◆算法研究与应用◆

密度峰值聚类算法研究现状与分析*

葛丽娜[1,2,3],陈园园[1],周永权[1,3**]

(1.广西民族大学人工智能学院,广西南宁 530006;2.广西民族大学,网络通信工程重点实验室,广西南宁 530006;3.广西混杂计算与集成电路设计分析重点实验室,广西南宁 530006)

摘要:密度峰值聚类(Clustering by Fast Search and Find of Density Peaks,DPC)算法是一种新型的基于密度的聚类算法,通过选取自身密度高且距离其他更高密度点较远的样本点作为聚类中心,再根据样本间的局部密度和距离进行聚类。一方面,虽然DPC算法参数唯一、简单、高效,但是其截断距离的取值是按经验策略设定,而截断距离值选取不当会导致局部密度和距离计算错误;另一方面,聚类中心的选取采用人机交互模式,对聚类结果的主观影响较大。针对DPC算法的这些缺陷,目前的改进方向主要有3个:改进截断距离的取值方式、改进局部密度和距离的计算方式以及改进聚类中心的选取方式。通过这3个方向的改进,使得DPC过程自适应。本文对DPC算法的自适应密度峰值聚类算法的研究现状进行比较分析,对进一步的工作进行展望并给出今后的研究方向:将DPC算法与智能算法有机结合实现算法自适应,对于算法处理高维数据集的性能也需要进一步探索。

关键词:密度峰 聚类算法 自适应 截断距离 聚类中心

中图分类号:TP391 **文献标识码**:A **文章编号**:1005-9164(2022)02-0277-10

DOI:10.13656/j.cnki.gxkx.20220526.007

随着现代信息技术的发展,生活中充斥着海量的数据信息,如医疗数据信息、个人消费记录、个人理财记录等,而数据信息的增多,也促使数据挖掘技术不断提高。聚类算法是数据挖掘的关键技术之一。聚类算法是根据数据之间的相似性将数据集样本划分为不同的类簇,每个类簇之间的数据相似性较高,不同的类簇中数据相似性较低。

传统的聚类算法分为基于划分的聚类算法、基于层次的聚类算法、基于密度的聚类算法、基于网格的聚类算法以及基于模型的聚类算法[1]。基于密度的聚类算法,如基于密度的噪声应用空间聚类(Density-Based Spatial Clustering of Applications with Noise,DBSCAN)算法,其对噪声不敏感,能够发现任意形状的簇,但是该算法对参数ε和Minpts

收稿日期:2021-06-11

*国家自然科学基金项目(61862007)和广西自然科学基金项目(2018GXNSFAA281269)资助。

【作者简介】

葛丽娜(1969-),女,博士,教授,主要从事信息安全、人工智能研究,E-mail:66436539@qq.com。

【**通信作者】

周永权(1962-),男,博士,教授,主要从事计算智能及应用、神经网络等研究,E-mail:yongquanzhou@126.com。

【引用本文】

葛丽娜,陈园园,周永权.密度峰值聚类算法研究现状与分析[J].广西科学,2022,29(2):277-286.

GE L N,CHEN Y Y,ZHOU Y Q. Research and Analysis of Adaptive Density Peak Clustering Algorithm [J]. Guangxi Sciences,2022,29(2):277-286.

设置敏感,且对于密度不均匀的数据集,该算法不适用[2]。基于密度的聚类算法是以数据集在空间分布上的稠密度为依据进行聚类,无需预先设定类簇数,适合对未知内容的数据集进行聚类。

本文所研究的密度峰值聚类(Clustering by Fast Search and Find of Density Peaks,DPC)算法是 2014 年意大利学者 Rodriguez 等[3]提出的。DPC 算法由于参数唯一、可以发现任意形状的数据、聚类过程简洁高效等优点,受到各界的广泛关注。目前,DPC 算法已经在医学图像处理[4]、分子动力学[5]、文档处理[6,7]、社区检测[8-10]等许多领域中展现出较好的性能。如在生物医学应用方面,为了确定在 300 K 基准温度下 T-REMD 模拟过程中采样的主要构象,Kührová 等[11]引入了 DPC 算法,与 εRMSD 结合,提出了新的算法;Chen 等[12]引入 DPC 算法来识别疾病症状,再利用 Apriori 算法分别对疾病诊断规则和疾病治疗规则进行关联分析。本文对 DPC 算法原理进行介绍、分析,并对自适应 DPC 算法的国内外研究现状进行比较总结,最后给出今后的研究方向。

1 DPC 算法原理

DPC 算法[3]基于以下假设:每一类簇的聚类中心被与其相邻的密度较低的样本点所包围,这些相邻的样本点距离其他局部密度相对较大的点较远。

设有数据集 $D=\{q_1,q_2,\cdots,q_n\}$,对于每一点 q_i,由公式(1)计算其局部密度 ρ_i,对于小规模数据集,采用公式(2)计算:

$$\rho_i=\sum_j\chi(d_{ij}-d_c),\text{其中},\chi(x)=\begin{cases}1,x<0\\0,\text{otherwise}\end{cases},\tag{1}$$

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

式中,d_c 是截断距离,d_{ij} 是点 q_i 到点 q_j 之间的欧氏距离。

再由公式(3)计算样本点 q_i 的距离 δ_i,δ_i 是样本点 q_i 到其他密度较高样本点之间的最短距离,若 q_i 是密度最高的样本点,则 δ_i 为 q_i 到其他样本的最大距离。

$$\delta_i=\begin{cases}\min\limits_j(d_{ij}),\rho_j>\rho_i\\\max\limits_j(d_{ij}),\text{其他}\end{cases}。\tag{3}$$

计算出 q_i 的局部密度和距离后,选取聚类中心。在 DPC 算法中,选取聚类中心的方法有两种,一种是决策图法,另一种是公式法。决策图法是根据样本点的局部密度和距离生成一个决策图,然后选取最佳的聚类中心点。例如,图 1 中的数据点按密度递减的顺序排列,图 2 是根据图 1 中的样本点计算局部密度和距离后得出的决策图[3]。由此可以得出 DPC 算法决策图选取聚类中心的一般规律:①位于决策图右上方的样本点适合选取为聚类中心,这些点拥有较高的局部密度且距离其他更高密度的点较远;②位于决策图 ρ 坐标轴附近的样本点具有较近的距离,认为是普通样本点,因为其附近存在更适合选取为聚类中心的样本点;③位于决策图 δ 坐标轴附近且距离 ρ 坐标轴相对较远的样本点识别为离群点,这些点拥有较低的密度且距离更高密度点较远。

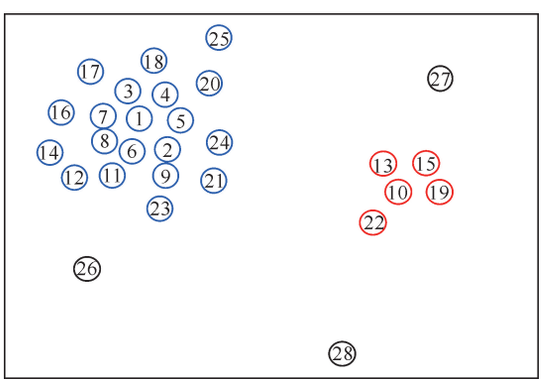

图 1 数据分布图

Fig. 1 Distribution map of data

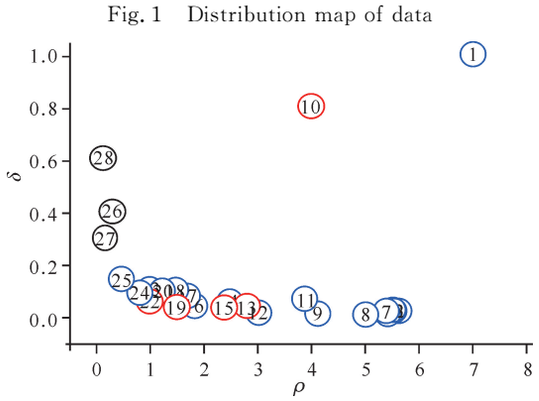

图 2 决策图

Fig. 2 Graph of decision

DPC 算法中选取聚类中心的另一种方法是公式法,根据公式(4)计算 γ 的值,并将其值进行降序排序,选取前 k 个样本作为聚类中心(k 为预先指定的簇数)。将局部密度值与距离相乘是为了寻找局部密度较高且距离较远的样本点。但是,该公式未考虑样本点邻域结构的影响。

$$\gamma_i=\rho_i\times\delta_i。\tag{4}$$

选出聚类中心后,将剩余样本点分配到距离其最近且

拥有较高密度的样本点所在的类簇。

DPC 算法的具体流程如算法 1 所示:

算法 1　密度峰值聚类算法流程

输入:数据集 $D=\{q_1,q_2,\cdots,q_n\}$,簇数 k

输出:聚类划分结果

1. 根据数据集样本点总数确定截断距离 d_c

2. 根据公式(1)或(2)计算样本局部密度 ρ

3. 根据公式(3)计算样本距离 δ

4. 由计算出的局部密度和距离生成决策图,根据决策图或公式(4)选取聚类中心

5. 将剩余样本点分配到距离其最近的局部密度较高点所在的类簇中

6. 返回聚类划分结果图

2　自适应 DPC 算法的优化

在 DPC 算法中,截断距离 d_c 并不是算法自动设定的,而是按照文献[3]中提出的经验策略设定 d_c 的值使得邻域样本点数为总样本点数的 1%－2%。而在实际应用中,按照文献[3]中所提的方法设定截断距离的值,并不是所有的聚类问题都适用。图 3 所示是 DPC 算法在不同的 d_c 取值下对同一数据集进行聚类的结果。由图 3 可以看出,虽然对类簇数没有影响,但是普通样本点和异常点的划分随着 d_c 的取值变化而发生变化。

在聚类中心选取阶段,虽然根据决策图选取聚类中心能够得到较好的聚类结果,但是若数据集较为复杂,人工难以选取合适的聚类中心,而聚类中心一旦选择错误,会导致非聚类中心点分配错误。图 4 为 DPC 算法对数据集 Aggregation 进行聚类时生成的决策图。由图 4 可以看出,符合聚类中心要求的点不容易确定,手动选取易造成聚类中心个数选取错误。由于 DPC 算法聚类无需迭代,若聚类中心选取错误,会引起剩余样本点分配出现错误,最终导致聚类效果不佳。

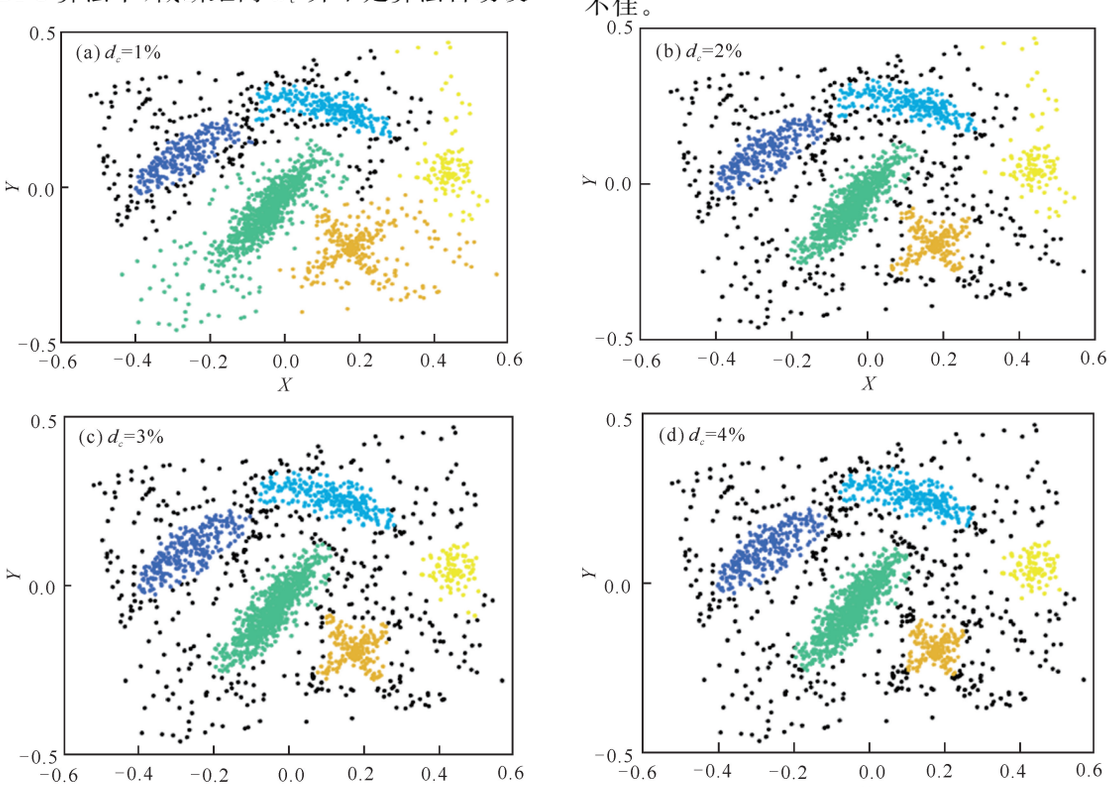

图 3　不同 d_c 取值下的聚类结果

Fig. 3　Clustering results when takes different values

目前,针对 DPC 算法过程不能实现自适应的问题,主要的改进方法有 3 种:①针对参数 d_c 的改进,使得 d_c 值能够自适应选取;②对计算局部密度 ρ 和距离 δ 的公式进行改进,避免参数 d_c 的使用;③在选取聚类中心时,采用不同的方式使得聚类中心自适应选取,不需要人为参与。

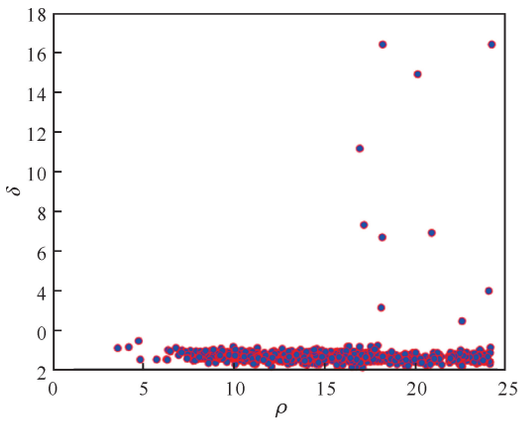

图 4　对 Aggregation 数据集聚类的决策图

Fig. 4　Decision graph of aggregation data set

2.1　参数 d_c 的改进

第 1 种改进方式主要是针对参数 d_c 的选取。由于原来的 d_c 值是人为设定的,淦文燕等[13]提出了 Improved Clustering Algorithm that Searches and Finds Density Peaks (ICADEP)算法。该算法引入密度估计熵,提出新的参数优化方法,使得参数 d_c 能够自适应选取最优值且聚类结果与核函数的类型无关,达到了更精确的聚类效果。但是该方法仍然需要人为参与选取聚类中心,为了解决这一问题,有学者引入 K 近邻思想[14-16],即在聚类过程中计算样本点的近邻密度,提出新的计算 d_c 的公式,实现 d_c 的自动计算取值。Liu 等[15]提出一种新的基于 K 近邻的计算 d_c 的算法。该算法不仅使得 d_c 的值自适应选择且聚类中心的选取准确、不遗漏,并能够更好地区分核心区域和边界区域。该算法的截断距离计算公式如下:

$$d_c=\mu^K+\sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(\delta_i^K-\mu^K)^2}\ , \tag{5}$$

$$\mu^K=\frac{1}{N}\sum_{i=1}^{N}\delta_i^K\ , \tag{6}$$

式中,N 为数据集的样本点总数,δ_i^K 为样本点 i 与其第 K 个最近邻样本点之间的距离,μ^K 为所有点的 δ_i^K 的均值。虽然引入 K 近邻思想能够使得截断距离 d_c 的值自适应选取,但是其输入参数 K 的值需要预先给定,而如何选取合适的 K 值也是一个需要研究的方向。

为了避免在改进算法的过程中出现需要选取参数的问题,王洋等[17]研究发现计算点势能的方法与 DPC 算法中计算 ρ 的方法相似,认为截断距离的最优值等价于电势能计算中的影响因子 σ 的最优值。而基尼指数 G 会随 σ 的改变而改变,因此,将基尼指数 G 最小时对应的 σ 作为截断距离的最优值;在聚类中心的选取上,根据 γ 的排序图中两点间的斜率差的变化来选取聚类中心。最终,该文算法实现了 DPC 算法的截断距离和聚类中心的自适应选取。

有研究将智能优化算法与 DPC 算法结合,如朱红等[18]将果蝇优化算法与 DPC 算法相结合,提出了 Density Peaks Clustering Based on Fruit Fly Optimization Algorithm (FOA-DPC)算法。该算法将截断距离 d_c 以及类簇数 k 作为决策变量,采用果蝇优化算法进行寻优,找到最优值后,采用公式(4)计算 γ_i 的值,选取前 k 个点作为聚类中心,对图像进行分割。

2.2　局部密度和距离的改进

第 2 种改进方法的主体是局部密度和距离的计算公式。DPC 算法的局部密度和距离的测量是基于截断距离的值,很难得到最优的参数。谢娟英等[19]提出的 K-Nearest Neighbors Optimized Clustering Algorithm by Fast search and Finding the Density Peaks (KNN-DPC)算法采用指数核函数,根据样本的 K 近邻信息重新定义局部密度的计算公式,使得局部密度的计算与参数 d_c 的取值无关,更准确地发现聚类中心。但是,其聚类中心的选择仍是人机交互模式。

Liu 等[20]提出了 Shared-Nearest-Neighbors-based Clustering by Fast Search and Find of Density Peaks (SNN-DPC)算法。该算法提出了共享最近邻 SNN 和共享最近邻相似度 Sim,将 Sim 引入局部密度的计算中,使得局部密度和距离的计算与截断距离无关,并且提出了新的剩余样本点分配方案,避免 DPC 算法一步分配策略易导致的“多米诺骨牌效应”的影响。从实验结果来看,SNN-DPC 算法的聚类准确性得到了提高。

虽然 KNN-DPC 算法和 SNN-DPC 算法避免了参数 d_c 对聚类结果的影响,但是对于稀疏密度相差较大的数据集,其聚类中心较难选取。因此,薛小娜等[21]提出了 Improved Density Peaks Clustering Algorithm (IDPCA)。该算法在计算局部密度时引入带有相似性系数的高斯核函数,既避免了截断距离对聚类结果的影响,又使得算法适用于任意数据集。

贾露等[22]提出的 Physics Improved Density Peak Clustering Algorithm (W-DPC)引入了物理学中的万有引力定律,用于重新定义局部密度的计算。样本间距离越小,吸引力越大,局部密度越大,从而易

于找到高密度点和选择聚类中心,同时还引入第一宇宙速度用于处理剩余样本点。

以上4种改进算法虽然都避免了截断距离对聚类结果的影响,但是都引入了新的参数,如KNN-DPC算法、SNN-DPC算法以及IDPCA算法中都需要预先给定样本近邻K的值,而W-DPC算法需要给出扫描半径 r 的值。除此之外,这4种算法的聚类中心选取方面均是采用决策图法,需要人为参与。

2.3 聚类中心选取方式的改进

第3种改进方式的主体是聚类中心的选取。王星等[23]提出了Fast Searching Clustering Centers Algorithm based on Linear Regression Analysis (LR-CFDP)算法,该算法利用线性回归模型和残差分析,实现了聚类中心自动选取,解决了算法聚类中心需要人机交互选择的问题,避免了主观影响。

同样是将数学理论用于DPC算法的改进,崔世琦等[24]将高斯核函数的数学性质用于DPC算法的局部密度度量优化,并在聚类中心选取时利用γ值的中位数和绝对中位差求取残差 R_i ,选取前 r 个作为潜在聚类中心,计算α显著水平下的检验临界值 λ_i ,将原来的潜在聚类中心中 $\lambda_i > R_i$ 的点作为最终的聚类中心,实现了聚类中心的自适应选取,但是对于高维数据集,该算法的性能不理想。因此,江平平等[25]提出了Improved Density Peak Clustering Algorithm based on Grid (G-DPC)算法。该算法采用网格划分法将样本空间划分为均等且不相交的网格单元,聚类中心的选取依据公式(7)和(8):

$$\rho_{C_i} - \mu(\rho_i) \geqslant 0, \tag{7}$$

$$(\delta_{C_i} - E(\delta_i))/2 \geqslant \sigma(\delta_i), \tag{8}$$

若网格代表点满足这两个公式,即为所寻聚类中心点,其中 ρ_{C_i} 为聚类中心的网格代表点的局部密度值,$\mu(\rho_i)$ 是所有网格代表点的局部密度均值,δ_{C_i} 则表示同一类簇中其他代表点与聚类中心的代表点间的最短距离,$E(\delta_i)$ 表示所有 δ_i 的期望。该算法实现了聚类中心自适应选取。

3 自适应DPC算法指标分析

3.1 聚类准确率(ACC)

准确率[26]是计算算法正确划分的样本数占总样本数的比例,如式(9)所示。准确率的取值区间为[0,1],其值越大,表示算法的聚类结果越接近于正确的划分。

$$\mathrm{ACC} = \frac{1}{N}\sum_{i=1}^{N}\psi(x_i^U, x_i^V), \text{其中,} \psi(x_i^U, x_i^V) = \begin{cases} 1, x_i^U = x_i^V \\ 0, x_i^U \neq x_i^V \end{cases}, \tag{9}$$

式中,$U=(U_1,U_2,\cdots,U_L)$ 是数据集 D 的标准划分,$V=(V_1,V_2,\cdots,V_L)$ 是优化算法的聚类结果,x_i^U 表示样本 x_i 在 U 中的标签类,x_i^V 表示样本 x_i 在 V 中的标签类。

表1为DPC算法及6种改进算法作用在UCI数据集上的聚类准确率。可以看出,KM-DPC和IDPCA算法在Seeds数据集中取得最优的聚类结果,在Segmentation数据集中表现最佳的是KM-DPC算法;在Iris数据集中,KNN-DPC和SNN-FKNN-DPC两种算法聚类结果最好;其余的3个数据集ACC值最大的均为SNN-FKNN-DPC算法。总体来说,从ACC值来看,6种改进算法均优于DPC算法,而SNN-FKNN-DPC算法则是几个数据集中聚类最优的算法。基于聚类中心自适应改进的AD-PC-WKNN和AKDP算法与原算法相比聚类性能有了一定程度的改进,但是与基于局部密度计算方式改进的其他算法相比,性能优势不够明显。

表1 7种算法在UCI数据集上的聚类准确率

Table 1 Clustering accuracy of 7 algorithms on the UCI data set

Algorithm	Iris	Wine	Seeds	Ionosphere	Segmentation	Dermatology
DPC[3]	0.887	0.882	0.900	0.681	0.684	0.697
KNN-DPC[19]	**0.973**	0.948	0.923	0.729	0.717	0.768
KM-DPC[27]	0.960	0.960	0.**938**	0.821	0.**776**	0.809
IDPCA[21]	–	–	0.**938**	0.769	0.767	0.842
SNN-FKNN-DPC[28]	**0.973**	**0.978**	0.924	0.**858**	–	0.**867**
AD-PC-WKNN[29]	0.942	0.917	0.917	–	–	–
AKDP[30]	0.954	0.893	0.922	–	–	–

注:“–”表示没有对应的数据,加粗数据表示在该数据集中的最优聚类准确率

Note:"–" indicates that there is no corresponding data,bold data indicates the optimal clustering accuracy in the data set

3.2 Adjusted Mutual Information (AMI)

AMI[31] 是基于信息论的聚类度量指标,通过互信息(Mutual information)度量两个事件集合的相关性,如式(10)所示:

$$\mathrm{AMI}(U,V)=\frac{MuI(U,V)-E\{MuI(U,V)\}}{\max\{H(U),H(V)\}-E\{MuI(U,V)\}},\tag{10}$$

式中,$U=(U_1,U_2,\cdots,U_L)$ 是数据集 D 的标准划分,$V=(V_1,V_2,\cdots,V_L)$ 是优化算法的聚类结果,$MuI(U,V)$ 表示事件 U 与事件 V 之间的互信息,如式(11)所示,互信息是一种对称度量,用于量化两个分布之间共享的统计信息。$E\{MuI(U,V)\}$ 是 U 和 V 之间的期望互信息,如式(12)所示。$H(U)$ 和 $H(V)$ 分别是 U 和 V 的熵。

$$MuI(U,V)=\sum_{i=1}^{K}\sum_{j=1}^{K'}P(i,j)\log\frac{P(i,j)}{P(i)P'(j)},\tag{11}$$

式中,K 和 K' 分别是标准划分 U 和聚类结果 V 中的类簇个数,$P(i)=\frac{|U_i|}{N}$ 表示任意选择的样本属于簇 U_i 的概率,$P(j)=\frac{|V_j|}{N}$ 表示任意选择的样本属于簇 V_j 的概率,$P(i,j)=\frac{|U_i\cap V_j|}{N}$ 表示任意选取的样本在 U 中属于 U_i 且在 V 中属于 V_j 的概率。

$$E\{MuI(U,V)\}=\sum_{i=1}^{K}\sum_{j=1}^{K'}\sum_{n_{ij}=\max(a_i+b_{j-N},0)}^{\min(a_i,b_j)}\frac{n_{ij}}{N}\log\left(\frac{N\cdot n_{ij}}{a_ib_j}\right)\times\frac{a_i!\,b_j!\,(N-a_i)!\,(N-b_j)!}{N!\,n_{ij}!\,(a_i-c_{ij})!\,(b_j-c_{ij})!\,(N-a_i-b_j+c_{ij})!},\tag{12}$$

式中,K 和 K' 分别为 U 和 V 中的类簇数,$a_i=\sum_{i=1}^{K}n_{ij}$,$b_j=\sum_{i=1}^{K'}n_{ij}$,$n_{ij}=|U_i\cap V_j|$,$n_{ij}$ 为选择的样本在 U 中属于 U_i 且在 V 中属于 V_j 的样本总数。

AMI 的取值范围是[-1,1],其值越接近 1,表示算法的聚类结果越优,越接近于真实结果。

由表 2 可以看出,5 种改进算法的 AMI 值大部分都优于原始的 DPC 算法。Wine 数据集中 AMI 值最优的是 SNN-FKNN-DPC 算法,Seeds 数据集最优的是 W-DPC 算法,Libras movement 和 Waveform 数据集中表现最佳的是 SNN-DPC 算法,Waveform (noise)数据集中 KM-DPC 算法取得最优的 AMI 值。关于 Iris 数据集,KNN-DPC、SNN-FKNN-DPC 以及 SNN-DPC 这 3 种算法的 AMI 值均为 0.912,原因是该数据集中的簇重叠严重,而这 3 种算法均是引入近邻思想,受该数据集的特殊邻域环境影响,这 3 种算法在 Iris 数据集的 AMI 值相等。

表 2 6 种聚类算法在各数据集上的 AMI 值

Table 2 AMI values of six clustering algorithms on each data set

Algorithm	Iris	Wine	Seeds	Libras Movement	Waveform	Waveform (noise)
DPC[3]	0.767 0	0.706 0	0.717 0	0.390 0	0.318 0	0.184 0
KNN-DPC[19]	**0.912 0**	0.829 0	0.785 0	0.523 0	0.313 0	0.245 0
SNN-FKNN-DPC[20]	**0.912 0**	**0.908 0**	0.767 0	0.507 0	0.382 0	0.296 0
KM-DPC[27]	0.883 0	0.860 0	0.777 0	0.505 0	0.386 0	**0.390 0**
W-DPC[22]	0.911 5	0.867 8	**0.820 9**	0.381 0	-	-
SNN-DPC[20]	**0.912 0**	0.874 0	0.751 0	**0.583 0**	**0.398 0**	0.326 0

注:"-"表示没有对应的数据,加粗数据表示在该数据集中的最优聚类准确率

Note:"-" indicates that there is no corresponding data,bold data indicates the optimal clustering accuracy in the data set

3.3 Adjusted Rand Index (ARI)

兰德指数(Rand Index,RI)只考虑表 3 所示的 a 和 d 两种聚类结果的情况,忽略了 b 和 c 两种聚类结果,评价方式较为片面并且没有区分度,其计算公式如式(13)。其中,$U=(U_1,U_2,\cdots,U_L)$ 是数据集 D 的标准划分,$V=(V_1,V_2,\cdots,V_L)$ 是优化算法的聚类结果:

$$\mathrm{RI}(U,V)=\frac{a+d}{a+b+c+d},\tag{13}$$

ARI[32] 是基于 RI 的改进,度量标准划分 U 和聚类结果 V 之间的相似程度,如式(14),也可用式(15)来表示。ARI 的取值范围为[-1,1],数值越高表示聚类划分效果越好。

$$\mathrm{ARI}(U,V)=\frac{\mathrm{RI}(U,V)-E\{\mathrm{RI}(U,V)\}}{\max\{\mathrm{RI}(U,V)\}-E\{\mathrm{RI}(U,V)\}},\tag{14}$$

$$\mathrm{ARI}(U,V)=$$

$$\frac{\sum_{i,j}\binom{n_{ij}}{2}-\left[\sum_i\binom{n_{i\cdot}}{2}\sum_j\binom{n_{\cdot j}}{2}\right]/\binom{n}{2}}{\frac{1}{2}\left[\sum_i\binom{n_{i\cdot}}{2}+\sum_j\binom{n_{\cdot j}}{2}\right]-\left[\sum_i\binom{n_{i\cdot}}{2}\sum_j\binom{n_{\cdot j}}{2}\right]/\binom{n}{2}}, \tag{15}$$

式中，$n_{ij}=|U_i\cap V_j|$为在U中属于U_i且在V中属于V_j的样本总数，$n_{i\cdot}$表示在U中属于类簇U_i的样本个数，$n_{\cdot j}$表示在V中属于类簇V_j的样本个数。

表 3　算法聚类的结果分布情况

Table 3　Result distribution of algorithm clustering

类型 Type	描述 Description
a	在U和V中均在同一类簇的样本对数目 The number of sample pairs in U and V are in the same class cluster
b	在U中划分在同一类簇，但在V中未划分在同一类簇的样本对数目 The number of sample pairs classified in the same cluster in U, but not in the same cluster in V
c	在U中未划分在同一类簇，但在V中划分在同一类簇的样本对数目 The number of sample pairs that are not classified in the same cluster in U, but are in the same cluster in V
d	在U和V中均未划分在同一类簇的样本对数目 The number of sample pairs where U and V are not classified in the same cluster

表 4 是 6 种改进算法和 DPC 算法在 UCI 数据集的 ARI 值。相比于 DPC 算法，各改进算法在 UCI 数据集的 ARI 值均有所改善，其中，在 Iris 数据集中，SNN-DPC 算法表现最佳；SNN-FKNN-DPC 算法在 Wine 和 Libras movement 两个数据集的聚类结果相比于其他算法较优；KM-DPC 算法在 Seeds 和 Segmentation 数据集的 ARI 值最大；在 WDBC 数据集中，聚类效果最优的是 SNN-DPC 算法。

表 4　6 种算法在 UCI 数据集的 ARI 值

Table 4　ARI values of 6 algorithms on UCI data set

Algorithm	Iris	Wine	Seeds	Libras Movement	WDBC	Segmentation
DPC[3]	0.720	0.672	0.734	0.214	−0.011	0.550
KNN-DPC[19]	0.922	0.844	0.788	0.331	0.783	0.539
KM-DPC[27]	0.886	0.884	**0.835**	0.291	0.818	**0.632**
SNN-DPC[26]	**0.922 2**	0.899 2	0.789 0	0.392 7	**0.850 3**	0.577 0
W-DPC[22]	0.868 1	0.800 6	0.732 0	0.323 2	0.805 1	–
SNN-FKNN-DPC[28]	0.922	**0.933**	0.791	0.**407**	–	–

注：“–”表示没有对应的数据，加粗数据表示在该数据集中的最优聚类准确率

Note: "–" indicates that there is no corresponding data, bold data indicates the optimal clustering accuracy in the data set

3.4　F-Measure

F-Measure[33]指标综合了查准率(Precision)和查全率(Recall)两种评价指标，其优势在于对聚类结果的整体区分能力。一般的聚类结果分布情况总结如表 3 所示。F-Measure 的取值范围为[0,1]，数值越高表示聚类效果越好。

查准率评估聚类结果的精确程度，计算方式如公式(16)所示。查全率评估实验结果的完备程度，计算方式如公式(17)所示。F-Measure 的计算方式如式(18)所示。

$$\text{precision}=\frac{a}{a+c}, \tag{16}$$

$$\text{recall}=\frac{a}{a+b}, \tag{17}$$

$$F_\beta=(1+\beta^2)\frac{\text{precision}\cdot\text{recall}}{\beta^2\cdot\text{precision}+\text{recall}}, \tag{18}$$

在此，$\beta=1$，即 $F_1=\dfrac{2\cdot\text{precision}\cdot\text{recall}}{\text{precision}+\text{recall}}$。

由表 5 可以看出，ADPC-KNN 算法在 Seeds 和 Libras Movement 两个数据集中的 F-Measure 值较

表 5　5 种算法在 UCI 数据集的 F-Measure 值

Table 5　F-Measure values of 5 algorithms on UCI data set

Algorithm	Iris	Wine	Seeds	Libras Movement	Ecoli	WDBC
DPC[3]	0.923 3	0.783 5	0.844 4	0.371 7	0.577 5	0.725 7
KNN-DPC[19]	0.935 5	0.866 7	0.827 6	0.397 6	0.691 9	0.765 8
ADPC-KNN[15]	0.900 0	0.720 0	**0.920 0**	**0.500 0**	0.640 0	–
SNN-DPC[20]	**0.947 9**	**0.933 0**	0.858 9	0.450 7	**0.824 3**	**0.930 5**
W-DPC[22]	0.911 5	0.867 8	0.820 9	0.404 0	0.659 2	0.910 0
E-DPC[24]	0.905 0	0.601 0	0.877	–	–	–

注：“–”表示没有对应的数据，加粗数据表示在该数据集中的最优聚类准确率

Note: "–" indicates that there is no corresponding data, bold data indicates the optimal clustering accuracy in the data set

其他算法大,即该算法在这两个数据集中的表现最佳;而在 Iris、Wine、Ecoli 以及 WDBC 4 个数据集中聚类结果最优的是 SNN-DPC 算法。

3.5 算法平均运行时间

由表 6 可以看出,3 种改进算法的平均运行时间均大于 DPC 算法,而由前面的 ACC、AMI、ARI 以及 F-Measure 4 个指标可以看出,这些算法的聚类结果都比 DPC 算法有所改善,但是其运行时间都比 DPC 算法慢。

表 6 4 种算法在 UCI 数据集上的平均运行时间(ms)

Table 6 Average running time of 4 algorithms on UCI data set (ms)

Algorithm	Iris	Wine	Seeds	WDBC	Ecoli	Libras Movement	Aggregation
DPC[3]	7.2	5.5	6.9	43.7	14.6	21.1	69.7
KNN-DPC[19]	18.1	19.1	21.9	94.5	36.3	48.1	159.7
SNN-DPC[20]	41.4	53.5	55.9	352.7	160.1	150.2	622.2
W-DPC[22]	30.5	59.7	55.4	421.8	36.3	306.2	519.9

由表 1、表 2、表 4、表 5 及表 6 的数据可以看出,3 种方向上的改进算法相比于原来的算法,聚类性能在一定程度上都得到了提升,但是从整体上来看,针对 d_c 值选取的改进算法以及针对聚类中心选取的改进算法,在数据集上的聚类效果不如基于局部密度计算公式的改进算法。5 个表格中的数据集均为规模较小的数据集,说明已改进的算法在处理规模较小、数据分布较为均匀的数据集时聚类效果比较理想。

4 展望

本文主要分析了目前针对 DPC 算法参数 d_c 及其聚类中心的选取不能自适应的缺陷,研究者对其进行改进的研究工作,并对改进算法的聚类结果指标进行分析。未来可从以下 3 个方面进行深入研究:

①将智能优化算法与 DPC 聚类算法有机结合,研究自适应 DPC 自动聚类算法:目前已有的对于 DPC 算法的自适应改进方式,主要是针对参数的自适应或者在选取聚类中心时无需人为参与,两者同时达到自适应效果的改进仍然较少,基于此,对 DPC 算法的自适应研究还可以更加完善;

②DPC 算法参数选取的数学理论依据分析:目前参数的选取主要依赖经验策略,缺乏数学理论的支撑;

③高维空间 DPC 聚类算法理论与方法研究:虽然 PDC 算法能够识别任意形状簇,但是对于高维数据集,该算法的处理性能不够理想,而现有的针对高维数据的改进方式主要是基于 PCA 的改进算法,因此,DPC 在高维空间的研究有待进一步探索。

参考文献

[1] JAIN A K. Data clustering: 50 years beyond K-means [J]. Pattern Recognition Letters,2010,31(8):651-666.

[2] 冯少荣,肖文俊. DBSCAN 聚类算法的研究与改进[J]. 中国矿业大学学报,2008,37(1):105-111.

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

[4] ZENG X H,CHEN A Z,ZHOU M. Color perception algorithm of medical images using density peak based hierarchical clustering [J]. Biomedical Signal Processing and Control,2019,48:69-79.

[5] LIU S,ZHU L Z,SHEONG F K,et al. Adaptive partitioning by local density-peaks: An efficient density-based clustering algorithm for analyzing molecular dynamics trajectories [J]. Journal of Computational Chemistry,2017,38(3):152-160.

[6] ZHANG Y,XIA Y Q,LIU Y,et al. Clustering sentences with density peaks for multi-document summarization [C]// Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, Colorado,2015:1262-1267.

[7] WANG B Y,ZHANG J,DING F G,et al. Multi-document news summarization via paragraph embedding and density peak clustering [C]//2017 International Conference on Asian Language Processing (IALP),Singapore: IEEE,2017.

[8] WANG M M,ZUO W L,WANG Y. An improved density peaks-based clustering method for social circle discovery in social networks [J]. Neurocomputing,2016,179: 219-227.

[9] XU M L,LI Y H,LI R X,et al. EADP: An extended adaptive density peaks clustering for overlapping community detection in social networks [J]. Neurocomputing, 2019,337:287-302.

[10] LU H,SHEN Z,SANG X S,et al. Community detection method using improved density peak clustering and nonnegative matrix factorization [J]. Neurocomputing,2020,415:247-257.

[11] KÜHROVÁ P,BEST R B,BOTTARO S,et al. Com-

puter folding of RNA tetraloops: Identification of key force field deficiencies [J]. Journal of Chemical Theory and Computation,2016,12(9):4534-4548.
[12] CHEN J G,LI K L,RONG H G,et al. A disease diagnosis and treatment recommendation system based on big data mining and cloud computing [J]. Information Sciences,2018,435:124-149.
[13] 淦文燕,刘冲.一种改进的搜索密度峰值的聚类算法[J].智能系统学报,2017,12(2):229-236.
[14] 蒋礼青,张明新,郑金龙,等.快速搜索与发现密度峰值聚类算法的优化研究[J].计算机应用研究,2016,33(11):3251-3254.
[15] 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.
[16] 罗军锋,锁志海,郭倩.一种基于k近邻的密度峰值聚类算法[J].软件,2019,41(7):185-188.
[17] 王洋,张桂珠.自动确定聚类中心的密度峰值算法[J].计算机工程与应用,2018,54(8):137-142.
[18] 朱红,何瀚志,方谦昊,等.基于改进密度峰值聚类的医学图像分割[J].徐州医科大学学报,2018,38(10):652-658.
[19] 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[J].中国科学:信息科学,2016,46(2):258-280.
[20] 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.
[21] 薛小娜,高淑萍,彭弘铭,等.结合K近邻的改进密度峰值聚类算法[J].计算机工程与应用,2018,54(7):36-43.
[22] 贾露,张德生,吕端端.物理学优化的密度峰值聚类算法[J].计算机工程与应用,2020,56(13):47-53.
[23] 王星,冯鹏程,王玉冰,等.基于线性回归分析的快速搜索聚类中心算法[J].系统工程与电子技术,2017,39(11):2614-2622.
[24] 崔世琦,刘冰,李勇.基于SH-ESD优化的密度峰值快速搜索聚类算法[J].长春工业大学学报,2020,41(2):149-156.
[25] 江平平,曾庆鹏.一种基于网格划分的密度峰值聚类改进算法[J].计算机应用与软件,2019,36(8):268-274,280.
[26] 薛小娜,高淑萍,彭宏铭,等.基于K近邻和多类合并的密度峰值聚类算法[J].吉林大学学报(理学版),2019,57(1):111-120.
[27] 刘奕志,程汝峰,梁永全.一种基于共享近邻的密度峰值聚类算法[J].计算机科学,2018,45(2):125-129,146.
[28] 杨震,王红军.基于加权K近邻的改进密度峰值聚类算法[J].计算机应用研究,2020,37(3):667-671.
[29] 钱雪忠,金辉.自适应聚合策略优化的密度峰值聚类算法[J].计算机科学与探索,2020,14(4):712-720.
[30] CARPANETO G,TOTH P. Algorithm 548: Solution of the assignment problem [H] [J]. ACM Transactions on Mathematical Software,1980,6(1):104-111.
[31] VINH N X,EPPS J R,BAILEY J. Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance [J]. The Journal of Machine Learning Research,2010,11:2837-2854.
[32] HUBERT L,ARABIE P. Comparing partitions [J]. Journal of Classification,1985,2(1):193-218.
[33] WANG C X,LIU L G. Feature matching using quasi-conformal maps [J]. Frontiers of Information Technology & Electronic Engineering,2017,18(5):644-657.

Research and Analysis of Adaptive Density Peak Clustering Algorithm

GE Lina[1,2,3],CHEN Yuanyuan[1],ZHOU Yongquan[1,3]
(1. School of Artificial Intelligence,Guangxi University for Nationalities,Nanning,Guangxi,530006,China;2. Key Laboratory of Network Communication Engineering,Guangxi University for Nationalities,Nanning,Guangxi,530006,China;3. Guangxi Key Laboratory of Hybrid Computation and IC Design Analysis,Nanning,Guangxi,530006,China)

Abstract: Clustering by fast search and find of density peak (DPC) is a new type of density-based clustering algorithm. It selects the sample points with high density and far from other higher density points as the clustering center, and then clustering is carried out according to the local density and distance between samples. Although the parameters of the DPC algorithm are unique, simple and efficient, the value of the cutoff distance is set according to the empirical strategy, and the improper selection of cutoff distance will lead to errors in the calculation of local density ρ and distance δ. On the other hand, the selection of clustering center adopts human-computer interaction mode, which has a great subjective influence on the clustering results. Aiming at these defects of DPC algorithm, there are three main improvement directions at present: improving the selection of cutoff distance, improving the calculation method of local density and distance, and improving the method of selecting cluster centers. Through the improvement of these three directions, the process of DPC is adaptive. Finally, the future work is prospected and the future research direction is given: DPC algorithm and intelligent algorithm are organically combined to realize algorithm adaptation, and the performance of the algorithm to deal with high-dimensional data sets needs to be further explored.

Key words: density peak; clustering algorithm; adaptive; cutoff distance; clustering center

责任编辑:陆　雁