



K-Means algorithm based on multi-feature-induced order

Benting Wan¹ · Weikang Huang¹ · Bilivogui Pierre² · Youyu Cheng¹ · Shufen Zhou¹

Received: 23 December 2023 / Accepted: 5 March 2024 / Published online: 9 April 2024
© The Author(s), under exclusive licence to Springer Nature Switzerland AG 2024

Abstract

The K-Means algorithm is a powerful tool for data analysis, but it faces several challenges when dealing with large multi-feature data. Centroid initialization and centroid determination are two significant hurdles that can reduce the performance of the K-Means algorithm. To address these challenges, based on partial-order relations, an enhanced K-Means algorithm, the multi-feature induced order K-Means algorithm (OWAK-Means) is developed which combines with a novel centroid initialization based on partial-order relations and a multi-feature induced ordered weighted average (MFIOWA) operator. By using a weighted iteration method based on partial-order relations, the OWAK-Means algorithm initializes centroids with greater precision. The MFIOWA operator is designed based on database indexing theory and the Sigmoid weight function that improves its information filtering ability. These techniques, combined with an ordered weighted distance metric and the MFIOWA operator, make the OWAK-Means algorithm an effective tool for multi-feature data analysis. In comparative analysis with the variants of the K-Means algorithm, the OWAK-Means algorithm has significant improvement in the adjusted rand score, normalized mutual information, and purity. Statistical tests, comprehensive evaluation methods, and sensitivity analysis prove that the OWAK-Means algorithm is effective and reliable.

Keywords Multi-feature-induced order · Sigmoid weight · Noise · OWAK-Means algorithm

1 Introduction

Clustering algorithms are widely used in data mining, pattern recognition, and other related fields (Goicovich et al. 2021; Peng et al. 2022; Fu et al. 2023; Le et al. 2023). Common clustering algorithms include partition clustering, hierarchical clustering, density clustering, etc. (Peng et al. 2022). Notably, since being introduced by MacQueen (1967), the K-Means algorithm has received significant attention for its simplicity and efficiency. Recently, centroid initialization and feature weight optimization of the K-Means algorithm have become a research focus. K-Means exhibits a strong dependence on the initial centroid vector, and the conventional random selection of the initial centroid will lead to unstable clustering results. To this end, the

researchers proposed different improved K-Means algorithms (Arthur and Vassilvitskii 2007; Li and Wu 2012; Mawati et al. 2014; Zhou et al. 2017; Rashidi et al. 2020; Ay et al. 2023). For example, Arthur and Vassilvitskii (2007) developed the K-Means++ algorithm, which determined the initial centroids by the probabilistic method and accelerated the convergence speed. Ay et al. (2023) improved the stability and convergence efficiency of centroid calculation under big-data samples by fixing partial initial centroids. Random selection of the initial centroids is easy to cause: (1) When the distance between the initial centroids is too large, the convergence speed of the K-Means algorithm slows down; (2) When the distance between the initial centroids is too small, it is not conducive to the calculation of the final centroids and determination of cluster boundaries during the clustering process. The random selection method is proved inadequate for multi-feature data. Moreover, in real life, partial-order relationship is common in multi-feature data (Wang et al. 2016). The traditional centroid initialization method cannot consider the partial-order relation. Therefore, how to determine the initial centroids of K-Means through the partial-order relationship between multi-feature data remains to be studied.

✉ Benting Wan
wanbenting@jxufe.edu.cn

¹ Department of Software and IoT, Jiangxi University of Finance and Economics, Nanchang 330013, China

² School of Finance, Zhongnan University of Economics and Law, Wuhan 430073, China

For multi-feature data clustering, existing research mainly focuses on the optimization of feature weights and distance metrics to improve the performance of the K-Means algorithm (Huang et al. 2005, 2018; Chen et al. 2012; De Amorim and Mirkin 2012). For instance, Huang et al. (2005), based on a global optimization strategy, designed a new K-Means algorithm that can automatically solve feature weights. Similarly, Renato (De Amorim and Mirkin 2012) employed the Minkowski distance metric to construct an objective optimization function. These approaches are based on the weighted feature distance metrics and construct objective optimization equations to improve the clustering performance of the K-Means algorithm. In recent years, more and more studies have begun to improve the K-Means algorithm from the perspective of multi-feature data fusion (Cheng et al. 2009; Pons-Vives et al. 2022). Researchers have employed operators such as the ordered weighted average (OWA) operator (Yager and Filev 1999) to aggregate data, thereby improving the performance of the K-Means algorithm. For example, based on the OWA operator, Pons-Vives et al. (2022) introduced the ordered weighted distance relationship (OWDr) metric to enhance the K-Means algorithm's sensitivity to different features and improve the neglect of differences between features which is the deficiency of Euclidean distance. Cheng et al. (2009) combined stepwise regression feature selection with the OWA operator to reduce multi-dimensional features and enhance the accuracy of K-Means in classification tasks. Despite these advancements, there remains an unmet need for research focused on weighted fusion at the feature level, particularly during the centroid calculation in clustering processes. At the same time, the weighted calculation can be further used for data level, that is, the important difference among cluster members should be considered in the centroid calculation process, and the noise data (Askari 2021) that will cause bias should be screened. The OWA operator family and its weight methods can solve the above, hence the combination of the OWA operator and K-Means centroid calculation needs to be studied.

To enhance the adaptability of the OWA operator in data fusion, Yager and Filev (1999) developed the induced ordered weighted average (IOWA) operator. Later, scholars did further research on the selection and acquisition methods of induced factors (Chiclana et al. 2007; Ma and Guo 2011; Yi et al. 2018; Ji et al. 2021). Building upon Yager's studies (1999), Chiclana et al. (2007) developed the Consistency-IOWA operator. Ma and Guo (2011) and Yi et al. (2018) introduced the Density-IOWA operator and Quantile-IOWA operator, respectively. Ji et al. (2021) developed Average-IOWA for expert opinions aggregation. However, these operators mainly rely on single-factor induction, which can lead to biases and often results in

numerous identical sorting outcomes when applied to big data. For multi-feature data such as vectors, there is not only one influencing factor but also the relationship between factors should be considered. Consequently, single-factor-induced sorting falls short of addressing the complexities of multi-feature data. Inspired by database indexing theory (Huang and Chen 2000), this paper proposed a new multi-feature-induced order weighted average (MFIOWA) operator to expand the adaptive range of the IOWA operator.

However, for the IOWA operator family, feature weights have always been a highly concerned problem. Different weight methods can make IOWA operators play different roles. For example, Yager (1988) obtained OWA weights through linguistic quantifiers and introduced Orness and Dispersion measures for weights optimization (Yager 1993; Yager and Filev 1994). O'Hagan (1988) and Xu and Da (2003) used the Orness measure to construct an objective optimization function to solve the weight. However, in the process of multi-feature data processing, scholars will screen samples and features by measure metrics, such as density, variance, Pearson correlation coefficient, and so on (Naik and Kiran 2021), or identify the importance of features by sensitivity analysis (Zhang 2019; Asheghi et al. 2020). How to use derived information to complete the filtering and aggregation of multi-feature data poses a challenge to IOWA's weight function. Considering the excellent characteristics, which are continuous, smooth, differentiable, and saturated at a certain value, the Sigmoid function has drawn considerable interest from researchers (Xu et al. 2021; Dombi and Jónás 2022). For example, Dombi and Jónás (2022) proposed a generalization of the Sigmoid function and used it for logistic regression and preference modeling. To enable the proposed MFIOWA operator to address data-fusion tasks with different information filtering requirements, and provide K-Means with more flexible and powerful data-fusion as well as noise-resistant tool, the weight method and solution model based on the Sigmoid function need to be studied.

According to the above investigations, the K-Means algorithm and IOWA operator as potent tools for data mining, but the K-Means algorithm combining with IOWA operator to improve cluster performances are challenges, the motivation of this paper is as follows.

1. Although multi-feature weighted K-Means have received high attention from researchers, K-Means combined with OWA operator have not been developed based on the induced order relationship between features.
2. The random selection centroid of K-Means causes unstable clustering results, and the partial-order

relationships between features need to be explored to improve the stability of K-Means.

3. The IOWA only takes the single-factor induction based on the OWA operator, multi-factor induction of the OWA operator has not been designed.

In this paper, we develop the OWAK-Means clustering algorithm which is based on the proposed MFIOWA operator and centroid initialization method. The major contributions of this research are as follows.

1. Inspired by the IOWA and database indexing theory (Yager and Filev 1999; Huang and Chen 2000), the MFIOWA operator is proposed. A new weight method is derived for the MFIOWA operator based on the Sigmoid function, and an objective optimization model is constructed to solve the feature weights.
2. Inspired by the K-Means algorithm (De Amorim and Mirkin 2012; Pons-Vives et al. 2022; Ay et al. 2023), an OWAK-Means algorithm is developed for the multi-feature clustering which integrates a new centroid initialization method and the MFIOWA operator.
3. The OWAK-Means is compared with variant K-Means on UCI real-world datasets. Comparison and analysis results show that the OWAK-Means has obvious advantages over the other algorithms in clustering performance.

The subsequent sections of this study are structured as follows: In Sect. 2, an extensive review is described. Section 3 introduces the MFIOWA operator. In Sect. 4, the OWAK-Means algorithm is elucidated. Section 5 describes the comparison and analysis of OWAK-Means with other cluster algorithms. Finally, Sect. 6 summarizes the contributions of our research and discusses potential avenues for future exploration in this domain.

2 Related work

2.1 IOWA operator

Definition 1 (Yager and Filev 1999) Let (a_1, a_2, \dots, a_n) be a set of real numbers, and $((u_1, a_1), (u_2, a_2), \dots, (u_n, a_n))$ be a set of numerical pairs. The IOWA operator with n -dimension is a mapping: $R^n \rightarrow R$, characterized by an n -dimensional weight vector $W = (w_1, w_2, \dots, w_n)^T$. The weight vector satisfies $w_i \geq 0, \sum_1^n w_i = 1 (i = 1, 2, \dots, n)$. The IOWA operator is shown as Eq. (1).

$$\text{IOWA}((u_1, a_1), (u_2, a_2), \dots, (u_n, a_n)) = \sum_{j=1}^n w_j c_j. \quad (1)$$

In Eq. (1), c_j represents a_i which is associated with the j -th largest induced value u_i in the sequence (u_1, u_2, \dots, u_n) . The sequence can be in descending or ascending order, depending on the sort rule. Notably, the Orness measure of the IOWA operator, as referenced in Yager’s paper (1988, 1993), is the same as that of the OWA operator, as shown in Eq. (2).

$$\text{Orness}(W) = \frac{1}{n-1} \sum_{i=1}^n (n-i)w_i. \quad (2)$$

The existing IOWA operator is characterized by a single-factor induction. However, real-world scenarios frequently involve a plurality of induced factors. To address this complexity, the MFIOWA operator is developed in Sect. 3.1.

2.2 K-Means clustering algorithm

K-Means algorithm can be distilled into a specific objective optimization function (De Amorim and Mirkin 2012). Suppose a dataset has N entities, each entity has M attributes. This dataset is represented by the matrix $X = (x_{iv})$, where x_{iv} denotes the value of the v -th attribute for the i -th entity ($i = 1, 2, \dots, N, v = 1, 2, \dots, M$). The K-Means algorithm divides the dataset into K clusters and the output of K-Means is represented by $C = \{C_1, C_2, \dots, C_k\}$, where the centroid of each cluster is represented by $\mu_k = (c_{kv})$. The minimization objective function of the K-Means is articulated in Eq. (3).

$$D = \sum_{k=1}^K \sum_{i=1}^N \sum_{v=1}^M s_{ik} (x_{iv} - c_{kv})^2 \quad (3)$$

where s_{ik} is an indicator function that $s_{ik} = 1$ if $x_i \in C_k$, and $s_{ik} = 0$ otherwise.

To improve the performances of the K-Means algorithm, there are several challenges (a) selecting initial centroids, (b) optimizing feature weight in the context of multi-feature big data, (c) optimizing centroid calculation based on multi-feature weights, (d) optimizing distance metric. Concerning issue (a), a detailed review is scheduled for Sect. 2.2.1. For issue (c), a comprehensible explanation will be provided in Sect. 2.2.2. This paper proposed an initial centroid selecting algorithm to address the issue (a) based on partial-order relations. For challenges (b) and (c), a new MFIOWA operator will be employed to filter interference data and noise data in the data fusion processing. Regarding issue (d), this study will employ the improved OWDr distance metric to calculate the distance between samples.

2.2.1 K-Means initial centroid selection method

To address the challenge of selecting centroids in K-Means clustering, researchers have made significant contributions (Arthur and Vassilvitskii 2007; Li and Wu 2012; Mawati et al. 2014; Wang et al. 2016; Zhou et al. 2017; Nainggolan et al. 2019; Khan et al. 2019; Rashidi et al. 2020; Ay et al. 2023). Arthur and Vassilvitskii (2007) proposed the K-Means++ algorithm, which utilizes a probabilistic method for computing distances to determine the initial centroids of K-Means. This method starts from a randomly chosen point, subsequently selects the farthest point as the initial centroid, and iteratively identifies K initial centroids, ensuring substantial distances between each cluster. Arthur and Vassilvitskii (2007) method has greatly improved the accuracy and convergence speed of K-Means, but the random selection of cluster centroid still brings challenges to the accuracy and stability of the K-Means algorithm. There are also many studies on centroid selection of clustering. For instance, Li and Wu (2012) employ the Max-Min distance algorithm to address the dependency of the K-Means algorithm on the initial centroid selection. Mawati et al. (2014) integrated a systematic method for initial centroid determination. This approach iteratively identifies $0.75 * (n/K)$ points in closest proximity to each other through an iterative algorithm and obtains the centroid using an arithmetic mean.

In the context of big data, Steinbach et al. (2000) introduced the Bisecting K-Means clustering algorithm to reduce the dependency of the clustering process on the selection of random points. The Bisecting K-Means algorithm employs a dichotomy to overcome K-Means convergence to the local optimal state to the local optimal state and reduce similarity computations to improve the overall execution speed. At the same time, in big data environments, the K-Means algorithm is also facing new challenges. Sculley (2010) developed the Mini-Batch K-Means algorithm, which employs mini-batch optimization methods to reduce clustering time, thereby improving the K-Means algorithm's adaptability to big data scenarios. Ay et al. (2023) introduced a new hybrid method, which combined the part-fixed clustering centroid method with the dynamic determination of the centroid. This method demonstrated effectiveness in high-dimensional feature spaces and big-data datasets and overcomes the problem of initial centroid selection and centroid solution under big data environment.

These studies have investigated various strategies to optimize the selection of initial centroids to enhance the stability and convergence efficiency of the K-Means algorithm. Nevertheless, in professional domains, the inherent partial-order relationships of data are often overlooked. Therefore, this paper proposes setting initial

centroids based on partial-order relationships to address this gap.

2.2.2 K-Means centroid optimization method

For the high-dimensional and multi-feature clustering problem, Eq. (3) can satisfy the clustering task when the feature weights are equal. However, in practical application, different features often possess different levels of importance. Recent researchers have developed K-Means algorithms for feature weight optimization (de Amorim 2016). For example, Modha and Spangler (2003) introduced the convex K-Means planning algorithm (CK-Means), based on Eq. (3). The CK-Means is designed to minimize intra-cluster dispersion while maximizing inter-cluster dispersion. The weight relationships among features are linear. Nonetheless, the CK-Means approach may not be apt for scenarios where feature relationships exhibit non-linear dependencies.

For the issue of non-linear relationships in feature weights, recent scholars have primarily focused on the exponentiation weights (w^β) (Makarenkov and Legendre 2001; Frigui and Nasraoui 2004) to refine the interaction among features. Makarenkov and Legendre (2001) utilized w^2 to improve the K-Means algorithm. Frigui and Nasraoui (2004) introduced a regularization method to adjust w^2 . Inspired by Makarenkov and Legendre (2001), Chan et al. (2004), and Huang et al. (2005) developed the Weighted K-Means (WK-Means) algorithm. The WK-Means automatically calculates the weights for each feature during the clustering process and integrates a step for computing feature weights in each iteration. The objective optimization function of WK-Means is shown in Eq. (4).

$$D = \sum_{k=1}^K \sum_{i=1}^N \sum_{v=1}^M s_{ik} w_v^\beta (x_{iv} - c_{kv})^2, \quad (4)$$

where $w_v = \frac{1}{\sum_{u \in V} (D_v/D_u)^{1/(\beta-1)}}$, and $D_v = \sum_{k=1}^K \sum_{i=1}^N s_{ik} (x_{iv} - c_{kv})^2$. In contrast to the approach of Modha and Spangler (2003), the WK-Means offers a defense against irrelevant or noisy features and emphasizes nonlinear relationships between feature weights. However, the WK-Means compromises the direct correlation between the scale of feature values and feature weights. Inspired by Chan et al. (2004), and Huang et al. (2005), De Amorim and Mirkin (2012) introduced weighted K-Means based on the Minkowski distance (MWK-Means), the optimization object function of MWK-Means is presented in Eq. (5).

$$D = \sum_{k=1}^K \sum_{i=1}^N \sum_{v=1}^M s_{ik} w_v^\beta (x_{iv} - c_{kv})^\beta \quad (5)$$

Equation (5) determines weights using feature scaling factors. The simplified expression of the MWK-Means algorithm is as follows:

$$D = \sum_{k=1}^K \sum_{i=1}^N \sum_{v=1}^M s_{ik} |w_v x_{iv} - w_v c_{kv}|^\beta = \sum_{k=1}^K \sum_{i=1}^N d^\beta |x_i' - c_i'|^\beta$$

, where $x_i' = (w_v x_{iv})$. Under the Minkowski distance metric, the MWK-Means algorithm transforms feature weights into scaling factors within the original K-Means criterion. The MWK-Means algorithm performs well in datasets with additional uniform random noise features, but determining reasonable parameter β is a challenge. Moreover, in big-data environments, the burden of calculating feature weights increases significantly. Compared with the method proposed by Huang et al. (2005), the method proposed by De Amorim and Mirkin (2012) is more flexible. While ensuring that the feature weights are non-linear, the distance metric is associated with the feature weights. However, the feature weights calculated by Huang and De Amorim may not be able to highlight representative features well. When the feature dimension increases, the weight difference between features is not obvious (Hashemzadeh et al. 2019). Therefore, the weight cluster algorithm needs to be improved.

Equations (4) and (5) are global feature optimization, so the computing cost will be high when there is a large amount of data. To address this problem, Hashemzadeh et al. (2019) employ weighted features within clusters and developed a new objective function: $D_k = \sum_{i=1}^N \sum_{v=1}^M u_{ik}^\alpha w_{kv}^\beta (x_{iv} - c_{kv})^2$. For high-dimensional sparse big data clustering, Jing et al. (2007) developed a K-Means algorithm with an entropy weight method and constructed the objective function:

$$D = \sum_{k=1}^K \left[\sum_{i=1}^N \sum_{v=1}^M w_{kv} (x_{iv} - c_{kv})^2 + \gamma \sum_{v=1}^M w_{kv} \log(w_{kv}) \right]$$

Jing’s method used local weighted features within clusters and entropy penalty terms to improve the algorithm’s performance. Inspired by Jing et al. (2007), Khan et al. (2019) introduced a weighted entropy penalty term and proposed a new objective optimization function.

The core of the above improvement method is to optimize the feature weights based on different weighting distances. Commonly used distance functions include Euclidean distance, Manhattan Distance, Chebyshev Distance, and so on. Euclidean distance is simple to calculate, Manhattan Distance is more suitable for datasets with independent features, and Chebyshev Distance has strong robustness. The Minkowski Distance (De Amorim and Mirkin 2012) is their generic form, with a lot of flexibility. Scholars have also improved K-Means from the

perspective of distance metric (Pons-Vives et al. 2022; Singh and Singh 2023; Savita and Siwch 2024). For example, Pons-Vives et al. (2022) utilized the OWA operator to define the ordered weight distance relative (OWDr) to further advance clustering methodologies. The distance formula Pons-Vives et al. (2022) proposed is $d_r(a, b) = |x - y|/\max(x, y)$. Subsequently, the optimization object function based on OWDr is shown in Eq. (6),

$$D = \sum_{k=1}^K \sum_{i=1}^N s_{ik} \text{OWDr}(x_{iv}, c_{kv}). \tag{6}$$

The OWDr significantly enhances the distinctiveness of information across varying feature dimensions, which further increases the likelihood and speed of convergence. OWA operator (Yager and Filev 1999) is an aggregation method that can effectively filter features. And Cheng et al. (2009) have adopted a stepwise regression feature selection method, employing the OWA operator for dimensionality reduction, which has been shown to improve the accuracy of K-Means in high-complexity and high-dimensional classification tasks.

Therefore, upon reviewing and analyzing existing literature, it becomes clear that current K-Means algorithms with weighted features primarily optimize performance through a focus on the weighted distance of features. At the same time, the important difference between data samples and interference samples in clusters is not taken into account, so how to screen and weighted aggregate multi-feature data has important research significance. In contrast, this study proposes a novel approach that emphasizes partial-order relations among different features and the significance of data samples within clusters. This perspective aims to further enhance the performance of the K-Means algorithm.

3 Multi-feature-induced ordered weighted averaging operator

3.1 MFOWA operator based on partial-order relation

Multi-feature order is one of the key technologies of database indexing which is often used to sort multi-feature or high-dimensional complex data (Huang and Chen 2000; Liang et al. 2021). Multi-feature sorting refers to determining the order of data based on the importance and internal rules of multiple features to meet different needs and scenarios, to achieve more accurate and comprehensive sorting results.

Data aggregation is critical in production and everyday life, particularly within big-data contexts. Relying on a

single feature is often inadequate for sorting tasks in real-world scenarios. Consider, for instance, there are five students' final grades in a university need to be comprehensively ranked, including five features: Advanced Mathematics, Python, Computer Networks, Sports, and Total Score. The score of Sports is classified into three levels (A, B, C) in descending order. The other features are numerical, ranging from 0 to 100, where higher values denote superior performance, as depicted in Table 1.

Employing the Total Score for initial sorting can select the top student (ID 15), but the priority relationship among students with IDs 11–14 cannot be distinguished. Therefore, by further using the Advanced Mathematics feature for induced sorting, the student with ID 11 can be placed at the lowest rank, but the students numbered 12, 13, and 14 still cannot be distinguished. Then, rest students are sequenced by the three features of Python, Computer Networks, and Sports until all students are sorted, and the final ranking result is (15, 13, 14, 12, 11).

As can be seen from Table 1, according to the database retrieval rules, an induced vector containing multiple features is formed by (Total Score, Advanced Mathematics, Python, Computer Networks, and Sports), adhering to the following relation: Total Score \succcurlyeq Advanced Mathematics \succcurlyeq Python \succcurlyeq Computer Networks \succcurlyeq Sports. In this context, the rank of student grades can be derived based on the induced vector. For instance, the data of student 15 is composed of ((263, 91, 90, 82, C), 263), where (263, 91, 90, 82, C) denotes the induced vector, and 263 represents the score corresponding to the induced vector. The rank result of student 15 is 1.

In this case, the sequence of multi features participating in the ranking is regarded as the induced partial-order relation. Drawing inspiration from database indexing theory and the IOWA operator, we designed the MFOWA as delineated in Definition 2.

Definition 2 Let $D = \{(U_i, a_i)\}$ be a two-dimensional dataset where $U_i = (u_{i1}, u_{i2}, \dots, u_{im})$ is an induced vector corresponding to the real number a_i and encompassing m features. These features uphold a partial-order relationship: $e_1 \succcurlyeq e_2 \succcurlyeq \dots \succcurlyeq e_m$. The MFOWA operator, characterized as a mapping: $R^n \rightarrow R$, is defined by an n -dimensional weight vector $W = (w_1, w_2, \dots, w_n)^T$, the weight vector satisfies

$w_i \geq 0, \sum_1^n w_i = 1 (i = 1, 2, \dots, n)$. The MFOWA operator is shown as Eq. (7).

$$\text{MFOWA}((U_1, a_1), (U_2, a_2), \dots, (U_n, a_n)) = \sum_{j=1}^n w_j d_j \tag{7}$$

In Eq. (7), d_j is defined as the j -th largest value of a_i after the application of a multi-feature partial order on the induced vector U_i . The behavior of the MFOWA operator is contingent on the composition of U_i . Specifically, when U_i encompasses a single feature, the MFOWA operator simplifies to an IOWA operator, and if U_i is an empty set, the MFOWA operator simplifies to an OWA operator. The sorting rule applied to the feature set (e_1, e_2, \dots, e_m) determines the type of multi-feature partial-order induction. If the feature set is all in ascending order, it is called ascending multi-feature partial-induced order, and vice versa, it is called descending multi-feature partial-induced order. If there are both ascending and descending order features, it is called mixed multi-feature partial-induced order. In this paper, ascending multi-feature partial-induced order is used. Compared with the IOWA operator, the MFOWA operator considers the effect of multiple features on data aggregation and the partial-order relationship between multiple features. MFOWA operator is like multi-feature index techniques (Liang et al. 2021). It utilizes expert-constructed sorting rules to rank and aggregate target data, with the feature's partial-order relationships significantly which will affect the sorting outcomes. Multi-feature sorting technique (Chen et al. 2022) offers the flexibility to modify, add, or remove features, and allows the definition of various feature combinations and partial-order relationships. So that it could comprehensively and accurately meet different sorting requirements and adapt to the change in decision-makers perspectives.

The MFOWA operator merges the idea of multi-feature partial induced order with the OWA operator's ability to reorder and aggregate data (Yager and Filev 1999) according to multiple features and their respective partial-order relationships. MFOWA operator exhibits attributes of comprehensiveness, stability, and heterogeneity, and the

Table 1 Final grades of sophomore students in a university

ID	Advanced Mathematics	Python	Computer networks	Sports	Total score	Rank
11	88	92	80	A	260	5
12	90	89	81	B	260	4
13	90	90	80	B	260	2
14	90	89	81	A	260	3
15	91	90	82	C	263	1

aggregated results can more accurately reflect the significance of features and the overall situation of samples. At the same time, MFOWA expands the application range of the IOWA operator by overcoming the limitations associated with a single induced factor in IOWA. Similar to the IOWA operator, the MFOWA operator satisfies properties of monotonicity, commutativity, boundedness, and idempotency which are derivable from operational rules.

Property 1 (Monotonicity) Let (U_i, a_i) and (U_i, a_i') ($i = 1, 2, \dots, n$) be two vector pairs, where U_i is an induced vector, and a_i and a_i' are two real numbers. For any $i \in [1, n]$, if a_i and a_i' satisfy the condition: $a_i \geq a_i'$, then Eq. (8) is established.

$$\text{MFOWA}((U_i, a_i)) \geq \text{MFOWA}((U_i, a_i')) \tag{8}$$

Proof Let d_j and d_j' be the j -th largest elements of a_i and a_i' after sorting induced by a multi-feature partial order ($j = 1, 2, \dots, n$). Since the induced vector U_i is the same, the rank results are identical, implying $d_j = a_i$ and $d_j' = a_i'$. There are non-negative weights w_j satisfying $\sum_1^n w_j = 1$ and $w_j \geq 0$ such that $\sum_1^n w_j d_j \geq \sum_1^n w_j d_j'$. Thus, Eq. (8) holds. \square

Property 2 (Commutativity) Let (U_i, a_i) be a vector pair, and a_i be a real number. If (U_i', a_i') is any arrangement of (U_i, a_i) ($i = 1, 2, \dots, n$), then Eq. (9) is established

$$\text{MFOWA}((U_i, a_i)) = \text{MFOWA}((U_i', a_i')). \tag{9}$$

Proof Let d_j and d_j' be the j -th largest elements of a_i and a_i' after sorting induced by a multi-feature partial order ($j = 1, 2, \dots, n$). Because (U_i', a_i') represent any ordering of (U_i, a_i) , (U_i', a_i') share the same ordering results as (U_i, a_i) . There exist non-negative weights w_j satisfying $\sum_1^n w_j = 1$ and $w_j \geq 0$, and for any j , $d_j = d_j'$. Therefore, the following equation can be obtained

$$\begin{aligned} \text{MFOWA}((U_i, a_i)) &= \sum_{j=1}^n w_j d_j \\ &= \sum_{j=1}^n w_j d_j' = \text{MFOWA}((U_i', a_i')) \end{aligned}$$

Equation (9) is proved. \square

Property 3 (Boundedness) Let (U_i, a_i) ($i = 1, 2, \dots, n$) be a vector pair, and a_i be a real number. For any $i \in [1, n]$, if $a_{\min} = \min(a_i)$, $a_{\max} = \max(a_i)$, then Eq. (10) is established.

$$a_{\min} \leq \text{MFOWA}((U_i, a_i)) \leq a_{\max} \tag{10}$$

Proof Because d_j is the j -th largest element of a_i after sorting induced by a multi-feature partial order

($j = 1, 2, \dots, n$), and d_j is a permutation of a_i , it follows that $a_{\min} \leq d_j \leq a_{\max}$. Therefore, the following equation can be obtained:

$$\begin{aligned} \sum_{j=1}^n w_j a_{\min} &\leq \text{MFOWA}((U_i, a_i)) \\ &= \sum_{j=1}^n w_j d_j \leq \sum_{j=1}^n w_j a_{\max} \end{aligned}$$

Also, since $\sum_{j=1}^n w_j = 1$, $\sum_{j=1}^n w_j a_{\min} = a_{\min} \sum_{j=1}^n w_j$, and $\sum_{j=1}^n w_j a_{\max} = a_{\max} \sum_{j=1}^n w_j$, it can be demonstrated that Eq. (10) holds. \square

Property 4 (Idempotency) Let (U_i, a_i) be a vector pair. If $a_1 = \dots = a_n = a$, then Eq. (11) is established

$$\text{MFOWA}((U_i, a_i)) = a. \tag{11}$$

Proof Since $\sum_{j=1}^n w_j = 1$ and for any i , $a_i = a$ it follows that $\sum_{j=1}^n w_j a_j = a \sum_{j=1}^n w_j = a$. Thus, Eq. (11) holds. \square

3.2 Derivation feature weights based on sigmoid function

The Sigmoid function is distinguished by its unique properties, such as monotonicity, continuity, differentiability, and the monotonicity of its inverse function. In addition, the Sigmoid function is characterized by a value range confined within $[0, 1]$ (Ying et al. 2021; Sun et al. 2021). The standard form of the Sigmoid function, expressed as $1/(e^{-x} + 1)$, is notable for its S-shaped curve, which is symmetrically centered at the point $(0, 0.5)$ (Ying et al. 2021). Inspired by Sun et al. (2021), the Sigmoid function is used to derive the feature weight of the MFOWA operator which involves the parameter λ , leading to the construction of the modified Sigmoid function, denoted as sigm^λ , as presented in Eq. (12). The λ allows for a modified curvature of the Sigmoid function, thereby, the modified Sigmoid function offers a more adaptable tool for weight allocation in data aggregation.

$$\text{sigm}^\lambda(r) = \frac{1}{e^{-\lambda(r-0.5)} + 1}, r \in [0, 1], \lambda \in R \tag{12}$$

In Eq. (12), while retaining the S-shaped curve of the traditional Sigmoid function, the sigm^λ function introduces a parameter λ to modulate the sharpness and direction of the weight curve. Notably, the sigm^λ function exhibits central symmetry, passing through the point $(0.5, 0.5)$. The parameter λ is instrumental in determining the orientation of the Sigmoid function curve. When $\lambda > 0$, the sigm^λ function takes on an S shape. Conversely, when $\lambda < 0$, the sigm^λ function takes on an inverted S shape. The magnitude of $|\lambda|$ is critical in controlling how changes in the input

influence the slope of the sigm^λ function curve. A larger $|\lambda|$ results in more pronounced slope changes, rendering the curve akin to a step function. Conversely, a smaller $|\lambda|$ yields more gradual slope changes, closely resembling a line parallel to the x-axis. In the special case, when $\lambda = 0$, the Sigmoid function simplifies to an equal-weight function curve. Furthermore, when $\lambda = n$, the Sigmoid function curve becomes a special instance of the sigm^λ function curve. By adjusting λ , the sigm^λ function can accommodate S-shaped curves with varying curvatures, thereby enhancing information filtering and processing capabilities. The flexibility of the Sigmoid function allows the weight method to be tailored to different application contexts.

Additionally, the sigm^λ function curve, as defined in Eq. (12), is centrally symmetric. To better fit real-world data, and improve the diversity and flexibility of the MFOWA weight method, a translation transformation is introduced into the sigm^λ function. This modification leads to the development of the $\text{sigm}^{\lambda,\theta}$ function, as shown in Eq. (13)

$$\text{sigm}^{\lambda,\theta}(r) = \frac{1}{e^{-\lambda(r-0.5-\theta)} + 1}, r \in [0, 1], \lambda \in \mathbb{R}. \quad (13)$$

Based on Eq. (12), Eq. (13) further adds θ to control the filtering proportion of sample data. The positive or negative value of θ determines the direction of the curve offset center. Assuming λ is positive, whether θ is greater than or less than 0, the $\text{sigm}^{\lambda,\theta}$ function values are increasing. Particularly, when $\lambda = n$ and $\theta = 0$, the $\text{sigm}^{\lambda,\theta}$ function degenerates into the standard Sigmoid function. The magnitude of $|\theta|$ controls the degree of the curve offset from the S-shaped curve. Assuming λ is positive, when $\theta > 0$, the larger $|\theta|$ is, the more the function curve resembles an exponential function curve, and the range of the function will increase accordingly. When $\theta < 0$, the larger $|\theta|$ is, the more the function curve tends towards an equidistant line, and the range of the function will gradually decrease. θ introduces subjective attitudes in weights and allows the $\text{sigm}^{\lambda,\theta}$ function to be adjusted based on the judgment of the important group size. The change of θ can control the degree and balance of information filtering. Thus, $\text{sigm}^{\lambda,\theta}$ function has greater flexibility and applicability.

Compared to Xu's weight method built upon normal distribution (Xu and Da 2003), the weight method using the $\text{sigm}^{\lambda,\theta}$ function can provide linear and nonlinear feature weights for requirements of S-shaped distribution. The new method makes it conducive for the OWA operator family to filter out important information and emphasize the proportion of important information. Therefore, using the $\text{sigm}^{\lambda,\theta}$ function can provide the weight vector with

monotonicity, boundedness, and information filtering capabilities. Inspired by the ideas of Xu and Da (2003), a weight method for the MFOWA operator is proposed based on the $\text{sigm}^{\lambda,\theta}$ function, as shown in Definition 3.

Definition 3 Let $W_{\text{Sigmoid}} = (w_1, w_2, \dots, w_n)^T$ be a weight vector, where $w_i \geq 0$ and $\sum_{i=1}^n w_i = 1$. The weight calculation method of the MFOWA operator based on the Sigmoid function is given by Eq. (14)

$$w_i = \frac{\text{sigm}^{\lambda,\theta}\left(\frac{i}{n}\right)}{\sum_{i=1}^n \text{sigm}^{\lambda,\theta}\left(\frac{i}{n}\right)}, i = 1, 2, \dots, n. \quad (14)$$

In Eq. (14), the calculation of $\text{sigm}^{\lambda,\theta}$ is as shown in Eq. (9). To facilitate control of the function shape, the input is usually normalized to form (i/n) , which range is $[0, 1]$. If the standard Sigmoid is used directly, when the input value is between 0 and 1, the output of Eq. (14) will become a straight line. Therefore, additional parameters are introduced to improve the Sigmoid function formula to solve this problem, as shown in Eq. (13). By adjusting the parameters θ and λ , Eq. (14) can fit the linear and nonlinear weight requirements.

When λ and θ take different values, the $\text{sigm}^{\lambda,\theta}$ function undergoes dynamic changes with the weight vector. Therefore, when solving the W_{Sigmoid} vector using Eq. (14) determining the values of λ and θ adds computing costs, and in practical applications, considerations need to be given to metrics such as Orness and Dispersion. However, for large datasets, the computing cost of Dispersion is higher than that of the Gini index (Breiman et al. 1984). Additionally, the Gini index applies to both discrete and continuous data. Hence, this paper employs Orness and the Gini index to measure the MFOWA weights. Let $W = (w_1, w_2, \dots, w_n)^T$ be a weight vector, and the Gini index of the weight vector W is given by Eq. (15).

$$\text{Gini}(W) = \sum_{i=1}^n \sum_{j=1, i \neq j}^n w_i w_j = 1 - \sum_{i=1}^n w_i^2 \quad (15)$$

It can be easily proved from Eq. (15) that the value range of the Gini index of the weight vector W is from 0 to 1. The closer the value of the Gini index is to 1, the higher the uncertainty within the dataset is, and the closer the value is near 0, the greater the degree of purity is. The Gini index serves as a valuable tool in evaluating the efficacy of weight distribution of the MFOWA operator, and the relative significance of each sample in the population.

In weight optimization of the OWA operator, the objective function proposed by O'Hagan offers substantial insight (O'Hagan 1988). Inspired by O'Hagan, a weight optimization model of the MFOWA is introduced that integrates Orness and the Gini index. The weight objective

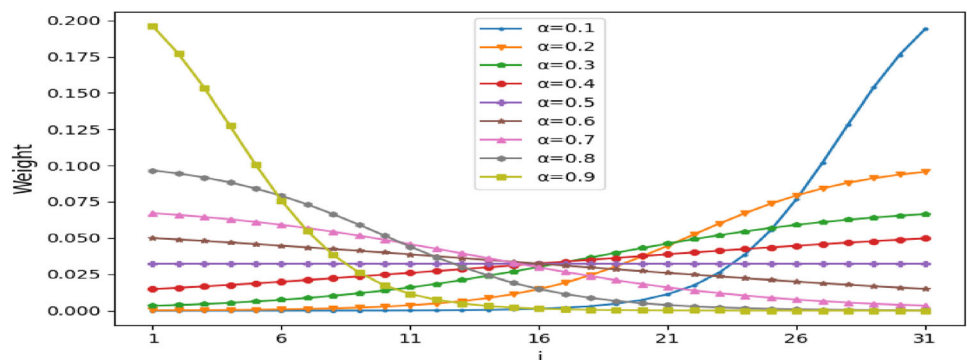
optimization function of the MFOWA operator is defined by the Gini index, which operates within the bounds set by a pre-determined level of Orness, and the weight optimization objective function is encapsulated in the model (M-1).

$$\begin{aligned} &\text{Minimize : } 1 - \sum_{i=1}^n w_i^2 \\ \text{Subject to : } &\frac{1}{n-1} \sum_{i=1}^n (n-i)w_i = \alpha \\ &w_i = \frac{\text{sigm}^{\lambda,\theta} \left(\frac{i}{n} \right)}{\sum_{i=1}^n \text{sigm}^{\lambda,\theta} \left(\frac{i}{n} \right)} \\ &\alpha \in [0, 1], w_i \in [0, 1], i = 1, 2, \dots, n \end{aligned}$$

In the model (M-1), the objective function is to solve the optimal weights w_i that minimize the Gini index for a given Orness measure, denoted as α . The Orness constraint is designed to ensure weight vector conforms to human subjective attitudes and enhances the model’s applicability in scenarios of human-like decision-making processes. The model (M-1) of the weight constraint function incorporates a $\text{sigm}^{\lambda,\theta}$ function, as detailed in Eq. (15). To solve model (M-1) efficiently, this study employs the Sequential Least Squares Quadratic Programming (SLSQP) method (Marques et al. 2021).

In model (M-1), the influence of different α on the weight vector is shown in Fig. 1. The parameter α is closely linked to parameters of weight method and weight measures, and simplifies parameter tuning processes. A smaller absolute difference $|\alpha - 0.5|$ is, the more balanced or neutral the subjective attitude is and the closer the curve is towards a linear representation. In contrast, the larger the value of $|\alpha - 0.5|$ is, the more pronounced or extremely subjective the attitude is and the greater the curvature of the graph is. Consequently, by modulating the α parameter, it is possible to determine the weight change of the MFOWA operator.

Fig. 1 MFOWA weight values at different α levels



4 OWAK-means algorithm based on MFOWA operator

Let x be a sample set comprising of n samples, characterized by m_1 -dimensional features. U is an induced set consisting of n samples of m_2 -dimensional features. The features of U can be derived from features of x (such as density, variance, etc.), or some part features of x . Data set x and induction set U together form the input set X , and the feature dimension of X is m meeting $m = m_1 + m_2$. After the OWAK-Means clustering, k subsets are obtained, represented as $C = \{C_1, C_2, \dots, C_k\}$. The following will detail the improvement in centroid initialization, distance metric, and centroid calculation in the OWAK-Means Algorithm.

4.1 Initialization centroids of OWAK-means algorithm

In real life, multi-feature data frequently have certain partial-order relationships (Wang et al. 2016). Random selection of the initial centroid of K-Means is not good for centroid calculation and membership assignment, while fixed selection is unsuitable for datasets with multi-feature and fails to consider the relationship between features. Therefore, given the distribution properties of partially ordered data, experts could evaluate the position of k initial centroid according to the distribution of observed data, and give the weight of $k - 1$ centroid intervals. The initial centroid selection method for OWAK-Means which considers partial-order relationship, is shown in Definition 4.

Definition 4 Let $\{x_1', x_2', \dots, x_n'\}$ be an order set, which is obtained by sorting the original set $\{x_1, x_2, \dots, x_n\}$ and satisfies the partial-order relationship. The expert weight vector is denoted as $(\varepsilon_1, \varepsilon_2, \dots, \varepsilon_{k-1})^T$. The computation of the initial centroid vector $(\mu_1, \mu_2, \dots, \mu_k)$ for the OWAK-Means algorithm, which is based on partial-order relationships, is shown in Eq. (16) and Eq. (17).

$$\mu_i = x_{id(i)'} \tag{16}$$

$$\text{id}(i) = \begin{cases} 1, & i = 1 \\ \text{id}(i-1) + \lfloor \varepsilon_{i-1}(n-1) \rfloor, & 2 \leq i \leq k \end{cases} \quad (17)$$

In Eq. (16), μ_i is defined as the i -th initial centroid derived from the partial-order set $\{x'_1, x'_2, \dots, x'_n\}$. The function id is an iterative function used to determine the index position of the i -th initial centroid in a partial-order set, as elaborated in Eq. (17). As can be seen from Eq. (17), the initial value of the id function is 1 and the index position of each ensuing centroid is determined by adding the current index position to the interval calculated based on expert-assessed. This interval is calculated as the product of $(n-1)$ and expert weights, with the result rounded down. The selection method of the initial centroid reflects the experts' estimation of the difference in index positions between consecutive centroids. The weights ε_i are evaluated considering the partial-order relationships and distribution characteristics of the dataset and represent the fraction of the interval between initial centroids relative to the total number of samples in the set. Thus, the initial centroid vector is an ordered vector. When $\varepsilon_1 = \varepsilon_2 = \dots = \varepsilon_{k-1}$, the index obtained is equivalent to the index extracted by the equidistant sampling method.

The initialization centroid method proposed in this paper adopts a deterministic approach based on index sorting, which avoids the problem of unstable results caused by the random selection of initial centroids. On the other hand, the weights introduced initialization centroid are constructed based on partial-order density, which has certain subjective preferences. Consequently, in practical implementations, it is recommended to engage individuals possessing extensive experience and professional expertise in evaluating partial-order density, to improve accuracy and reliability.

4.2 OWAK-means clustering algorithm

To improve the performance of the K-Means algorithm, the proposed OWAK-Means algorithm uses the MFIOWA operator, which employs the Sigmoid weight method to find the prime centroid vectors and integrates into Eq. (3) to calculate the distances between each point and its centroid vector. The OWAK-Means proceeds through iterations until it reaches convergence. Let $C_j (j = 1, 2, \dots, k)$ denote a cluster in the clustering process, and $D = (x_1, x_2, \dots, x_{|C_j|})$ represents the clustered set. Correspondingly, $(U_1, U_2, \dots, U_{|C_j|})$ represent the induced set. The update method for the centroid vector in the OWAK-Means Algorithm, based on the MFIOWA operator, is shown in Eq. (18).

$$\mu^l = \text{MFIOWA}((U_i, x_i) | (U_i, x_i) \in C_j) = W_S^T X^l \quad (18)$$

In Eq. (18), X^l is defined as an array comprising j -th largest elements x_i , which is determined following the application of multi-feature-induced order using U_i . The weight vector W_S^T is solved by the (M-1) model. In this context, $W_S^T X^l$ represents the vector form of the updated formula for centroids. The new centroid calculation method can reduce the influence of sample noise or uneven distribution by filtering the samples in the cluster with multi-feature induced factors.

In the K-Means algorithm, the distance function will significantly affect the performance of clustering results. Inspired by the OWD_r distance metric (Pons-Vives et al. 2022), the squaring distance is introduced in this paper to enhance the distance difference between different points. Let x and y be two n -dimension vectors, and $(w_1, w_2, \dots, w_n)^T$ be a weight vector meeting $w_i \geq 0$ and $\sum_{i=1}^n w_i = 1$. The weighted squaring distance metric is shown as Eq. (19).

$$\text{OWD}(x, y) = \sum_{j=1}^n w_j d_{rs}(x_{(j)}, y_{(j)}), \quad (19)$$

where x, y are feature vectors with $x = (x_1, x_2, \dots, x_n)$, $y = (y_1, y_2, \dots, y_n)$, $d_{rs}(x_{(j)}, y_{(j)})$ denotes the j -th distance element in the distance set $\{d_{rs}(x_1, y_1), \dots, d_{rs}(x_n, y_n)\}$. And $d_{rs} : R \times R \rightarrow [0, 1]$ is the squaring distance given by Eq. (20).

$$d_{rs}(a, b) = \begin{cases} (a - b)^2 / (\max(a, b))^2, & \text{otherwise} \\ 0, & a = b = 0 \end{cases} \quad (20)$$

Observe that the input vectors x and y represent the data instances involved in the clustering process. Moreover, like the same as OWD_r distance metric, OWD distance is sensitive to differences in the scales of the variables. Subsequently, by replacing μ^l and OWD distance metric in Eq. (3), the optimization objective function of OWAK-Means is presented in Eq. (21).

$$D = \sum_{j=1}^k \sum_{i=1}^n \sum_{v=1}^{m_1} s_{ij} w_v d_{rs}(x_{iv}, (W_{S_j}^T X'_j)_v), \quad (21)$$

where d_{rs} is shown as Eq. (20). In Eq. (21), the $\mu'_{jv} = (W_{S_j}^T X'_j)_v$ represents the v -th feature values of the j -th centroid. x_{iv} denotes the v -th feature value of the i -th sample in the sample set. The OWAK-Means algorithm, as outlined in Table 2, initiates with the construction of the input set X with two subsets: the sample set x , containing m_1 features, and the induced set U encompassing m_2 features. This paper selects the sample features from the original datasets, and the induced features are obtained by calculating based on multi-density levels. The process of calculating multi-level density is as follows. First, calculate the mean point of sample set x and obtain the distance

Table 2 OWAK-means algorithm

Description of OWAK-Means Algorithm	
Input:	Let $x = \{x_{ij} i = 1, 2, \dots, n, j = 1, 2, \dots, m_1\}$ be the sample set, $(\varepsilon_1, \varepsilon_2, \dots, \varepsilon_{k-1})$ be the expert weight vector and k be the number of clusters. α is a weight parameter of the MFIOWA operator.
1:	Construct the m -dimension input set X with the sample set $x = \{x_i\}$ and the density-based induced set $U = \{U_i\}$, where x contains m_1 dimensions of features, U contains m_2 dimensions of features, and $m = m_1 + m_2$.
2:	Apply multi-feature sorting to $\{x_i\}$ to obtain the ordered set $\{x'_i\}$, and use Eq. (16) and Eq. (17) to select the initial centroid vectors $\{\mu_j j = 1, 2, 3, \dots, k\}$.
3:	Repeat
4:	Set $C_j = \emptyset$ for the cluster corresponding to the centroid vector $\mu_j (j \in [1, k])$.
5:	for $i = 1, 2, \dots, n$ do
6:	Calculate the OWD distance between x_i and μ_j for $j \in [1, k]$ using Eqs. (19) and (20).
7:	Determine the cluster label for (U_i, x_i) : $label_i = argmin_{j \in [1, k]} d_{ij}$, based on the nearest centroid vector μ_j .
8:	Assign the sample data (U_i, x_i) to the corresponding cluster $C_{label_i} = C_{label_i} \cup \{(U_i, x_i)\}$.
9:	end for
10:	for $j = 1, 2, \dots, k$ do
11:	Set $\frac{1}{ C_j }$ and α into the (M-1) model to solve for the weight vector W_S^T . Substitute W_S^T into Eq. (18) to compute the new centroid vector: $\mu'_j = MFIOWA(\{ \langle U, x \rangle x \in C_j \})$.
12:	if $\mu'_j \neq \mu_j$ then update μ_j to μ'_j , else keep the current centroid vector μ_j unchanged. end if
13:	end for
14:	Until the objective function in Eq. (21) is minimized or all centroid vectors remain unchanged.
Output:	The partition of clusters is $C = \{C_1, C_2, \dots, C_k\}$, where $C_x \cap_{x \neq y} C_y = \emptyset$ and $D = \cup_{i=1}^k C_j$.

between the farthest point and the mean point as the baseline distance (*base*). Secondly, given the truncation rate σ and the number of levels m_2 , the critical distances for different density levels are calculated with the formula: $(\sigma \cdot base \cdot 1/m_2, \dots, \sigma \cdot base \cdot l/m_2)$, where $l \in [1, m_2]$. Last, with each point as the center, a circle with the radius of l -th critical distance $\sigma \cdot base \cdot l/m_2$ is constructed, and the number of sample points existing in the circle is calculated as the density of the point under the l -th critical distance. The data of multi-density features are constructed into induced set U . The number of points around each point within different critical distances as the induced set for that point is calculated. In the experiment in Sect. 5, the values of σ and m_2 are fixed after, where $\sigma = 0.1$ and $m_2 = 5$. Following this, a multi-feature sorting is applied to x , and

the initial centroid vector μ_j is computed based on the ordered set and expert-assigned weights. Cluster C_j is initially established as an empty set. The algorithm then assigns each sample point to its corresponding cluster based on the OWD distance metric between μ_j and x_j . Utilizing the MFIOWA operator, which incorporates both clustering and induced features of the cluster members, a new centroid vector μ'_j is determined. The necessity of updating the centroid vector is evaluated by comparing the OWD distance between μ'_j and μ_j . The iterative process of updating centroids continues until the objective function reaches its minimum value.

The introduction of MFIOWA increases the characteristics that can be used for clustering tasks. The sample data set is used to calculate the distance and determine

memberships, while the induced set is used to filter interference and noise data. Multi-feature-induced order rules support the sorting of continuous data and discrete data, and support custom sorting rules for non-numerical data so that non-continuous features participate in the iterative process of clustering centroid. MFOWA influences the updating process of cluster centroid through feature induction and order-weighted aggregation, so that there is a certain preference in determining cluster centroid, and the degree of preference is determined by the parameter α of the Sigmoid weight model (M-1). In the algorithm proposed in this paper, this preference is expressed in the retention of data points with a certain degree of density. Utilizing the weight method from the (M-1) model, the parameter α can be manipulated to alter the direction, form, and equilibrium of the weight curve. As delineated in Sect. 3.2, an $\alpha < 0.5$ imparts a pessimistic bias through W_S^T filtering information from lower rankings. Consequently, the calculation results of MFOWA exhibit an upward bias relative to the WA operator, leading to a consistent positive bias in each cluster centroid calculation. In contrast, when $\alpha > 0.5$, W_S^T adopts an optimistic bias, emphasizing lower positioned information. This results in the MFOWA's calculations being downwardly biased compared to the WA operator, thereby introducing a negative bias in the centroid calculations. The application of the S-shaped weight method within the MFOWA operator framework enables sophisticated filtering and screening of data during the iterative centroid vector update process, in conjunction with induced features. The α parameter affords subjective control over the cluster centroids, ensuring that each iteration strategically adjusts centroid calculations, thus enhancing the overall clustering performance and accuracy. We will analyze the relationship between parameters and the OWAK-Means algorithm in Sect. 5.

The OWAK-Means distinct from the conventional K-Means algorithm, the proposed OWAK-Means algorithm necessitates the input of expert weights $(\varepsilon_1, \varepsilon_2, \dots, \varepsilon_{k-1})$ and a subjective attitude parameter α . The expert weights $(\varepsilon_1, \varepsilon_2, \dots, \varepsilon_{k-1})$ ensure the rationality of the initial centroid positions in OWAK-Means. Given the Orness measure α , the optimal weights for the MFOWA operator can be determined, thereby inducing the optimization of centroids in OWAK-Means. This additional input expedites centroid convergence during the iterative clustering process, culminating in final clustering outcomes reflecting subjective judgment and objective data patterns.

5 OWAK-means algorithm implementation

5.1 Datasets and evaluation indicators

The experimental study employed eight real-world datasets from the UCI, Wine, Iris, Drug, Obesity, Customer Segment, Heart, Balance, and User. The basic characteristics of these datasets are succinctly summarized in Table 3, which includes information on the sample size, the number of feature dimensions, and the number of target categories for classification. Notably, the Customer Segment dataset, with sample sizes of 8068, is a multi-sample dataset. Furthermore, the Wine and Heart datasets have 13 features. The scope of classification within these datasets is diverse, encompassing categories ranging from 2-class to 5-class.

To assess the performances of the clustering algorithm, three metrics are employed: adjusted rand score (ARS) (Khan et al. 2019), Normalized Mutual Information (NMI), and purity (PUR) (Huang et al. 2021). The ARS ranges from -1 to 1 , where values approaching 1 signify superior clustering performance. The NMI spans from 0 to 1 , with values nearing 1 indicating more precise clustering outcomes, while values closer to 0 suggest results akin to random clustering. Lastly, a greater value in PUR is indicative of more effective clustering performance.

5.2 Analysis of implementation result

The proposed OWAK-Means algorithm is compared with seven clustering algorithms: K-Means (MacQueen 1967), K-Means+ (Arthur and Vassilvitskii 2007), Mini Batch K-Means (Sculley 2010), Bisecting K-Means (Steinbach et al. 2000), WK-Means (Chen et al. 2012), MWK-Means (De Amorim and Mirkin 2012), and K-Means (OWDr) (Pons-Vives et al. 2022). In the implementation phase, the OWAK-Means uniquely partitioned the multi-feature data of each dataset into sample sets and induced sets. In contrast, the other clustering algorithms utilized only attribute

Table 3 Basic characteristics of the datasets

Item	Dataset	Samples	Features	Class
1	Wine	178	13	3
2	Iris	150	4	3
3	Drug	200	5	5
4	Obesity	108	5	4
5	Customer segment	8068	9	4
6	Heart	270	13	2
7	Balance	625	4	3
8	User	258	5	4

sets for the clustering process. To ensure the reliability of the experimental results and prevent accidental results from being seen as final, each algorithm is subjected to 50 iterations of experimentation. The average of multiple experiments is used as the result. To effectively assess the performance of these methodologies, the Technique for Order of Preference by Similarity to the Ideal Solution (TOPSIS) based on the entropy weight method is used to generate a comprehensive score and ranking based on the mean results. Assuming a scenario with n objects, each characterized by m attributes, the TOPSIS method, underpinned by the entropy weight method, entails the following procedural steps: (1) Standardization to procure a matrix, denoted as $Z = \{z_{ij}\}$. (2) Computation of information entropy for each indicator: $e_j = -\frac{\sum_{i=1}^n p_{ij} \ln(p_{ij})}{\ln(n)}$, where $p_{ij} = z_{ij} / \sqrt{\sum_{i=1}^n z_{ij}^2}$. And obtain the weight for each indicator: $W_j = (1 - e_j) / (\sum_{j=1}^m (1 - e_j))$. (3) Calculate the distance to the positive ideal solution: $D_i^+ = \sqrt{\sum_{j=1}^m W_j (Z_j^+ - z_{ij})^2}$, and the distance to the negative ideal solution: $D_i^- = \sqrt{\sum_{j=1}^m W_j (Z_j^- - z_{ij})^2}$, where the positive and negative ideals are defined as $Z^+ = \{Z_j^+ = \max(z_{ij})\}$ and $Z^- = \{Z_j^- = \min(z_{ij})\}$, respectively. (4) Calculate the comprehensive score as $S_i = D_i^- / (D_i^+ + D_i^-)$. Where $i \in [1, n], j \in [1, m]$.

Comprehensive performance comparison results of different K-Means algorithms are given in Table 4, where italic represents the best method for a given dataset and a certain metric. And the Friedman test and Nemenyi test (Ma et al. 2022) results are conducted, as shown in Table 5 and Fig. 2. Stability performance analysis is given in Fig. 3 which details variance based on uncertainty assessment (Abbaszadeh Shahri et al. 2022) for ARS, NMI, and PUR. An Outlier comparison of ARS, NMI, and PUR is given in Fig. 4. Z-Score methodology is used to evaluate the outlier, and the outlier threshold is established at 2. If the Z-Score of ARS, NMI, and PUR exceeds 2, the evaluating index is classified as an outlier. Robustness analysis is given in Fig. 5 which shows the performance of each algorithm under noise.

From Table 4, it can be revealed that the OWAK-Means algorithm exhibits superior overall performance. Detailed observations are as followed. (1) For the ARS metric, OWAK-Means is better than the second-best clustering algorithm on the Wine, Iris, Drug, Customer Segment, Heart, Balance, and User datasets, exhibiting improvements of 0.0159 (1.9487%), 0.1258 (16.5589%), 0.0036 (4.9557%), 0.0051 (4.6460%), 0.1535 (100.7730%), 0.0047 (3.3107%), 0.1732 (69.5556%), respectively. (2) Regarding the NMI metric, OWAK-Means outperforms on

the Wine, Iris, Drug, Obesity, Customer Segment, Heart, Balance, and User datasets, and closely approaches the optimal results on the Drug dataset. (3) For PUR metric, OWAK-Means ranks highest on all datasets and shows improvements over the second-best clustering algorithm of 0.0055 (0.5868%), 0.0629 (7.0155%), 0.0167 (3.2221%), 0.0096 (1.4254%), 0.0151 (3.4579%), 0.0800 (11.4650%), 0.0121 (1.8379%), 0.0029 (0.4927%), respectively. (4) In terms of TOPSIS comprehensive evaluating result, OWAK-Means consistently ranks first, surpassing K-Means, K-Means++, Mini Batch K-Means, Bisecting K-Means, WK-Means, MWK-Means, and K-Means (OWDr) algorithms.

To further evaluate whether the OWAK-Means has significant advantages over other algorithms, the Friedman test and Nemenyi test are used to evaluate whether the ranks of the methods significantly differ or not. The Friedman test statistics value F_F and the corresponding critical values for each evaluation criterion are shown in Table 5. In Table 5, k is the number of comparing methods and N is the number of datasets. Taking a significance level of $\varphi = 0.05$, the null hypothesis that the compared methods perform equally is rejected for all evaluation criteria.

The Nemenyi test further investigates whether each of the methods performs equally well against the others. The performance of the two methods significantly differs if the difference in their average is over the critical difference $CD = q_\varphi \sqrt{\frac{k(k+1)}{6N}}$. q_φ is 3.0310 and the CD is calculated as 3.7122 ($k = 8, N = 8$) for $\varphi = 0.05$. The CDs are outlined in Fig. 2. The compared methods with average ranks within a CD to that of OWAK-Means are covered by a red line. In other words, uncovered methods thus have a significantly worse performance than OWAK-Means.

By looking at ARS, for example, the average rank for OWAK-Means is 1.25 and the critical value is 4.281 after adding the CD. Given that the average ranks of K-Means, K-Means++, Mini Batch K-Means, Bisecting K-Means, WK-Means, and MWK-Means, are than 4.281, they are classified as worse methods in this case. However, there is no statistical evidence to assert that OWAK-Means outperforms the rest of the compared methods under ARS. The same applies to NMI and PUR.

From Fig. 3, it can be revealed that the OWAK-Means algorithm demonstrates a stable result and lower uncertainty, with the smallest mean standard deviation when the OWAK-Means algorithm compared to K-Means, K-Means++, Mini Batch K-Means, Bisecting K-Means, WK-Means, MWK-Means, and K-Means (OWDr) algorithms. This observation suggests that OWAK-Means superior stability and reliability during the clustering process.

Table 4 Performance comparison of variant K-Means clustering algorithms (50 experiments)

Datasets	Algorithms	Evaluation metrics			TOPSIS score	TOPSIS rank
		ARS	NMI	PUR		
Wine	Proposed method	0.8309	0.8229	0.9438	1.0000	1
	K-Means (MacQueen 1967)	0.3711	0.4288	0.7022	0.0451	5
	K-Means++ (Arthur and Vassilvitskii 2007)	0.3711	0.4288	0.7022	0.0451	5
	Mini Batch K-Means (Sculley 2010)	0.3645	0.4289	0.6930	0.0384	6
	Bisecting K-Means (Steinbach et al. 2000)	0.3873	0.3971	0.7183	0.0654	3
	WK-Means (Chen et al. 2012)	0.3636	0.4255	0.6958	0.0350	7
	MWK-Means (De Amorim and Mirkin 2012)	0.3709	0.4305	0.7026	0.0472	4
	K-Means (OWDr) (Pons-Vives et al. 2022)	0.8150	0.8150	0.9383	0.9734	2
Iris	Proposed method	0.8857	0.8642	0.9600	1.0000	1
	K-Means (MacQueen 1967)	0.7302	0.7582	0.8933	0.4470	5
	K-Means++ (Arthur and Vassilvitskii 2007)	0.7302	0.7582	0.8933	0.4470	5
	Mini Batch K-Means (Sculley 2010)	0.7378	0.7553	0.8971	0.4546	4
	Bisecting K-Means (Steinbach et al. 2000)	0.6772	0.6892	0.8713	0.2258	6
	WK-Means (Chen et al. 2012)	0.7401	0.7703	0.8901	0.4832	3
	MWK-Means (De Amorim and Mirkin 2012)	0.6160	0.6972	0.8052	0.0301	7
	K-Means (OWDr) (Pons-Vives et al. 2022)	0.7599	0.7912	0.8797	0.5424	2
Drug	Proposed method	0.0757	0.2319	0.5350	0.9939	1
	K-Means (MacQueen 1967)	0.0579	0.2309	0.5038	0.6060	4
	K-Means++ (Arthur and Vassilvitskii 2007)	0.0601	0.2329	0.5054	0.6328	3
	Mini Batch K-Means (Sculley 2010)	0.0487	0.1865	0.5008	0.4081	5
	Bisecting K-Means (Steinbach et al. 2000)	0.0299	0.1417	0.4843	0.0159	7
	WK-Means (Chen et al. 2012)	0.0285	0.1434	0.4842	0.0097	8
	MWK-Means (De Amorim and Mirkin 2012)	0.0401	0.1813	0.4904	0.2716	6
	K-Means (OWDr) (Pons-Vives et al. 2022)	0.0721	0.1944	0.5183	0.7020	2
Obesity	Proposed method	0.2952	0.4637	0.6852	0.9253	1
	K-Means (MacQueen 1967)	0.3150	0.4059	0.6750	0.7580	3
	K-Means++ (Arthur and Vassilvitskii 2007)	0.3133	0.4076	0.6756	0.7645	2
	Mini Batch K-Means (Sculley 2010)	0.2695	0.3724	0.6239	0.3807	5
	Bisecting K-Means (Steinbach et al. 2000)	0.2496	0.3306	0.6057	0.2015	7
	WK-Means (Chen et al. 2012)	0.2946	0.4064	0.6483	0.6228	4
	MWK-Means (De Amorim and Mirkin 2012)	0.2672	0.3603	0.6239	0.3513	6
	K-Means (OWDr) (Pons-Vives et al. 2022)	0.1944	0.3688	0.5991	0.1529	8
Customer Segment	Proposed method	0.1141	0.1141	0.4504	1.0000	1
	K-Means (MacQueen 1967)	0.1042	0.0943	0.4219	0.5125	6
	K-Means++ (Arthur and Vassilvitskii 2007)	0.1048	0.0944	0.4217	0.5162	5
	Mini Batch K-Means (Sculley 2010)	0.1018	0.0932	0.4188	0.4578	7
	Bisecting K-Means (Steinbach et al. 2000)	0.1090	0.0959	0.4207	0.5519	3
	WK-Means (Chen et al. 2012)	0.0834	0.0752	0.4014	0.0000	8
	MWK-Means (De Amorim and Mirkin 2012)	0.1062	0.0949	0.4200	0.5175	4
	K-Means (OWDr) (Pons-Vives et al. 2022)	0.1064	0.0985	0.4354	0.6753	2
Heart	Proposed method	0.3059	0.2313	0.7778	1.0000	1
	K-Means (MacQueen 1967)	0.0302	0.0199	0.5925	0.0064	7
	K-Means++ (Arthur and Vassilvitskii 2007)	0.0302	0.0199	0.5926	0.0067	6
	Mini Batch K-Means (Sculley 2010)	0.0338	0.0229	0.5970	0.0238	5
	Bisecting K-Means (Steinbach et al. 2000)	0.0289	0.0191	0.5907	0.0000	8
	WK-Means (Chen et al. 2012)	0.0684	0.0477	0.6284	0.1595	3
	MWK-Means (De Amorim and Mirkin 2012)	0.0436	0.0309	0.6081	0.0681	4

Table 4 continued

Datasets	Algorithms	Evaluation metrics			TOPSIS score	TOPSIS rank
		ARS	NMI	PUR		
Balance	K-Means (OWDr) (Pons-Vives et al. 2022)	0.1523	0.2060	0.6978	0.6192	2
	Proposed method	<i>0.1466</i>	<i>0.1592</i>	<i>0.6720</i>	<i>1.0000</i>	<i>1</i>
	K-Means (MacQueen 1967)	0.1419	0.1229	0.6599	0.6619	2
	K-Means + + (Arthur and Vassilvitskii 2007)	0.1321	0.1143	0.6522	0.5288	5
	Mini Batch K-Means (Sculley 2010)	0.1267	0.1112	0.6438	0.4345	7
	Bisecting K-Means (Steinbach et al. 2000)	0.0985	0.0888	0.6214	0.0000	8
	WK-Means (Chen et al. 2012)	0.1312	0.1109	0.6507	0.4970	6
User	MWK-Means (De Amorim and Mirkin 2012)	0.1386	0.1208	0.6578	0.6265	4
	K-Means (OWDr) (Pons-Vives et al. 2022)	0.1405	0.1226	0.6588	0.6505	3
	Proposed method	<i>0.4222</i>	<i>0.5201</i>	<i>0.6008</i>	<i>1.0000</i>	<i>1</i>
	K-Means (MacQueen 1967)	0.1806	0.2478	0.5805	0.2514	3
	K-Means++ (Arthur and Vassilvitskii 2007)	0.1840	0.2563	0.5731	0.2045	5
	Mini Batch K-Means (Sculley 2010)	0.1846	0.2560	0.5560	0.0854	6
	Bisecting K-Means (Steinbach et al. 2000)	0.2095	0.2908	0.5725	0.2508	4
	WK-Means (Chen et al. 2012)	0.2490	0.3413	0.5978	0.4829	2
	MWK-Means (De Amorim and Mirkin 2012)	0.1868	0.2579	0.5522	0.0812	7
	K-Means (OWDr) (Pons-Vives et al. 2022)	0.1749	0.2126	0.5539	0.0167	8

Table 5 Summary of the Friedman Statistics F_F ($k = 8, N = 8$) and the critical value

Evaluation criteria	F_F	Critical value ($\varphi = 0.05$)
ARS	19.8090	14.0671
NMI	25.1582	
PUR	26.3572	

From Fig. 4, it can be indicated that among the eight datasets, the OWAK-Means algorithm records the lowest number of outlier results across the evaluation metrics. In comparison, the rest algorithms show a higher prevalence of outlier value results, especially in Drug, Obesity, Balance, and User datasets, where OWAK-Means registers fewer outliers than the other seven algorithms. This disparity is particularly pronounced in multi-featured and

large-sample datasets such as Obesity, Customer Segment, and Heart datasets. These results show that the OWAK-Means algorithm has better stability and reliability.

From Fig. 5, after randomly increasing 5% uniform noise to simulate extreme values and outliers, OWAK-Means algorithm remained advantages in ARS, NMI, and PUR metrics. It shows that the OWAK-Means algorithm has better robustness under the influence of noise. For ARS metric, the OWAK-Means algorithm is significantly better than the second-best algorithm in the Wine, Drug, and User datasets, exhibiting improvement of 0.1164 (19.09%), 0.0534 (64.48%) and 0.1036 (32.98%), respectively. As for the NMI metric, OWAK-Means outperforms on eight datasets. For PUR metric, except User and Customer Segment datasets, OWAK-Means ranks top and shows improvements over the second-best algorithm. The noise will affect the determination of centroids, which deviates from the real centroid. OWAK-Means filters noise by

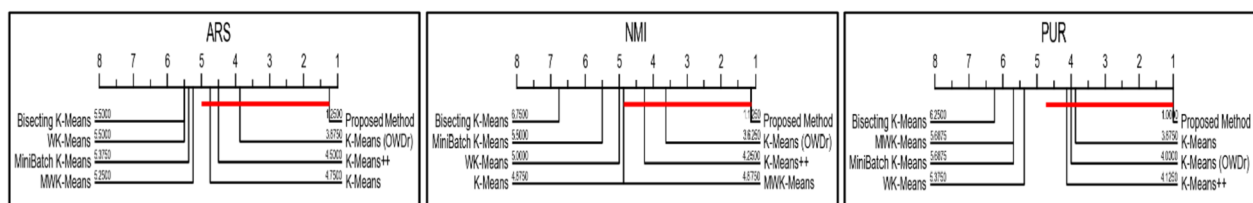


Fig. 2 Nemenyi test of variant K-Means algorithms

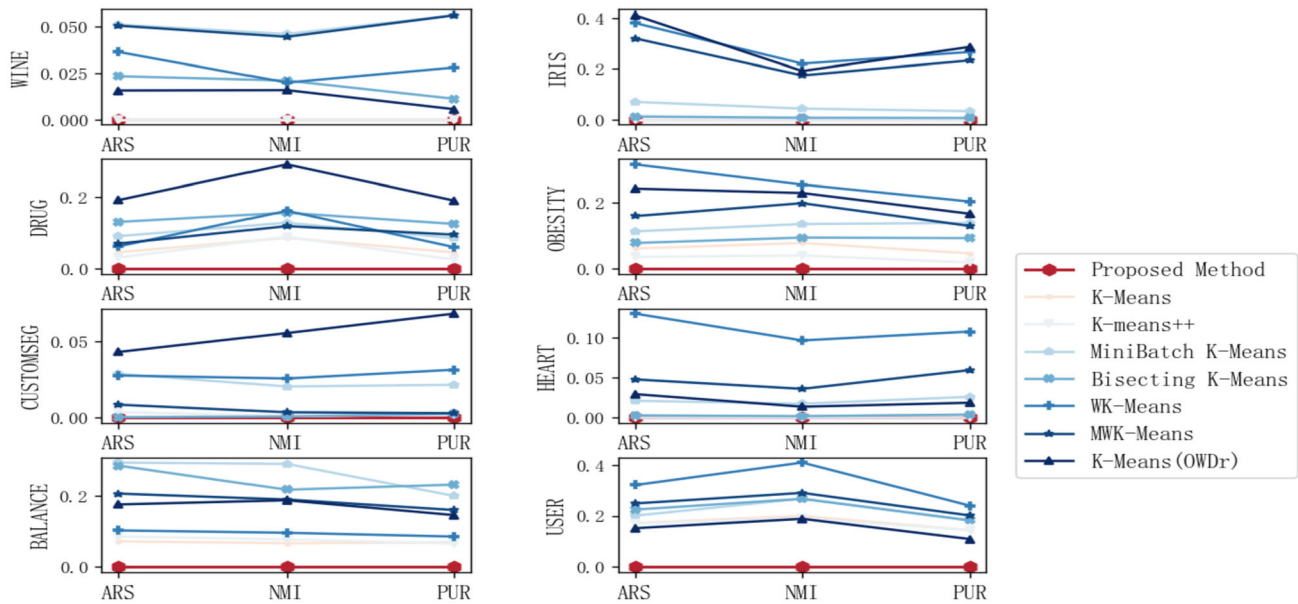


Fig. 3 Stability comparison of variant K-Means algorithms on eight datasets

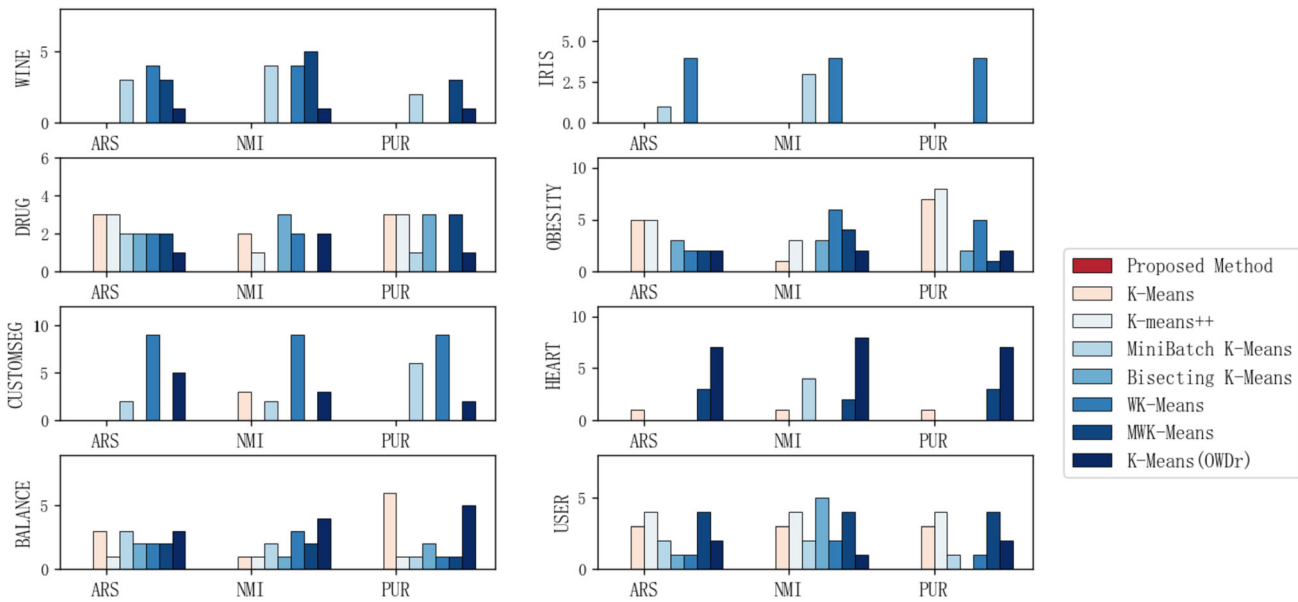


Fig. 4 Outliers comparison of variant K-Means algorithms

MFIOWA operator and Sigmoid weights according to density metric, thus it can be well adapted to the noise, especially in dealing with noise at cluster edges. These results show that the OWAK-Means algorithm has better robustness.

Through comparative analysis, the proposed OWAK-Means algorithm has better comprehensive performance, stability, and robustness. But there are some limitations and difficulties. Multiple subjective parameters are involved in OWAK-Means, which increases the cost of decision-making for experts. In the part of centroid initialization, the

OWAK-Means clustering algorithm based on expert weight and partial-order relationship has high stability, but it also brings the problem that may fall into a local optimal solution. In MFIOWA, this paper takes multi density levels as the induced factors to explore dense-induced in-cluster sample filtering and centroid calculation. The value of the critical distance and the number of density levels used to calculate multi-density has a great impact on MFIOWA, and this part is not related to the data distribution and integrated into the algorithm. In this experiment, the feature weight optimization was not carried out, and the

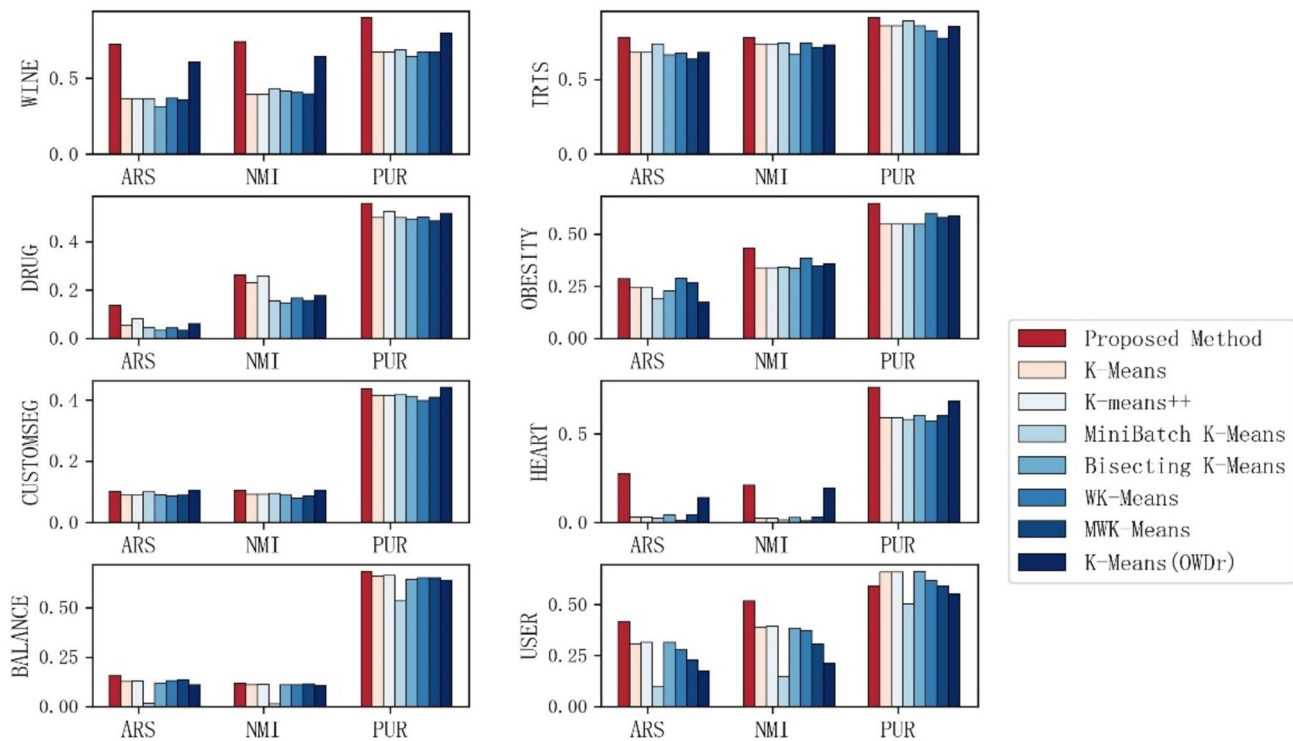


Fig. 5 Robustness comparison of variant K-Means algorithms

feature weights of the OWD distance metric were set as equal weights. The OWAK-Means algorithm proposed in this paper still has a large space for improvement, and we will continue to explore it in future research.

5.3 Sensitivity analysis of OWAK-Means algorithm

To further explore the impact of the parameter settings of MFIOWA’s weighting method on clustering results, this study conducted a sensitivity analysis on the parameter α . Multiple experiments were conducted on OWAK-Means at different α levels, and the results were averaged to avoid the occurrence of random outcomes, as shown in Fig. 6. In OWAK-Means, the multi-feature-induced order rule of MFIOWA is in ascending multi-feature partial order. As mentioned in Sect. 3.2, the weight values exhibit an ascending order, indicating that OWAK-Means places more emphasis on high-density samples. Conversely, for OWAK-Means focusing on low-density samples, the weight values exhibit a descending order. As shown in Fig. 6, different data samples have different data distributions and densities, and the optimal parameter values are also different. The datasets Wine, Iris, and Customer Segment show minor variations. Obesity, Balance, and Heart datasets exhibit obvious optimal regions, indicating that the datasets have uneven sample distributions. The result of the User dataset shows a clear monotonicity,

suggesting that high-density samples better reflect the concentration of data. From Fig. 6, it can be observed that the changing trends of all indicators are consistent, with many optimal solutions concentrated in the ranges of 0.3 to 0.5 and 0.5 to 0.7. This suggests that appropriately filtering intra-cluster members can effectively enhance the accuracy, feasibility, and reliability of the clustering algorithm. However, the relationship between the selection of density-induced features and the characteristics of the dataset is not well-reflected, and this is a focus for future research.

6 Conclusion

This paper introduces the OWAK-Means algorithm, which incorporates a new initial centroid optimization method and the newly proposed MFIOWA operator. The initial centroid optimization method is designed based on partial-order relationships of multiple features. The MFIOWA operator, developed from the existing IOWA operator, integrates concepts from database index theory and multi-feature induced order. Furthermore, we construct an objective optimization model based on the Gini index and Orness measure to determine the feature weights for the MFIOWA operator using a modified Sigmoid function. This approach reduces tuning costs while enhancing flexibility and variability.

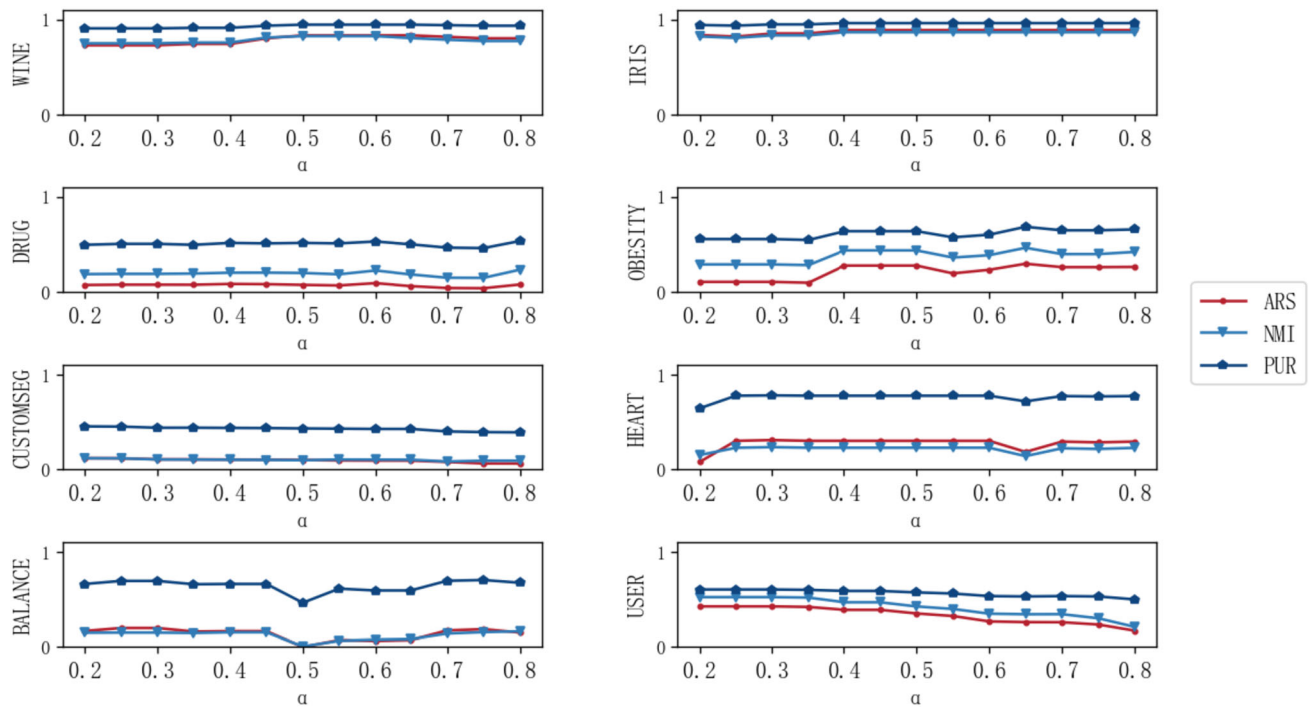


Fig. 6 Performance comparison of OWAK-Means at different α levels

Comparative analysis with state-of-the-art clustering algorithms reveals that OWAK-Means outperforms in critical metrics such as ARS, NMI, and PUR on eight real-world datasets from the UCI repository. Comprehensive evaluation using the TOPSIS method consistently ranks OWAK-Means as the top performer. Significance through the Friedman and Nemenyi tests and stability assessment based on variance and outlier further affirm OWAK-Means' superiority over other K-Means algorithm variants. Additionally, the comparative experiment with adding noise shows that OWAK-Means has better robustness depending on the information filtering ability from the MFOWA operator and Sigmoid weights. Therefore, the proposed OWAK-Means algorithm demonstrates superior performance in key metrics such as the ARS, NMI, and PUR, along with robustness and stability.

While experimental validation on eight multi-feature UCI datasets confirms OWAK-Means' high performance, future research will further explore OWAK-Means' adaptability across a broader spectrum of datasets and examine the performance of clustering algorithms under different dataset distribution scenarios. The integration of centroid initialization, feature weight optimization method, and automatic selection of cluster k based on the IOWA operator will be further studied.

Acknowledgements This work is supported in part by the Humanities and Social Science Planning Project of the Ministry of Education under Grant 22YJA880051, the Science and Technology Project of Jiangxi Provincial Education Department under Grant GJJ2200535,

and the 18th Student Research Project of Jiangxi University of Finance and Economics under Grant 20231016140352946.

Author contributions Benting Wan: Conception and design of study, Acquisition of data, Analysis and/or interpretation of data, Writing—original draft, Writing—review & editing. Weikang Huang: Analysis and/or interpretation of data, Writing—original draft, Writing—review & editing. Bilivogui Pierre: Analysis and/or interpretation of data, Writing—review & editing. Youyu Cheng: Writing—review & editing. Shufen Zhou: Analysis and/or interpretation of data. Writing—review & editing.

Data availability Data will be made available on request.

Declarations

Conflict of interest All authors declare that they have no conflict of interest.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

References

- Abbaszadeh Shahri A, Shan C, Larsson S (2022) A novel approach to uncertainty quantification in groundwater table modeling by automated predictive deep learning. *Nat Resour Res* 31:1351–1373. <https://doi.org/10.1007/s11053-022-10051-w>
- Arthur D, Vassilvitskii S (2007) K-means++: the advantages of careful seeding. In Proceedings of the eighteenth annual ACM-SIAM symposium on discrete algorithms, pp 1027–1035
- Asheghi R, Hosseini SA, Saneie M, Shahri AA (2020) Updating the neural network sediment load models using different sensitivity

- analysis methods: a regional application. *J Hydroinf* 22:562–577. <https://doi.org/10.2166/hydro.2020.098>
- Askari S (2021) Noise-resistant fuzzy clustering algorithm. *Granul Comput* 6:815–828. <https://doi.org/10.1007/s41066-020-00230-6>
- Ay M, Özbakır L, Kulluk S, Gülmez B, Öztürk G, Özer S (2023) FC-Kmeans: fixed-centered K-means algorithm. *Expert Syst Appl* 211:118656. <https://doi.org/10.1016/j.eswa.2022.118656>
- Breiman L, Friedman J, Stone CJ (1984) Classification and regression trees. CRC Press, Boca Raton
- Chan EY, Ching WK, Ng MK, Huang JZ (2004) An optimization algorithm for clustering using weighted dissimilarity measures. *Pattern Recogn* 37:943–952. <https://doi.org/10.1016/j.patcog.2003.11.003>
- Chen X, Ye Y, Xu X, Huang JZ (2012) A feature group weighting method for subspace clustering of high-dimensional data. *Pattern Recogn* 45:434–446. <https://doi.org/10.1016/j.patcog.2011.06.004>
- Chen Y, Li W, Gao F, Wen Q, Zhang H, Wang H (2022) Practical attribute-based multi-keyword ranked search scheme in cloud computing. *IEEE Trans Serv Comput* 15:724–735. <https://doi.org/10.1109/TSC.2019.2959306>
- Cheng C-H, Wang J-W, Wu M-C (2009) OWA-weighted based clustering method for classification problem. *Expert Syst Appl* 36:4988–4995. <https://doi.org/10.1016/j.eswa.2008.06.013>
- Chiclana F, Herrera-Viedma E, Herrera F, Alonso S (2007) Some induced ordered weighted averaging operators and their use for solving group decision-making problems based on fuzzy preference relations. *Eur J Oper Res* 182:383–399. <https://doi.org/10.1016/j.ejor.2006.08.032>
- De Amorim RC (2016) A survey on feature weighting based k-means algorithms. *J Classif* 33:210–242. <https://doi.org/10.1007/s00357-016-9208-4>
- De Amorim R, Mirkin B (2012) Minkowski metric, feature weighting and anomalous cluster initializing in K-Means clustering. *Pattern Recogn* 45:1061–1075. <https://doi.org/10.1016/j.patcog.2011.08.012>
- Dombi J, Jónás T (2022) Generalizing the sigmoid function using continuous-valued logic. *Fuzzy Sets Syst* 449:79–99. <https://doi.org/10.1016/j.fss.2022.02.010>
- Frigui H, Nasraoui O (2004) Unsupervised learning of prototypes and attribute weights. *Pattern Recogn* 37:567–581. <https://doi.org/10.1016/j.patcog.2003.08.002>
- Fu Q, Li Y, Albathan M (2023) A supervised method to enhance distance-based neural network clustering performance by discovering perfect representative neurons. *Granul Comput* 8:1051–1065. <https://doi.org/10.1007/s41066-023-00370-5>
- Goicovich I, Olivares P, Román C, Román C, Vázquez A, Poupon C, Mangin J, Guevara P, Hernández C (2021) Fiber clustering acceleration with a modified kmeans++ algorithm using data parallelism. *Front Neuroinf* 15:727859. <https://doi.org/10.3389/fninf.2021.727859>
- Hashemzadeh M, Golzari Oskouei A, Farajzadeh N (2019) New fuzzy C-means clustering method based on feature-weight and cluster-weight learning. *Appl Soft Comput* 78:324–345. <https://doi.org/10.1016/j.asoc.2019.02.038>
- Huang YF, Chen JM (2000) The study of indexing techniques on object oriented databases. *Inf Sci* 130:109–131. [https://doi.org/10.1016/S0020-0255\(00\)00088-8](https://doi.org/10.1016/S0020-0255(00)00088-8)
- Huang JZ, Ng MK, Rong H, Li Z (2005) Automated variable weighting in k-means type clustering. *IEEE Trans Pattern Anal Mach Intell* 27:657–668. <https://doi.org/10.1109/TPAMI.2005.95>
- Huang X, Yang X, Zhao J, Xiong L, Ye Y (2018) A new weighting k-means type clustering framework with an l2-norm regularization. *Knowl-Based Syst* 151:165–179. <https://doi.org/10.1016/j.knosys.2018.03.028>
- Huang W, Peng Y, Ge Y, Kong W (2021) A new Kmeans clustering model and its generalization achieved by joint spectral embedding and rotation. *PeerJ Comput Sci* 7:e450. <https://doi.org/10.7717/peerj-cs.450>
- Ji C, Lu X, Zhang W (2021) Development of new operators for expert opinions aggregation: average-induced ordered weighted averaging operators. *Int J Intell Syst* 36:997–1014. <https://doi.org/10.1002/int.22328>
- Jing L, Ng MK, Huang JZ (2007) An entropy weighting k-means algorithm for subspace clustering of high-dimensional sparse data. *IEEE Trans Knowl Data Eng* 19:1026–1041. <https://doi.org/10.1109/TKDE.2007.1048>
- Khan IK, Luo Z, Huang JZ, Shahzad W (2019) Variable weighting in fuzzy k-means clustering to determine the number of clusters. *IEEE Trans Knowl Data Eng.* <https://doi.org/10.1109/TKDE.2019.2911582>
- Le K-NT, Nguyenthinhong D, Vovan T (2023) Fuzzy cluster analysis algorithm for image data based on the extracted feature intervals. *Granul Comput* 8:2067–2081. <https://doi.org/10.1007/s41066-023-00420-y>
- Li Y, Wu H (2012) A clustering method based on k-means algorithm. *Phys Procedia* 25:1104–1109. <https://doi.org/10.1016/j.phpro.2012.03.206>
- Liang Y, Li Y, Zhang K, Ma L (2021) DMSE: dynamic multi-keyword search encryption based on inverted index. *J Syst Architect* 119:102255. <https://doi.org/10.1016/j.sysarc.2021.102255>
- Ma F-M, Guo Y-J (2011) Density-induced ordered weighted averaging operators. *Int J Intell Syst* 26:866–886. <https://doi.org/10.1002/int.20500>
- Ma J, Xia D, Wang Y, Niu X, Jiang S, Liu Z, Guo H (2022) A comprehensive comparison among metaheuristics (MHs) for geohazard modeling using machine learning: insights from a case study of landslide displacement prediction. *Eng Appl Artif Intell* 114:105150. <https://doi.org/10.1016/j.engappai.2022.105150>
- MacQueen J (1967). Some methods for classification and analysis of multivariate observations. In Proceedings of the fifth Berkeley symposium on mathematical statistics and probability, vol 1, no 14, pp 281–297
- Makarenkov V, Legendre P (2001) Optimal variable weighting for ultrametric and additive trees and k-means partitioning: methods and software. *J Classif* 18:245–271. <https://doi.org/10.1007/s00357-001-0018-x>
- Marques JPPG, Cunha DC, Harada LMF, Silva LN, Silva ID (2021) A cost-effective trilateration-based radio localization algorithm using machine learning and sequential least-square programming optimization. *Comput Commun* 177:1–9. <https://doi.org/10.1016/j.comcom.2021.06.005>
- Mawati R, Sumertajaya IM, Afendi F (2014) Modified centroid selection method of k-means clustering. *IOSR J Math* 10:49–53. <https://doi.org/10.9790/5728-10234953>
- Modha DS, Spangler WS (2003) Feature Weighting in k-Means clustering. *Mach Learn* 52:217–237. <https://doi.org/10.1023/A:1024016609528>
- Naik DL, Kiran R (2021) A novel sensitivity-based method for feature selection. *J Big Data* 8:128. <https://doi.org/10.1186/s40537-021-00515-w>
- Nainggolan R, Perangin-angin R, Simarmata E, Tarigan AF (2019) Improved the performance of the k-means cluster using the sum of squared error (SSE) optimized by using the Elbow method. *J Phys Conf Ser* 1361:012015. <https://doi.org/10.1088/1742-6596/1361/1/012015>
- O'Hagan M (1988). Aggregating template or rule antecedents in real-time expert systems with fuzzy set logic. In: Twenty-second

- asilomar conference on signals, systems and computers, IEEE, vol 2, pp 681–689. <https://doi.org/10.1109/ACSSC.1988.754637>
- Peng D, Gui Z, Wang D, Ma Y, Huang Z, Zhou Y, Wu H (2022) Clustering by measuring local direction centrality for data with heterogeneous density and weak connectivity. *Nat Commun* 13:5455. <https://doi.org/10.1038/s41467-022-33136-9>
- Pons-Vives PJ, Morro-Ribot M, Mulet-Forteza C, Valero O (2022) An application of ordered weighted averaging operators to customer classification in hotels. *Mathematics* 10:1987. <https://doi.org/10.3390/math10121987>
- Rashidi R, Khamforoosh K, Sheikahmadi A (2020) An analytic approach to separate users by introducing new combinations of initial centers of clustering. *Physica A* 551:124185. <https://doi.org/10.1016/j.physa.2020.124185>
- Savita KN, Siwch A (2024) Fuzzy clustering based on distance metric under intuitionistic fuzzy environment. *Granul Comput* 9:20. <https://doi.org/10.1007/s41066-023-00446-2>
- Sculley D (2010) Web-scale k-means clustering. In: Proceedings of the 19th international conference on world wide web. association for computing machinery, New York, NY, USA, pp 1177–1178. <https://doi.org/10.1145/1772690.1772862>
- Singh S, Singh K (2023) Novel fuzzy similarity measures and their applications in pattern recognition and clustering analysis. *Granul Comput* 8:1715–1737. <https://doi.org/10.1007/s41066-023-00393-y>
- Steinbach M, Karypis G, Kumar V (2000) A comparison of document clustering techniques. Steinbach2000ACO. <https://api.semanticscholar.org/CorpusID:12808608>
- Sun S, Duan L, Xu Z, Zhang J (2021) Blind deblurring based on sigmoid function. *Sensors* 21:3484. <https://doi.org/10.3390/s21103484>
- Wang S, Li T, Luo C, Fujita H (2016) Efficient updating rough approximations with multi-dimensional variation of ordered data. *Inf Sci* 372:690–708. <https://doi.org/10.1016/j.ins.2016.08.044>
- Xu Z, Da Q (2003) Approaches to obtaining the weights of the ordered weighted aggregation operators. *J Southeast Univ* 33(1):94–96
- Xu S, Li X, Xie C, Chen H, Chen C, Song Z (2021) A high-precision implementation of the sigmoid activation function for computing-in-memory architecture. *Micromachines* 12:1183. <https://doi.org/10.3390/mi12101183>
- Yager RR (1988) On ordered weighted averaging aggregation operators in multicriteria decisionmaking. *IEEE Trans Syst Man Cybern.* <https://doi.org/10.1016/B978-1-4832-1450-4.50011-0>
- Yager RR (1993) Families of OWA operators. *Fuzzy Sets Syst* 59:125–148. [https://doi.org/10.1016/0165-0114\(93\)90194-M](https://doi.org/10.1016/0165-0114(93)90194-M)
- Yager RR, Filev DP (1994) Parameterized and-uke and or-like owa operators. *Int J Gen Syst* 22(3):297–316. <https://doi.org/10.1080/03081079408935212>
- Yager RR, Filev DP (1999) Induced ordered weighted averaging operators. *IEEE Trans Syst Man Cybern B* 29:141–150. <https://doi.org/10.1109/3477.752789>
- Yi P, Li W, Guo Y, Zhang D (2018) Quantile induced heavy ordered weighted averaging operators and its application in incentive decision making. *Int J Intell Syst* 33:514–528. <https://doi.org/10.1002/int.21945>
- Ying Y, Zhang N, Shan P, Miao L, Sun P, Peng S (2021) PSigmoid: improving squeeze-and-excitation block with parametric sigmoid. *Appl Intell* 51:7427–7439. <https://doi.org/10.1007/s10489-021-02247-z>
- Zhang P (2019) A novel feature selection method based on global sensitivity analysis with application in machine learning-based prediction model. *Appl Soft Comput* 85:105859. <https://doi.org/10.1016/j.asoc.2019.105859>
- Zhou X, Gu J, Shen S, Ma H, Miao F, Zhang H, Gong H (2017) An automatic k-means clustering algorithm of GPS data combining a novel niche genetic algorithm with noise and density. *ISPRS Int J Geo Inf* 6:392. <https://doi.org/10.3390/ijgi6120392>

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.