Computer Science and Application 计算机科学与应用, 2025, 15(2), 118-131
Published Online February 2025 in Hans. https://www.hanspub.org/journal/csa
https://doi.org/10.12677/csa.2025.152039

Hans汉斯

基于离群点检测的优化初始中心的三支K-Means算法

樊有明，李志聪

哈尔滨师范大学计算机科学与信息工程学院，黑龙江 哈尔滨

收稿日期：2025年1月12日；录用日期：2025年2月14日；发布日期：2025年2月25日

摘　要

针对传统的k-means算法的聚类数目k无法确定、初始聚类中心随机给定、容易受到离群点影响等问题，该算法使用LOF (Local Outlier Factor)离群点检测算法计算数据集中每个数据对象的离群因子，并去除离群因子大于指定阈值的数据对象，使用手肘法来确定符合数据集的最佳k值，根据最大密度和最大距离的思想结合每个点的离群因子来选取初始聚类中心并进行后续聚类中心的迭代，聚类完成后结合三支决策的思想对聚类结果的每个簇内的数据对象进行进一步优化。实验结果表明ODT-kmeans算法能合理选取k值、减少离群点的影响并且可以消除随机选择初始聚类中心的问题，提高了k-means聚类算法的准确率。

关键词

K-Means算法，三支聚类，LOF离群点检测算法，聚类中心

Three-Branch K-Means Algorithm with Optimized Initial Center Based on Outlier Detection

Youming Fan, Zhicong Li

School of Computer Science and Information Engineering, Harbin Normal University, Harbin Heilongjiang

Received: Jan. 12th, 2025; accepted: Feb. 14th, 2025; published: Feb. 25th, 2025

Abstract

In view of the problems of the traditional k-means algorithm, such as the number of clusters k

文章引用: 樊有明, 李志聪. 基于离群点检测的优化初始中心的三支 K-Means 算法[J]. 计算机科学与应用, 2025, 15(2): 118-131. DOI: 10.12677/csa.2025.152039

cannot be determined, the initial cluster center is randomly given, and it is easily affected by outliers, this algorithm uses the LOF (Local Outlier Factor) outlier detection algorithm to calculate the outlier factor of each data object in the data set and remove the data objects whose outlier factor is greater than the specified threshold. The elbow method is used to determine the best k value that meets the data set. The initial cluster center is selected based on the idea of maximum density and maximum distance combined with the outlier factor of each point and the subsequent cluster center iterations are performed. After clustering is completed, the idea of three-way decision is combined to further optimize the data objects in each cluster of the clustering results. Experimental results show that the ODT-kmeans algorithm can reasonably select the k value, reduce the influence of outliers, and eliminate the problem of randomly selecting the initial cluster center, thereby improving the accuracy of the k-means clustering algorithm.

Keywords

K-Means Algorithm, Three-Branch Clustering, LOF Outlier Detection Algorithm, Cluster Center

1. 引言

随着大数据时代的到来，数据分析和处理技术快速发展，如何从海量的数据中提取有价值的信息成为了研究者们关注的焦点。聚类分析作为数据挖掘领域的重要分支，已被广泛应用于统计学[1]、图像处理[2]、生物信息学、市场分析、信息检索、模式识别等多个领域。k-means 聚类算法以其简洁高效、易于实现的特点，成为了最为经典的聚类算法之一。

对于 k-means 聚类算法面临的一些挑战，如初始聚类中心选择的随机性、k 值预先无法确定、聚类结果不稳定以及易受离群点的影响，许多研究者对传统 k-means 聚类算法进行了改进和优化，提高其在实际应用中的性能。岳珊等[3]对高维样本数据在最大限度保证原始样本数据特征的前提下进行降维处理，再使用初始簇心求解模型选取初始中心。Hu H [4]提出了一种基于 Lévy 飞行轨迹的均值聚类算法 Lk-means，在算法的迭代过程中，利用 Lévy 飞行搜索新位置，避免聚类时过早收敛，增加簇的多样性，增强均值算法的全局搜索能力，避免过早陷入局部最优值，但在 Lévy 飞行优化过程中并没有增加混合算法的复杂度。Shrifan [5]等人提出了一种基于 Tukey 规则结合新的距离度量的 k-means 算法，在计算质心之前应用了异常值的消除以最小化异常值的影响，同时提出了一种新的距离度量来将每个数据点分配给最近的聚类。孙林等[6]提出一种引入平均样本距离和误差平方和构造初始聚类中心的选取方法，然后基于最近簇中心进行簇的合并，基于中位数构造轮廓系数，设计基于中位数的平均轮廓系数评价指标，判断簇合并之后的最佳 k；最后设计了基于优化初始聚类中心和轮廓系数的 k-means 聚类算法。郭文娟等[7]提出了一种基于优化初始聚类中心的 k-means 算法，该算法通过量化样本间距离和聚类的紧密性来确定聚类数目 k 值；根据数据集的分布特征来选取相距较远的数据作为初始聚类中心，避免了传统 k-means 算法的聚类数目和聚类中心的随机选取。张亚迪等[8]提出了一种基于密度参数和中心替换的 DC-k-means 算法，采用数据对象的密度参数来逐步确定初始类簇中心，使用中心替换方法更新偏离实际位置的初始中心。刘美玲等[9]提出一种基于离散量改进 k-means 初始聚类中心选择的算法，首先将所有对象作为一个大类，然后不断从对象数目最多的聚类中选择离散量最

大与最小的两个对象作为初始聚类中心，再根据最近距离将这个大聚类中的其他对象划分到与之最近的初始聚类中，直到聚类个数等于指定的 k 值。唐东凯等[10]提出了一种使用离群因子和最大最小算法的思想来优化初始中心的选取以及排除离群点的影响的方法。廖纪勇等[11]提出一种通过计算相异性来构造相异性矩阵，然后通过定义均值相异性和总体相异性的度量准则来确定初始聚类中心，并利用各簇中数据点的中位数代替均值以进行后续聚类中心的迭代，消除离群点对聚类准确率的影响的改进 k-means 算法。

本文提出一种基于离群点检测的优化初始聚类中心的 k-means 算法，简称 ODT-kmeans 算法。该算法首先通过 LOF 离群点检测算法计算出每个数据对象的离群因子，然后给出一个阈值，如果一个数据对象的离群因子大于该阈值则将其剔除，目的是把最有可能为离群点的数据对象排除掉，减少对初始聚类中心选择的影响。使用手肘法得到最佳 k 值后，选取处于最大密度区域且离群因子最小的数据对象并结合最大距离的思想来选取初始聚类中心，利用 k-means 算法迭代后续聚类中心，得到聚类结果后再使用三支决策的思想进一步划分每个类中的数据对象。该算法消除了 k-means 算法对初始聚类中心的依赖并且每次聚类结果完全一致，保证了聚类结果的稳定性。

2. 使用须知相关工作

2.1. K-Means 算法

K-means 算法是一种常见的基于划分的聚类算法，它的基本目标是对于给定的数据集，按照数据对象之间的相似性将其划分为 k 个不同的簇，使得同一个簇内的数据对象之间的相似度较高，不同的簇内的数据对象之间具有较大的差异。设一个样本集合 D，其中包含 n 个数据对象，指定所需要的聚类中心个数 k，首先从给定的数据集中随机选择 k 个数据对象作为初始的聚类中心；然后分别计算其余每个数据对象到各个聚类中心的距离，将数据对象分配到与其距离最近的聚类中心所对应的簇中；将数据对象划分完毕后，对于每个簇，计算其内部所有数据对象的平均值并将其作为新的聚类中心以便于后续的迭代过程；不断重复前两步过程直至聚类中心不再显著变化或者达到指定的最大迭代次数。本文计算距离时采用的是欧式距离。样本集合 $D=\{x_1,x_2,\cdots,x_n\}$，聚类中心集合 $C=\{C_1,C_2,\cdots,C_k\}$，n 为数据对象的个数，k 为聚类中个数，每个数据对象的维数为 m。

定义 1 欧式距离。数据对象 x_i 和数据对象 x_j 之间的欧式距离。

$$d\left(x_i,x_j\right)=\sqrt{\sum_{t=1}^{m}\left(x_{it}-x_{jt}\right)^2} \tag{1}$$

其中 x_i 和 x_j 为样本集合 D 中任意两个数据对象，x_{it} 和 x_{jt} 为数据对象第 t 维的值，$i=1,2,\cdots,n;\ j=1,2,\cdots,n;\ l=1,2,3,\cdots,m$。

定义 2 数据对象之间的平均距离。

$$d=\frac{1}{n(n-1)}\sum_{i=1}^{n}\sum_{j=i+1}^{n}d\left(x_i,x_j\right) \tag{2}$$

其中 d 表示指定个数的数据对象之间的平均距离。

2.2. LOF 算法

对于数据集中的离群点问题，Hodge V [12]等人将离群点检测方法分成了统计方法、距离方法、密度方法、模型方法、集成方法和机器学习方法等类，并介绍了这些方法的基本原理、优缺点以及适用场景。DU Xusheng [13]提出了基于图上随机游走的离群点检测算法，该算法在利用图结构信息和捕

捉局部异常性方面表现出显著优势，并且能够处理非欧几里得数据，具有较好的鲁棒性和自动化特性。然而，该算法也存在计算复杂度高、参数选择依赖、图构建困难、对图结构敏感以及解释性问题等不足之处。

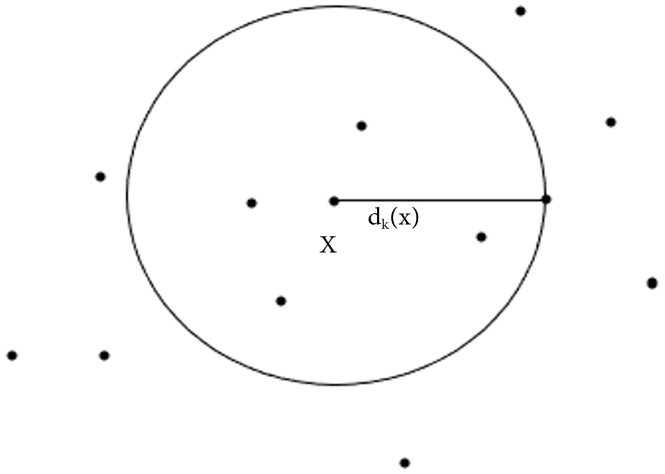


Figure 1. 5th distance of data object *x*
图 1. 数据对象 *x* 的第 5 距离

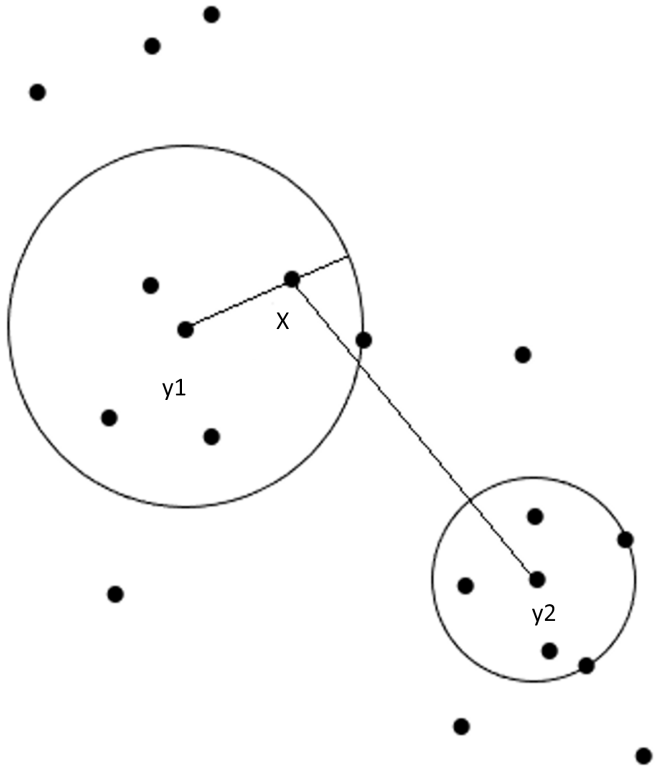


Figure 2. The fifth reachable distance of data objects *y*1, *y*2 to *x*
图 2. 数据对象 *y*1、*y*2 到 *x* 的第五可达距离

LOF 离群点检测算法是一种基于密度的离群点检测方法，由 Breunig [14]等人在 2000 年提出，旨在从数据集中识别出那些与其他数据点明显不同的离群点。这种方法通过比较一个数据对象与它的邻近数

据对象的局部密度差异来识别离群点。在此算法中，离群点被定义为局部密度显著低于其邻近数据对象的局部密度的数据对象。

下面是 LOF 算法中涉及的一些概念。

定义 4 数据对象 x 的第 k 距离，记作 $d_k(x)$。对于一个正整数 k，在样本集合 D 中数据对象 x_i 的第 k 距离是指数据对象 x_i 与另一个数据对象 x_j 之间的距离，这个距离满足两个条件：

1) 至少存在 k 个数据对象 a，使得它们与 x_i 的距离 $d(xi,a)$小于或等于 $d(x_i,x_j)$。

2) 至多存在 k-1 个数据对象 a，使得它们与 x_i 的距离 $d(x_i,a)$小于 $d(x_i,x_j)$。例如，数据对象 x 的第 5 距离的几何意义如图 1 所示。

定义 5 数据对象 x_i 的第 k 距离邻域 $N_k(x_i)$。数据对象 x_i 的第 k 距离邻域包括数据对象 x_i 的第 k 距离内的所有点的集合。

定义 6 数据对象 x_i 关于 x_j 的可达距离，记作 reach-dist_k(x_i,x_j)。如果 x_j 是 $N_k(x_i)$内的数据对象，那么 x_i 关于 x_j 的可达距离是 $d_k(x_i)$和 $d(x_i,x_j)$的最大值。可以表示为 reach-dist_k(x_i,x_j) = max{$d_k(x_j)$,$d(x_i,x_j)$}。

例如，$y1$、$y2$ 到数据对象 x 的第 5 可达距离如图 2 所示，在此图中 reach-dist_k(x,$y1$)=$d5(y1)$，reach-dist_k(x,$y2$) = $d(x,y2)$。

定义 7 数据对象 x_i 的局部可达密度。数据对象 x_i 的局部可达密度是 $N_k(x_i)$内$|N_k(x_i)|$个数据对象的可达距离的平均值的倒数，计算方法如下：

$$Irdk\left(xi\right)=\frac{1}{\frac{\sum_{x_j\in N_k(x_i)}reach-dist_k\left(x_i,x_j\right)}{\left|N_k\left(x_i\right)\right|}} \tag{3}$$

其中，$|Nk(x_i)|$指的是 x_i 的 k 个邻近数据对象。

定义 8 离群因子。数据对象 x_i 的离群因子是 $N_k(x_i)$内的数据对象 x_j 的局部可达密度的平均值与 x_i 的 $lrd_k(xi)$的比值，计算方法如下：

$$LOF_k\left(x_i\right)=\frac{\sum_{x_j\in N_k(x_i)}\frac{lrd_k\left(x_j\right)}{\left|N_k\left(x_i\right)\right|}}{lrd_k\left(x_i\right)} \tag{4}$$

通过以上的定义，可以逐步得出数据集中每个数据对象的离群因子，即 LOF 值，从而得知数据集的离群点的分布情况，在后续选取初始聚类中心时可以减少离群点的影响。这里的比值越大于 1，说明该数据对象越有可能是离群点，反之，越可能是正常点。

2.3. 三支聚类

K-means 聚类算法属于硬聚类算法，即每个数据对象只属于一个类别且类的交集为空集。换句话说，一个数据对象不能同时属于多个类别，每个数据对象都有一个明确的归属。如果一个数据对象在进行分类时具有不确定性，k-means 会将其强制性地分配到一个类中，此时便会导致错误率的提高。为了解决上述 k-means 存在的问题，Yao [15]-[17]等提出了三支决策的思想。Yu [18]等人在聚类中结合了三支决策理论，将一个整体分为核心域(L 域)、边界域(M 域)以及琐碎域(R 域)，将确定属于该类簇的数据对象分到核心域，不确定是否属于该类簇的数据对象分到边界域，确定不属于该类簇的数据对象划分到琐碎域，会在一定程度上提高聚类过程的准确率。唐欣[19]从样本间关系的角度出发，利用邻域密度和最远欧氏距离的方法来选取 k 值，得到二支聚类结果，然后通过 Q 近邻计算二支聚类中数据对象到各个聚类中心的距离，将二支聚类中的数据对象进一步划分核心域和边界域，得到三支 k-means 聚类结果。李浩博等[20]

提出了一种结合孤立森林和鲸鱼优化算法的三支 k-means 算法(iF-W-TWKM)。利用孤立森林算法将数据集划分为正常数据子集和异常数据子集，使用正常数据子集进行后续算法步骤，待算法结束后使用得到的聚类中心将异常数据子集中的样本划分到各类簇的边界域。利用鲸鱼优化算法建立以 STDI 为目标函数的优化问题进行全局寻优实现聚类中心的选取。Wang [21]等人对三支聚类的发展进行了概述，并指出了三支聚类未来的挑战和研究主题。

三支聚类结果的每一个簇，对于每一个簇不使用传统的单集合表示方法，而是把它用两个集合来表示。聚类结果可以表示为 $C_i\left\{\left(C_1^L,C_1^M\right),\left(C_2^L,C_2^M\right),\cdots\left(C_k^L,C_K^M\right)\right\}$。其中，如果一个数据对象在一个簇 $C_i\left(i=1,2,\cdots,k\right)$ 的 L 域中，那么该数据对象属于该簇的典型对象，若在 M 域中，则该数据对象与该簇联系紧密，属于非典型对象。对于用该方法表示的类簇，具有以下特点：1) 每个类簇的 L 域一定不为空。2) 每个数据对象都至少会被分到一个簇中。3) 两个不同类簇的 L 域不会有任何交集。

3. ODT-kmeans 算法

3.1. 基本定义

定义 9 所有数据对象之间的最大距离(MaxDistance)。

$$\text{MaxDistance}=\max d\left(x_i,x_j\right)\quad 1\le i<j\le n \tag{5}$$

定义 10 所有数据对象之间的最小距离(MinDistance)。

$$\text{MinDistance}=\min d\left(x_i,x_j\right)\quad 1\le i<j\le n \tag{6}$$

定义 11 最大距离和最小距离的平均值。

$$\text{AveDistance}=\frac{\text{MaxDistance}+\text{MinDistance}}{2} \tag{7}$$

定义 12 数据对象间的均方差 s。

$$s=\frac{\sqrt{\sum_{i=1}^{k}\sum_{x\in C_i}\left\|x-u_i\right\|^2}}{n} \tag{8}$$

其中 x 代表每个簇中的数据对象，u_i 为簇 C_i 的聚类中心，n 为数据集 D 中的数据对象个数。

3.2. 数据集预处理

通过 1.2 小节的定义可以计算出给定数据集中所有数据对象的 LOF 值，它反映了数据对象是否是离群点的可能性，一个数据对象 LOF 的值越大说明该点越可能是离群点。

对数据集进行以下操作：

1) 计算出每个数据对象的离群因子 LOF。

2) 根据数据集的特点给定一个相对较高的阈值 t，若一个数据对象的 LOF 值大于该阈值，则将其视为该数据集的离群点，然后将其从数据集中剔除。若数据对象的 LOF 值小于等于该阈值，则视为正常点，不做任何处理。

3) 将剔除离群点之后的数据集作为初始聚类中心候选集。

上述过程引入了一个阈值 t，如果 t 的取值过小，则有较多的数据对象被视为离群点，导致聚类效果较差；如果 t 的取值过大，则可能会有较少的离群点甚至没有剔除离群点，因此阈值 t 的给定需要比较剔除离群点之后的数据集和原始数据集的大小，多次调整阈值，使处理之后的数据集在具有较好聚类效果

的同时没有剔除过多的数据对象。本文阈值 t 的取值对于不同的数据集会有不同的取值，因此需要根据数据集的特点而定。

3.3. 选取最佳的 k 值

在 k-means 算法中，数据集的聚类中心数 k 是人为给定的，增加了 k 值选择的随机性，当数据集较大时或者数据对象特征不明显时容易使算法陷入局部最优。如果 k 值选取过大，会导致每个簇之间的数据对象区别不明显。如果 k 值选取过小会导致每个簇内的数据对象的相似度过低，不利于获得最佳聚类结果。因此本文采用手肘法来选取 k 值，它通过分析不同 k 值对应的误差平方和 SSE (即误差平方和)的变化趋势来找出最佳的聚类数量。SSE 表示数据对象与其所属聚类中心之间的距离平方和，SSE 越小，聚类的效果越好。当 k 值增加时，每个簇中的数据对象会更加接近其聚类中心，SSE 通常会减小，但是随着 k 值的进一步增加，SSE 的减小幅度会逐渐放缓，因为增加更多的簇可能会导致一些数据对象成为离群点，从而使得 SSE 的值不再显著减小。手肘法的目标就是找到一个 k 值，使得当 k 继续增加时，SSE 的减小幅度不再显著，这个 k 值通常就是最佳的聚类簇数。

定义 13 误差平方和 SSE。

$$SSE = \sum_{i=1}^{k} \sum_{x \in C_i} \left\| x - u_i \right\|^2 \tag{9}$$

其中，x 为每个簇内的数据对象，u_i 为每个簇的聚类中心，k 为簇数。

3.4. 初始聚类中心的选取

在选取初始聚类中心时，采取以下步骤：

1) 根据定义 11 计算出最大距离和最小距离的平均值 AveDistance。

2) 根据定义 12 计算样本间的均方差 s。

3) 对于数据集 D，分别画出以每个数据对象为球心，以距离 AveDistance 作为半径的球体，计算每个球体范围内所包含的数据对象个数 $r1, r2, \cdots$。

4)计算 3) 中的所有数据对象个数的平均值 avgnum，即 $r1, r2, \cdots$ 的平均值。

5) 创建一个集合 Y，依次判断 3) 中每个球体中包含的数据对象的个数是否大于等于 avgnum，若大于等于 avgnum，则把作为球心的数据对象加入 Y；否则不加入。

6) 在 Y 中找出以其为球心的球包含数据对象最多的数据对象作为第一个初始聚类中心 C1。

7) 从 Y 中去除点 C1，并在 Y 的剩余点中找出点 C2，满足该点到 C1 的距离大于均方差且该点的 LOF 值最小，并将其作为第二个初始聚类中心，并从集合 Y 中删除。

8) 按照 7)依次求取，直到找出第 k 个初始聚类中心。

3.5. 算法流程

在 2.4 小节得到初始聚类中心的基础上，进行 k-means 聚类之后，将三支决策的思想运用到其中。本文将 k-means 的聚类结果的每个簇中的数据分为 L 域(核心域)和 M 域(边界域)，使数据对象实现最优的划分。通过设置一个阈值 f，然后通过计算阈值 f 与每个簇中的距离簇中心最远的数据对象距离的乘积，把乘积内的数据对象视为核心域，乘积之外的数据对象视为边界域。对于每一类的边界域，任取一点 z，寻找距离 z 点最近的 5 个点，如果这五个点中有三个及以上的点与 z 点不属于同一类，则把 z 点归入公共边界域，否则 z 点只属于该类的边界域。重复上述操作，直至所有边界域的点划分完成。阈值 f 和给定数据集中的数据对象的分布有关，本文的 f 值根据数据集的特点给定。因此可以提高核心域和边界域划

分的准确率。

算法描述如下：

Input：数据集 D，聚类个数 k，阈值 t，阈值 f

Output：聚类结果的 k 个簇

1) 根据 1.2 小节中的定义计算数据集 D 中每个数据对象的离群因子 LOF 的值。

2) 根据 2.2 小节的方法对数据集进行处理。

3) 对处理之后的数据集，根据 2.4 小节的方法选取出 k 个初始聚类中心。

4) 在选出的 k 个聚类中心的基础上执行 k-means 算法，得到该数据集的 k-means 的聚类结果。

5) 对 4) 的聚类结果中的每一个簇的数据对象分别计算其与自身所在聚类中心的距离，若该距离小于给定的阈值 f 与每个簇中的距离簇中心最远的数据对象距离的乘积，则划分到 L 域，若该距离大于给定的阈值 f 与每个簇中的距离聚类中心最远的数据对象距离的乘积，则划分到 M 域。对于每一类的边界域，任取一点 z，寻找距离 z 点最近的 5 个点，如果这五个点中有三个及以上的点与 z 点不属于同一类，则把 z 点归入公共边界域，否则 z 点只属于该类的边界域。重复上述操作，直至所有边界域的点划分完成。

6) 通过 5) 得到的结果，返回 $\left\{\left(C_1^L,C_1^M\right),\left(C_2^L,C_2^M\right),\cdots,\left(C_k^L,C_K^M\right)\right\}$。

本算法通过使用 LOF 离群点检测算法来处理数据集，并进行减少了 k-means 对初始聚类中心的敏感程度以及离群点的对聚类结果的影响。

4. 聚类评价指标

本文采用 ACC、DBI 以及 NMI 三种评价指标来衡量聚类算法性能。

4.1. 准确率

准确率 ACC (Accuracy)是一种常见的用来评价聚类算法性能好坏的指标。它是指在所有分类中，分类正确的样本数与总样本数之比。聚类结果的 ACC 越高，聚类效果就越好。

定义 14 准确率 ACC [22]。

$$ACC=\frac{1}{N}\sum_{i=1}^{k}n_i \tag{10}$$

其中，N 表示数据集中数据对象的个数，k 代表聚类中心数，n_i 表示被正确划分到簇 i 的数据对象个数，$i=1,2,\cdots,k$。

4.2. 戴维森堡丁指数

戴维森堡丁指数 DBI (Davies-Bouldin Index) [23]。DBI 指数是一种用于评估聚类算法性能的外部评价指标，它衡量的是聚类内聚度和分离度的比值。DBI 指数越小，表示聚类效果越好。

定义 14 DBI (Davies-Bouldin Index)。

$$R_{ij}=\frac{\left(S_i+S_j\right)}{d_{ij}} \tag{11}$$

$$DBI=\frac{1}{k}\sum_{i=1}^{k}\max_{i\neq j}R_{ij} \tag{12}$$

其中 S_i 代表这个类的直径；d_{ij} 表示类 i 与 j 的聚类中心之间的距离。

4.3. 标准化互信息

标准化互信息(Normalized Mutual Information，简称 NMI) [24]是一种用于度量两个聚类结果之间相似度的指标。它主要用于聚类分析和社区发现中，以评估聚类算法的性能。NMI 的取值范围在 0 到 1 之间，值越大表示两个聚类结果越相似，即聚类效果越好。

定义 15 标准化互信息 NMI。

$$NMI = \frac{\sum_{i=1}^{k}\sum_{j=1}^{e} h_{i,j}^{a,b} \log \frac{n * h_{i,j}^{a,b}}{h_i^a * h_j^b}}{\sqrt{\left(\sum_{i=1}^{k} h_i^a \log \frac{h_i^a}{n}\right)\left(\sum_{j=1}^{e} h_j^b \log \frac{h_j^b}{n}\right)}} \tag{13}$$

其中，k 表示簇数，e 表示数据集本质聚类类别数，$h_{i,j}^{a,b}$ 表示数据对象属于真实标签类 j 但被划分到聚类结果簇 i 中的个数，h_i^a 表示聚类结果簇 i 中数据对象的个数，h_j^b 表示真实标签类 j 中数据对象的个数，n 为数据对象个数。

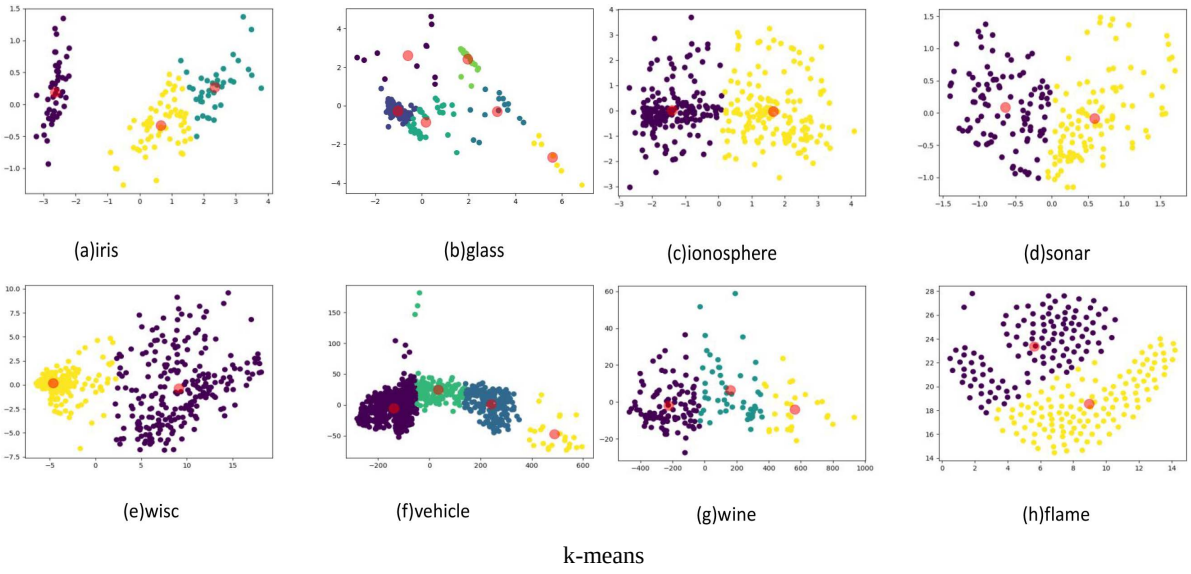
5. 实验结果与分析

为了验证提出的 ODT-kmeans 算法的有效性，分别在 iris、glass、heart-statlog、ionosphere、sonar、vehicle、wine、flame 数据集上进行聚类，将本文算法聚类得到的实验结果与传统 kmeans、IK-DM、OFMMK-kmeans、kmeansPA [25]算法的聚类结果进行比较，并使用 ACC、DBI、NMI 评价指标进行衡量。

5.1. UCI 数据集实验结果

下图为本文算法和其他算法在各种数据集上的效果图，使用不同颜色带表不同簇，效果如图 3~5 所示。

图 6 给出了五种算法在各种数据集上的 ACC 条形图。



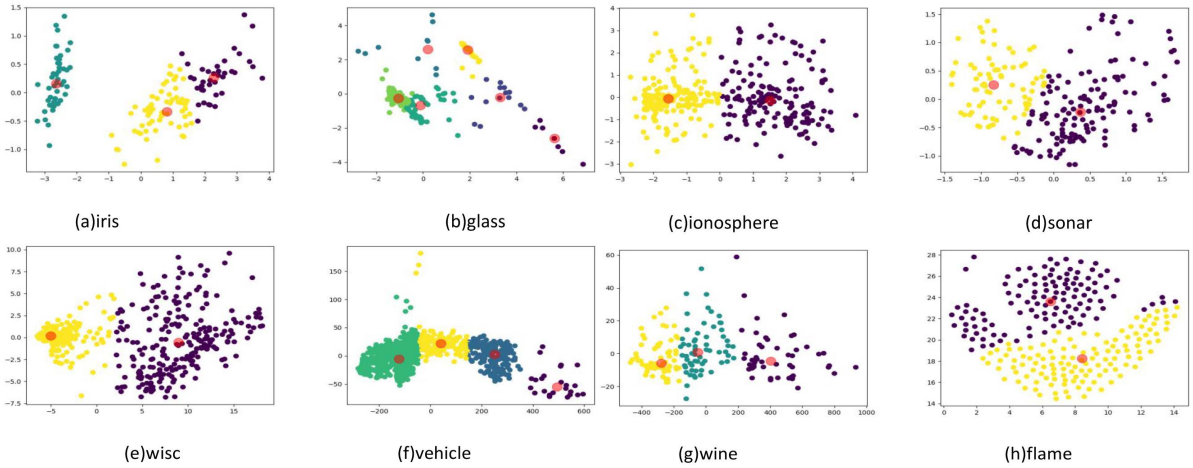
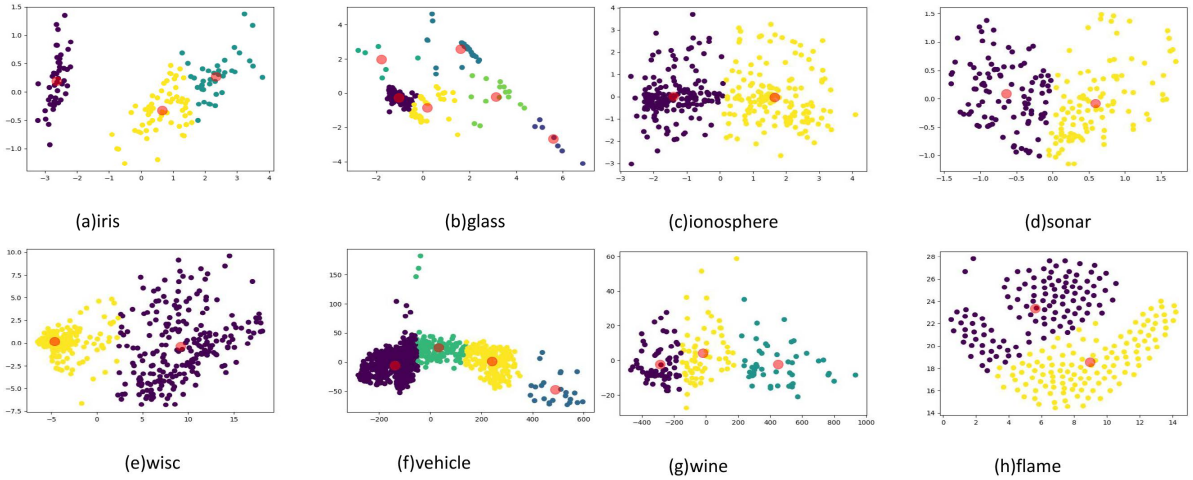


Figure 3. Comparison of algorithm effects Figure 1
图 3. 算法效果对比图 1



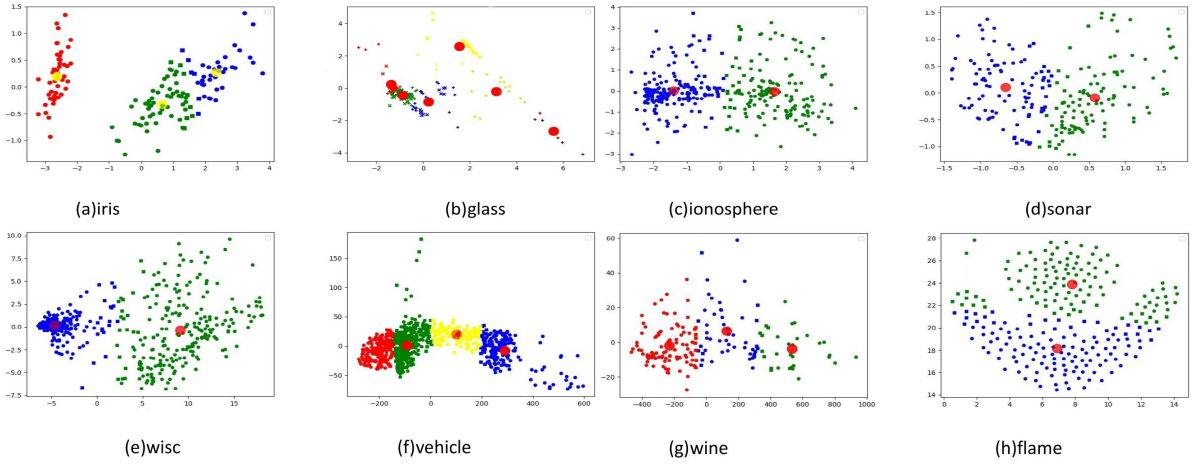


Figure 4. Comparison of algorithm effects Figure 2
图 4. 算法效果对比图 2

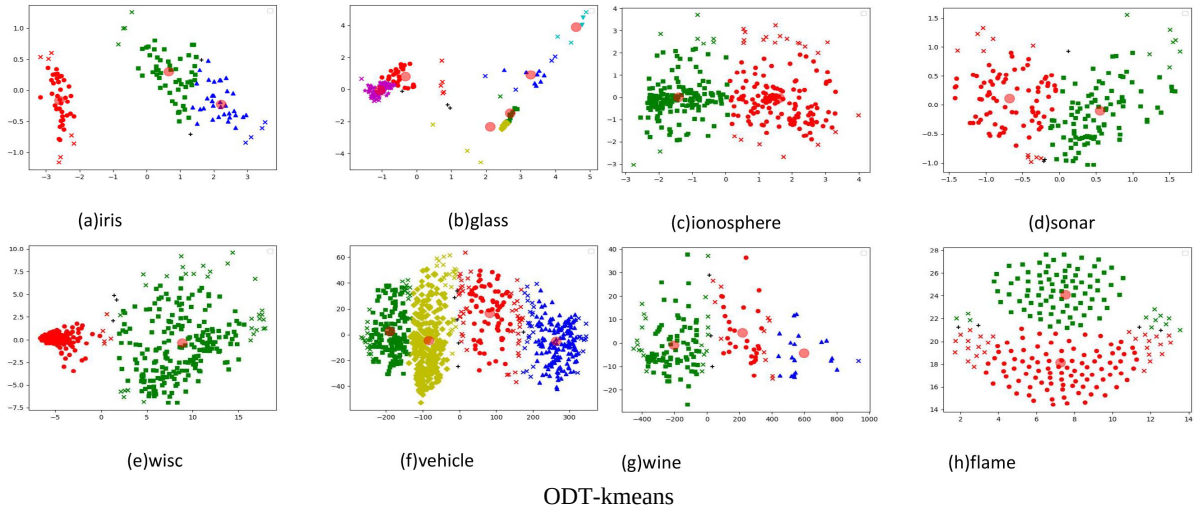


Figure 5. Comparison of algorithm effects Figure 3
图 5. 算法效果对比图 3

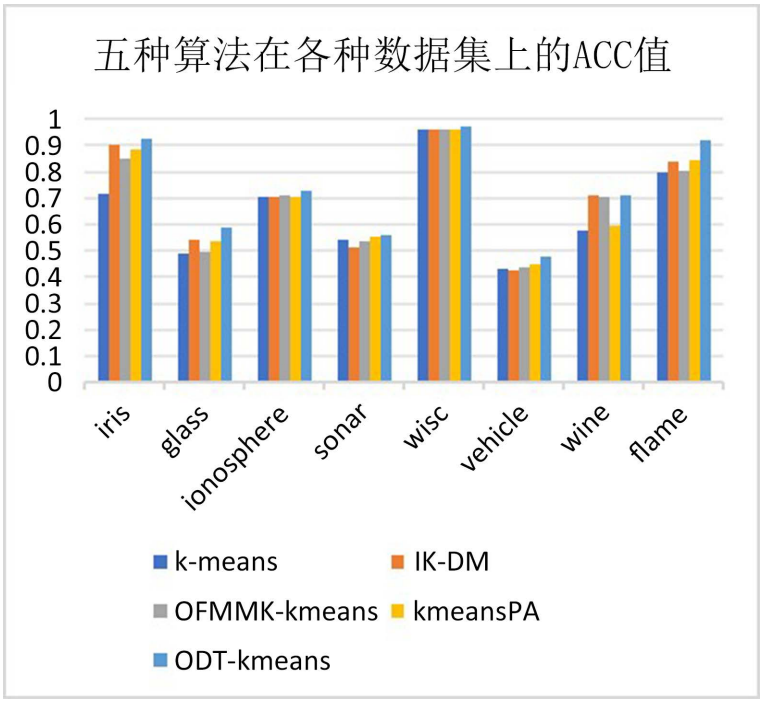


Figure 6. ACC bar chart of five algorithms
图 6. 五种算法的 ACC 条形图

表 1 给出了各种算法在数据集上的聚类评价指标值，加粗的表示不同的算法在同一数据集上的相同指标的最佳值。

Table 1. Algorithm clustering evaluation index value
表 1. 算法聚类评价指标值

数据集	评价指标	k-means	IK-DM	OFMMK-kmeans	kmeansPA	ODT-kmeans
Iris	ACC	0.718 1	0.9000	0.8492	0.8867	**0.9236**

续表

Iris	DBI	0.5983	**0.5651**	0.5895	**0.5651**	0.5893
	NMI	0.7431	0.7661	0.7235	0.7419	**0.7842**
glass	ACC	0.4907	0.5421	0.4968	0.5374	**0.5860**
	DBI	0.7357	0.7484	**0.6655**	0.7426	0.7378
	NMI	0.2862	0.4879	**0.5128**	0.3874	0.4775
ionosphere	ACC	0.7066	0.7037	0.7123	0.7066	**0.7294**
	DBI	0.7241	0.7230	**0.7206**	0.7241	0.7296
	NMI	0.1270	0.1264	0.1349	0.1270	**0.1688**
sonar	ACC	0.5433	0.5144	0.5385	0.5529	**0.5606**
	DBI	1.0269	0.9693	1.0275	1.0228	**0.9610**
	NMI	0.0058	0.0018	0.0058	0.0088	**0.0113**
wisc	ACC	0.9585	0.9628	0.9613	0.9613	**0.9708**
	DBI	0.4369	0.4370	0.4370	0.4370	**0.4131**
	NMI	0.7361	0.7564	0.7495	0.7495	**0.8008**
vehicle	ACC	0.4338	0.4279	0.4350	0.4515	**0.4766**
	DBI	0.5219	**0.5166**	0.5388	0.6252	0.5848
	NMI	0.1908	0.1889	0.1892	0.1846	**0.2143**
wine	ACC	0.5787	0.7079	0.7022	0.5955	**0.7126**
	DBI	0.5487	**0.5304**	0.5343	0.5572	0.5337
	NMI	0.4140	0.4193	0.4288	0.4102	**0.4325**
flame	ACC	0.7985	0.8375	0.8038	0.8458	**0.9220**
	DBI	1.1184	1.1193	1.1171	1.1171	**0.9553**
	NMI	0.3989	0.3989	0.4521	0.4343	**0.6058**

5.2. 实验结果分析

通过图 3 的对比可以看出 ODT-kmeans 算法的聚类结果更加符合预期，这是因为 ODT-kmeans 算法在选取初始聚类中心时有效排除了离群点的影响，同时选取了处于密度最大的区域的数据对象作为初始聚类中心，有效避免了选取到密度较低区域数据对象的可能性，提高了聚类的准确率。同时 ODT-kmeans 算法把得到的聚类结果进一步细分成了核心域和边界域，使数据的区分度有了提高。而 k-means 算法、OFMMK-kmeans 算法、kmeansPA 算法的结果相对较差，原因可能是这些算法在选取初始聚类中心时是随机选取的，可能会使选取的初始中心点偏离实际的初始聚类中心点，最终导致聚类效果降低。通过图四可知，ODT-kmeans 算法所选取的数据集上相对于其他几种算法都表现出明显的优势，尤其在 iris、glass、vehicle、wine、flame 数据集上的聚类准确率相对于 kmeans 算法都有了大幅度提升，相对于其他三种算法也有着不同幅度的提升。在 ionosphere、sonar、wisc 数据集上，各算法的准确率都比较接近，但 ODT-kmeans 的准确率仍然比其他算

法高，这充分说明了 ODT-kmeans 算法性能的稳定，体现了该算法的优越性，可以适用于多种数据集。

从表 1 的指标来看，k-means 算法在多个数据集上展现出了稳定的性能，特别是在 iris 和 wine 数据集上，其准确率(ACC)分别达到了 0.9236 和 0.7126，显示出较高的聚类效果。然而，k-means 算法对初始聚类中心的选择敏感且难以处理非球形簇或大小差异显著的簇，这在 ionosphere 数据集上体现得尤为明显，其 ACC 相对较低，同时 DBI 值较高，表明聚类间的分离度不够理想。IK-DM 算法在 glass 数据集上表现尤为突出，ACC 高达 0.5860，且 DBI 和 NMI 指标也相对较好，显示出该算法在处理具有复杂结构的数据集时具有一定的优势。此外，IK-DM 算法的计算复杂度通常较高，可能不适用于大规模数据集。OFMMK-kmeans 算法在 ionosphere 数据集上，其 ACC 达到了 0.7294，略高于 k-means 算法，且 NMI 值也较高，表明聚类结果与真实标签之间的信息重叠度较好。然而在 vehicle 数据集上 OFMMK-kmeans 的 ACC 和 NMI 均较低，可能与其对特定数据特征的敏感性有关。kmeansPA 算法在特定数据集上如 flame 上表现优异，ACC 高达 0.9708，且 DBI 和 NMI 值均较低，显示出极高的聚类质量和良好的聚类间分离度。然而在 wine 数据集上，其 ACC 仅为 0.5955，远低于其他算法。从 ACC 这一指标来看，ODT-kmeans 算法在 iris 数据集上的 ACC 达到了 0.9236，相较于 k-means 算法(0.7181)和其他对比算法如 IK-DM (0.8492)、OFMMK-kmeans (0.8867)等，有着明显的提升。这种提升不仅体现了 ODT-kmeans 算法在特征提取和聚类划分上的精准度，也反映了算法在处理复杂数据集时的强大能力。在 flame 数据集上，IK-DM 算法性能的提升并不大，体现出了该算法处理非球形簇时存在一定的局限性。而 ODT-kmeans 相对于其他聚类算法在指标上有较大提升。类似地，在 glass、ionosphere 等数据集上，ODT-kmeans 算法同样展现出了较高的 ACC 值，进一步验证了其广泛的适用性和优越性。在多个数据集上，ODT-kmeans 算法在保持较低 DBI 值的同时，也获得了较高的 NMI 值。在 glass 数据集上，ODT-kmeans 算法的 DBI 值为 0.7378，相较于其他算法如 k-means (0.7484)等有所降低，而 NMI 值则达到了 0.4775，显著高于部分对比算法。这种在聚类质量和纯度上的双重优化，使得 ODT-kmeans 算法在复杂数据集的聚类任务中更具竞争力。在不同数据集上，ODT-kmeans 算法的聚类性能均保持在一个相对稳定的水平，没有出现大幅度的波动。ODT-kmeans 算法在各种数据集上的聚类效果都有了显著提高，在 iris、glass、wine、flame 数据集上相对于传统 k-means 聚类算法的准确率至少提高了 10%，同时本文算法可以确定符合数据集的最佳 k 值，可以进一步提高聚类效果，因此可以验证 ODT-kmeans 算法是有效可行的。

6. 结论

本文提出了 ODT-kmeans 算法，使用 LOF 离群点检测算法排除了数据集中的离群点的干扰，使用手肘法确定了符合数据集的最佳 k 值后，采用密度最大方法和最大距离原则以及结合每个点的 LOF 值来选取初始中心点并进行 k-means 聚类，最后结合三支决策的思想聚类结果进行优化，有效解决了传统 k-means 算法初始中心随机选择的问题和聚类簇数需要人为给定、容易受到离群点影响、类别划分的问题。在各种数据集上的实验结果表明 ODT-kmeans 算法相对于 kmeans 算法的准确率和稳定性具有显著提高，但是本文在对离群点进行筛选时的对阈值的选择还有待讨论，如何选取最优的离群点排除阈值是今后研究的重点。

基金项目

哈尔滨师范大学双一流－提高人才培养质量项目(1504120015)；哈尔滨师范大学计算机科学与信息工程学院教育教学改革项目(JKYJGY202205)。

参考文献

[1] Brown, D., Japa, A. and Shi, Y. (2019). A Fast Density-Grid Based Clustering Method. 2019 *IEEE* 9*th Annual*

Computing and Communication Workshop and Conference (*CCWC*), Las Vegas, 7-9 January 2019, 48-54. https://doi.org/10.1109/ccwc.2019.8666548

[2] Xu, C., Lin, R., Cai, J. and Wang, S. (2022) Deep Image Clustering by Fusing Contrastive Learning and Neighbor Relation Mining. *Knowledge-Based Systems*, **238**, Article 107967. https://doi.org/10.1016/j.knosys.2021.107967

[3] 岳珊, 雍巧玲. 基于确定初始簇心的优化 K-Means 算法[J]. 数字技术与应用, 2023, 41(11): 140-142.

[4] Hu, H., Liu, J., Zhang, X. and Fang, M. (2023) An Effective and Adaptable K-Means Algorithm for Big Data Cluster Analysis. *Pattern Recognition*, **139**, Article 109404. https://doi.org/10.1016/j.patcog.2023.109404

[5] Shrifan, N.H.M.M., Akbar, M.F. and Isa, N.A.M. (2022) An Adaptive Outlier Removal Aided K-Means Clustering Algorithm. *Journal of King Saud University—Computer and Information Sciences*, **34**, 6365-6376. https://doi.org/10.1016/j.jksuci.2021.07.003

[6] 孙林, 刘梦含, 徐久成. 基于优化初始聚类中心和轮廓系数的 K-Means 聚类算法[J]. 模糊系统与数学, 2022, 36(1): 47-65.

[7] 郭文娟. 基于优化初始聚类中心的 K-Means 聚类算法[J]. 科技风, 2022(4): 63-65.

[8] 张亚迪, 孙悦, 刘锋, 等. 结合密度参数与中心替换的改进 K-Means 算法及新聚类有效性指标研究[J]. 计算机科学, 2022, 49(1): 121-132.

[9] 刘美玲, 黄名选, 汤卫东. 基于离散量优化初始聚类中心的 K-Means 算法[J]. 计算机工程与科学, 2017, 39(6): 1164-1170.

[10] 唐东凯, 王红梅, 胡明, 等. 优化初始聚类中心的改进 K-Means 算法[J]. 小型微型计算机系统, 2018, 39(8): 1819-1823.

[11] 廖纪勇, 吴晟, 刘爱莲. 基于相异性度量选取初始聚类中心改进的 K-Means 聚类算法[J]. 控制与决策, 2021, 36(12): 3083-3090.

[12] Hodge, V. and Austin, J. (2004) A Survey of Outlier Detection Methodologies. *Artificial Intelligence Review*, **22**, 85-126. https://doi.org/10.1023/b:aire.0000045502.10941.a9

[13] Du, X., Yu, J., Ye, L., *et al*. (2020) Outlier Detection Algorithm Based on Graph Random Walk. *Journal of Computer Applications*, **40**, Article 1322.

[14] Breunig, M.M., Kriegel, H., Ng, R.T. and Sander, J. (2000) LOF: Identifying Density-Based Local Outliers. *Proceedings of the* 2000 *ACM SIGMOD International Conference on Management of Data*, Dallas, 15-18 May 2000, 93-104. https://doi.org/10.1145/342009.335388

[15] Yao, Y. (2007) Decision-Theoretic Rough Set Models. In: *Lecture Notes in Computer Science*, Springer, 1-12. https://doi.org/10.1007/978-3-540-72458-2_1

[16] Yao, Y. (2009) Three-Way Decision: An Interpretation of Rules in Rough Set Theory. In: *Lecture Notes in Computer Science*, Springer, 642-649. https://doi.org/10.1007/978-3-642-02962-2_81

[17] Yao, Y. (2012) An Outline of a Theory of Three-Way Decisions. In: *Lecture Notes in Computer Science*, Springer, 1-17. https://doi.org/10.1007/978-3-642-32115-3_1

[18] Yu, H., Chu, S. and Yang, D. (2012) Autonomous Knowledge-Oriented Clustering Using Decision-Theoretic Rough Set Theory. *Fundamenta Informaticae*, **115**, 141-156. https://doi.org/10.3233/fi-2012-646

[19] 唐欣. 三支 K-means 聚类算法及其应用研究[D]: [硕士学位论文]. 银川: 北方民族大学, 2023.

[20] 李浩溥, 李志聪. 结合孤立森林和鲸鱼优化算法的三支 K-Means [J]. 长江信息通信, 2023, 36(2): 48-50.

[21] Wang, P., Yang, X., Ding, W., Zhan, J. and Yao, Y. (2024) Three-Way Clustering: Foundations, Survey and Challenges. *Applied Soft Computing*, **151**, Article 111131. https://doi.org/10.1016/j.asoc.2023.111131

[22] Chen, W., Song, Y., Bai, H., Lin, C. and Chang, E.Y. (2011) Parallel Spectral Clustering in Distributed Systems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **33**, 568-586. https://doi.org/10.1109/tpami.2010.88

[23] Davies, D.L. and Bouldin, D.W. (1979) A Cluster Separation Measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **1**, 224-227. https://doi.org/10.1109/tpami.1979.4766909

[24] Wang, Y. and Chen, L. (2016) K-MEAP: Multiple Exemplars Affinity Propagation with Specified *K* Clusters. *IEEE Transactions on Neural Networks and Learning Systems*, **27**, 2670-2682. https://doi.org/10.1109/tnnls.2015.2495268

[25] 蔺艳艳, 陆介平, 王郁鑫, 等. 改进的 K-Means 算法在三支决策中的应用研究[J]. 计算机与数字工程, 2020, 48(6): 1294-1299+1353.