Computer Science and Application 计算机科学与应用, 2025, 15(1), 246-255
Published Online January 2025 in Hans. https://www.hanspub.org/journal/csa
https://doi.org/10.12677/csa.2025.151025

Hans汉斯

基于密度聚类的三支K-Means聚类算法

李志聪，晏啸昊

哈尔滨师范大学计算机科学与工程学院，黑龙江 哈尔滨

收稿日期：2024年12月24日；录用日期：2025年1月23日；发布日期：2025年1月29日

摘　要

本文提出了一种基于密度聚类的三支K-Means算法。针对传统的K-Means算法在选取初始聚类中心时往往依赖于随机选择和无法处理不确定性数据对象的问题，本文采用基于密度聚类算法优化初始聚类中心的选择，并优化了截断距离的选取，最后使用三支决策的方法对聚类结果进行处理。实验结果表明，与传统的K-Means算法相比，改进的K-Means算法在聚类中表现出更高的聚类精度和稳定性。

关键词

K-Means算法，密度聚类，三支决策

Three-Way K-Means Algorithm Based on Density Clustering

Zhicong Li, Xiaohao Yan

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

Received: Dec. 24th, 2024; accepted: Jan. 23rd, 2025; published: Jan. 29th, 2025

Abstract

This paper proposes a three-branch K-Means algorithm based on density clustering. In view of the problem that the traditional K-Means algorithm often relies on random selection and cannot handle uncertain data objects when selecting initial clustering centers, this paper uses a density-based clustering algorithm to optimize the selection of initial clustering centers, and optimizes the selection of truncation distance. Finally, a three-branch decision method is used to process the clustering results. The experimental results show that the improved K-Means algorithm exhibits higher clustering accuracy and stability in clustering compared to the traditional K-Means algorithm.

Keywords

K-Means Algorithm, Density Clustering, Three-Way Decision-Making

文章引用: 李志聪, 晏啸昊. 基于密度聚类的三支 K-Means 聚类算法[J]. 计算机科学与应用, 2025, 15(1): 246-255.
DOI: 10.12677/csa.2025.151025

1. 引言

在数据挖掘和机器学习领域中，聚类分析一直是一个重要的研究方向。聚类算法旨在将一组无标签的数据点按照某种相似性或距离度量标准划分为多个簇(Cluster)，使得同一簇内的数据点彼此相似，而不同簇间的数据点则差异显著[1]。这一特性使得聚类分析在数据探索、模式识别、图像分割、市场细分、生物信息学等多个领域得到了广泛应用。

聚类算法的研究从早期的层次聚类、划分聚类到后来的密度聚类、网格聚类、模型聚类等，各种算法层出不穷，各有千秋[2]。其中，K-Means 算法以其简单高效、易于实现的特点成为最经典的聚类算法之一，但其对初始聚类中心选择的敏感性、对噪声和异常值的鲁棒性不足等问题也限制了其应用范围[3]。此外，传统的 K-Means 算法属于硬聚类算法，往往要求数据对象的边界清晰，一个数据对象最多只能属于一个类别。然而，在实际应用中，数据往往包含大量的不确定性。为了解决上述存在的问题，研究者们不断探索新的聚类算法和改进策略，以期在保持算法效率的同时，提高聚类结果的准确性和稳定性。

Yoder 等[4]提出了半监督的 K-Means++算法，该算法在随机选取一个初始聚类中心后，对数据集中的每一个样本，计算其与已选择的聚类中心之间的最短距离。然后，根据距离的平方为每个样本分配一个权重，权重越大表示该样本被选为下一个聚类中心的概率越高。最后，根据这些权重随机选择一个样本作为下一个聚类中心。孟子健等[5]提出了一种选择初始聚类中心的算法。该算法首先计算数据对象两两之间的相异度函数，构造一种新的相异度矩阵，然后选取 k 个与其他数据对象相异度较低且个数最多的数据对象作为初始聚类中心。张亚迪等[6]提出了一种基于密度参数和中心替换的改进 K-Means 算法 DC-Kmeans。该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，因而比传统的聚类算法更加精确。QIAN Jin、Pingxin Wang 等[7] [8]提出在聚类算法中引入三支决策。在三支聚类中，对象与簇之间的关系不仅仅是属于或不属于，还可能包括一种中间状态，如部分属于或具有某种程度的关联性，通过引入第三种可能性，三支聚类能够更好地处理数据中的不确定性和模糊性，从而提供更丰富、更细致的聚类结果[9]。孙旭阳等[10]提出了基于离群点和自适应参数的三支 DBSCAN 算法，该算法将三支决策思想与离群点检测 LOF 算法进行结合，有效地处理了数据集内不确定信息的聚类问题，并对聚类后产生的离群点做出了进一步的判断，显著提高了聚类的准确率。朱金等[11]提出了基于蚁群算法的三支 K-Means 聚类算法，利用蚁群算法中随机概率选择策略和信息素的正负反馈机制，动态调整权重的方法，对三支 K-Means 聚类算法进行优化。

依据上述问题，本文提出了基于密度聚类的三支 K-Means 聚类算法，本文简称为 D3W-Kmeans 算法。该算法首先通过密度峰值聚类算法(Density Peak Clustering Algorithm，简称 DPC)优化初始聚类中心的选择，并采用中位数代替均值计算新的类簇中心，最后利用三支决策思想对聚类结果进行处理。本文首先介绍了 K-Means 算法基本原理和存在的问题，然后详细阐述了基于密度聚类的初始聚类中心优化方法和三支决策思想。最后，通过实验验证了所提方法的有效性和优越性。

2. 相关工作

2.1. 密度峰值聚类(DPC)算法

密度峰值聚类算法(Density Peak Clustering Algorithm，简称 DPC)是一种无监督的聚类算法，由 Alex

Rodriguez 和 Alessandro Laio [12]于 2014 年提出。该算法的核心思想是通过自动发现数据中的密度峰值点，并将这些峰值点作为聚类中心，进而将剩余的数据点分配到最近的聚类中心所在的类簇中。DPC 算法因其简单而高效的特点，在数据挖掘、模式识别、图像处理、社交网络分析等多个领域得到了广泛应用。

下面将介绍 DPC 算法的相关概念，如图 1 所示。

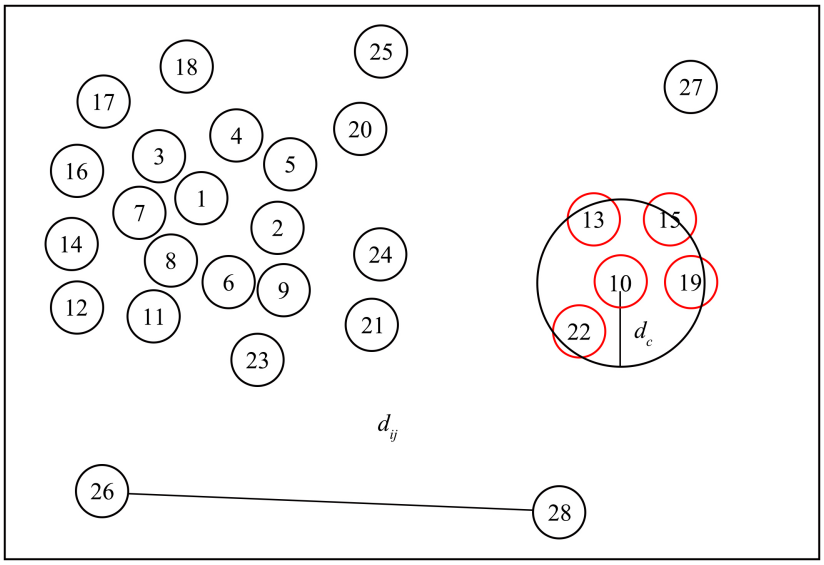

Figure 1. DPC algorithm-related concepts
图 1. DPC 算法相关概念

DPC 算法基于两个关键概念：局部密度 ρ 和相对距离 δ 。

定义 1 局部密度

局部密度 ρ 表示一个数据点周围一定半径范围内的数据点数量，用于描述该点的密集程度。局部密度通常采用截断核(Cut-off kernel)的计算方式，对于数据集中的每一个点 x_i ，其局部密度 ρ_i 的定义为：

$$\rho_i = \sum_{j \neq i} \chi\left(d_{ij} - d_c\right) \tag{1}$$

局部密度 ρ 表示一个数据点周围一定半径范围内的数据点数量，用于描述该点的密集程度。其中，d_{ij} 是点 x_i 和点 x_j 之间的距离(多半采用欧氏距离)，d_c 是截断距离(cutoff distance)，是一个需要预先设定的参数，$\chi(x)$ 是单位阶跃函数，如果 $d_{ij} < d_c$ ，则 $\chi\left(d_{ij} < d_c\right) = 1$ ，否则为 0。因此，ρ_i 实际上是点 x_i 的 d_c 邻域内其他点的数量。

定义 2 相对距离

对于数据集中的每一个点 x_i ，其相对距离 δ_i 定义为：

$$\delta_i = \min_{j:\rho_j > \rho_i} \left(d_{ij}\right) \tag{2}$$

即，δ_i 是点 x_i 到所有局部密度大于 ρ_i 的点 x_j 之间的最短距离。如果点 x_i 的局部密度是数据集中最大的，则通常将 δ_i 定义为数据集中所有点对之间的最大距离，以确保该点被选为聚类中心之一。

定义 3 截断距离

截断距离 d_c 是一个需要预先设定的参数，但在实际应用中，它首先需要计算所有点对之间的距离，并构建一个距离矩阵，并对距离矩阵中所有的距离进行排序，最后选择一个百分比(如 2%)的最大距离作

为截断距离 d_c。这个百分比的选择是经验性的，并且可能需要根据具体数据集进行调整。

2.2. K-Means 聚类算法

K-Means 是一种基于划分的无监督聚类算法，能将数据集分成 k 类，其中 k 是事先假定的。K-Means 算法随机产生 k 个聚类中心，根据最近邻原则将数据点归类离其最近的聚类中心，形成 k 个类，并重新计算各类的聚类中心，重复上述步骤直到聚类中心不再改变位置或达到规定的迭代次数为止[13]。

定义 4 欧氏距离

在 K-Means 算法中，通常使用欧氏距离来计算数据点与簇中心之间的距离。对于数据点 x_i 和簇中心 c_j，其欧氏距离 $d\left(x_i, c_j\right)$ 定义为：

$$d\left(x_i, c_j\right)=\sqrt{\sum_{k=1}^{n}\left(x_{ik}-c_{jk}\right)^2} \tag{3}$$

其中，n 是数据点的维度，x_{ik} 是数据点 x_i 在第 k 维上的值，c_{jk} 是簇中心 c_j 在第 k 维上的值。

在每次迭代中，每个簇的质心 c_i 会根据簇内所有点的均值进行更新。对于第 j 个簇，其新的质心 c_j' 的定义为：

$$c_j'=\frac{1}{\left|C_j\right|}\sum_{x_i\in C_j} x_i \tag{4}$$

其中，C_j 是第 j 个簇中所有点的集合，$\left|C_j\right|$ 是该簇中点的数量，x_i 是簇 C_j 中的点。

K-Means 算法的停止条件可以基于多种因素，常见的有以下几种：

质心变化：如果连续两次迭代中所有簇的质心变化非常小(例如，小于某个预设的阈值)，则算法停止。

迭代次数：达到预设的最大迭代次数后停止。

目标函数：K-Means 算法的目标是最小化所有点到其簇中心的距离平方和(SSE, Sum of Squared Errors)。当 SSE 的变化非常小或达到某个阈值时，算法可以停止。在 K-Means 算法中，SSE (Sum of Squared Errors，误差平方和)是衡量聚类效果的一个重要指标。它表示的是每个数据点到其所属簇中心点的距离的平方和。SSE 越小，表示数据点与簇中心点的距离越近，聚类的效果越好。SSE 的计算公式如下：

$$\mathrm{SSE}=\sum_{k=1}^{K}\sum_{i\in C_k}\left(x_i-c_k\right)^2 \tag{5}$$

其中，K 表示簇的数量；C_k 表示第 k 个簇，即包含所有属于第 k 个簇的数据点的集合；x_i 表示数据集中的每个数据点；c_k 表示第 k 个簇的中心点。

2.3. 三支聚类

Yao [14] [15]在决策粗糙集和概率粗糙集的假设中提出了三只决策理论，它是一种基于人类认知的决策模式，主要用于处理不确定性和模糊性的决策问题。三支决策理论将决策空间划分为三个互不重叠的区域：核心域(Positive Region，R-域)、琐碎域(Negative Region，L-域)和边界域(Boundary Region，M-域)。这三个区域分别对应了决策中的三种状态：接受、拒绝和不确定。

Yu [16]等将三支决策理论结合到聚类方法中，提出了三支聚类方法。三支聚类将一个论域划分为三个不相交的区域：核心域(Core Region)、边界域(Fringe Region)和琐碎域(Trivial Region)，分别对应聚类中的肯定、不确定和否定状态。各区域的具体定义如下：

定义 5 核心域

核心域($Co(C)$)包含那些明确属于某个聚类的对象，这些对象与聚类中心的相似度或距离满足一定的

阈值条件。

定义 6 边界域

边界域($Fr(C)$)包含那些可能属于某个聚类但又不完全确定的对象，这些对象与聚类中心的相似度或距离处于一定的模糊范围内。

定义 7 琐碎域

琐碎域($Tr(C)$)包含那些明确不属于任何聚类的对象，这些对象与所有聚类中心的相似度或距离都低于某个阈值。

三支聚类通常使用一对集合来表示一个聚类：

$$C=\left(Co(C),Fr(C)\right) \tag{6}$$

其中 $Co(C)$和 $Fr(C)$分别表示聚类的核心区域和边缘区域，而外部区域 $Tr(C)$则是论域 U 中除了 $Co(C)$和 $Fr(C)$之外的所有对象组成的集合，即

$$Tr(C)=U-Fr(C)-Co(C) \tag{7}$$

3. 基于密度聚类的三支 K-Means 算法

3.1. 基于密度聚类的初始聚类中心选择算法

本文所提出的初始聚类中心选择算法，旨在通过计算数据集中各数据对象的密度来明确初始类簇中心，从而有效规避 K-Means 算法因随机选择初始类簇中心而导致的聚类结果不稳定问题。二支聚类与三支聚类的区别如图 2 所示。

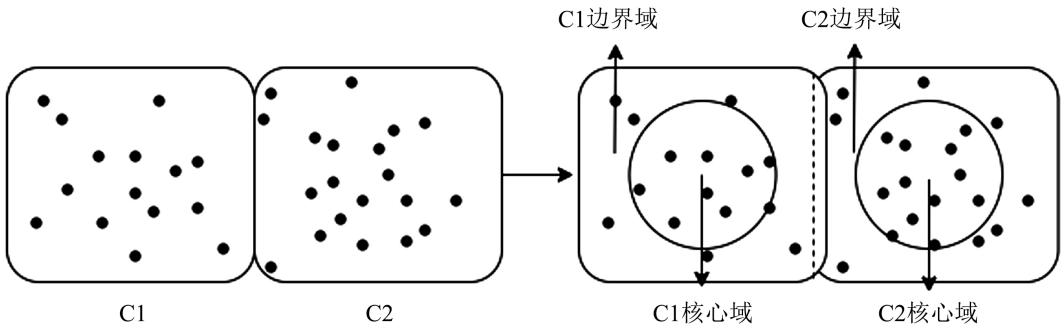

Figure 2. Two-branch clustering is different from three-branch clustering
图 2. 二支聚类与三支聚类区别

本文所提出的初始聚类中心选择算法，旨在通过计算数据集中各数据对象的密度来明确初始类簇中心，从而有效规避 K-Means 算法因随机选择初始类簇中心而导致的聚类结果不稳定问题。二支聚类与三支聚类的区别如图 2 所示。

定义 8 基于欧氏距离，所有数据对象之间距离的中位数定义为 *MedianDist*。

定义 9 动态中位距离

对于不同的数据集，聚类后所生成的簇的数量以及各类簇中包含的数据对象的数目可能会存在差异。伴随着类簇数量以及各个类簇中数据对象的变化，数据对象之间的距离也会相应地发生变化。为此，定义基于所有数据对象之间距离中位数和簇数的动态中位距离 *DyMeDist* 作为截断距离 d_c：

$$DyMeDist=MedianDist/K \tag{8}$$

$$d_c = DyMeDist \tag{9}$$

其中，K 为数据集 D 被划分的类簇的数目。

3.2. 类簇中心的替换

针对 K-Means 算法对离群值敏感的问题，本文采用中位数代替均值计算新的类簇中心的方法，并根据数据集规模采取了两种不同的计算方式。当数据集规模较小时，计算所有数据对象欧氏距离的中位数；当数据集规模较大时，随机选取设定数量的数据对象并计算其欧氏距离的中位数，以此提升处理大规模数据集时的效率。

3.3. 三支聚类

在 K-Means 聚类后，每个数据点都被分配到了一个簇中。然而，由于数据的复杂性和噪声的存在，某些数据点的簇分配可能并不明确，即存在不确定性。本文引入三支决策思想来处理聚类结果的不确定性。

为了量化三支决策的决策区域，本文定义以下公式：设 x_i 为数据点，c_i 为第 i 个簇的簇中心，$d\left(x_i,c_i\right)$ 为点 x_i 到簇中心 c_i 的欧氏距离，δ 为划分核心域与边界域的阈值。

如果 $\forall j \neq i$，$d\left(x_i,c_i\right) < d\left(x_i,c_j\right)$，则 x_i 属于 c_i 的核心域。

如果 $\exists j \neq i$，$d\left(x_i,c_i\right) < d\left(x_i,c_j\right) - \delta$，则 x_i 不属于 c_i 的核心域。

且如果 $d\left(x_i,c_i\right) > \max_k d\left(x_i,c_k\right) + \delta$，则 x_i 属于 c_i 的琐碎域。

不满足核心域和琐碎域条件的则属于边界域。

3.4. D3W-K-Means 算法流程

输入：数据集 $D=\{x_1,x_2,\cdots,x_n\}$；类簇数 K

输出：数据集 $C=\{C_1,C_2,\cdots,C_K\}$

Step1：计算数据集 D 中任意一对数据对象 $\left(x_i,x_j\right)$ 之间的动态中位距离(*DyMeDist*)作为截断距离 d_c

Step2：for $i=1,2,\cdots,n$ do

计算数据 x_i 对象的密度 $\rho\left(x_i,d_c\right)$

Step3：for $k=1,2,\cdots,K$ do //搜索并确定 k 个初始类簇中心，随后将它们添加到初始类簇中心集合 V 中

Step3.1：从数据集 D 中选取密度最高的数据对象 x，并删除从数据集 D 中 x 邻域内的所有数据对象。设 x 为第 k 个初始类簇中心 v_k

Step3.2：$V \leftarrow v_k$；//将 v_k 放入初始类簇中心的集合 V

Step4：初始化各个类簇 C_k

Step5：将 D 中的各个数据对象放入相应的类簇中。计算数据对象 x_i 和 V 中各个类簇中心之间的距离，将 x_i 放入距离最近的类簇中

Step6：用中位数替代均值计算新的类簇中心 v_k 并在更新类簇中心后初始化类簇 C_k。当 v_k 不再改变，此时中止迭代

Step7：利用三支决策思想划分每个类簇 C_k 的核心域 $Co\left(C\right)$ 和边界域 $Fr\left(C\right)$，此时聚类完成

4. 聚类结果评价指标

4.1. ACC (Accuracy)

ACC，即聚类准确率，是聚类算法外部评价指标之一[17]。它通过比较聚类算法为每个样本分配的聚类标签与样本的真实标签之间的匹配程度，来评估聚类算法的性能。ACC 的取值范围在 0 到 1 之间，值

越接近 1，表示聚类结果越准确，即聚类算法的性能越好。

定义 10 ACC

ACC 的计算公式基于样本的聚类标签与真实标签之间的对应关系。具体来说，假设有 N 个样本，第 i 个样本聚类产生的标签是 p_i，真实标签是 y_i，则 ACC 的计算公式可以表示为：

$$\mathrm{ACC}=\frac{1}{N}\sum_{i=1}^{N}\delta\left(y_i,map\left(p_i\right)\right) \tag{10}$$

其中，$\delta(x,y)$ 是一个 delta 函数，当 $x=y$ 时，$\delta(x,y)=1$，否则 $\delta(x,y)=0$。$map(p_i)$ 是一个映射函数，用于将聚类标签 p_i 映射到真实标签 y_i 相对应的最佳排列上。这是因为聚类算法可能会产生与真实标签顺序不一致的聚类标签，但只要聚类结果中的样本分配是正确的，就可以通过映射函数找到与真实标签相对应的聚类标签排列。

4.2. F 值

F 值，也称为 F-Measure 或 F-Score，是信息检索和机器学习领域中常用的一种性能评价指标，它综合了查准率和查全率的优点，用于衡量聚类算法生成的聚类结果与真实标签之间的匹配程度[18]。F 值越高，表示聚类效果越好，即聚类算法的性能越优。

定义 11 F 值

F 值的计算公式如下：

$$F_\beta=\frac{\left(\beta^2+1\right)\times P\times R}{\beta^2\times P+R} \tag{11}$$

其中，F_β 表示 F 值，β 是一个调整参数，用于平衡查准率(P)和查全率(R)的重要性。当 $\beta=1$ 时，F 值退化为最常见的 F1 值，此时查准率和查全率的重要性相等。

查准率和查全率的公式分别为：

$$\text{查准率：}\ P=\frac{\left|P_j\cap C_i\right|}{\left|C_i\right|} \tag{12}$$

表示聚类算法将样本正确分配到某个簇的比例。其中，P_j 表示真实标签中的第 j 个簇，C_i 表示聚类算法生成的第 i 个簇，$\left|P_i\cap C_i\right|$ 表示真实簇 P_j 和聚类簇 C_i 的交集大小，$\left|C_i\right|$ 表示聚类簇 C_i 的大小。

$$\text{查全率：}\ R=\frac{\left|P_j\cap C_i\right|}{\left|P_j\right|} \tag{13}$$

表示真实簇 P_j 中的样本被聚类算法正确分配到簇 C_i 的比例。其中，$\left|P_j\right|$ 表示真实簇 P_j 的大小。

F 值广泛应用于聚类效果的评估中，特别是在有外部标签信息的情况下，可以通过将聚类结果与真实标签进行比较来计算 F 值。通过 F 值，我们可以评估聚类算法在不同参数设置下的性能表现，选择最优的聚类算法和参数设置。

4.3. Purity

Purity 的基本思想是将聚类结果中的每个簇分配给与其包含最多相同真实标签的类别，然后计算所有簇分配正确的样本数占总样本数的比例[19]。这个比例越高，表示聚类结果与真实标签的一致性越好，即聚类算法的纯度越高。

定义 12 Purity

Purity 的计算公式可表示为：

$$Purity = \frac{1}{N}\sum_{i=1}^{k}\max_{j}\left|\omega_i \cap c_j\right| \tag{14}$$

其中，N 是样本总数；k 是聚类结果中簇的个数；ω_i 表示第 i 个簇中的样本集合；c_j 表示真实标签中第 j 个类别的样本集合；$\left|\omega_i \cap c_j\right|$ 表示第 i 个簇与第 j 个真实类别之间的交集大小，即同时属于该簇和该类别的样本数；$\max_j \left|\omega_i \cap c_j\right|$ 表示第 i 个簇与所有真实类别交集大小中的最大值，即该簇分配给与其包含最多相同真实标签的类别。

5. 实验结果与分析

本节选取了 5 组 UCI 数据集(表 1)对基于密度聚类的三支 K-Means 聚类算法的性能进行测试及评估。

Table 1. Data set description
表 1. 数据集描述

ID	数据集	样本数	属性数	类别数
1	Iris	150	4	3
2	Seeds	210	7	3
3	Wine	178	13	3
4	Glass	214	9	6
5	Wdbc	569	30	2

为了对所提出的算法验证其有效性，对聚类结果分别计算 ACC、F 值、Purity 来评价聚类效果。数值越大，表明聚类效果越好。为了能更好地体现出 D3W-Kmeans 算法在 ACC、F 值、Purity 性能指标上有所提升，将所提出的算法与传统 K-Means 算法和 DC-Kmeans 算法进行聚类性能指标比较，实验结果如表 2。

Table 2. Results on the UCI data set
表 2. UCI 数据集上的结果

ID	算法	ACC	F 值	Purity
1	K-Means	0.7603	0.7321	0.7533
	DC-Kmeans	0.8455	0.8346	0.8842
	D3W-Kmeans	0.8741	0.8566	0.9084
2	K-Means	0.6573	0.6405	0.7534
	DC-Kmeans	0.7576	0.7433	0.8708
	D3W-Kmeans	0.7666	0.7504	0.9253
3	K-Means	0.6475	0.6683	0.6714
	DC-Kmeans	0.7159	0.7152	0.7345
	D3W-Kmeans	0.6943	0.6836	0.7259

续表

4	K-Means	0.5981	0.5874	0.6124
	DC-Kmeans	0.6357	0.6145	0.6896
	D3W-Kmeans	0.6782	0.6317	0.7021
5	K-Means	0.8541	0.8268	0.8562
	DC-Kmeans	0.8764	0.8467	0.8865
	D3W-Kmeans	0.8863	0.8475	0.9043

根据表 2 的实验结果可以看出，与 K-Means 和 DC-Kmeans 算法比较，本文中算法在 5 个数据集上对性能指标 ACC、F 值、Purity 上有所提高。实验结果表明，基于密度聚类的 D3W-Kmeans 聚类算法在大部分数据集上相比传统 K-Means 聚类算法和 DC-Kmeans 算法拥有更高的聚类精度和准确性。鉴于 UCI 真实数据集的数据分布往往错综复杂，具有高度的多样性和非均匀性，这种复杂性对传统 K-Means 聚类算法构成了严峻挑战，导致其在处理这些真实数据时表现欠佳。相比之下，D3W-Kmeans 算法通过引入密度聚类，规避了传统 K-Means 算法随机选取初始聚类中心的影响，利用三支决策思想将样本更精确地划分为核心域、边界域、琐碎域，更加灵活地适应了复杂数据分布的特点，从而在真实数据集中实现了显著的准确率提升。这种改进不仅增强了算法对数据局部特性的敏感性和适应性，还有效提升了聚类结果的准确性和鲁棒性，为处理复杂数据集提供了一种更为高效和可靠的解决方案。

6. 结束语

本文首先提出改进 K-Means 算法来解决传统 K-Means 算法的不足。在算法的初始阶段，采用了密度聚类的方法来确定初始类簇中心，这一策略有效地规避了传统 K-Means 算法随机选择初始中心点可能导致的聚类效果不佳的问题。进入迭代阶段后，进一步优化了类簇中心的更新机制。传统的 K-Means 算法使用均值来计算新的类簇中心，但在某些情况下，均值可能会受到极端值或噪声数据的影响，导致聚类结果偏离实际。为了解决这个问题，该算法引入了使用中位数代替均值的方法来计算新的类簇中心。中位数作为一种更加稳健的统计量，能够更好地反映数据的中心趋势，减少噪声数据的干扰，从而提高聚类的准确性和稳定性。此外，针对传统 K-Means 算法在处理不确定性数据时存在的局限性，本文引入了三支决策思想。三支决策思想将聚类结果划分为核心域和边界域，其中核心域包含确定性较高的数据点，而边界域则包含不确定性较高的数据点。

为了验证本文提出的方法的有效性，本文进行了大量的实验测试。测试结果表明，与传统 K-Means 算法相比，本文提出的方法在聚类准确率、稳定性和鲁棒性等方面均表现出显著的优势，在数据挖掘、机器学习等领域的广泛应用前景。

参考文献

[1] Xu, R. and WunschII, D. (2005) Survey of Clustering Algorithms. *IEEE Transactions on Neural Networks*, **16**, 645-678. https://doi.org/10.1109/tnn.2005.845141

[2] Sinaga, K.P. and Yang, M. (2020) Unsupervised K-Means Clustering Algorithm. *IEEE Access*, **8**, 80716-80727. https://doi.org/10.1109/access.2020.2988796

[3] Zhao, W., Deng, C. and Ngo, C. (2018) K-Means: A Revisit. *Neurocomputing*, **291**, 195-206. https://doi.org/10.1016/j.neucom.2018.02.072

[4] Yoder, J. and Priebe, C.E. (2017) Semi-Supervised K-Means++. *Journal of Statistical Computation and Simulation*, **87**,

2597-2608. https://doi.org/10.1080/00949655.2017.1327588

[5] 孟子健, 马江洪. 一种可选初始聚类中心的改进 K-Means 算法[J]. 统计与决策, 2014(12): 12-14.

[6] 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.

[7] 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

[8] 钱进, 郑明晨, 周川鹏, 等. 多粒度三支决策研究进展[J]. 数据采集与处理, 2024, 39(2): 361-375.

[9] 钱进, 汤大伟, 洪承鑫. 多粒度层次序贯三支决策模型研究[J]. 山东大学学报(理学版), 2022, 57(9): 33-45.

[10] 李志聪, 孙旭阳. 基于离群点检测和自适应参数的三支 DBSCAN 算法[J]. 计算机应用研究, 2024, 41(7): 1999-2004.

[11] 朱金, 徐天杰, 王平心. 基于蚁群算法的三支 K-Means 聚类算法[J]. 江苏科技大学学报(自然科学版), 2024, 38(3): 63-69.

[12] Rodriguez, A. and Laio, A. (2014) Clustering by Fast Search and Find of Density Peaks. *Science*, **344**, 1492-1496. https://doi.org/10.1126/science.1242072

[13] 王森, 刘琛, 邢帅杰. K-Means 聚类算法研究综述[J]. 华东交通大学学报, 2022, 39(5): 119-126.

[14] Yao, Y. (2010) Three-Way Decisions with Probabilistic Rough Sets. *Information Sciences*, **180**, 341-353. https://doi.org/10.1016/j.ins.2009.09.021

[15] Yao, Y. (2011) The Superiority of Three-Way Decisions in Probabilistic Rough Set Models. *Information Sciences*, **181**, 1080-1096. https://doi.org/10.1016/j.ins.2010.11.019

[16] 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

[17] Penrose, D.M. and Glick, B.R. (2003) Methods for Isolating and Characterizing ACC Deaminase-Containing Plant Growth-Promoting Rhizobacteria. *Physiologia Plantarum*, **118**, 10-15. https://doi.org/10.1034/j.1399-3054.2003.00086.x

[18] Chung, C.Y., Liu, C., Wang, K. and Zykaj, B.B. (2015) Institutional Monitoring: Evidence from the F-Score. *Journal of Business Finance & Accounting*, **42**, 885-914. https://doi.org/10.1111/jbfa.12123

[19] Detlefsen, M. and Arana, A. (2011) Purity of Methods. https://philpapers.org/rec/DETPOM