Journal of Network Intelligence
Taiwan Ubiquitous Information
 ISSN 2414-8105 (Online)
Volume **9**, Number **1**, February **2024**

Enhancing Clustering Performance: A Fuzzy Subspace Clustering Method with Local Correlation and Sparse Feature Selection

Fei Yan

College of Computer and Information Engineering
Xiamen University of Technology
Xiamen, 361024, China
fyan@xmut.edu.cn

Xiaodong Wang*

College of Computer and Information Engineering
Xiamen University of Technology
Xiamen, 361024, China
xdwangjsj@xmut.edu.cn

Longfu Hong

College of Computer and Information Engineering
Xiamen University of Technology
Xiamen, 361024, China
1393895211@qq.com

*Corresponding author: Xiaodong Wang

Received June 12, 2023, revised September 17, 2023, accepted November 22, 2023.

Abstract. *Despite the growing popularity of recent clustering techniques, most conventional clustering methods often suffer from overlooking the local correlation among the low-dimensional feature space of data points. In this scenario, they may encounter challenges in managing high-dimensional data. To address this issue, this article proposes a fuzzy subspace clustering method that jointly combines feature selection and local learning. Concretely, the proposed method incorporates local relationships among data points into the procedure of the clustering center calculation, which is optimized according to the local tightness of data points, thereby improving the clustering accuracy and robustness. Simultaneously, to filter out noisy features or abundant features, a sparse feature selection module is dynamically integrated into the clustering process. After that, the "local learning, feature selection, and clustering" procedure is repeated until the objective function converges. We apply our method to several widely used datasets, showing superior results to other state-of-the-art methods. Specifically, our method surpasses the second best compared method by over 4% ACC on the Glass dataset, validating its effectiveness.*
Keywords: Fuzzy clustering, Feature selection, Local learning.

1. **Introduction.** Clustering is a fundamental data processing technique that organizes data into distinct clusters based on the principle that similar entities tend to group together, as seen in the proverb "birds of a feather flock together." This method permits a high degree of similarity within each cluster while minimizing the similarity between

samples from distinct clusters. Over time, clustering methods have been extensively investigated and successfully implemented in a variety of fields, including data mining [1, 2], pattern recognition [3, 4, 5], and image processing [6, 7, 8].

TABLE 1. Comparison of recent KM type subspace clustering methods

Methods	type	subspace	local	W/O EVD	NR
KM	hard	✗	✗	✓	✗
Xu et al. [8]	hard	✓	✗	✗	✗
Bezde et al. [10]	soft	✗	✗	✓	✗
Li et al. [11]	hard	✓	✗	✗	✗
Hou et al. [12]	hard	✓	✗	✗	✗
Wang et al. [14]	hard	✓	✗	✗	✗
Xu et al. [17]	soft	✓	✗	✗	✗
Ours	soft	✓	✓	✓	✓

Existing clustering methods can be divided into different groups, such as partition-based methods, hierarchical methods, density-based methods, and model-based methods, depending on the rules of data aggregation in their cluster process [9]. Among them, partition-based methods, such as K-means (KM), have garnered significant attention from researchers due to their simplicity and low complexity. However, conventional KM is known to be sensitive to initial clustering centers and susceptible to noise, making it prone to local optimization and difficult to handle high-dimensional data, which often contain noisy features. To solve this problem, some scholars have proposed to integrate the subspace learning algorithm into KM, leading to the development of a range of subspace K-means (KM) variants. For example, Li et al. [11] introduced the maximum margin criterion (MMC) into KM, which can quickly extract low-dimensional feature subspaces and effectively avoid model instability caused by the singularity of the total scattered matrix, achieving better results on multiple high-dimensional datasets. Hou et al. [12] proposed a framework that integrates primary component analysis (PCA) into conventional KM and can flexibly balance the contribution of the matrix of the within-class scatter matrix and the between-class scatter matrix, thus achieving high clustering performance on different types of datasets. Although the aforementioned KM type subspace clustering methods can effectively reduce data dimensions and mitigate noise interference by transforming high-dimensional data into low-dimensional space, their sub-spatial solution procedure requires complex eigenvalue decomposition, thus posing difficulties in handling high-dimensional data [13]. In order to address this challenge, Wang et al. [14] proposed a fast KM type subspace algorithm that introduces feature selection into the KM model. This approach allows for the identification of representative features during the clustering process and achieves superior clustering performance with reduced computational time across diverse high-dimensional datasets.

However, the above methods use a "hard" cluster mechanism, that is, one data point could only belong to one single cluster. In this case, when there are noise samples in the dataset, the cluster performance of these methods may become unstable [15, 16]. To address this problem, researchers have proposed to use the "soft" clustering strategy to loosen the membership of each data point so that it can belong to multiple clusters at the same time, thereby improving the model's resistance to interference with noisy data. Among them, fuzzy c-means (FCM) is the most representative "soft" clustering method and has achieved superior performance than KM in a variety of application fields [17]. Nevertheless, similar to KM, FCM only considers the global geometry information among data during its membership calculation process and is difficult to handle data samples

with varying densities. To address this limitation, Xu et al. [17] proposed a robust FCM clustering algorithm that utilizes the $l_{2,1}$ norm-based distance calculation between data points and their cluster centers. This approach can effectively suppress the impact of noise on the model. Building upon this foundation, Zhang et al. [18] proposed to combine FCM with non-negative spectral clustering to enhance the robustness of the clustering process. This method is built on the similarity between samples, simultaneously optimizing the clustering indicator matrix and the Laplacian matrix of the data. As a result, it achieves better results across multiple datasets. However, they still simply average all the samples within each cluster to determine the cluster centers, which may bias the clustering results. Besides, the clustering process for the aforementioned "soft" clustering methods remains confined to the data's original feature space, which consumes expensive computational resources when processing high-dimensional data. Lastly, the presence of noisy features in high-dimensional data substantially compromises the clustering performance of these methods. Table 1 shows the comparison of recent KM type subspace clustering methods, where "W/O EVD" refers to "without eigenvalue decomposition" and "NR" refers to "noise resistance".

To solve the above issues, this article proposes a fuzzy clustering approach that jointly integrates feature selection and local learning. Our approach incorporates local neighboring relationships among data points into the calculation of cluster centers, thereby optimizing them according to the local tightness among data points and improving the clustering accuracy. In addition, our method combines feature selection with sparse structure into the dynamic clustering process to avoid the interference of noisy features. Experimental results on multiple datasets verify the effectiveness of our method. The technical contribution of our work lies in its pioneering approach to local fuzzy subspace clustering, which addresses the limitations of existing methods. By leveraging local data relationships and feature selection, our approach offers a comprehensive solution to enhance the quality of clustering outcomes.

Below is the outline of this article. Section 2 provides an overview of related works. Section 3 explains our local fuzzy subspace clustering method along with the optimization strategy. Experimental results and ablation studies are reported in Section 4, followed by the conclusion in Section 5.

2. **Related Works.** Given a dataset $X = [x_1, x_2, \ldots, x_n] \in R^{d\times n}$ containing n data points with d features, KM type clustering methods aims to divide X into K groups, making a higher similarity degree of samples in the same cluster and a lower similarity degree of samples between different clusters. Let $V = [v_1, v_2, \ldots, v_c] \in \mathbb{R}^{d\times c}$ be the cluster center matrix and v_i be the center for the ith cluster, where c refers to the number of clusters. Suppose $Y = [y_1, y_2, \ldots, y_n]^T \in \{0,1\}^{n\times c}$ is the cluster indicator matrix for X. $y_{ij} = 1$ if the ith sample belongs to the jth cluster, $y_{ij} = 0$ otherwise. Then, the objective function of KM can be formulated as:

$$\min_{Y\in\{0,1\}^{n\times c}, v_j} \sum_{i=1}^{n}\sum_{j=1}^{c} y_{ij} \left\|x_i - v_j\right\|_2^2 . \tag{1}$$

As indicated in previous studies [19], KM is sensitive to the initial cluster centers for its hard membership determination strategy and is prone to local optimization. To solve this issue, FCM, which incorporates fuzziness into the membership degree of data points, was introduced, allowing one data point to belong to multiple categories at the same time. Specifically, in FCM, the fuzzifier parameter m (ranging from $[1, \infty]$) determines the extent to which each data point belongs to a certain cluster. A higher value of the fuzzifier parameter m indicates a greater degree of fuzziness in the clusters. This means

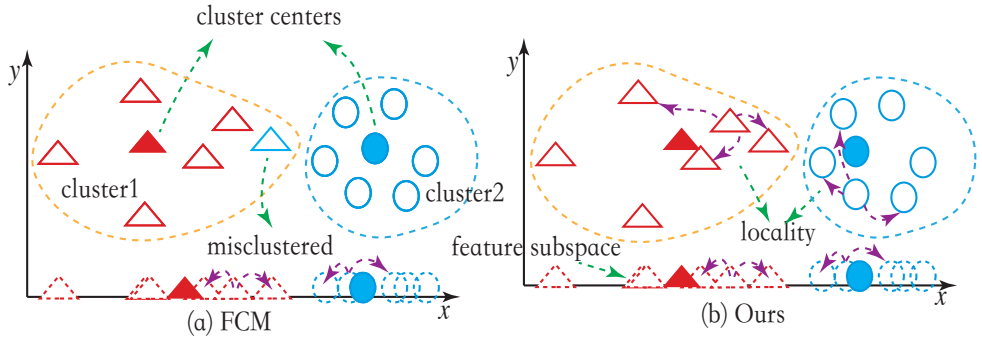


FIGURE 1. Clustering mechanism comparison of our method and FCM

that data points can have membership degrees spread across multiple clusters, resulting in softer boundaries between clusters. Conversely, when m is smaller, it will lead to sharper boundaries between clusters, with data points having higher membership degrees in a single cluster. This flexible framework of FCM makes it applicable to a wide range of datasets. Mathematically, FCM relaxes KM's class indicator matrix Y from discrete $\{0,1\}$ to continuous $[0,1]$, where y_{ij} represents the extent to which the ith data point belongs to the center of the jth cluster. It should be noted that FCM requires the sum of each row of Y to be 1, that is, $Y\mathbf{1}_c = \mathbf{1}_n$, where $\mathbf{1}_c$ and $\mathbf{1}_n$ are the c-dimensional and n-dimensional column vectors of all ones, respectively. In other words, by ensuring $Y\mathbf{1}_c = \mathbf{1}_n$, the membership degrees of FCM can be viewed as probabilities or likelihoods of a data point belonging to different clusters. The objective function of FCM can be formulated as follows:

$$\min_{Y\mathbf{1}_c=\mathbf{1}_n, Y>0, v_j} \sum_{i=1}^{n}\sum_{j=1}^{c} y_{ij}^m \left\|x_i - v_j\right\|_2^2, \tag{2}$$

where $Y > 0$ indicates that all of the elements in Y are strictly greater than zero.

3. Our Method.

3.1. Motivations.

While FCM is effective in capturing global correlation relationships between samples and cluster centers, it often overlooks the significance of local adjacency relationships among data points. Consequently, in the presence of outliers within the input data, FCM may exhibit noticeable clustering errors, as depicted in Figure 1. In particular, Figure 1 (a) demonstrates that when the data to be processed contains dense regions or outliers, such as the blue triangle, FCM tends to calculate distances between each data point and the global cluster centers, leading to erroneous clustering outcomes. Based on the above analysis and drawing inspiration from local learning techniques, we argue that FCM can benefit from incorporating local relationships (purple dashed lines) among samples into its clustering procedure. Concretely, when a sample has neighboring samples that belong to a specific cluster, it is more likely to share similar characteristics with its neighbors and should be influenced accordingly during the membership assignment process. By considering the similarity in cluster assignments of neighboring samples, we can enhance the FCM's ability to capture the local structure of the data and better handle datasets with complex structures, reducing the impact of outliers and improving the overall clustering accuracy. For example, as shown in Figure 1 (b), after incorporating local inter-sample correlations, our method effectively mitigates the interference of noisy data and attains superior clustering results. Moreover, from Figure 1, it is evident that these two-dimensional data points exhibit a discriminative cluster structure in the

low-dimensional feature subspace (x-axis), indicating the existence of abundant features among the data. Motivated by this observation, we propose integrating a feature selection technique into FCM. By identifying the most informative features, we can focus on the relevant aspects of the data, thereby improving clustering accuracy and efficiency. We substantiate these claims by demonstrating the efficacy of our approach on a synthetic dataset, as detailed in Section 4.2.

3.2. **Formulation.** In this section, we formulaically introduce our proposed method. Given an input dataset $X \in \mathbb{R}^{d\times n}$, we suppose x_i and x_j are the ith and the jth samples, respectively. Follow the previous studies [20], we impose a knn-based distance matrix $\widetilde{D} \in R^{n\times n}$, where $\widetilde{D}_{ij} = \|x_i - x_j\|_2^2$ if x_j is in the k-nearest neighbor of x_i, $\widetilde{D}_{ij} = 0$ otherwise. We want to pay more attention to the local similarity of samples. To achieve this goal, for D_{ij} at the ith row and jth column of $\widetilde{D}$, we determine it by normalizing $\widetilde{D}_{ij}$ as follows:

$$D_{ij} = \frac{exp^{(-\frac{\widetilde{D}_{ij}}{t})}}{\sum_{l\in\mathcal{N}(x_i)} exp^{(-\frac{\widetilde{D}_{il}}{t})}}, \tag{3}$$

where $\mathcal{N}(x_i)$ is an index set which consists of the indexes of the k-nearest samples of x_i. t is a parameter to control the spread range of similarity. Following previous works [21], we set t as the mean value of the ith row of $\widetilde{D}$, that is, $t = \frac{1}{n}\sum_{l=1}^{n}\widetilde{D}_{il}$. Obviously, for D_{ij} in the ith row and the jth column of D, D_{ij} ranges from $[0, 1]$ and represents the local similarity of x_i and x_j.

Based on the definition of FCM in previous section, we combine the local similarity information into the membership of FCM as $\sum_{r=1}^{n} d_{ri}y_{ij}^m$. Clearly, for the ith sample x_i, the possibility that x_i belongs to the jth cluster is closely related to its neighbors. In other words, if most of x_i's neighbors belong to the jth cluster, then x_i itself also probably belongs to the jth cluster. Formally, we propose to integrate the local information encoded in D into the objective function in Equation (2) as follows:

$$\begin{aligned} \min_{Y,v_j} &\sum_{i=1}^{n}\sum_{j=1}^{c}\sum_{r=1}^{n} d_{ri}y_{ij}^m \|x_i - v_j\|_2^2 \\ \text{s.t. } &\sum_{j=1}^{c} y_{ij} = 1, y_{ij} > 0. \end{aligned} \tag{4}$$

As mentioned earlier, FCM's clustering performance is limited when processing high-dimensional data due to the influence of noisy features. To overcome this limitation, researchers proposed embedding subspace learning into clustering to obtain a low-dimensional feature subspace during the clustering process. In these methods, clustering and subspace learning are jointly optimized to determine the optimal feature subspace, resulting in enhanced clustering performance. However, most of these subspace clustering methods require complex feature decomposition operations and consume a large amount of computational resources. To address this issue, inspired by previous works [22], this article proposes a dynamic combination of feature selection with Equation (4), introducing a special sparse constraint into the construction process of the feature selection matrix. The incorporation of this sparse structure allows the optimization process of our proposed model to operate efficiently without relying on complex feature decomposition operations. As a result, our method can effectively deal with high-dimensional data while mitigating the computational burden.

Suppose $W = [w_1, w_2, \ldots, w_p] \in \{0,1\}^{d\times p}$, where p represents the reduced number of features and $w_i \in \{0,1\}^{d\times 1}$ refers to the ith column of W. In order to facilitate efficient feature selection, we assume that matrix W maintains a sparse structure, wherein each column contains only one nonzero entry. These nonzero entries in W correspond to the indices of the selected features for the input data. Formally, W should satisfy $W^T\mathbf{1}_d = \mathbf{1}_p$ and there are p entries of $W\mathbf{1}_p$ equal to 1 and the remaining elements equal to 0, that is, $\|W\mathbf{1}_p\|_0 = p$, where $\mathbf{1}_d$ and $\mathbf{1}_p$ are the d-dimensional and p-dimensional column vectors of all ones, respectively. After integrating W into Equation (4), we arrive at:

$$\begin{aligned} &\min_{W,Y,v_j} \sum_{i=1}^{n}\sum_{j=1}^{c}\sum_{r=1}^{n} d_{ri} y_{ij}^m \left\|W^T x_i - W^T v_j\right\|_2^2 \\ &\text{s.t.} \sum_{j=1}^{c} y_{ij} = 1, y_{ij} > 0, \\ &W \in \{0,1\}^{d\times p}, W^T\mathbf{1}_d = \mathbf{1}_p, \|W\mathbf{1}_p\|_0 = p. \end{aligned} \tag{5}$$

The advantages of the similarity matrix D. According to the construction process of D above, it effectively captures the similarity vectors that depict the relationships between each sample and its neighbors, allowing for a representation of the local relationships within the dataset. The importance of one sample can be inferred from the density of connections it exhibits with its neighbors. Specifically, by examining the sums of columns in matrix D, we can assess the importance of each sample. Samples reflecting large sums in D indicate a higher contribution to the overall structure and should be emphasized during the clustering procedure. In contrast, samples with smaller sums may indicate potential noise or less relevance to the underlying patterns and should be treated with caution as they could be noisy or less informative data points. For instance, given the following input matrix X with one outlier, $x_1 = [-5\ \ -5]$:

$$X = \begin{bmatrix} -5 & 2 & 5 & 5 & 5 & 6 & 8 & 8 & 7.5 & 8.5 & 8 & 9 \\ -5 & 3 & 1 & 3 & 3.5 & 3.5 & 3 & 4 & 5 & 5 & 6 & 5 \end{bmatrix}. \tag{6}$$

We depict X in Figure 2. We can see that X is composed of two distinct classes, which are labeled as blue and green. Additionally, there exists one outlier in the dataset, which is labeled orange. From the discussion in Section 1 and the conventional objective functions of KM and FCM in Section 2, this outlier may bias the cluster center, resulting in a degraded clustering performance.

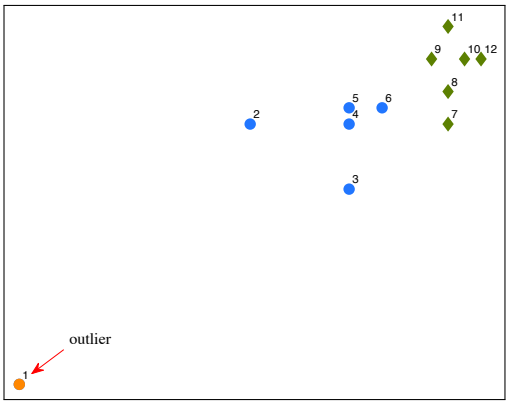


FIGURE 2. Illustration of X in Equation (6). Clearly, X consists of two classes (labeled in blue and green) and one outlier (labeled in orange)

We can construct the similarity matrix D with 5-nearest neighbors as follows:

$$D = \begin{bmatrix} 0 & 0.351 & 0.246 & 0.160 & 0.141 & 0.102 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.192 & 0.346 & 0.334 & 0.119 & 0 & 0 & 0.009 & 0 & 0 & 0 \\ 0 & 0.040 & 0 & 0.473 & 0.254 & 0.193 & 0.040 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.006 & 0.083 & 0 & 0.565 & 0.339 & 0.006 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.008 & 0.032 & 0.558 & 0 & 0.391 & 0 & 0 & 0.011 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.403 & 0.490 & 0 & 0.038 & 0.038 & 0.031 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.086 & 0 & 0.689 & 0.086 & 0.086 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.362 & 0 & 0.264 & 0.264 & 0.008 & 0.102 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.007 & 0.272 & 0 & 0.367 & 0.272 & 0.082 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.001 & 0.126 & 0.183 & 0 & 0.126 & 0.564 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.002 & 0.055 & 0.363 & 0.363 & 0 & 0.217 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.005 & 0.111 & 0.085 & 0.688 & 0.111 & 0 \end{bmatrix}, \tag{7}$$

where the column associated with sample x_1 is labeled in red. From Equation (7), we can see that the outlier exhibits relatively loose relations with other samples (the elements in $D_{:1}$ are all zero). By incorporating the definition of Equation (5), it becomes evident that our proposed method systematically assigns a weight of 0 ($\sum_{r=1}^{n} d_{r1} = 0$) to outliers, thus successfully mitigating their interference.

3.3. **Optimization.** Our clustering model in Equation (5) contains unknown variables W, Y, and V, which are jointly non-convex, and W is a discrete variable, making it difficult to directly obtain its closed solution. In light of this difficulty and inspired by the methodology outlined in reference [14], this article proposes the following iterative optimization approach for solving this problem. Specifically, we alternatively optimize one of these unknown variables by considering the remaining variables as constant values. The detailed optimization procedure is listed as follows:

1) **Firstly, fix Y and W to solve V.** The objective function in Equation (5) can be converted to:

$$\min_{V} \sum_{i=1}^{n} \sum_{j=1}^{c} \sum_{r=1}^{n} d_{ri} y_{ij}^{m} \left\| W^T x_i - W^T v_j \right\|_2^2. \tag{8}$$

Equation (8) is equivalent to solving each column of matrix V separately. Take its jth column vector v_j as an example, to optimize v_j, we take the derivative of Equation (8) w.r.t. j and make the derivative result zero. Then, we have:

$$\sum_{i=1}^{n} \sum_{r=1}^{n} d_{ri} y_{ij}^{m} W W^T v_j = \sum_{i=1}^{n} \sum_{r=1}^{n} d_{ri} y_{ij}^{m} W W^T x_i. \tag{9}$$

In Equation (9) above, due to the special structure of W, the $\sum_{i=1}^{n} \sum_{r=1}^{n} d_{ir} y_{rj}^{m} W W^T$ on the left-hand side of Equation (9) is singular and cannot be directly used to solve v_j. However, if we take a careful observation, we can find out that the elements of v_j correspond one-to-one with the elements of x_i that are set to zero by WW^T. In other words, v_j and x_i share the same index set of zero entries. Therefore, we only need to solve the non-zero entries of v_j. Concretely, we can directly remove the WW^T on the left side of Equation (9) to solve v_j, and get the following formula:

$$v_j = \frac{\sum_{i=1}^{n} \sum_{r=1}^{n} d_{ri} y_{ij}^{m} W W^T x_i}{\sum_{i=1}^{n} \sum_{r=1}^{n} d_{ri} y_{ij}^{m}}. \tag{10}$$

2) **Secondly, fix V and W to solve Y.** The Lagrange function of the objective function in Equation (5) can be expressed as:

$$\mathcal{L}\left(y_{ij},\lambda_i,\Lambda\right)=\sum_{i=1}^{n}\sum_{j=1}^{c}\sum_{r=1}^{n}d_{ri}y_{ij}^{m}\left\|W^Tx_i-W^Tv_j\right\|_2^2+\sum_{i=1}^{n}\lambda_i\left(1-\sum_{j=1}^{c}y_{ij}\right)-\Lambda Y, \quad (11)$$

where λ_i and Λ are Lagrange multiplier. By combining the KKT condition of the Lagrange function in Equation (11), we can obtain:

$$\begin{aligned}\Lambda&=0\\ \lambda_i&=\left(\sum_{j=1}^{c}\left(m\mu_{ij}^2d_{ii}\right)^{\frac{1}{1-m}}\right)^{1-m}\\ y_{ij}&=\frac{\left(m\mu_{ij}^2\right)^{\frac{1}{1-m}}}{\left(\sum_{j=1}^{c}\left(m\mu_{ij}^2\right)^{\frac{1}{1-m}}\right)},\end{aligned} \quad (12)$$

where $\mu_{ij}=\|W^Tx_i-W^Tv_j\|_2^2$.

3) **Finally, fix Y and V to solve W.** Let $\hat{y}_{ij}=\sum_{r=1}^{n}d_{ri}y_{ij}^m$, the objective function in Equation (5) can be converted to:

$$\begin{aligned}&\min_W\sum_{i=1}^{n}\sum_{j=1}^{c}Tr\left(W^T\hat{y}_{ij}(x_i-v_j)^T(x_i-v_j)W\right)\\ &\text{s.t. } W\in\{0,1\}^{d\times p},W^T\mathbf{1}_d=\mathbf{1}_p,\|W\mathbf{1}_p\|_0=p.\end{aligned} \quad (13)$$

Considering the special structure of W in Equation (13), it can be found that the optimal solution of W, that is, the entries of non-zero elements, is the subscript corresponding to the first p smallest elements on the diagonal of matrix M, where M can be represented as:

$$M=\sum_{i=1}^{n}\sum_{j=1}^{c}\hat{y}_{ij}(x_i-v_j)^T(x_i-v_j). \quad (14)$$

Based on the above optimization analysis for our objective function, we summarize our algorithm in Algorithm 1.

In this discussion, we offer a brief overview of the complexity associated with Algorithm 1. The primary computational components involve the generation of the similarity matrix D, the determination of the clustering center matrix, the computing of the cluster indicator matrix Y, and the updating of the feature selection matrix. The complexity of the aforementioned operations is $O(dnlog(n))$ (K-D tree solution), $O(dn)$, $O(nc)$, and $O(dnc)$. Therefore, our method can be applied to large-scale high-dimensional data and is efficient for real-world applications.

4. Experiments.

4.1. Experimental Setting. Compared methods:

In order to verify the effectiveness of our method, seven popular clustering methods are selected for comparison. The relevant comparison methods are described as follows:

1) **KM**: A classic hard clustering algorithm, which is chosen as the baseline for this article.

2) **FCM** [10]: A classic fuzzy clustering algorithm that introduces a "soft" membership calculation strategy.

Algorithm 1 The optimization procedure for Equation (5)

Input:

The input data $X \in R^{d\times n}$
The number of clusters c
The number of neighbors k
The reduced feature numbers p
The fuzziness parameter m

Output:

The cluster indicator matrix Y

1: Randomly initialize the cluster indicator matrix Y and make it satisfy $Y^T\mathbf{1}_c = \mathbf{1}_n$
2: Construct the similarity matrix D
3: Randomly initialize feature selection matrix W and and make it satisfy $W^T\mathbf{1}_d = \mathbf{1}_p, \|W\mathbf{1}_p\|_0 = p$
4: **repeat**
5: Compute the cluster center matrix V, that is, $v_j = \frac{\sum_{i=1}^{n}\sum_{r=1}^{n} d_{ri}y_{ij}^m WW^T x_i}{\sum_{i=1}^{n}\sum_{r=1}^{n} d_{ri}y_{ij}^m}$
6: Update Y according to $y_{ij} = \frac{\left(m\mu_{ij}^2\right)^{\frac{1}{1-m}}}{\left(\sum_{j=1}^{c}\left(m\mu_{ij}^2\right)^{\frac{1}{1-m}}\right)}, \forall i, 1 \le i \le n, \forall j, 1 \le j \le c$
7: Set the subscript corresponding to the first p smallest elements on the diagonal of M in Equation (14) to 1 for the corresponding elements in W, and 0 for the rest
8: **until** Convergence

3) **MMCKM** [11]: A subspace clustering algorithm that imposes the maximum margin criterion to solve "small sample" problems.

4) **DEC** [12]: A discriminative subspace learning framework, which jointly optimizes PCA and KM while incorporating a single balancing parameter to regulate the influence of both within-class scatter and between-class scatter matrices on the clustering procedure.

5) **RSFKM** [17]: An algorithm that introduces sparse constraints into the fuzzy C-means clustering process, which can effectively reduce the impact of noisy data on clustering results.

6) **SRDEKM** [8]: An enhanced KM type clustering method that leverages a re-weighted optimization strategy to mitigate the impact of noise by imposing non-square constraints on the distance calculation of each sample and its cluster center.

7) **FAKM** [14]: A fast clustering method that combines feature selection and K-means clustering. It also uses the adaptive loss function to obtain the cluster indicator matrix, which can quickly process large-scale high-dimensional data.

Datasets. We select various types of experimental data validation to verify the efficacy of the proposed method, including one synthetic dataset and six publicly available datasets. Among them, the synthetic dataset is artificially designed to analyze the underlying mechanism of the proposed method. This dataset consists of two distinct sets of data points distributed in a two-dimensional space. The first set, referred to as Cluster1, consists of 200 randomly generated bottle-shaped data points with X-axis coordinates ranging from $[-1, 1]$ and Y-axis coordinates ranging from $[-3, 3]$; The second set, referred to as Cluster2, comprises 200 circular data points. To generate Cluster2, we initially create a 2D data point set of 200 samples distributed within a radius of $[0.5, 1]$. After that, we apply an offset of 8.8 in the X and Y coordinate directions, resulting in the final data point set for Cluster2. In addition to this synthetic dataset, we also introduce six popular

publicly available datasets, including three UCI datasets (Glass, Breast, and Vehicle)[1], Umist[2], Yale[3], and WebKB [23]. Table 2 lists the detailed descriptions of the selected datasets.

TABLE 2. Detailed description of our selected six publicly available datasets

Dataset	number of classes	number of samples	dimensionality	tuned feature range
Glass	6	214	9	{3,4,5,6,7,8}
Breast	2	699	10	{2,3,4,5,6,7,8,9}
Vehicle	4	846	18	{3,4,5,6,7,8,9,10}
Umist	20	575	644	{100,200,300,400,500}
Yale	15	165	1024	{100,200,...,900,1000}
WebKB	7	814	4029	{50,100,150,200,250,300}

To ensure a fair experimental comparison, the hyperparameters (if any) of all compared methods in this article are set to the optimal values recommended in the original paper of each method. In cases where the original paper does not recommend optimal parameters, the grid-search approach introduced in [14] is applied to determine the optimal parameters. Concretely, the search range is defined as $\{10^{-6}, 10^{-4}, 10^{-2}, \ldots, 10^{2}, 10^{4}, 10^{6}\}$. For fuzzy type clustering methods like FCM, RSFKM, and our method, the fuzziness parameter m needs adjustment, and the search range is set to $[1.0, 1.2]$. All experiments in this article were implemented in an environment with a 3.7GHz CPU, 16GB of memory, and a 64-bit Windows 10 operating system.

In this experiment, two evaluation metrics, that is, accuracy (ACC) and normalized mutual information (NMI) are used to measure clustering performance [14]. Higher values of both of these evaluation metrics indicate superior performance. Each compared method is repeated 20 times on each dataset, and the average along with the standard deviation of the experimental results corresponding to the optimal parameters are reported.

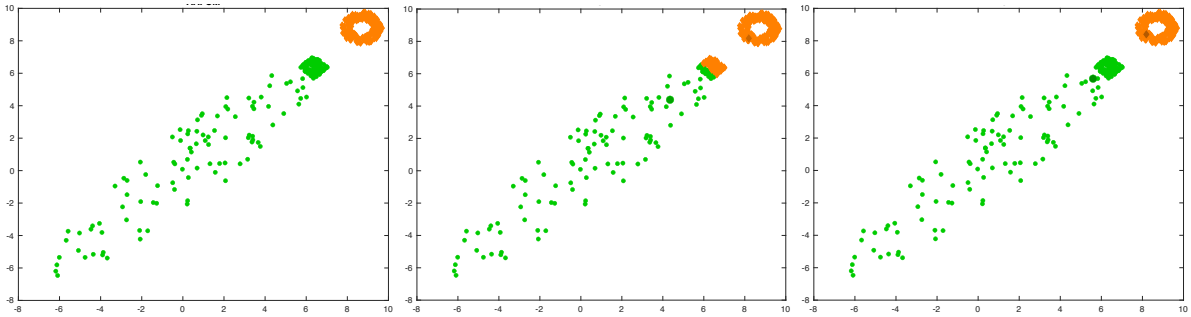
4.2. **Results on the synthetic dataset.** Firstly, a comparison and analysis with FCM on the synthetic dataset were conducted, and the clustering results are shown in Figure 3. Figure 3 (a) shows the ground truth labels, and the cluster centers of the clustering results are highlighted with dark large circles and diamonds. The figure reveals that the data points on the mouth of the bottle-shaped dataset in Cluster1 are more densely concentrated, whereas the data points on the body of the bottle-shaped dataset are scattered. The data points in Cluster2 exhibit an overall concentration. From the clustering results of these two compared methods, it can be observed that FCM has deviations in the calculation of its clustering centers, resulting in misclustering of bottle mouth and circle into a single cluster, as shown in Figure 3 (b). On the contrary, our method in Figure 3 (c) achieves perfect clustering results. The reason may be that our method not only calculates the membership between the data points and the cluster centers, but also considers the adjacent relationships among data. In this case, the clustering center calculated by our method is more inclined towards the data points with larger weight values in each cluster (such as the bottle mouth (dense area)), which can obtain more accurate clustering results. After we repeat the two compared methods 20 times on this dataset, the average ACC of FCM is 74.2%, while our method is 100%. In this experiment, the parameter k of our method is set to 200, the fuzziness parameter m is 1.35, and p is 2.

To comprehensively analyze the clustering performance of our method across diverse application scenarios, we apply it to six open-source datasets and compare it with various

[1]http://archive.ics.uci.edu/ml

[2]http://images.ee.umist.ac.uk/danny/database.html

[3]http://www.cvc.yale.edu/projects/yalefaces/yalefaces.html



(a)Ground truth (b)FCM (c)Ours

FIGURE 3. The visualization clustering results of FCM (b) and our method (c) on the synthetic dataset

popular clustering methods. Table 3 and Table 4 show the comparison results under the ACC and NMI evaluation metrics, respectively. From the results, we have the following observations: 1) On most datasets, FCM achieves better clustering performance than KM, indicating that incorporating the fuzziness mechanism into KM can enhance clustering performance. 2) All subspace clustering methods, such as MMCKM, DEC, SRKEKM, FAKM, etc., consistently outperform KM on most datasets, and most of their clustering results are also more stable compared to KM (with smaller variance), indicating that mapping high-dimensional data to low-dimensional subspaces helps reduce feature noise and improve clustering performance. 3) Compared to traditional FCM, RSFKM achieves superior results on some datasets with low-dimensional features such as Glass, Break, and Vehicle. This suggests that introducing a sparse constraint mechanism into the cluster center update process can improve FCM's anti-interference ability against sample-level noise. However, RSFKM has poorer clustering performance than FCM when processing high-dimensional data such as Umist, Yale, and WebKB. The discrepancy can be attributed to the clustering update process of RSFKM still employing the Euclidean distance calculation method in high-dimensional feature space, which is susceptible to noisy features. 4) Our method consistently achieves the best clustering results across most datasets. For instance, under the ACC evaluation criterion, our method surpasses the second-highest method by over 4% on the Glass dataset, over 2.6% on the Yale dataset, and nearly 1.9% on the WebKB dataset, illustrating the effectiveness of our method.

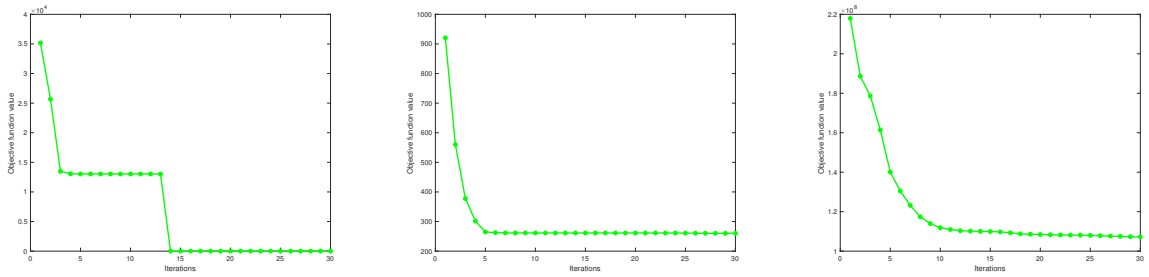
TABLE 3. Performance comparison using the ACC criterion with different clustering methods on six datasets. We observe that our method obtains the best results in most cases

	Glass	Breast	Vehicle	Umist	Yale	WebKB
KM	46.79 ± 4.07	95.28 ± 0.00	36.95 ± 0.75	42.79 ± 2.17	43.68 ± 4.37	56.64 ± 1.18
FCM	47.99 ± 3.71	95.27 ± 0.00	37.17 ± 0.72	44.30 ± 1.93	46.72 ± 1.72	55.17 ± 1.24
MMCKM	46.73 ± 2.47	95.42 ± 0.00	38.06 ± 0.00	43.74 ± 2.01	45.50 ± 2.86	53.28 ± 2.64
DEC	47.24 ± 2.82	95.42 ± 0.00	40.40 ± 0.90	43.56 ± 2.45	47.73 ± 3.14	57.13 ± 3.53
SRDEKM	47.87 ± 3.73	83.12 ± 0.00	42.07 ± 1.98	43.85 ± 2.38	46.06 ± 2.77	52.93 ± 2.99
RSFKM	49.35 ± 5.70	96.28 ± 0.00	38.89 ± 1.63	40.66 ± 1.61	43.52 ± 2.61	53.00 ± 1.60
FAKM	49.53 ± 5.96	95.57 ± 0.00	44.13 ± 2.38	44.35 ± 3.27	48.56 ± 4.82	67.09 ± 2.42
Ours	53.73 ± 2.41	96.42 ± 0.01	43.15 ± 0.54	44.45 ± 1.26	51.17 ± 2.33	68.92 ± 0.00

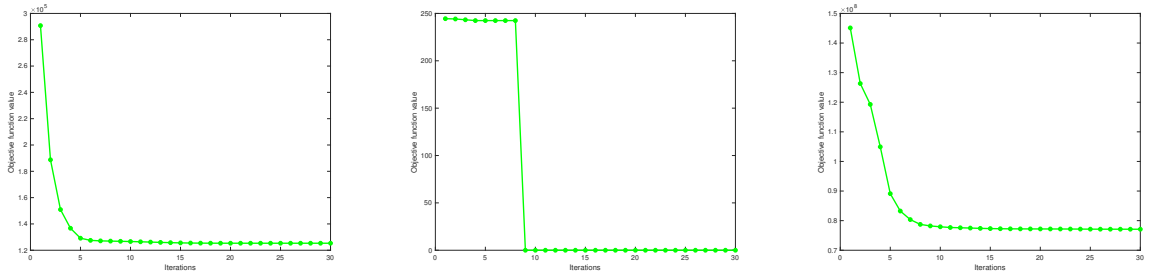
4.3. **Convergence Analysis.** To demonstrate that our iterative optimization algorithm proposed in Algorithm 1 in Section 3.3 can monotonically reduce the objective function value of Equation (5) until the model converges, we record the objective function values throughout the iterative optimization process of our method on six datasets, as depicted

TABLE 4. Performance comparison using the NMI criterion with different clustering methods on six datasets. We observe that our method obtains the best results in most cases

	Glass	Breast	Vehicle	Umist	Yale	WebKB
KM	32.77 ± 2.49	70.49 ± 0.00	11.32 ± 2.28	64.57 ± 1.70	51.44 ± 3.88	15.12 ± 1.37
FCM	33.68 ± 2.35	70.59 ± 0.00	11.38 ± 2.14	64.57 ± 1.14	53.81 ± 1.74	18.86 ± 1.09
MMCKM	34.25 ± 1.75	71.17 ± 0.00	10.41 ± 0.03	65.23 ± 1.59	51.86 ± 2.70	17.45 ± 0.94
DEC	33.62 ± 1.37	71.17 ± 0.00	10.20 ± 0.00	65.53 ± 1.41	54.19 ± 2.20	18.53 ± 0.92
SRDEKM	33.92 ± 4.10	29.72 ± 0.00	11.85 ± 3.68	65.02 ± 1.79	52.44 ± 3.01	17.34 ± 0.91
RSFKM	32.68 ± 3.51	75.60 ± 0.00	12.17 ± 1.62	60.16 ± 1.04	50.58 ± 2.26	14.10 ± 3.07
FAKM	33.81 ± 3.83	71.92 ± 0.00	17.87 ± 2.41	63.89 ± 2.39	54.63 ± 3.55	16.72 ± 1.42
Ours	37.44 ± 0.38	76.29 ± 0.40	15.98 ± 0.98	63.90 ± 0.72	53.66 ± 2.11	16.23 ± 0.00



(a)Breast (b)Glass (c)Umist



(d)Vehicle (e)WebKB (f)Yale

FIGURE 4. Convergence analysis of our method on different datasets

in Figure 4. From the results, it can be seen that our method converges on all the selected datasets, albeit with varying convergence speeds. Generally, our method converges within 30 optimization iterations, indicating its efficiency.

4.4. **Ablation Study.** Our method consists of three core modules: the local learning module, the feature selection module, and the membership fuzziness module. These three modules are controlled by three hyperparameters, namely: 1) The parameter k, which is used to build the similarity matrix, and governs the range of local relations among data. A larger value of k directs our clustering model towards analyzing more global information. 2) The parameter p, which determines the sparsity of the feature selection matrix W, and controls the number of retained features during the clustering process. A smaller value of p, leads to a reduced number of features and lower data dimensionality. 3) The parameter m, which sets the fuzziness level of our model. In this section, we only focus on verifying the effectiveness of the first two modules, that is, the parameters k and p. The sensitivity analysis of the parameter m will be discussed in Section 4.5.

In order to validate the efficacy of the local learning module and the feature selection module, we examine their impacts on our clustering model in Equation (5) by individually removing each module. Specifically, the local learning module is removed from our

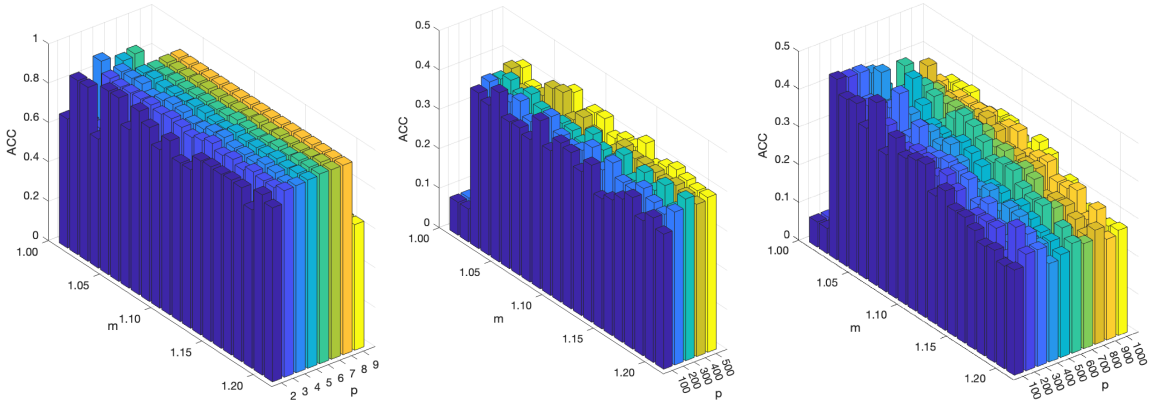
TABLE 5. Performance comparison with our proposed two modules, that is, local learning and feature selection, on the Glass dataset

Locality	Feature selection	ACC	NMI
✗	✗	47.86 ± 3.76	34.04 ± 1.79
✓	✗	52.99 ± 0.42	36.53 ± 0.91
✗	✓	52.15 ± 2.31	36.42 ± 0.75
✓	✓	53.73 ± 2.41	37.44 ± 0.38

clustering model by setting $k = 1$, transforming the similarity matrix D into an identity matrix. Consequently, our model solely focuses on each sample individually and ignores the impact of its adjacent samples. Similarly, by setting $p = d$ and initializing W as an identity matrix, we remove the feature selection module. The clustering performance of our method with different module combinations on the Glass dataset is shown in Table 5. It is important to note that according to Equation (5) and the analysis in Section 2, when the local learning module and the feature selection module are removed (for brevity, we call it NON-LF), our method will degrade to FCM. From the results (the second row) of Table 5, we observe that after removing both of these two modules, the performance of our method is also close to FCM, confirming the aforementioned inference. Subsequently, we remove only one module at a time and observe performance improvements. For example, after adding the local learning module (the third row), our method outperforms NON-LF by over 5.1% in terms of ACC. After adding the feature selection module (the fourth row), our method outperforms NON-LF by nearly 2.4% in terms of NMI. Besides, after we add both of these two modules, we obtain the best clustering performance. These results and observations affirm the effectiveness of each module within our method.

4.5. **Parameter Sensitivity.** As described in Section 4.4, our method incorporates three hyperparameters to control the effect of different modules on clustering. This section focuses on discussing the sensitivity of each parameter separately. Firstly, we fix the parameter $k = 5$ and analyze the parameters m and p. Three datasets, that is, Breast, Umist, and Yale, are selected for analysis. From the results in Figure 5, it can be seen that the performance of our model with different parameters m and p varies on the selected datasets. For example, our model obtains the best ACC on the Breast dataset when p ranges from $[1.0, 1.05]$ and p ranges from $[2, 4]$. On the Yale dataset, our method performs well when m ranges from $[1.02, 1.05]$ and p ranges from $[100, 400]$. In addition, it is important to note that the parameter p is used to control the number of retained features, where a larger value of p implies more features being retained. Interestingly, from the results on these three datasets, we can find out that increasing p does not always lead to higher ACC, indicating the presence of redundant features in the data. This once again demonstrates the importance of feature selection in this article.

Next, we fix the parameters $m = 1.1$ and $p = d/2$, and analyze the clustering performance changes of our model with different k on different datasets. The results are shown in Figure 6, with darker colors indicating higher values. From the results, it can be seen that our method is greatly influenced by the parameter k. For example, on the Umist dataset, a smaller value of k, such as $k = 1$, results in lower clustering performance, and a similar trend also occurs on the Yale dataset. Additionally, when k becomes too large, the clustering performance of our model may also deteriorate. For example, on the Yale dataset, when $k > 15$, our clustering performance shows a significant downward trend compared to that of $k = 5$. The above analysis highlights the fact that our method has different optimal values of k on different datasets. Generally speaking, across the six datasets of this experiment, our model approaches the optimal solution when $k = 5$.



(a)Breast (b)Umist (c)Yale

FIGURE 5. Sensitivity analysis of parameters m and p of our method on different datasets

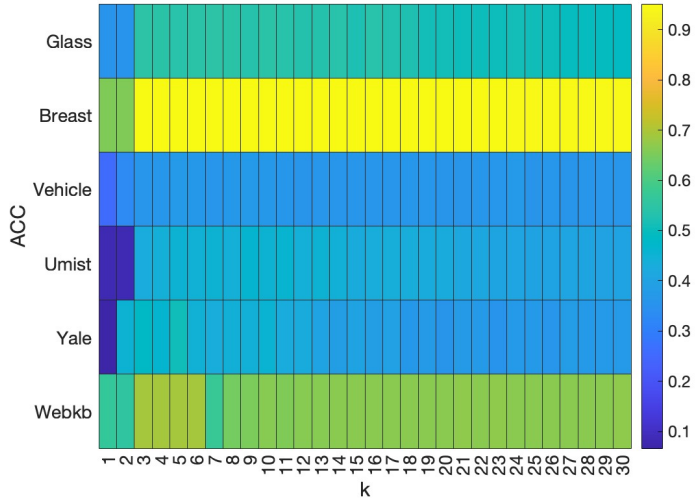


FIGURE 6. Sensitivity analysis of parameters k of our method on different datasets

5. **Conclusions.** This article presents a local fuzzy clustering method with feature selection, which can effectively address the challenges encountered in existing subspace clustering methods. These challenges include the neglect of local correlations among samples and vulnerability to noisy features. The proposed method overcomes these limitations by integrating local adjacency relationships among samples into the clustering center calculation process, thereby improving clustering accuracy. Moreover, our method can dynamically combine sparse feature selection to mitigate the influence of noise interference. The experimental results show that our proposed method can achieve excellent clustering performance across multiple datasets, verifying its effectiveness.

In future research, we aim to further optimize the construction process of similarity matrices. This can be achieved by integrating adaptive learning algorithms to dynamically construct similarity matrices and reduce their impacts on noisy features in high-dimensional data. In addition, we will also explore the application of the proposed method in other fields, such as image segmentation, natural language processing, and more.

Acknowledgment. This paper was supported by National Natural Science Foundation of China (Grant No. U1805264), National Natural Science Foundation of Fujian Province

(Grant Nos. 2021J011186, 2023J011428), Program of XMUT for High-level Talents Introduction Plan (Grant No. YKJ20016R), Scientific Research Fund of Fujian Provincial Education Department (Grant Nos. JAT200486, JAT200478), University Industry Research Fund of Xiamen (Grant Nos. 2023CXY0413, 2022CXY0416).

REFERENCES

[1] T.-Y. Wu, J. C.-W. Lin, Y. Zhang, and C.-H. Chen, "A grid-based swarm intelligence algorithm for privacy-preserving data mining," *Applied Sciences*, vol. 9, no. 4, 774, 2019.

[2] M. Ahmed, R. Seraj, and S. M. S. Islam, "The k-means algorithm: A comprehensive survey and performance evaluation," *Electronics*, vol. 9, no. 8, 1295, 2020.

[3] F. Zhang, T.-Y. Wu, J.-S. Pan, G. Ding, and Z. Li, "Human motion recognition based on svm in vr art media interaction environment," *Human-centric Computing and Information Sciences*, vol. 9, no. 1, 40, 2019.

[4] R. Chen, C. Dewi, S. Huang, and R. E. Caraka, "Selecting critical features for data classification based on machine learning methods," *Journal of Big Data*, vol. 7, no. 1, 52, 2020.

[5] K. Wang, Z. Chen, X. Dang, X. Fan, X. Han, C.-M. Chen, W. Ding, S.-M. Yiu, and J. Weng, "Uncovering hidden vulnerabilities in convolutional neural networks through graph-based adversarial robustness evaluation," *Pattern Recognition*, vol. 143, 109745, 2023.

[6] L. Xing, H. Zhao, Z. Lin, and B. Chen, "Mixture correntropy based robust multi-view k-means clustering," *Knowledge-Based Systems*, vol. 262, 110231, 2023.

[7] K. Wang, C.-M. Chen, M. S. Hossain, G. Muhammad, S. Kumar, and S. Kumari, "Transfer reinforcement learning-based road object detection in next generation iot domain," *Computer Networks*, vol. 193, 108078, 2021.

[8] J. Xu, J. Han, F. Nie, and X. Li, "Re-weighted discriminatively embedded k -means for multi-view clustering," *IEEE Transactions on Image Processing*, vol. 26, no. 6, pp. 3016–3027, 2017.

[9] X. Wang, P. Wu, Q. Xu, Z. Zeng, and Y. Xie, "Joint image clustering and feature selection with auto-adjoined learning for high-dimensional data," *Knowledge-Based Systems*, vol. 232, 107443, 2021.

[10] J. C. Bezdek, R. Ehrlich, and W. Full, "Fcm: The fuzzy c-means clustering algorithm," *Computers & Geosciences*, vol. 10, no. 2, pp. 191–203, 1984.

[11] H. Li, T. Jiang, and K. Zhang, "Efficient and robust feature extraction by maximum margin criterion," *IEEE Transactions on Neural Networks*, vol. 17, no. 1, pp. 157–165, 2006.

[12] C. Hou, F. Nie, D. Yi, and D. Tao, "Discriminative embedded clustering: A framework for grouping high-dimensional data," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 6, pp. 1287–1299, 2015.

[13] J. M.-T. Wu, Q. Teng, S. Huda, Y.-C. Chen, and C.-M. Chen, "A privacy frequent itemsets mining framework for collaboration in iot using federated learning," *ACM Transactions on Sensor Networks*, vol. 19, no. 2, 27, 2023.

[14] X.-D. Wang, R.-C. Chen, F. Yan, Z.-Q. Zeng, and C.-Q. Hong, "Fast adaptive k-means subspace clustering for high-dimensional data," *IEEE Access*, vol. 7, pp. 42639–42651, 2019.

[15] X. Zhu, S. Zhang, R. Hu, Y. Zhu, and J. Song, "Local and global structure preservation for robust unsupervised spectral feature selection," *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 3, pp. 517–529, 2018.

[16] T.-Y. Wu, J. C.-W. Lin, U. Yun, C.-H. Chen, G. Srivastava, X. Lv, V. E. Balas, and L. C. Jain, "An efficient algorithm for fuzzy frequent itemset mining," *Journal of Intelligent and Fuzzy Systems*, vol. 38, no. 5, pp. 5787–5797, 2020.

[17] J. Xu, J. Han, K. Xiong, and F. Nie, "Robust and sparse fuzzy k-means clustering," in *Twenty-Fifth International Joint Conference on Artificial Intelligence*, AAAI Press, 2016, pp. 2224–2230.

[18] R. Zhang, F. Nie, M. Guo, X. Wei, and X. Li, "Joint learning of fuzzy k-means and nonnegative spectral clustering with side information," *IEEE Transactions on Image Processing*, vol. 28, no. 5, pp. 2152–2162, 2019.

[19] X.-D. Wang, R.-C. Chen, and F. Yan, "High-dimensional data clustering using k-means subspace feature selection." *Journal of Network Intelligence*, vol. 4, no. 3, pp. 80–87, 2019.

[20] J. Lai, H. Chen, T. Li, and X. Yang, "Adaptive graph learning for semi-supervised feature selection with redundancy minimization," *Information Sciences*, vol. 609, pp. 465–488, 2022.

[21] J. Han, K. Xiong, and F. Nie, "Orthogonal and nonnegative graph reconstruction for large scale clustering," in *Twenty-Sixth International Joint Conference on Artificial Intelligence*, AAAI Press, 2017, pp. 1809–1815.

[22] P. Zhao, Y. Zhang, Y. Ma, X. Zhao, and X. Fan, "Discriminatively embedded fuzzy k-means clustering with feature selection strategy," *Applied Intelligence*, vol. 53, pp. 18959–18970, 2023.

[23] Z. Li, Y. Yang, J. Liu, X. Zhou, and H. Lu, "Unsupervised feature selection using nonnegative spectral analysis," in *Twenty-Sixth AAAI Conference on Artificial Intelligence*, AAAI Press, 2012, pp. 1026–1032.