软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2020,31(11):3321−3333 [doi: 10.13328/j.cnki.jos.005813]

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

基于不相似性度量优化的密度峰值聚类算法*

丁世飞[1,2], 徐 晓[1], 王艳茹[1]

[1](中国矿业大学 计算机科学与技术学院,江苏 徐州 221116)

[2](中国科学院 计算技术研究所 智能信息处理重点实验室,北京 100190)

通讯作者: 丁世飞, E-mail: dingsf@cumt.edu.cn

摘 要: 密度峰值聚类(clustering by fast search and find of density peaks,简称 DPC)是一种基于局部密度和相对距离属性快速寻找聚类中心的有效算法.DPC通过决策图寻找密度峰值作为聚类中心,不需要提前指定类簇数,并可以得到任意形状的簇聚类.但局部密度和相对距离的计算都只是简单依赖基于距离度量的相似度矩阵,所以在复杂数据上 DPC 聚类结果不尽如人意,特别是当数据分布不均匀、数据维度较高时.另外,DPC 算法中局部密度的计算没有统一的度量,根据不同的数据集需要选择不同的度量方式.第三,截断距离 d_c 的度量只考虑数据的全局分布,忽略了数据的局部信息,所以 d_c 的改变会影响聚类的结果,尤其是在小样本数据集上.针对这些弊端,提出一种基于不相似性度量优化的密度峰值聚类算法(optimized density peaks clustering algorithm based on dissimilarity measure,简称 DDPC),引入基于块的不相似性度量方法计算相似度矩阵,并基于新的相似度矩阵计算样本的 K 近邻信息,然后基于样本的 K 近邻信息重新定义局部密度的度量方法.经典数据集的实验结果表明,基于不相似性度量优化的密度峰值聚类算法优于 DPC 的优化算法 FKNN-DPC 和 DPC-KNN,可以在密度不均匀以及维度较高的数据集上得到满意的结果;同时统一了局部密度的度量方式,避免了传统 DPC 算法中截断距离 d_c 对聚类结果的影响.

关键词: 密度峰值聚类;局部密度;决策图;不相似性度量;密度不均匀

中图法分类号: TP301

中文引用格式: 丁世飞,徐晓,王艳茹.基于不相似性度量优化的密度峰值聚类算法.软件学报,2020,31(11):3321−3333. http://www.jos.org.cn/1000-9825/5813.htm

英文引用格式: Ding SF, Xu X, Wang YR. Optimized density peaks clustering algorithm based on dissimilarity measure. Ruan Jian Xue Bao/Journal of Software, 2020,31(11):3321−3333 (in Chinese). http://www.jos.org.cn/1000-9825/5813.htm

Optimized Density Peaks Clustering Algorithm Based on Dissimilarity Measure

DING Shi-Fei[1,2], XU Xiao[1], WANG Yan-Ru[1]

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

[2](Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China)

Abstract: Clustering by fast search and find of density peaks (DPC) is an efficient algorithm for finding cluster centers quickly based on local-density and relative-distance. DPC uses the decision graph to find the density peaks as cluster centers. It does not need to specify the number of clusters in advance and clusters with arbitrary shapes can be obtained. However, the calculation of local-density and relative-distance depends on the similarity matrix which is based on distance metrics simply, thus, DPC is not satisfactory on complex datasets, especially when the datasets with uneven density and higher dimensions. In addition, the measurement of the local-density is not unified and different methods correspond to different datasets. Third, the measurement of d_c only considers the global distribution of

* 基金项目: 国家自然科学基金(61672522, 61379101); 国家重点基础研究发展计划(973)(2013CB329502)

Foundation item: National Natural Science Foundation of China (61672522, 61379101); National Program on Key Basic Research Project of China (973) (2013CB329502)

收稿时间: 2018-04-19; 修改时间: 2018-07-25; 采用时间: 2019-01-15

datasets, ignoring the local information of the data, so the change of d_c will affect the results of clustering, especially on small scale datasets. Aiming at these shortcomings, this study proposes an optimized density peaks clustering algorithm based on dissimilarity measure (DDPC). DDPC introduces a mass-based dissimilarity measure to calculate the similarity matrix, and calculates the *k*-nearest neighbor information of the sample based on the new similarity matrix. Then local-density is redefined by the *k*-nearest neighbor information. Experimental results show that the optimized density peaks clustering algorithm based on dissimilarity measure is superior to the optimized FKNN-DPC and DPC-KNN clustering algorithms, and can be satisfied on datasets with uneven density and higher dimensions. As a result, the local-density measurement method is unified at the same time, which avoids the influence of d_c on the clustering results in the traditional DPC algorithm.

Key words: density peaks clustering; local-density; decision graph; dissimilarity measure; uneven density

聚类分析是数据挖掘的有效技术之一,因为聚类分析不依赖于类的预先定义以及数据样本的标签,所以被称为无监督学习[1].目前,聚类分析在市场分析、模式识别、基因研究、图像处理等领域具有一定的应用价值[2−4].

聚类分析的主要目标是:根据某种相似性(不相似性)度量把数据对象分成多个类,尽可能使同一类簇内样本的相似度较大,不同类簇之间样本的相似度较小[5].然而,相似性的度量至今没有统一的定义,不同的数据类型对应不同的相似性度量定义,得到不同的聚类结果[6].类似地,不同的聚类目标对应不同的聚类算法,目前,聚类算法主要被分为基于划分的聚类、基于密度的聚类、基于网格的聚类、基于层次以及基于模型的聚类[7].这 5 大类算法各有利弊,不同的算法适用于不同类型的数据集.例如,经典的 K 均值[8]聚类算法在凸球形结构的数据集上具有满意的聚类结果,但却对初始值的设置敏感;尽管 DBSCAN[9]在不规则簇簇上提供了良好的聚类结果,并且不需要预先设定类簇数,但对于密度分布不均匀和高维较高的数据集,聚类结果不尽如人意.

2014 年,Rodriguez 等人[10]提出一种快速寻找聚类中心的密度峰值聚类算法(clustering by fast search and find of density peaks,简称 DPC).DPC 算法利用数据的局部密度以及相对距离属性快速确定聚类中心,可以用于任意形状数据的聚类分析,并有效进行样本点分配,得到比较满意的聚类结果[11].但是它存在以下不足.

(1) 局部密度和相对距离的计算基于数据之间的相似性,而相似性的度量却只是简单依赖于数据之间的几何距离,所以在复杂数据上,DPC 无法得到满意的聚类结果,特别是当数据分布不均匀和数据维度较高时[12].

(2) 局部密度的计算没有统一的度量,根据不同的数据集大小,需要选择不同的度量方式[13].

(3) 截断距离 d_c 的度量只考虑数据的全局分布,在小数据集上,d_c 的改变会影响聚类的结果[14].

针对以上不足,本文提出一种基于不相似性度量优化的密度峰值聚类算法(optimized density peaks clustering algorithm based on dissimilarity measure,简称 DDPC).DDPC 算法的主要创新点包括:(1) 考虑数据分布的周围环境,利用概率块代替几何距离度量数据之间的相似性,提高数据在较高维度以及分布不均匀数据集上的聚类精度;(2) 利用样本的 K 近邻信息重新定义局部密度,统一不同大小数据集上局部密度的度量方式;(3) 改进的局部密度度量,使 DDPC 算法的聚类结果不受截断距离 d_c 变化的影响.

1 DPC 聚类算法

一种基于局部密度和相对距离的密度聚类算法 DPC 由 Rodrigue 等人在 Science 上提出.DPC 算法基于一个重要假设:聚类中心的局部密度大于周围邻居的局部密度;聚类中心的距离密度比其高的点的距离相对较远[15].DPC 算法分为两个步骤完成聚类:(1) 确定聚类中心;(2) 分配剩下的点[16].

DPC 算法首先根据聚类中心的特点为每一个数据点赋予局部密度ρ_i和相对距离δ_i属性.ρ_i的物理意义表示与点 x_i 的距离小于 d_c 的点的个数,定义为

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

其中,d_{ij} 表示数据点 x_i 和 x_j 的距离.d_c 是唯一的输入参数,表示截断距离,定义为两两数据点之间相似度按从小到大排列后 2%位置处的值.另外,当数据集较小时,局部密度ρ_i以高斯核函数的形式被定义.

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

数据点 x_i 的 δ_i 是点到所有比其局部密度大的点的距离的最小值,其公式为

$$\delta_i = \min_{j:\rho_j>\rho_i}(d_{ij}) \tag{3}$$

对于密度最大的点,我们可以得到:

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

DPC 算法选择 ρ_i 和 δ_i 均较大的数据点作为聚类中心.为了可以自动确定聚类中心,DPC 算法借助决策图选择聚类中心.决策图的绘制以 ρ_i 为横坐标,δ_i 作为纵坐标.在实际情况中,为了有助于更准确地确定聚类中心,算法定义一个参数 γ_i.

$$\gamma_i=\rho_i\cdot\delta_i \tag{5}$$

此时,DPC 算法将根据 γ_i 绘制决策图,选择 γ_i 大的点作为聚类中心.

DPC 算法确定好聚类中心后,需要将剩余的点分配到相应的类簇中.剩余的点分为一般的数据点以及噪声点.DPC 算法首先将所有剩下的点归于局部密度等于或者高于其最近点一类,而后为每一个类簇定义一个边界阈值,边界阈值即为划分为该类但是距离其他类簇的点的距离小于 d_c 的点,然后将局部密度最大的点的值作为阈值,小于该阈值的点将作为噪声点去除完成聚类.

DPC 算法简单有效,能够很好地处理噪声孤立点,而且可以得到任意形状的簇聚类,同时不需要提前指定数据中类簇的数量,并且需要用户指定的参数比较少,因此近几年得到了很多学者的关注.在 DPC 的应用方面,2015 年,Zhang 等人[17]利用密度峰值聚类算法提取多文档摘要,提出了一个同时测量代表性和多样性的统一句子评分模型,优于现有的 MDS 方法;2016 年,Chen 等人[18]使用 DPC 算法来获得基于人脸图像的可能年龄估计;2017 年,Shi 等人[19]将 DPC 算法应用于场景图像聚类,提出了一种新颖的基于聚类的图像分割方法,能够捕捉图像的固有结构并检测非球形簇.

虽然 DPC 算法在大部分数据集上的聚类结果令人满意,但是 DPC 的缺点也非常明显.

- 首先,局部密度和相对距离的计算都依赖简单的距离度量,所以当数据分布不均匀和数据维度较高时,DPC 无法得到满意的聚类结果.例如图 1 所示,在 3 类密度不均匀的数据集上,DPC 的聚类结果不尽如人意,无法得到准确的 3 类.
- 其次,局部密度的计算没有统一的度量,根据不同的数据集大小,需要选择公式(1)或者公式(2)度量.同时,局部密度的计算依赖于截断距离 d_c 的选择,但是 d_c 只考虑数据的全局分布,忽略了数据的局部信息,所以 d_c 的改变会影响聚类的结果,尤其是在小数据集上.例如图 2 所示,在小数据集 Flame 上,图 2(a)是采用公式(2)计算局部密度,并且截断距离 d_c 取 4%,可以很好地将数据集分成两类;而图 2(b)中,d_c 取值和图 2(a)一样,取 4%,但是局部密度的度量采用公式(1),则并不能得到满意的聚类结果;图 2(c)采用和图 2(a)相同的方法计算局部密度,但是截断距离取 2%,得到的聚类结果也不是很理想.

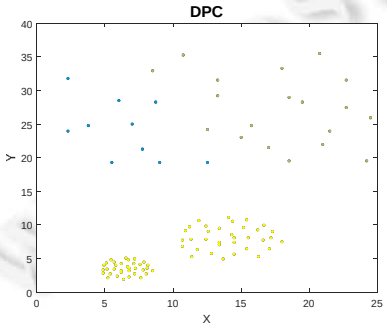
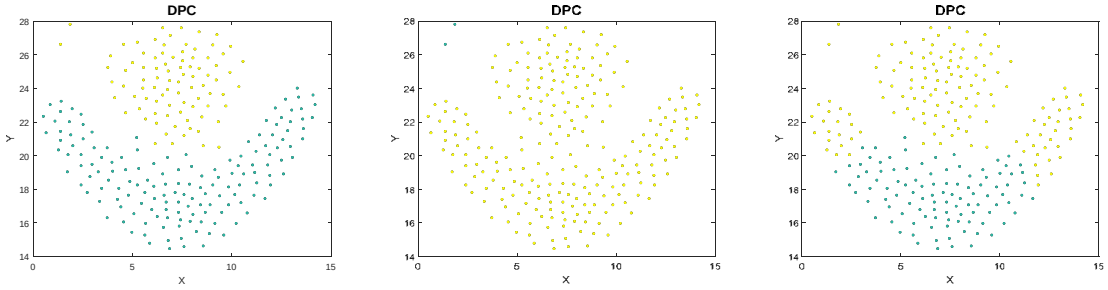


Fig.1 Clustering result of DPC on non-uniform density dataset

图 1 DPC 在密度不均匀数据集上的聚类结果



(a) 采用公式(2),d_c=4%的聚类结果 (b) 采用公式(1),d_c=4%的聚类结果 (c) 采用公式(2),d_c=2%的聚类结果

Fig.2 Clustering results of DPC on the Flame with different local density and different cutoff distance d_c

图 2 DPC 在 Flame 上采用不同局部密度和截断距离 d_c 的聚类结果

从图 2 可以看出,局部密度的计算影响了 DPC 在数据集上的聚类效果.

许多学者针对局部密度的计算方式进行了 DPC 优化.在文献[20]中,Du 等人提出了基于 K 近邻的密度峰值聚类(DPC-KNN)算法,将 K 近邻的概念引入到 DPC 中,考虑数据的局部分布,为计算局部密度提供了另一种选择,从而统一局部密度的度量方式,减少截断距离 d_c 对聚类结果的影响.在文献[21]中,Xie 等人提出了一种基于模糊加权 K 近邻(FKNN-DPC)的密度峰值搜索和点分配算法,使用 K 近邻信息来定义点的局部密度并搜索和发现聚类中心,从而统一局部密度的度量方式,并减少截断距离 d_c 对聚类结果的影响.在文献[22]中,谢娟英等人提出了 K 近邻优化的密度峰值快速搜索聚类算法,利用样本的 K 近邻信息重新定义局部密度属性和分配策略,有效地改进了局部密度度量不理想的弊端.

受以上优化算法的启发,本文提出了基于不相似性度量优化的密度峰值聚类算法(DDPC).DDPC 算法首先利用概率块代替原来的几何距离计算新的相似度矩阵,而后利用新的相似度矩阵获取数据样本的 K 近邻信息,再根据样本的 K 近邻重新定义局部密度的度量方式.由于 DDPC 算法基于新的不相似性度量方式计算相似度矩阵,充分考虑了数据分布的局部环境,所以可以反映更准确的数据结构,在高维数据集以及密度变化的数据集上可以得到更优的聚类结果.同时,由于 DDPC 算法定义了新的局部密度,统一了局部密度在任意大小数据集上的度量方式,避免了 DPC 算法中截断距离 d_c 对聚类结果的影响.

2 基于不相似性度量优化的密度峰值聚类算法

2.1 基于块的不相似性度量

由上一节分析得出,DPC 算法由于直接采用几何距离度量数据样本之间的相似度从而计算局部密度以及相对距离属性,忽略了数据分布的周围环境,所以在复杂数据集,特别是高维数据以及密度不均匀数据集上,DPC 算法聚类效果不尽如人意.自 1970 年代开始,心理学家就表示,两个实例的相似性不能简单地由几何模型表征,相似性的度量受到测量的背景以及实例附近的点的影响[23].基于这个事实,可以定义更合适的相似性度量方式,在这里称为基于块的不相似性度量[24].

基于块的不相似性度量的基本思想是,密集区域的两个实例的相似度小于同等间隔但位于低密度区域的两个实例[25].基于几何模型的相似度计算仅依赖于几何位置推导距离;相反,基于块的不相似性的度量方式主要取决于数据分布,即覆盖两个实例的最小区域的概率块[26].假设 D 表示概率密度函数 F 中的数据样本,$H\in\mathcal{H}(D)$ 表示一个层次划分,将空间划分成不重叠的非空域.

定义 1. $R(x,y|H;D)$表示关于 H 和 D 覆盖 x 和 y 的最小域,定义为

$$R(x,y\mid H;D)=\underset{r\subset H\ \text{s.t.}\ \{x,y\}\in r}{\arg\min}\sum_{z\in D}I(z\in r) \tag{6}$$

其中,$I(\cdot)$表示指数函数.

定义 2. x 和 y 关于 D 和 F 的基于块的不相似性定义为 $R(x,y|H,D)$的期望概率.

$$m(x,y|D,F)=E_{\mathcal{H}(D)}[R_F(R(x,y|H,D))] \tag{7}$$

其中,$R_F(\cdot)$是关于 F 的概率,期望需要取 $\mathcal{H}(D)$中的所有模型.而在实际中,基于块的不相似性是从有限的模型 $H_i\in\mathcal{H}(D),i=1,\ldots,t$ 中估计得到的.

$$m_e(x,y\mid D)=\frac{1}{t}\sum_{i=1}^{t}\tilde{P}(R(x,y\mid H_i;D)),\ \tilde{P}(R)=\frac{1}{|D|}\sum_{z\in D}I(z\in D) \tag{8}$$

注意,$R(x,y|H;D)$是覆盖 x 和 y 的最小区域,类似于 x 和 y 几何模型中最短的距离.

基于块的不相似性度量共定义两个参数 t 和 ψ,t 表示 iTrees 的数目,ψ表示每棵 iTree 的大小,每棵树的高度最高为 $h=\lceil\log_2\psi\rceil$.例如,存在一个大小为 8 的数据集 $X=\{x_1,x_2,x_3,\ldots,x_8\}$.

- 第 1 步,构建一个由 t 个 iTrees 组成的 iForest 作为分区结构 R.

每个 iTree 都使用子集 $\mathcal{D}\subset D$ 独立构建,其中,$|\mathcal{D}|=\psi$.假设定义 t=100 和 ψ=256,即设置 iTrees 的数目为 100,每棵 iTree 的大小为 256,但是由于 256 大于样本总数目 8,所以随机采样样本数为$|\mathcal{D}|$=8 代替$|\mathcal{D}|$=256.然后,在每棵 iTree 的每个内部节点处采用轴平行分割算法将节点处的样本集分为两个非空子集,直到每个点被隔离或达到最大树高度 h.轴平行分割 iTree 构建过程如算法 1 所示.

算法 1. 轴平行分割算法.

输入:数据集 $\boldsymbol{X}$,最高限制 h.

输出:一棵 iTree.

步骤:

Step 1. 随机选择一个属性 q.

Step 2. 随机选择一个分裂的点 p,此点的属性 q 需要在属性 q 最小和最大值之间.

Step 3. 将在属性 q 取值小于 p 点的点归为一类,将在属性 q 取值大于等于 p 点的点归为一类.

Step 4. 如果树的高度大于最高限制 h,或者点都成了独立的节点,则停止,否则返回 Step1.

Step 5. 返回 iTree 树结构.

假设通过算法 1 分割第 1 棵 iTree,得到如图 3 所示的树结构.依次分割 100 棵 iTree 得到类似 100 个图 3 的树构成 iForest.X 中的所有实例遍历每棵树,并记录每个节点的块.

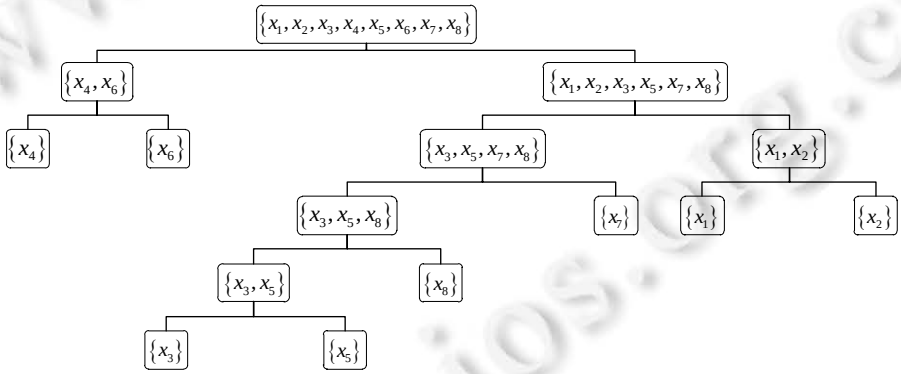


Fig.3 Structure of one iTree

图 3 iTree 的树结构

- 第 2 步,进行块的评估.

通过每个 iTree 解析测试点 x 和 y,计算包含 x 和 y 两者的最低节点的块总和,即 $\sum_i|R(x,y\mid H_i)|$.

- 最后,$m_e(x,y)$是这些块的均值.

$$m_e(x,y)=\frac{1}{t}\sum_{i=1}^{t}\frac{|R(x,y\mid H_i)|}{|D|} \tag{9}$$

假设求解点 x_3 和 x_7 的不相似性.在图 3 的 iTree 中,x_3 和 x_7 的块值为包含两者的最小的区域$|\{x_3,x_5,x_7,x_8\}|$=4.

以相同的方式构建并遍历所有的树,得到 x_3 和 x_7 的块值总和,求平均即得到 x_3 和 x_7 的不相似性.

2.2 DDPC算法介绍

本文 DDPC 算法将继续采用 DPC 的中心思想,快速寻找局部密度以及相对距离属性均较大的点作为聚类中心,但是相似度计算以及局部密度的度量方式我们将进行改进.首先,样本间相似性度量将使用基于块的不相似性度量代替简单的几何距离度量,根据公式(9)得出新的相似度矩阵;然后根据新的相似度矩阵,找到样本的 K 个最接近的匹配近邻,并定义新的局部密度.

$$\rho_i = \sum_{j\in KNN(i)} \exp(-m_e(x_i,x_j)) \tag{10}$$

其中,$KNN(i)$是点 i 的 k 个近邻点.同时,数据样本的相对距离属性也不再依赖几何距离度量的相似度,而是利用公式(9)计算得到的相似度.

$$\delta_i = \begin{cases} \min\limits_{j:\rho_j>\rho_i}(m_e(x_i,x_j)), \text{ if } \exists j \text{ s.t. } \rho_i > \rho_j \\ \max\limits_{j}(m_e(x_i,x_j)), \quad \text{otherwise} \end{cases} \tag{11}$$

DDPC 算法具体步骤如算法 2.

算法 2. DDPC 聚类算法.

输入:数据集 $\boldsymbol{X}=\{\boldsymbol{x}_1,\boldsymbol{x}_2,\ldots,\boldsymbol{x}_n\}$,iTree 的数目 t,每棵 iTree 的大小 ψ,近邻数 k.

输出:聚类结果 $\boldsymbol{Y}$.

步骤:

Step 1. 将数据集 $\boldsymbol{X}$ 通过随机采样分为 t 个大小为 ψ的集合,如果 $n<\psi$值,则采样的大小取 n.

Step 2. 对每个集合,根据算法 1 进行轴平行分割,构成 1 棵 iTree,t 棵 iTree 构成 1 个 iForest.

Step 3. 遍历 iForest,根据公式(9),基于块的不相似性度量方式计算样本相似度矩阵.

Step 4. 根据重新定义的局部密度的计算公式(10),计算每个样本的ρ_i 值.

Step 5. 根据公式(11)计算每个样本的δ_i 值.

Step 6. 根据由ρ_i 和δ_i 构成的决策图,自动选择聚类中心.

Step 7. 将数据集中的其余数据点归于密度等于或者高于“当前点”的最近点一类.

Step 8. 返回结果矩阵 $\boldsymbol{Y}$.

DDPC 算法保留了 DPC 算法寻找聚类中心的主要步骤,快速寻找密度峰值作为聚类中心.但是本文 DDPC 算法局部密度以及相对距离属性的度量依赖于更符合人类心理的基于块的不相似性度量方式计算得到的样本间相似度,代替了传统 DPC 算法中简单的距离度量得到的相似度,使 DDPC 在高维数据以及密度不均匀数据集上更高效.另外,利用基于块的不相似性度量计算的数据样本间的相似性,定义了基于样本 K 近邻的新的局部密度.与 DPC 算法相比,DDPC 算法避免了过度依赖截断距离 d_c,并且局部密度度量适合于任意大小的数据集.

3 实验与分析

3.1 实验设计

为了验证 DDPC 算法的聚类性能,实验采用人工数据集和真实数据集对本文算法进行测试.聚类精度(Acc)被用来测量聚类结果的质量,Acc 的值越高,聚类性能越好.Acc 的计算公式如下.

$$Acc = \sum_{i=1}^{N}\delta(y_i,map(z_i))\Big/n \tag{12}$$

对于数据集 $X=\{x_1,x_2,\ldots,x_n\}$,y_i,z_i 分别是固有类标签和聚类结果标签,$map(\cdot)$通过 Hungarian 算法将每个类标签映射到类别标签,并且该映射是最优的.除了 DPC 算法,我们将 DDPC 算法与 DPC 的优化算法 FKNN-DPC 以及 DPC-KNN 进行对比,对比算法的实验结果均采用 Acc 统一标准进行准确率度量.由于我们的算法没有考虑噪声点处理,所以为了公平起见,我们选择无噪声数据集,见表 1 和表 2,对比算法也均不处理噪声点.另外,DPC 算法的参数取值参考文献[10],为[0.2%,0.4%,0.6%,1%,2%,4%,6%];FKNN-DPC 算法以及 DPC-KNN 算法中的 k 值参考

文献[20,21],取 5 到 7;DDPC 中的 t 和 ψ都取文献[24]中指出的默认值 100 和 256,k 值同样取 5~7.本文算法和各对比算法均通过实验尝试获取最优参数以及最优值.由于 DDPC 算法中采用基于块的不相似性度量具有随机性,所以结果同时给出最优值并给出 20 次实验的平均值.

Table 1 Synthetic datasets

表 1 人工数据集

Datasets	Samples	Attributes	Categories
D	97	2	3
Flame	240	2	2
R15	600	2	15
Forty	1 000	2	40
S2	5 000	2	15

Table 2 Real-world datasets

表 2 真实数据集

Datasets	Samples	Attributes	Categories
Iris	150	4	3
Seeds	210	7	3
Wine	178	13	3
WDBC	569	30	2
Ionosphere	351	34	2
Soybean	47	35	4

3.2 实验结果分析

3.2.1 人工数据集实验结果分析

本节对 5 组人工数据集进行 DDPC 测试,实验数据特征见表 1.其中,人工数据集 D 是典型的包含密度不均匀的 3 个类的数据集,其余 4 个数据集包含了规模较小的 Flame 和规模较大的 S2;同时包含了分布均匀的 Forty 数据集,又包含了分布紧密、有交叉分布的 S2 数据集.

实验对二维数据集的结果采用可视化展示,一个颜色代表一个类.分别将本文 DDPC 算法和 DPC 算法以及 FKNN-DPC 算法、DPC-KNN 算法在以上 5 个数据集上进行了聚类,结果如图 4~图 8 所示.

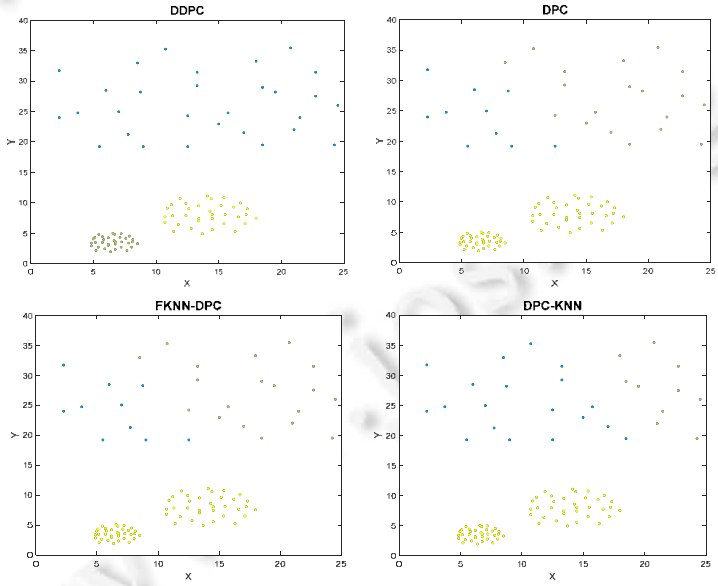


Fig.4 Clustering results of different algorithms on D dataset

图 4 不同算法在 D 数据集上的聚类结果

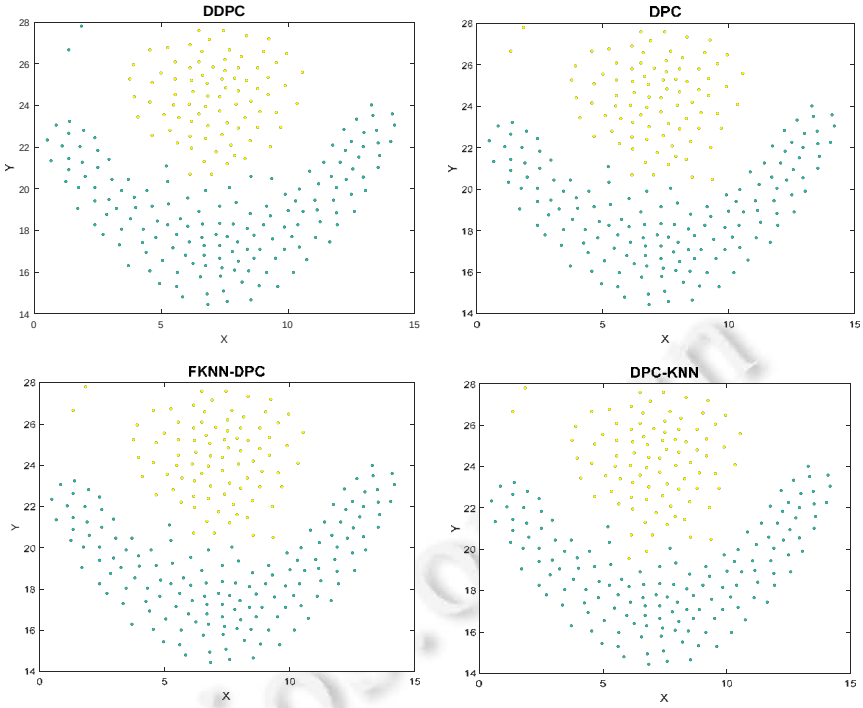


Fig.5 Clustering results of different algorithms on Flame dataset

图 5 不同算法在 Flame 数据集上的聚类结果

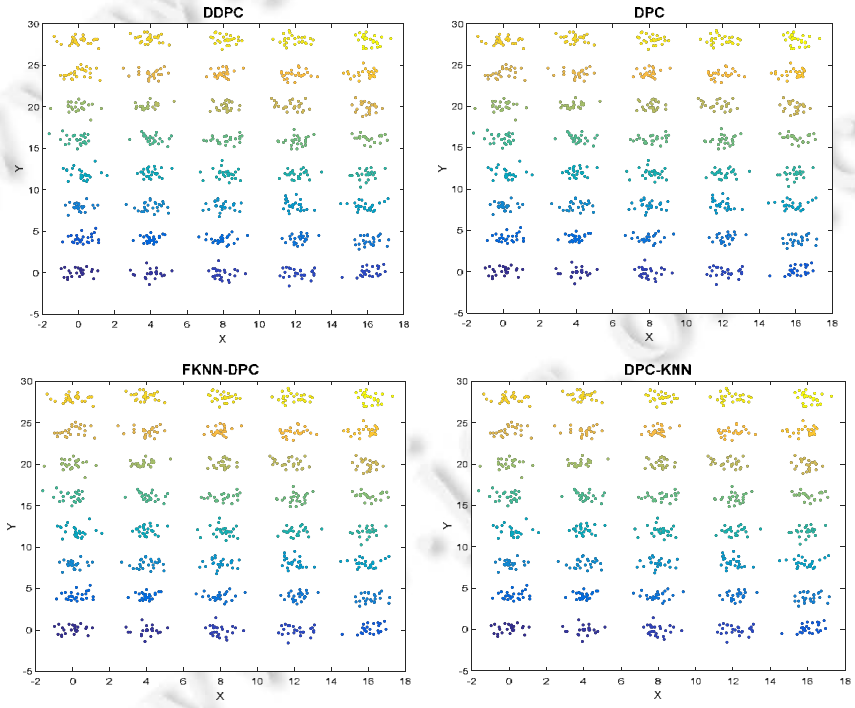


Fig.6 Clustering results of different algorithms on Forty dataset

图 6 不同算法在 Forty 数据集上的聚类结果

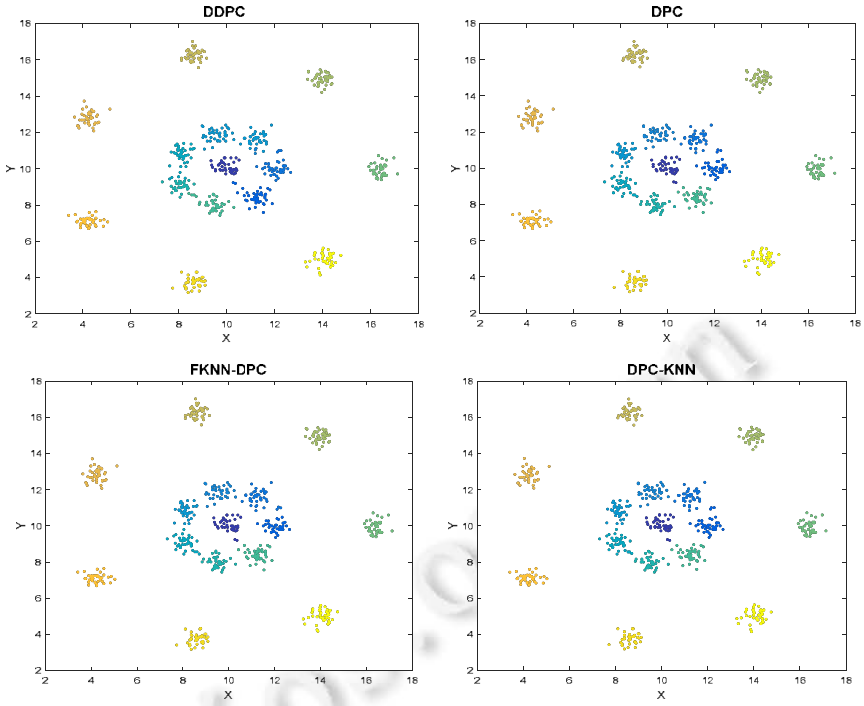


Fig.7 Clustering results of different algorithms on R15 dataset

图 7 不同算法在 R15 数据集上的聚类结果

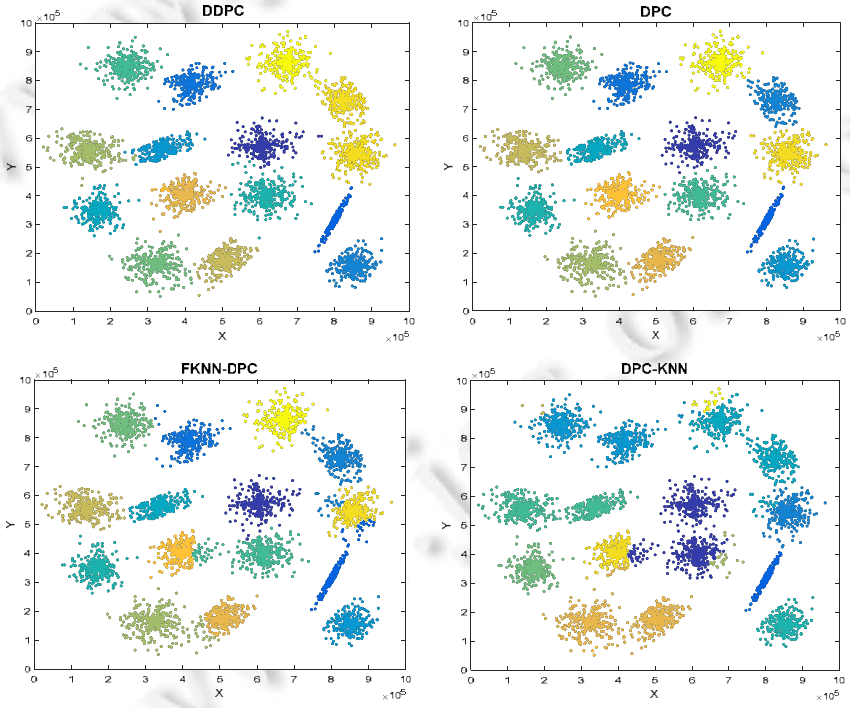


Fig.8 Clustering results of different algorithms on S2 dataset

图 8 不同算法在 S2 数据集上的聚类结果

从图 4 中可以看出,DDPC 可以很好地处理密度不均匀的数据集.在数据集 D 上,DDPC 算法可以很好地聚类成 3 类.但是 DPC 不能很好地将数据集分成 3 类,因为 DPC 只是简单地用数据间的几何距离来度量相似度,计算局部密度和相对距离属性,所以对于数据集分布不均匀的数据集,DPC 不能很好地识别所有的类.而 FKNN-DPC 算法虽然考虑了样本的 K 近邻度量局部密度,但是 K 近邻的判断依旧是根据简单的几何距离度量的相似性判断的近邻点,所以针对密度不均匀的分布数据集,FKNN-DPC 算法的聚类结果也不是很理想.同样,DPC-KNN 算法也引入的 K 近邻信息是依据样本间的相似性判断的,但是相似性的判断只是依赖数据的几何距离,所以聚类效果不尽如人意.

从图 5~图 7 中可以看出:虽然在分布相对均匀的数据集上,DDPC 以及 DPC 和 FKNN-DPC,DPC-KNN 算法均可以通过选择合适的参数得到较为满意的聚类结果,但是 DPC 算法中局部距离的度量方式不统一,并且需要选择合适的 d_c.而 DDPC 算法以及 FKNN-DPC 和 DPC-KNN 统一了局部密度的度量方式,并且不再需要选择参数 d_c.虽然依旧有参数 k,但是 k 值的变化对聚类中心的选择影响不大,即对聚类结果的影响不是很大.

从图 8 中可以看出:DDPC 和 DPC 算法可以得到令人满意的聚类结果,但是 FKNN-DPC 和 DPC-KNN 算法的聚类结果不是很理想.DPC 算法考虑聚类中心的特点,从全局角度出发选择聚类中心,所以可以准确处理类之间有交叉的数据集.虽然 FKNN-DPC 和 DPC-KNN 引入数据样本的 K 近邻统一局部密度的度量方式,不再需要参数 d_c,但是 FKNN-DPC 和 DPC-KNN 只是简单进行几何距离的度量选择 K 近邻,忽略了数据分布的周围的环境,所以当数据集类之间有交叉,分布紧密的时候结果不理想.而本文 DDPC 算法虽然也是考虑样本的 K 近邻度量局部密度,但是由于 K 近邻的度量不再简单根据几何距离度量,而是使用了基于块的不相似性度量,考虑了数据分布的周围环境,所以 DDPC 可以和 DPC 一样,针对 FKNN-DPC 和 DPC-KNN 聚类结果不理想的类有交叉的数据集上聚类的结果还是相对较满意.

各算法虽然在大部分分布规则、密度均匀的数据集上聚类效果都差强人意,但是 DPC 算法局部密度和相对距离简单依赖几何距离度量的相似度,使得 DPC 算法在密度不均匀的数据集上聚类效果不尽如人意,同时在高维数据集上的聚类结果也不是很理想,将在下节实验证明.另外,局部密度的度量方式不统一,d_c 的选择对聚类结果的影响,使得 DPC 算法在实际操作中需要一定的先验知识.而 FKNN-DPC 和 DPC-KNN 虽然统一了局部密度的度量方式,同时去除了 d_c 改变对聚类结果的影响,但是由于引入的 K 近邻信息也只是简单依靠几何距离度量寻得,而忽略了数据分布的周围环境,所以当数据分布复杂,有交叉、密度不均匀时,FKNN-DPC 和 DPC-KNN 的聚类结果不是很理想.而本文提出的 DDPC 聚类算法在引入 K 近邻的基础上又考虑数据分布的周围环境度量样本间的相似性,所以本文 DDPC 算法在处理密度不均匀以及类间有交叉点时效果较理想,同时统一了局部密度的度量方式,克服了 d_c 改变对聚类结果的影响,并且 DDPC 本身只有一个参数 k 需要选择,而由于 DDPC 依旧采用 DPC 中选择聚类中心的特点选取密度峰值,所以聚类中心一定处于密度较大的区域,所以 k 的微小改变对聚类的结果影响不大.

3.2.2 真实数据集实验结果分析

本节对 6 组人工数据集进行 DDPC 测试,实验数据特征见表 2.由于 DPC 算法中 d_c 的改变对小样本数据集上聚类的结果影响较大,同时,在数据维度较高的数据集上 DPC 的聚类结果不尽如人意,所以本节实验挑选了经典的小样本数据集,并且包含较高的维度.

实验分别将本文 DDPC 算法和 DPC 算法以及 FKNN-DPC 算法和 DPC-KNN 算法在以上 6 个数据集上进行了聚类,聚类结果见表 3,并且给出了对应的最佳参数.粗体即为各算法中最优结果,而 DDPC 同时在括号中给出了 20 次测试均值.

从表 3 可以看出,本文 DDPC 算法整体的聚类效果较 DPC 以及 FKNN-DPC 和 DPC-KNN 更好.而 DPC 在维度较高的数据集上聚类结果不是很满意,而且 d_c 需要合适的选择.虽然 FKNN-DPC 和 DPC-KNN 算法避免了传统 DPC 算法中参数 d_c 的选择,结果较 DPC 相比差强人意,但是与 DDPC 算法相比,聚类结果不是很理想.本文 DDPC 算法由于考虑数据分布的周围环境,用基于块的不相似性度量代替了简单的几何距离度量相似度的方式,所以本文 DDPC 算法在针对数据维度较高的数据集效果较好.另外,由于 DDPC 也选择考虑 K 近邻来度量局

部密度,所以避免了 DPC 算法中参数 d_c 的选择.而本文参数 k 的选择因为聚类中心处于密度紧密的区域的特点,所以在本文实验中 k 的选择对聚类结果的影响不是很大.

Table 3 Clustering accuracy of different algorithms on different datasets

表 3 各算法在不同数据集上的聚类准确率

Datasets	DDPC	DPC	FKNN-DPC	DPC-KNN
Iris	**96**(90.491) (k=7)	94 (d_c=2%)	90.667 (k=7)	90.667 (k=7)
Seeds	**92.857**(89.85) (k=7)	89.524 (d_c=1%)	89.524 (k=7)	89.524 (k=7)
Wine	**96.067**(94.086) (k=7)	69.101 (d_c=0.2%)	69.101 (k=7)	53.933 (k=7)
WDBC	**94.903**(92.249) (k=6)	62.917 (d_c=2%)	62.917 (k=7)	79.438 (k=7)
Ionosphere	**78.063**(73.564) (k=6)	73.504 (d_c=0.5%)	68.091 (k=7)	68.091 (k=7)
Soybean	**100**(100) (k=6)	89.362 (d_c=2%)	89.362 (k=7)	91.49 (k=7)

我们选取参数 k=5,k=6,k=7 分别进行了测试,取最优解并绘制了对比图,如图 9 所示.从图中可以看出,本文算法在 k 变化的时候,聚类结果波动不大,即说明 DDPC 算法中参数具有鲁棒性.

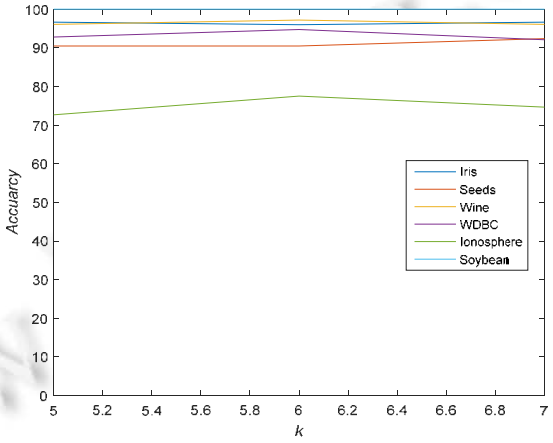


Fig.9 Accuracy on different datasets with different k values

图 9 不同 k 值的在不同数据集上的准确率

4 结束语

本文提出一种基于不相似度量优化的密度峰值聚类算法,引入基于块的不相似性度量计算样本间的相似度,并将此度量得到的样本间相似度引入样本的 K 近邻度量,结合样本的 K 近邻信息定义新的局部密度计算方式,统一局部密度的度量方式,避免了小样本数据集上参数 d_c 选择问题,并提高 DPC 算法在复杂数据集,尤其是维度较高以及密度不均匀数据集上的缺陷.同时,本文算法虽然增加了参数 k,但是由于 DDPC 算法是对 DPC 算法的优化,保留了传统 DPC 算法选取聚类中心的方法,所以参数 k 的选择具有鲁棒性.本文从理论以及实验证明分析了优化后的密度峰值聚类算法 DDPC 优于传统的 DPC 算法以及优化的 FKNN-DPC 和 DPC-KNN 算法.

本文 DDPC 算法如何合理分配剩下的点而不是采用一步式分配策略,并有效处理噪声点,需要进一步探索.

References:

[1] Zhang W, Du L, Li L, Zhang X, Liu H. Infinite Bayesian one-class support vector machine based on Dirichlet process mixture clustering. Pattern Recognition, 2018,78:56−78.

[2] Shi Y, Otto C, Jain A. Face clustering: Representation and pairwise constraints. IEEE Trans. on Information Forensics and Security, 2018,13(7):1626−1640.

[3] Ivannikova E, Park H, Hämäläinen T, Lee K. Revealing community structures by ensemble clustering using group diffusion. Information Fusion, 2018,42:24−36.

[4] Wang L, Shao Y. Crack fault classification for planetary gearbox based on feature selection technique and *K*-means clustering method. Chinese Journal of Mechanical Engineering, 2018,31:Article No.4. [doi: 10.1186/s10033-018-0202-0]

[5] Slimen Y, Allio S, Jacques J. Model-based co-clustering for functional data. Neurocomputing, 2018,291:97−108.

[6] Bai X. Similarity Measures in Cluster Analysis and Its Applications. Beijing: Beijing Jiaotong University, 2012 (in Chinese).

[7] Shi QY, Liang JY, Zhao XW. A clustering ensemble algorithm for incomplete mixed data. Journal of Computer Research and Development, 2016,53(9):1979−1989 (in Chinese with English abstract).

[8] Zhao W, Deng C, Ngo C. *k*-means: A revisit. Neurocomputing, 2018,291:195−206.

[9] Ienco D, Bordogna G. Fuzzy extensions of the DBScan clustering algorithm. Soft Computing, 2018,22(5):1719−1730.

[10] Rodríguez A, Laio A. Clustering by fast search and find of density peaks. Science, 2014,344(6191):1492−1496.

[11] 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).

[12] Liu R, Wang H, Yu X. Shared-nearest-neighbor-based clustering by fast search and find of density peaks. Information Sciences, 2018,450:200−226.

[13] Mehmood R, Zhang G, Bie R, Dawood H. Clustering by fast search and find of density peaks via heat diffusion. Neurocomputing, 2016,208:210−217.

[14] Zhou L, Pei C. Delta-distance based clustering with a divide-and-conquer strategy: 3DC clustering. Pattern Recognition Letters, 2016,73:52−59.

[15] Ding S, Du M, Sun T, 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.

[16] Bai L, Cheng X, Liang J, Shen H, Guo Y. Fast density clustering strategies based on the *k*-means algorithm. Pattern Recognition, 2017,71:375−386.

[17] Shi Y, Chen Z, Qi Z, Meng F, Cui L. A novel clustering-based image segmentation via density peaks algorithm with mid-level feature. Neural Computing and Applications, 2017,28(1):29−39.

[18] Zhang Y, Xia Y, Liu Y, Wang W. Clustering sentences with density peaks for multi-document summarization. In: Proc. of the NAACL HLT 2015. Denver: ACL, 2015. 1262−1267.

[19] Chen Y, Lai D, Qi H, Wang J, Du J. A new method to estimate ages of facial image for large database. Multimedia Tools and Applications, 2016,75(5):2877−2895.

[20] Du M, Ding S, Jia H. Study on density peaks clustering based on *k*-nearest neighbors and principal component analysis. Knowledge-based Systems, 2016,99:135−145.

[21] Xie J, Gao H, Xie W, Liu X, Grant P. Robust clustering by detecting density peaks and assigning points based on fuzzy weighted *K*-nearest neighbors. Information Sciences, 2016,354:19−40.

[22] 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).

[23] Krumhansl C. Concerning the applicability of geometric models to similarity data: The interrelationship between similarity and spatial density. Psychological Review, 1978,85(5):445−463.

[24] Kai M, Zhu Y, Carman M, Zhu Y, Zhou Z. Overcoming key weaknesses of distance-based neighbourhood methods using a data dependent dissimilarity measure. In: Proc. of the KDD 2016. San Francisco: ACM, 2016. 1205−1214.

[25] Aryal S, Kai MT, Haffari G, Washio T. m_p-dissimilarity: A data dependent dissimilarity measure. In: Proc. of the ICDM 2014. Shenzhen: IEEE, 2014. 707−712.

[26] Chen B, Ting K, Washio T, Haffari G. Half-Space mass: A maximally robust and efficient data depth method. Machine Learning, 2015,100(2-3):677−699.

附中文参考文献:

[6] 白雪.聚类分析中的相似性度量及其应用研究.北京:北京交通大学,2012.

[7] 史倩玉,梁吉业,赵兴旺.一种不完备混合数据集成聚类算法.计算机研究与发展,2016,53(9):1979−1989.

[11] 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.

[22] 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.

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

徐晓(1992－),女,博士生,主要研究领域为机器学习,聚类分析.

王艳茹(1991－),女,博士生,CCF 学生会员,主要研究领域为机器学习,聚类分析.