻图优化和微簇合并密度峰值聚类算法。首先, 基于自然近邻图, 利用顶点间的测地距离设计了一种新的局部密度度量方法, 精准刻画流形数据的内部结构与分布特性; 其次, 通过分析自然近邻图中顶点的连接关系, 自动识别样本代表并确定局部核心, 用以指导微簇划分; 最后, 利用样本间测地距离定义一种新的微簇间相似性度量准则, 用以指导微簇合并, 从而优化聚类效果。将该算法与 4 个改进密度峰值聚类算法以及密度峰值聚类算法进行对比, 实验结果表明, 本文算法能有效应用于流形数据和真实数据的聚类分析当中。

关键词: 密度峰值聚类; 聚类算法; 流形数据; 自然近邻图; 测地距离; 微簇合并

中图分类号: TP311 **文献标志码:** A **文章编号:** 1671-5497(2026)08-2219-10

DOI: 10.13229/j.cnki.jdxbgxb.20250043

Density peaks clustering algorithm based on natural neighbor graph optimization and micro-cluster merging for manifold data

ZHAO Jia^{1,2}, HE Chao-fan^{1,2}, XIAO Ren-bin³, FAN Tang-huai^{1,2}, PAN Zheng-xiang⁴

(1. School of Information Engineering, Jiangxi University of Water Resources and Electric Power, Nanchang 330099, China; 2. Jiangxi Province Engineering Research Center for Intelligent Processing and Early Warning Technology of Water Conservancy Big Data, Nanchang 330099, China; 3. School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan 430074, China; 4. School of Artificial Intelligence, Nanjing University of Information Science and Technology, Nanjing 210044, China)

Abstract: Density Peak Clustering algorithm faces challenges in detecting density peaks in manifold data, and its assignment strategy often misallocates samples far from density peaks. To address these issues, this paper proposes a Natural Neighbor Graph Optimization and Micro-Cluster Merging Density Peak Clustering algorithm for manifold data. First, based on the natural neighbor graph, a novel local density measurement method is designed using geodesic distances between vertices, which accurately characterizes

收稿日期: 2025-01-13.

基金项目: 国家自然科学基金项目 (62466037, 62463021, 52069014).

作者简介: 赵嘉 (1981-), 男, 教授, 博士. 研究方向: 智能计算与计算智能, 模式识别, 大数据挖掘.

E-mail: zhaojia@juwp.edu.cn

the internal structure and distribution properties of manifold data. Second, by analyzing the connection relationships between vertices in the natural neighbor graph, the algorithm automatically identifies representative samples and determines local cores to guide micro-cluster partitioning. Finally, a new similarity measurement criterion for micro-clusters is defined based on geodesic distances between samples, which optimizes clustering performance through micro-cluster merging. The proposed algorithm is compared with four improved density peaks clustering algorithms and the original density peaks clustering algorithm. Experimental results show that the proposed algorithm can be effectively applied to clustering analysis of manifold data and real-world datasets.

Key words: density peak clustering; clustering algorithm; manifold data; natural neighbor graph; geodesic distance; micro-cluster merging

0 引言

聚类作为一种典型的无监督学习方法,将对象分为彼此相似的类簇,是数据挖掘和模式识别领域的热点。聚类算法问世至今,学者们提出了许多具有不同特征的聚类算法,按照它们的数据规模、应用场景、数据类型可以概括为以下几类:划分聚类^[1]、密度聚类^[2]、网格聚类^[3]、层次聚类^[4]等。随着聚类算法的逐渐成熟,其在交通工程^[5]、图像分割^[6]、生物医疗^[7]、推荐系统^[8]等领域均有广泛应用。

密度峰值聚类(Density peak clustering, DPC)算法^[9]是2014年由两位科学家提出,并发表在Science上的一种新颖聚类技术。该方法主要依据样本数量在特定区域的集中程度和样本之间的距离关系来构建决策图,通过决策图直观选取密度峰值,然后将非密度峰值按照密度高且距离近的规则分配到各类簇。DPC算法可以在没有先验知识的情况下快速识别具有任意形状类簇,获得不错的聚类性能。但DPC算法仍存在以下不足:①定义局部密度时未考虑样本的整体分布特征,导致易错选密度峰值^[10];②数据集呈流形分布时,由于有较多样本距离密度峰值较远,分配非密度峰值时可能有大量样本被错误分配^[11]。

针对DPC算法易错选密度峰值的问题,Du等^[12]综合了DPC聚类算法的快速性和模糊连接点算法的鲁棒性,可以较好发现不同形状数据的密度峰值;Xie等^[13]引入局部标准差测量密度,可同时在稀疏和密集类簇中找到密度峰值;吕莉等^[14]通过自然近邻信息,调整样本在不同区域内的局部密度,从而减少稀疏与密集区域之间的密度差异,以便更精确识别稀疏区域的特征;Chen

等^[15]提出了一种借助 K 近邻的域适应密度评估方式,能够有效识别各个密度区域中的密度峰值。这些算法虽相对DPC算法有更好的聚类精度,但面对复杂形态的数据集,如流形数据时仍存在不足,难以保证聚类效果。

针对DPC算法错误分配非密度峰值的问题,Zhang等^[16]使用 K 归纳法分配剩余样本,可有效避免错误信息的传递,防止“链式反应”现象发生;Li等^[17]利用经典的图形化标签传播机制,结合对称密度邻居图进行样本分配,使聚类过程中的标签传递更加稳定和准确,从而减少了“链式错误”的风险;赵嘉等^[18]采用邻近样本的信息界定样本之间的相似性,避免欧氏距离对非密度峰值样本分配的影响;Zang等^[19]将数据点分为“已确认依赖点”和“不确定依赖点”,并采用混合策略进行分配,减少了错误传播。这些算法虽可避免DPC算法分配剩余样本时产生的链式分配错误,提高DPC算法聚类分析的表现,但未能充分考虑样本在整体空间中的分布特点,因此在处理流形类型的数据时结果并不理想。

流形数据是由许多弯曲或环形的类簇组成,这类数据在自然界中常见,例如数字识别^[20]、医疗图像分析^[21]及语音处理^[22]等领域。DPC算法在计算局部密度时只聚焦于一定范围内的样本,但流形数据通常是复杂、非凸的,欧氏距离通常难以准确衡量样本间的全局关系;分配策略未考虑流形数据剩余样本远离密度峰值的特性,对于尚未分配的样本,仅将其分配到距离较近且密度较高的样本所在的簇,然而,这一策略可能导致许多样本被误分到其他类簇,从而影响聚类结果的准确性。

为提高DPC算法对流形数据的适用性,本文

提出一种面向流形数据的自然近邻图优化和微簇合并密度峰值聚类 (Density peaks clustering based on natural neighbor graph optimization and micro-cluster merging for manifold data, DPC-NGMM) 算法。本文的主要贡献如下: ①提出了一种利用自然近邻图中样本间测地距离的新型局部密度计算方法, 能够准确反映样本间的全局关系与分布特性; ②分析自然近邻图中邻近样本间的关系, 识别数据的代表, 并以此确定局部核心, 以局部核心指导微簇划分; ③设计了一种基于测地距离的微簇间相似性度量准则, 用于动态合并微簇, 避免了将远离密度峰值的样本错误地分配到其他簇中, 提升了流形数据的聚类效果和鲁棒性。

1 DPC 算法

DPC 算法是一种基于密度的聚类方法, 旨在发现数据集中的密度峰值, 并根据密度峰值对数据进行聚类。其基于以下假设: ①类簇的局部密度中心大边缘小; ②密度峰值间的相对距离远。由此, DPC 算法使用式(1)(2)分别定义了针对大规模数据集的截断核和针对小规模数据集的高斯核的局部密度 ρ_i 的计算方法, 使用式(3)定义了样本 x_i 到更高局部密度样本的最小距离, 即相对距离 δ_i 。

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

$$\rho_i = \sum_{i \neq j} \exp\left(-\frac{d_{ij}^2}{d_c^2}\right) \quad (2)$$

$$\delta_i = \begin{cases} \min_{j: \rho_j > \rho_i} (d_{ij}), & \rho_i \neq \rho_{\max} \\ \max_{i \neq j} (d_{ij}), & \rho_i = \rho_{\max} \end{cases} \quad (3)$$

式中: d_{ij} 为样本 x_i 与 x_j 之间的欧氏距离; d_c 为截断距离; $\rho_{\max}$ 为所有样本局部密度的最大值。

DPC 算法通过绘制决策图选取密度峰值, 决策图以 ρ_i 为横坐标, δ_i 为纵坐标。为提高密度峰值的识别能力, DPC 算法引入了一个决策参数 γ_i , 即:

$$\gamma_i = \rho_i \cdot \delta_i \quad (4)$$

DPC 算法中, 密度峰值被定义为具有较大 ρ_i 和 δ_i 的样本, 即将具有较高 γ_i 值的样本识别为密度峰值。一旦检测到密度峰值, 剩余样本将被分配到具有更高密度的最近类簇中。

2 DPC-NGMM 算法

2.1 自然近邻图的局部密度

引用 DPC 算法的两种局部密度定义方法均与截断距离紧密相关, 局部密度的计算依赖于截断距离内的样本个数, 而截断距离外的样本对密度几乎没有影响。DPC 算法通过欧氏距离计算局部密度和相对距离, 能较好地处理团状数据并准确识别密度峰值, 但在处理流形数据时往往难以准确定位密度峰值。为此, 本文提出了一种自然近邻图的局部密度定义方法。该方法首先构建自然近邻图, 并通过图中顶点间的测地距离重新定义样本的局部密度。

定义 1 自然近邻 $NNN(x_i)$ 。如果样本 x_i 是样本 x_j 的 K 近邻且样本 x_j 是样本 x_i 的 K 近邻, 则称 x_i 和 x_j 互为自然近邻, 表示为:

$$NNN(x_i) = \{x_j | x_j \in KNN(x_i) \wedge x_i \in KNN(x_j)\} \quad (5)$$

定义 2 自然近邻图 G_n , 表示为:

$$G_n = (V, E, \omega) \quad (6)$$

式中: 顶点集合 $V = \{x_1, x_2, \dots, x_n\}$; 边集合 $E \subseteq \{e(x_i, x_j) | x_i, x_j \in V \wedge x_j \in NNN(x_i)\}$;

$e(x_i, x_j)$ 表示顶点 x_i 和 x_j 之间的边, 边权重 $\omega(x_i, x_j) = d_{ij}$ 。

相对直接使用 K 近邻构建近邻图的方法, 利用自然近邻构建近邻图能更有效消除近邻图非对称性带来的干扰, 只保留具备强关联顶点之间的边, 使得近邻图更具稳定性和对称性。

定义 3 自然近邻图的测地距离 $d_G(x_i, x_j)$ 。在自然近邻图 G_n 中顶点 x_i 到顶点 x_j 的加权最短路径长度称为测地距离, 表示为:

$$d_G(x_i, x_j) = \min_{p \in \text{Path}(x_i, x_j)} \sum_{e(u, v) \in p} \omega(u, v) \quad (7)$$

式中: $\text{Path}(x_i, x_j)$ 为顶点 x_i 到顶点 x_j 之间所有可能路径的集合。

定义 4 自然近邻图的局部密度 $\rho(x_i)$, 表示为:

$$\rho(x_i) = \sum_{x_j \in S(x_i, 1)} \exp\left(\frac{-d_G(x_i, x_j)^2}{\deg(x_i)} \times \alpha\right) + \sum_{x_j \in S(x_i, 2)} \frac{\exp(-d_G(x_i, x_j)^2)}{\exp(-\deg(x_i)^2) + \deg(x_i)} \quad (8)$$

$$\alpha = \exp\left(\frac{-d_G(x_i, x_j)}{\text{mean}(x_i)}\right) + 1 \quad (9)$$

式中: α 为权重系数; $S(x_i, n)$ 表示到顶点 x_i 需要经过的最少数目为 n 的节点集合; $\deg(x_i)$ 表示顶点 x_i 的度, 即与这个顶点相连接的边的数目; $\text{mean}(x_i)$ 表示 $S(x_i, 1)$ 集合中所有顶点到顶点 x_i 的平均距离。

式(8)的第一部分用于计算顶点 x_i 相邻顶点 $S(x_i, 1)$ 的贡献, 引入自适应的权重系数 α 动态调整不同距离范围内邻居节点的影响; 第二部分用于计算顶点 x_i 间接相邻顶点 $S(x_i, 2)$ 的贡献。该方法结合近邻图中顶点的局部拓扑特性和节点间的加权最短路径, 使得算法在处理流形数据时更加稳健。

为验证自然近邻图的局部密度优势, 使用 Jain 数据集进行实验。图 1(彩图参见电子版, 下同)展示了不同局部密度定义方式找到的密度峰值, 其中图 1(a)为截断核的局部密度找到的密度峰值, 图 1(b)为高斯核的局部密度找到的密度峰值, 图 1(c)为自然近邻图的局部密度找到的密度峰值, 其中红色六角星表示密度峰值。

如图 1 所示, DPC 算法定义的两局部密度计算方法得到的密度中心均落在同一类簇上; 本文提出的自然近邻图的局部密度通过自然近邻图中顶点的测地距离定义局部密度, 可以准确识别密度峰值。

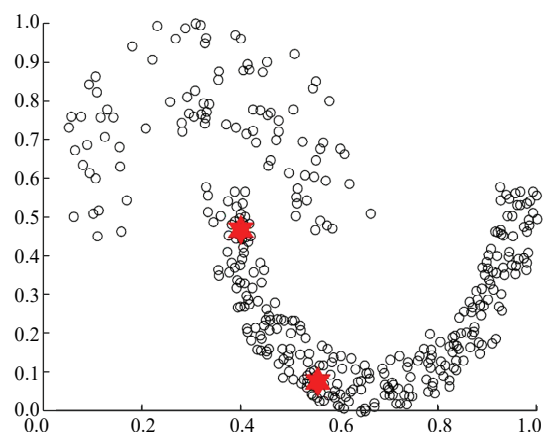
2.2 微簇合并策略

在分配非密度峰值时, DPC 算法基于欧氏距离的度量方法难以准确反映数据在流形上的真实邻近关系, 使 DPC 算法聚类流形数据时难以将样本正确分配至类簇; 并且, 其分配过程依赖于链式分配, 导致一旦出现样本分配错误, 会出现错误连带效应, 造成大量样本分配错误。本文利用测地距离代替欧氏距离, 同时引入代表和局部核心的概念, 通过微簇合并策略进行非密度峰值样本的分配。

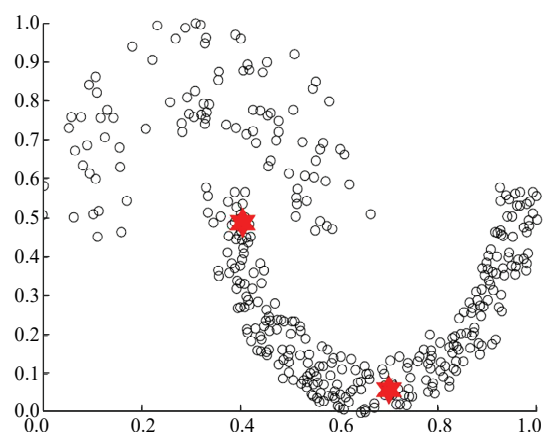
定义 5 代表。如果样本 $x_j \in S(x_i, 1) \cup \{x_i\}$, $x_k \in S(x_i, 1) \cup \{x_i\}$ 且 $\rho(x_j) > \rho(x_k)$, 则称 x_j 为 x_i 的代表, 表示为: $\text{Rep}(x_i) = x_j$ 。

定义 6 代表的传递性规则。如果 $\text{Rep}(x_i) = x_j$, $\text{Rep}(x_j) = x_k$, 且 $x_k \in S(x_i, 1) \cup \{x_i\}$, 则 $\text{Rep}(x_i) = x_k$ 。

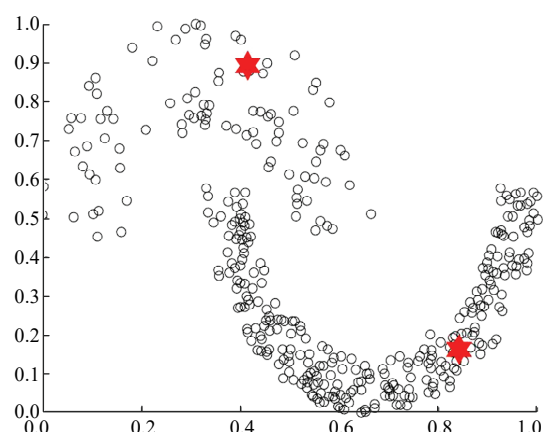
定义 7 代表的竞争性规则。如果 $\text{Rep}(x_i) = x_j$, $\text{Rep}(x_i) = x_k$, 则 $\text{Rep}(x_i) = \arg \min_{x_i \in \{x_j, x_k\}} \{d_G(x_i, x_i)\}$ 。



(a) 截断核的局部密度



(b) 高斯核的局部密度



(c) 自然近邻图的局部密度

图 1 不同局部密度定义方式在 Jain 数据集找到的密度峰值

Fig. 1 Density peaks obtained on Jain dataset by different local density definitions

定义 8 局部核心。根据代表的传递性及代表的竞争性规则, 如果样本的代表是其自身, 即 $\text{Rep}(x_i) = x_i$, 则称样本 x_i 为局部核心。

定义 9 微簇 c_i 。一个局部核心及该核心代表的所有样本组成一个微簇。

测地距离可以很好地反映流形数据集中样本的关系,所以本文采用测地距离定义样本的相似度,再通过微簇间所有样本的平均相似度度量微簇的相似性。

定义 10 测地距离的样本相似度 $Sim(x_i, x_j)$, 表示为:

$$Sim(x_i, x_j) = \exp[-d_G^2(x_i, x_j)] \quad (10)$$

定义 11 微簇的相似性 $CSim(c_i, c_j)$ 。表示为:

$$CSim(c_i, c_j) = \frac{\sum_{x_k \in c_i, x_l \in c_j} sim(x_k, x_l)}{num(c_i) + num(c_j)} \quad (11)$$

式中: $num(c_i)$ 为微簇 c_i 中的样本数量。

微簇合并策略流程如图 2 所示,过程如下:首先,确定每个样本的初始代表,并通过传递性规则不断更新代表关系,使所有样本的代表逐步收敛至一个核心节点;同时,依据竞争性规则优化代表关系,当某个样本的代表存在多路径竞争时,优先选择测地距离更近的样本作为新的代表;待所有样本的代表关系趋于稳定后,标记代表为自身的局部核心节点,进而由这些核心节点吸引所有以其为代表的样本,形成完整的微簇;最后,计算测地距离的样本相似度,再评估微簇间的相似性,并按照相似性大小逐步将微簇合并至包含密度峰值的微簇,直至所有微簇均包含密度峰值,最终构成完整的类簇。

2.3 算法步骤

DPC-NGMM 算法流程如图 3 所示,具体的聚类过程如下:

输入:数据集 $X = \{x_1, x_2, \dots, x_n\}$, 近邻数 K 。

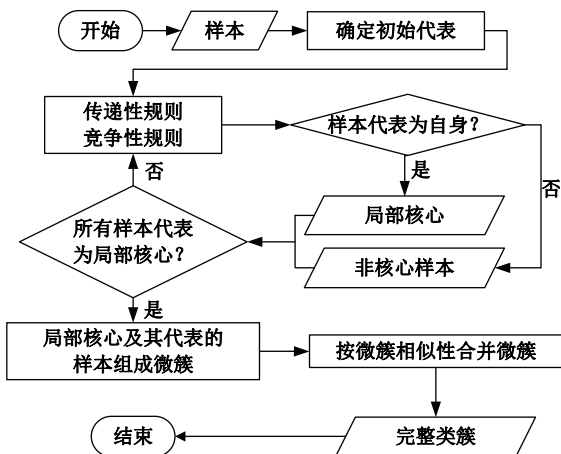


图 2 微簇合并流程

Fig. 2 Flowchart of micro-cluster merging

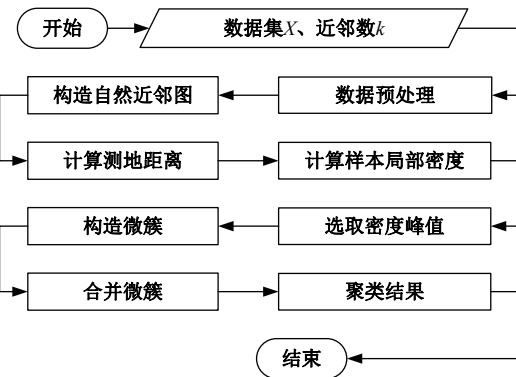


图 3 DPC-NGMM 算法流程

Fig. 3 Flowchart of DPC-NGMM algorithm

输出:聚类结果 $C = \{c_1, c_2, \dots, c_m\}$ 。

步骤 1 数据预处理,对数据进行归一化处理;

步骤 2 利用式(5)(6)构建自然近邻图;

步骤 3 利用式(7)计算测地距离 d_G ;

步骤 4 利用式(8)计算样本的局部密度 ρ_i ;

步骤 5 用测地距离代替式(3)中欧氏距离计算代表的相对距离 δ_i ,绘制决策图,选取密度峰值;

步骤 6 利用定义 5 获取所有样本的代表,并通过代表的传递性规则和竞争性规则获取所有局部核心及其代表的所有样本,局部核心及各自代表的所有样本凝聚成微簇;

步骤 7 根据式(10)(11)计算微簇的相似度 $CSim(c_i, c_j)$;

步骤 8 按照微簇间相似性大小将微簇合并到包含密度峰值的微簇中,直到所有微簇均包含密度峰值;

步骤 9 输出最终的聚类结果。

3 实验结果及分析

3.1 实验设置

本文采用 10 个经典的流形数据集和 8 个 UCI 真实数据集进行实验。流形数据集包含 Line-Blobs、Pathbase、Jain、Compound、Db、Cmc、Cth3、Ls、Circle3、Complex9; 真实数据集包含 Iris、Seeds、Wine、Ecoli、Ionosphere、Libras、WDBC、Waveform。

采用 SDPC^[24]、DPC-CE^[25]、IDPC-FA^[26]、DPCSA^[27] 和 DPC^[9] 作为对比算法。除 DPC-CE、IDPC-FA、DPCSA 算法无参数外,本实验对其余各算法进行参数调优。DPC-NGMM 算法 K

值在 2~60 中选取;SDPC 算法参数 α 值按原文推荐取 0.1;DPC 算法截断距离设定在 0.01%~5% 之间。

本文选用调整兰德系数(Adjusted rand index, ARI)^[28]、调整互信息(Adjusted mutual information, AMI)^[28]和 Fowlkes-Mallows 指数(Fowlkes-Mallows index, FMI)^[29]三个独立于标签绝对值的评价指标作为聚类性能的评判标准,这 3 个评价指标的上限均为 1,该值越接近 1 表示聚类效果越好。

3.2 流形数据集实验结果分析

为验证 DPC-NGMM 算法的聚类性能,采用 10 个流形数据集进行实验,实验得到的 ARI、

AMI、FMI 指标展示在表 1 中。各聚类算法的最优参数用 Arg- 表示,若没有对应参数,则以“-”表示。

根据表 1 数据,DPC-NGMM 算法对 10 个流形数据集在 ARI、AMI 和 FMI 指标上均能取到较好的结果,除 Compound 数据集的 AMI 指标以外,其取值均在 0.9 以上,在 LineBlobs、Jain、Db、Cmc、Ls、Circle3、complex9 数据集上均能得到最佳值 1。

表 2 采用 Friedman 检验分析 6 种算法在流形数据集上的综合性能,其中,秩均值越大表示算法的性能相较于其它算法越优。如表 2 所示,DPC-NGMM 算法的秩均值高于对比算法,且均大于

表 1 6 种聚类算法在 10 个流形数据集上的聚类性能

Table 1 Clustering results of six algorithms on ten manifold datasets

算法	LineBlobs				Pathbase			
	ARI	AMI	FMI	Arg-	ARI	AMI	FMI	Arg-
DPC-NGMM	1	1	1	2	0.990 5	0.983 0	0.993 7	2
SDPC	0.541 9	0.623 0	0.699 9	0.1	0.484 3	0.522 1	0.666 6	0.1
DPC-CE	1	1	1	—	0.473 8	0.486 4	0.693 8	—
IDPC-FA	1	1	1	—	0.859 3	0.844 2	0.906 7	—
DPCSA	1	1	1	—	0.613 3	0.707 3	0.751 1	—
DPC	0.823 7	0.837 5	0.884 2	4.2	0.508 2	0.557 3	0.684 8	0.4
算法	Jain				Compound			
	ARI	AMI	FMI	Arg-	ARI	AMI	FMI	Arg-
DPC-NGMM	1	1	1	4	0.905 6	0.882 2	0.930 2	10
SDPC	1	1	1	0.1	0.702 2	0.707 4	0.782 1	0.1
DPC-CE	1	1	1	—	0.808 2	0.614 1	0.706 0	—
IDPC-FA	1	1	1	—	0.832 7	0.792 2	0.881 5	—
DPCSA	0.044 2	0.216 7	0.592 4	—	0.828 4	0.839 2	0.870 7	—
DPC	0.714 6	0.618 3	0.881 9	0.8	0.636 2	0.825 0	0.723 6	4.6
算法	Db				Cmc			
	ARI	AMI	FMI	Arg-	ARI	AMI	FMI	Arg-
DPC-NGMM	1	1	1	7	1	1	1	7
SDPC	0.396 1	0.544 1	0.641 4	0.1	0.638 5	0.563 2	0.785 8	0.1
DPC-CE	0.675 8	0.558 8	0.739 5	—	0.669 4	0.736 2	0.835 2	—
IDPC-FA	0.503 3	0.652 6	0.699 9	—	0.842 1	0.809 3	0.902 7	—
DPCSA	0.109 6	0.413 6	0.468 9	—	0.576 1	0.665 6	0.745 4	—
DPC	0.279 4	0.518 5	0.585 3	4	0.266 1	0.385 7	0.537 7	5
算法	Cth3				Ls			
	ARI	AMI	FMI	Arg-	ARI	AMI	FMI	Arg-
DPC-NGMM	0.997 1	0.994 8	0.997 9	10	1	1	1	11
SDPC	0.471 5	0.652 8	0.620 4	0.1	0.589 2	0.669 7	0.702 3	0.1
DPC-CE	0.825 5	0.715 8	0.793 5	—	0.743 5	0.639 2	0.741 5	—
IDPC-FA	0.832 7	0.875 8	0.878 6	—	0.627 4	0.707 6	0.732 5	—
DPCSA	0.653 8	0.789 1	0.754 7	—	0.599 9	0.725 2	0.712 9	—
DPC	0.513 5	0.686 6	0.647 3	0.1	0.689 4	0.766 5	0.777 9	0.9
算法	Circle3				Complex9			
	ARI	AMI	FMI	Arg-	ARI	AMI	FMI	Arg-
DPC-NGMM	1	1	1	16	1	1	1	16
SDPC	0.146 2	0.221 7	0.483 4	0.1	0.556 1	0.679 7	0.638 9	0.1
DPC-CE	0.529 0	0.255 5	0.627 9	—	0.461 5	0.695 8	0.552 7	—
IDPC-FA	0.438 5	0.462 9	0.765 2	—	0.957 7	0.951 3	0.965 8	—
DPCSA	0.083 3	0.295	0.524 2	—	0.422 1	0.682 1	0.518 8	—
DPC	0.301 5	0.359 6	0.604 8	0.3	0.609 4	0.740 2	0.683 9	2

表 2 6 种算法在流形数据集上指标的秩均值
Table 2 Mean ranks on the manifold dataset

算法	ARI	AMI	FMI
DPC-NGMM	5.70	5.70	5.70
SDPC	2.25	1.95	2.15
DPC-CE	3.80	2.80	3.60
IDPC-FA	4.50	4.50	4.60
DPCSA	2.35	3.15	2.45
DPC	2.40	2.90	2.50

5,说明 DPC-NGMM 算法在 ARI、AMI 和 FMI 3 个指标上优于其他算法。

受论文篇幅所限,图 4 至图 6 仅展示 6 种聚类算法在 Db、Cth3 和 Complex9 的 3 个流形数据集上的聚类结果。图中不同颜色的点被分配到不同的聚类,白色六角星表示类簇的中心。

图 4 为 6 种聚类算法对 Db 数据集的聚类结果。Db 数据集一共 630 个样本,4 个弧形类簇组成,由于其中的“S”形类簇几乎贯穿整个样本空间,并且密度高,与另外的两个类簇相互环绕,导致 Db 数据集的聚类难度很高。对于该数据集,仅 DPC-NGMM 算法能准确聚类;IDPC-FA 算法虽然能在每个类簇都找到密度峰值,但由于其基于欧氏距离的度量方法难以准确反映数据在流形上的真实邻近关系,导致大量样本错误分配;其余算法均未能准确找到密度峰值;SDPC 算法和 DPC-SA 算法将邻近中间“S”形类簇的两个密度峰值



图 4 六种聚类算法对 Db 数据集的聚类结果

Fig. 4 Clustering results of six algorithms on Db dataset

均误识别到中间类簇中,使得最终聚类结果并不理想;DPC-CE 算法在寻找密度峰值时发生错误,将右上方区域样本归为一类,导致聚类质量降低;DPC 算法同样无法获取准确的密度峰值,错将右侧中间类簇的密度峰值识别到“S”形类簇的右下部分,使聚类结果不佳。

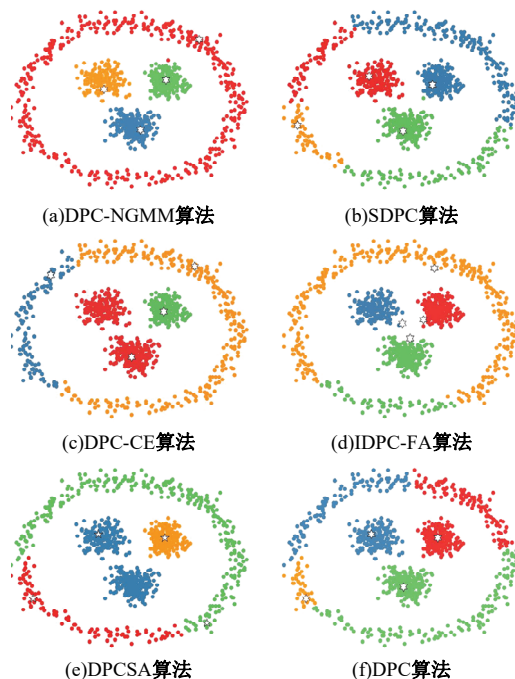


图 5 六种聚类算法对 Cth3 数据集的聚类结果

Fig. 5 Clustering results of six algorithms on Cth3 dataset

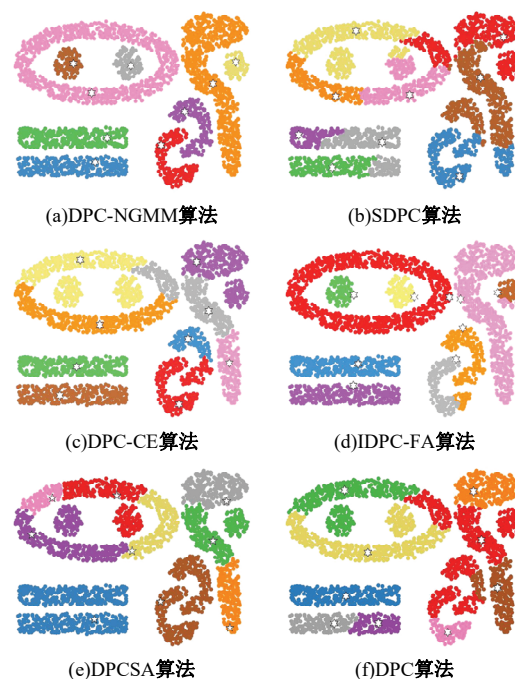


图 6 六种聚类算法对 Complex9 数据集的聚类结果

Fig. 6 Clustering results of six algorithms on Complex9 dataset

图 5 为 6 种聚类算法对 Cth3 数据集的聚类结果。Cth3 数据集一共 1 016 个样本,4 个类簇组成,其中包括 3 个团状类簇和 1 个环绕四周的圆形类簇。对该数据集,SDPC 和 DPCSA 算法没能找到准确的密度峰值导致聚类结果不佳;DPC-CE、IDPC-FA 和 DPC 算法由于其分配策略缺陷,没能处理好圆形类簇,错将其中部分样本分配到中间的团状类簇中。

图 6 为 6 种聚类算法对 Complex9 数据集的聚类结果。Complex9 数据集一共 3 031 个样本,9 个类簇组成,其中包含多个流形类簇,且类簇间的疏密程度并不相同,同时部分类簇较为接近。该数据集仅有 DPC-NGMM 算法可以得到准确的聚类结果;IDPC-FA 算法错将右下两个弧形类簇识别为一个类簇,导致聚类精度稍差;其余算法均无法很好地处理右侧贯穿上下的类簇,均在其中找到 2~3 个密度峰值,除此之外,在样本分配时无法处理较接近类簇邻近部位的样本分配,导致一

些样本被错误分配。

3.3 真实数据集实验结果分析

为进一步验证 DPC-NGMM 算法在聚类任务中的表现,本文采用 8 个来自 UCI 机器学习库的数据集进行实验,实验得到的 ARI、AMI、FMI 指标呈现在表 3 中,可以看出,DPC-NGMM 算法的整体聚类精度最高。在鸢尾花卉数据集 Iris 中,DPC-NGMM 算法的 ARI、AMI、FMI 指标分别与最好的结果相差 2.29%、3.47%、1.45%;在红酒数据集 Wine 中,DPC-NGMM 算法的聚类效果仅次于 SDPC 算法,但相比其他算法 DPC-NGMM 算法均有明显优势;在波形数据集 Waveform 中,DPC-NGMM 算法在 AMI 指标取得最优,但 ARI、FMI 指标稍逊于 SDPC。在剩余的种子数据集 Seeds、大肠杆菌数据集 Ecoli、电离层数据集 Ionosphere、手势数据集 Libras、威斯康星乳腺癌诊断数据集 WDBC 等 5 个数据集中,DPC-NGMM 算法均取得最优的聚类效果。

表 3 6 种聚类算法在 8 个真实数据集上的聚类性能
Table 3 Clustering results of six algorithms on eight real datasets

算法	Iris				Seeds			
	ARI	AMI	FMI	Arg-	ARI	AMI	FMI	Arg-
DPC-NGMM	0.883 1	0.852 5	0.921 6	3	0.775 5	0.732 4	0.849 8	35
SDPC	0.885 7	0.862 3	0.923 3	0.1	0.765 7	0.721 5	0.843 2	0.1
DPC-CE	0.663 4	0.727 7	0.782 4	—	0.744 8	0.714 4	0.829 7	—
IDPC-FA	0.885 7	0.862 3	0.923 3	—	0.767	0.729 9	0.844 4	—
DPCSA	0.903 8	0.883 1	0.935 5	—	0.687 3	0.660 9	0.791 8	—
DPC	0.903 8	0.883 1	0.935 5	3.2	0.767	0.729 8	0.844 4	0.7
算法	Wine				Ecoli			
	ARI	AMI	FMI	Arg-	ARI	AMI	FMI	Arg-
DPC-NGMM	0.864 9	0.819 7	0.910 3	26	0.757 8	0.675 0	0.834 3	4
SDPC	0.870 8	0.855 0	0.914 0	0.1	0.679 9	0.574 22	0.762 26	0.1
DPC-CE	0.536 2	0.584 1	0.694 5	—	0.114 5	0.070 4	0.580 2	—
IDPC-FA	0.771 3	0.767 5	0.847 8	—	0.756 1	0.663 8	0.828 4	—
DPCSA	0.741 4	0.748 0	0.828 3	—	0.459 3	0.440 6	0.646 7	—
DPC	0.770 3	0.769 5	0.847 4	2.4	0.410 9	0.522 3	0.554 5	2.1
算法	Ionosphere				Libras			
	ARI	AMI	FMI	Arg-	ARI	AMI	FMI	Arg-
DPC-NGMM	0.419 5	0.305 4	0.760 3	11	0.386 5	0.583 5	0.435 8	11
SDPC	0.217 6	0.150 4	0.627 7	0.1	0.255 7	0.451 0	0.308 4	0.1
DPC-CE	0.114 5	0.070 4	0.580 2	—	0.353 1	0.557 0	0.419 2	—
IDPC-FA	0.218 3	0.135 5	0.643 2	—	0.381 6	0.573 3	0.424 7	—
DPCSA	0.213 5	0.133 5	0.639 0	—	0.309 5	0.538 8	0.379 1	—
DPC	0.230 6	0.148 4	0.644 9	1.8	0.362 6	0.583 2	0.419 0	0.5
算法	WDBC				Waveform			
	ARI	AMI	FMI	Arg-	ARI	AMI	FMI	Arg-
DPC-NGMM	0.811 4	0.704 3	0.913 0	6	0.319 6	0.367 3	0.574 4	55
SDPC	0.773 0	0.676 4	0.897 7	0.1	0.356 1	0.364 8	0.593 8	0.1
DPC-CE	0.435 5	0.374 2	0.774 3	—	0.283 6	0.327 4	0.545 6	—
IDPC-FA	0.773 0	0.676 4	0.897 7	—	0.311 4	0.295 6	0.533 1	—
DPCSA	0.377 1	0.336 1	0.759 5	—	0.223 6	0.251 0	0.532 7	—
DPC	0.754 8	0.637 5	0.887 6	0.7	0.269 8	0.326 1	0.529 2	0.1

表4采用Friedman检验分析了6种算法在真实数据集上的性能差异。如表4所示,DPC-NGMM算法的秩均值高于其他算法,且都不小于5,说明DPC-NGMM算法在ARI、AMI和FMI三个指标上优于其他算法。

表4 6种算法在真实数据集上指标的秩均值

Table 4 Mean ranks on the real dataset			
算法	ARI	AMI	FMI
DPC-NGMM	5.25	5.38	5.25
SDPC	3.94	4.06	3.81
DPC-CE	2.06	2.25	2.44
IDPC-FA	3.81	3.19	3.69
DPCSA	2.19	2.06	2.44
DPC	3.75	4.06	3.38

4 结束语

面对流形数据,DPC算法的局部密度依赖于欧氏距离,仅考虑截断范围内的样本,难以准确衡量复杂流形数据的全局关系;分配策略未考虑远离密度峰值的样本特性,导致大量样本被错误分配到邻近类簇,影响聚类效果。针对这些问题,本文提出一种面向流形数据的自然近邻图优化和微簇合并密度峰值聚类算法。DPC-NGMM算法利用自然近邻图中顶点间的测地距离定义的局部密度,可以有效揭示流形数据的全局关系,准确找到类簇的密度峰值;利用自然近邻图中顶点的关系找出局部核心及其代表的样本,以此指导微簇划分可以更贴合流形形状,减少划分错误;利用样本间测地距离定义一种新的微簇间相似性度量准则,用以指导微簇合并,能够更好地区分流行数据中相近但不相连的微簇,确保合并结果更符合数据的本质结构。通过实验证明,本文算法通过自然近邻图重新定义的局部密度及微簇合并策略提高了密度峰值聚类算法对流形数据集的识别能力,在流形数据集及真实数据集均有较好的聚类结果。但受限于测地距离计算的高复杂度,本文算法仍存在时间复杂度较高的问题,如何提高算法的效率将是下一步的努力方向。

参考文献:

[1] Nan F, Tang Y, Yang P, et al. A novel sub-Kmeans based on co-training approach by transforming single-view into multi-view[J]. Future Generation Computer Systems, 2021, 125: 831-843.

[2] Hou J, Lin H, Yuan H, et al. Flexible density peak

clustering for real-world data[J]. Pattern Recognition, 2024, 156: 110772.

- [3] Gagniere S, Han Y, Chen Y, et al. A robust grid-based meshing algorithm for embedding self-intersecting surfaces[J]. Computer Graphics Forum, 2024, 43(1): e14986.
- [4] Wang F, Li L, Liu Z. Stratification-based semi-supervised clustering algorithm for arbitrary shaped datasets[J]. Information Sciences, 2023, 639: 119004.
- [5] 谢坤,董宏辉,卢玲玉,等. 基于CFSFDP-BP的车辆出行群体分类及识别方法[J]. 吉林大学学报:工学版, 2026, 56(3): 725-733.
- Xie Kun, Dong Hong-hui, Lu Ling-yu, et al. Classification and recognition method of vehicle travel groups based on CFSFDP-BP[J]. Journal of Jilin University (Engineering and Technology Edition), 2026, 56(3): 725-733.
- [6] Kim B, Yuvaraj N, Tse K T, et al. Pressure pattern recognition in buildings using an unsupervised machine-learning algorithm[J]. Journal of Wind Engineering and Industrial Aerodynamics, 2021, 214: 104629.
- [7] Corneo E, Garbelotto R, Prestes G, et al. Coagulation biomarkers and coronavirus disease 2019 phenotyping: a prospective cohort study[J]. Thrombosis Journal, 2023, 21(1): 80.
- [8] Chang J Q, Yu F S. Design matrix factorization and knowledge-guided clustering for trust-aware cross-domain recommendation systems[C]//7th International Conference on Software Engineering and Computer Science, Taicang, China, 2025: 1-6.
- [9] Rodriguez A, Laio A. Clustering by fast search and find of density peaks[J]. Science, 2014, 344: 1492-1496.
- [10] Xiong J, Zang W, Zhao Y, et al. Density peaks clustering algorithm with connected local density and punished relative distance[J]. The Journal of Supercomputing, 2024, 80(5): 6140-6168.
- [11] Wang Y. McDPC: multi-center density peak clustering[J]. Neural Computing and Applications, 2020, 32(17): 13465-13478.
- [12] Du M, Ding S, Xue Y. A robust density peaks clustering algorithm using fuzzy neighborhood[J]. International Journal of Machine Learning and Cybernetics, 2018, 9(7): 1131-1140.
- [13] Xie J, Jiang W, Ding L. Clustering by searching density peaks via local standard deviation[C]//International conference on Intelligent Data Engineering and Au-

- tomated Learning, Guilin, China, 2017: 295-305.
- [14] 吕莉, 朱梅子, 康平, 等. 面向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.
- Lv Li, Zhu Mei-zi, Kang Ping, et al. Multiplex neighbor density peaks clustering for uneven density data sets[J]. Control Theory & Applications, 2024, 41(10): 1821-1830.
- [15] Chen J, Yu P S. A domain adaptive density clustering algorithm for data with varying density distribution[J]. IEEE Transactions on Knowledge and Data Engineering, 2021, 33(6): 2310-2321.
- [16] Zhang S, Li K. A novel density peaks clustering algorithm with isolation kernel and k -induction[J]. Applied Sciences, 2022, 13(1): 322.
- [17] Li Y, Sun L, Tang Y. DPC-DNG: Graph-based label propagation of k -nearest higher-density neighbors for density peaks clustering[J]. Applied Soft Computing, 2024, 161: 111773.
- [18] 赵嘉, 陈磊, 吴润秀, 等. K 近邻和加权相似性的密度峰值聚类算法[J]. 控制理论与应用, 2022, 39(12): 2349-2357.
- Zhao Jia, Chen Lei, Wu Run-xiu, et al. Density peaks clustering algorithm with K -nearest neighbors and weighted similarity[J]. Control Theory & Applications, 2022, 39(12): 2349-2357.
- [19] Zang W, Liu X, Ma L, et al. DPC-MFP: an adaptive density peaks clustering algorithm with multiple feature points[J]. Neurocomputing, 2025, 618: 129060.
- [20] Du M. Density peaks clustering using geodesic distances[J]. International Journal of Machine Learning and Cybernetics, 2018, 9(8): 1335-1349.
- [21] Zhang D, Wang Y, Zhou L, et al. Multimodal classification of Alzheimer's disease and mild cognitive impairment[J]. NeuroImage, 2011, 55(3): 856-867.
- [22] Sahraeian R, Van Compernelle D. Crosslingual and multilingual speech recognition based on the speech manifold[J]. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 2017, 25(12): 2301-2312.
- [23] Ding S, Li C, Xu X, et al. A sampling-based density peaks clustering algorithm for large-scale data[J]. Pattern Recognition, 2023, 136: 109238.
- [24] Guo W, Wang W, Zhao S, et al. Density peak clustering with connectivity estimation[J]. Knowledge-Based Systems, 2022, 243: 108501.
- [25] Zhao J, Tang J, Shi A, et al. Improved density peaks clustering based on firefly algorithm[J]. International Journal of Bio-Inspired Computation, 2020, 15(1): 24-42.
- [26] Sun L, Liu R, Xu J, et al. An adaptive density peaks clustering method with fisher linear discriminant[J]. IEEE Access, 2019, 7: 72936-72955.
- [27] Vinh N X, Epps J, Bailey J. Information theoretic measures for clusterings comparison: variants, properties, normalization and correction for chance[J]. Journal of Machine Learning Research, 2010, 11: 2837-2854.
- [28] Fowlkes E B, Mallows C L. A method for comparing two hierarchical clusterings[J]. Journal of the American Statistical Association, 1983, 78: 553-569.