

基本

学业/其他

联系/简历

学历

学前

论文/获奖

毕业/学位

奖/惩

注册/异动

培养计划

成绩

贷款/交费

保存

学号	S240231203	年级	2024	<div></div>	
姓名	金孟泽	曾用名			
姓名拼音	jinmengze 提	英文名称			
身高、体重	170 cm 68 kg	出生日期	2001-12-17		
性别	男	血型	未知		
婚姻状况	未婚	民族	汉族		
证件号码	610502200112177215	证件类型	居民身份证	国家/地区	中国
户口地	610502 陕西省西安市临渭区	政治面貌	共青团员	健康状况	健康或良好
户口类型	农村				
籍贯	610502 陕西省西安市	入团日期	2014-09-03	入团地点	陕西省临潼区秦陵?
出生地	610502 陕西省西安市临渭区	入党日期		入党地点	
现役军人	非军人	是否侨属	否	备注	

个人基本信息管理

基本

学业/其他

联系/简历

学历

学前

论文/获奖

毕业/学位

奖/惩

注册/异动

培养计划

成绩

贷款/交费

保存

入学日期	2024-09-09	生源地	陕西省	培养方式	非定向
学生类别	硕士	学生分类	专硕(全日)	入学方式	硕士统考
培养院系	计算机科学与技术学院	学生来源	应届本科	学习方式	脱产
专业	085404 计算机技术	培养层次	硕士研究生	办学形式	国家任务
研究方向	大数据智能计算	主修外语	英语	外语水平	一般
指导老师	程东东(1号)、王国胤(2号)	攻读学位	电子信息硕士	学制	3
学位院系	计算机科学与技术学院			预计答辩	2027-05-01
教育基地		技术职务	正高级	留学人员	<input type="checkbox"/>
工作时学历	--请选择--	参加工作		工作年限	(年)
行政职务	--请选择--	职业(工种)	原工资级别	单位电话	
工作单位	13224016169 单位类别: --请选择--	单位地址		单位邮编	
假期乘火车区间信息		起始车站	西安北站	终点车站	重庆北站
银行卡号	6228210214001419872	开户地银行	中国农业银行	所属银行	农业银行

学号：120241315009

姓名：姚光禹

- 基本信息
- 学籍信息
- 其他信息
- 联系方式
- 家庭成员
- 奖惩信息
- 成绩信息
- 选课信息
- 个人培养计划
- 学籍异动
- 学习简历
- 入学成绩
- 工作经历
- 其他主要社会活动
- 录取信息

学年	2025-2026	学期	1		
年级	2024	培养层次	硕士研究生	培养形式	专业学位型
学习形式	全日制	招生角度	正常报考统招	拿证程度	双证
学生类别	全日制专业学位硕士	学位类别		导师	程东东
学院名称	计算机科学与工程学院	系名称		专业名称	农业工程与信息技术（农业信息化）
专业方向		班级名称	2024级农业工程与信息技术（农业信息化）1班	学籍状态	在校生
				是否在校	是
报到注册状态	已注册	报到注册备注	学生自主报到注册	撤销报到注册原因	
报到时间	2024-09-07 17:01:22	注册时间	2024-09-07 17:01:22	未报到原因	
未注册原因		招生年度	2024	招生季度	秋季
招生学院	计算机科学与工程学院	招生专业	农业工程与信息技术（农业信息化）	培养方式	
				学制	3

Pseudo-label-Based Unsupervised Granular-ball Division and Fast Spectral Clustering for High-dimensional Data

1st Dongdong Cheng

College of Big Data and Intelligent Engineering
Yangtze Normal University
Chongqing, China
cdd@yznu.edu.cn

2th Xiaocui Jiang

Chongqing Key Laboratory of Computational Intelligence
Key Laboratory of Big Data Intelligent Computing
Chongqing University of Posts and Telecommunications
Chongqing, China
S230233020@stu.cqupt.edu.cn

3rd Shuyin Xia*

Key Laboratory of Cyberspace Big Data Intelligent Security
Chongqing University of Posts and Telecommunications
Chongqing, China
xiasy@cqupt.edu.cn

4th Guoyin Wang

National Center for Applied Mathematics in Chongqing
Chongqing Normal University
Chongqing, China
wanggy@cqupt.edu.cn

Abstract—With the swift advancement of information technology, vast amounts of high-dimensional data have accumulated across various domains. Clustering such data presents a significant challenge, as existing methods often suffer from slow execution speeds and reduced clustering accuracy. To tackle these issues, we introduce the granular-ball approach, which aims to decrease the number of sample points and enhance processing speed, while also improving clustering accuracy through feature selection. Granular-ball computing, a coarse-grained data representation technique, has demonstrated its advantages in enhancing classification and clustering models in recent studies. However, current granular-ball division techniques are inadequate for high-dimensional data. To confront the complexities arising from clustering high-dimensional data and improve upon existing granular-ball methods, this paper proposes a novel granular-ball division approach that leverages pseudo-labels and feature selection. This new method enables the identification of anchor points through an improved granular-ball division process, leading to the development of a fast spectral clustering algorithm for high-dimensional data, termed PLGB-FSC. Specifically, we initially employ weighted K-Means for feature to generate pseudo-labels. Subsequently, we conduct a primary stage of feature selection by utilizing the mutual information between pseudo-labels and features, thereby eliminating the interference caused by irrelevant features. We further refine the feature selection by combining standard deviation and pearson correlation coefficients to choose mutually independent features. Using these pseudo-labels, we then perform granular-ball division

to obtain anchor points. Lastly, we construct a similarity matrix between all sample points and the anchor points, and leveraging spectral clustering for definitive clustering outcomes. Experimental evaluations reveal that PLGB-FSC surpasses state-of-the-art algorithms such as W-KMeans, WGB, GB-USC, RC-PCA-SC, GLUFC, FGO, SFESA, SPCAFS, and LLSRFS, and it achieves higher accuracy and faster execution speed. The source code is available at <https://github.com/DongdongCheng/PLGB-FSC>.

Index Terms—Granular-ball Computing, Feature Selection, Pseudo-label, High-dimensional data clustering

I. INTRODUCTION

In the era of big data, fueled by rapid advancements in information technology, the scale and dimensionality of diverse data have been experiencing an exponential growth. Extracting useful information efficiently from high-dimensional datasets poses a crucial challenge within the realm of data science. High-dimensional data pervade numerous application domains, encompassing gene expression data [1], machine learning [2], data mining [3], and image processing [4], where traditional analytical and processing techniques often fall short. Consequently, the development of efficient clustering algorithms tailored for high-dimensional data is of paramount importance.

To overcome the challenges posed by high-dimensional data, feature selection [5] has emerged as a pivotal dimensionality reduction technique, attracting substantial attention. Li et al. [6] reframed PCA as a task focused on minimizing reconstruction errors, incorporating an $l_{2,p}$ -norm regularization to induce sparsity in the projection matrix. Subsequently, this sparse orthogonal projection matrix was leveraged to identify and select salient features. When running on data with 7,000-dimensional features, the processing time reached

This work is supported by project of National Natural Science Foundation of China under Grant 62221005, 62376045, 62222601, 62176033, 62006029 in part by Postdoctoral Innovative Talent Support Program of Chongqing under Grant CQBX2021024, in part by Natural Science Foundation of Chongqing (China) under Grant CSTB2022NSCQ-MSX0258, cstc2019cyj-cxttX0002, cstc2021ycjh-bgzxm0013, CSTB2022TIAD-KPX0196, and in part by Project of Chongqing Municipal Education Commission, China under Grant KJZD-K202301402, KJQN202201413, HZ2021008, HZ2021014, KJZD-M202201401.

*: Corresponding author.

2,300 seconds. Wang et al. [7] first built an adaptive anchor-neighbor graph and preserved data manifold via spectral analysis. They then regularized the projection to match the low-dim embedding. Finally, a constraint of $l_{2,0}$ -norm enhanced subspace sparsity. Running the algorithm on features with the same dimensionality took 180 seconds. It can be observed that the runtime for feature selection of high-dimensional data is generally not ideal.

Traditional spectral clustering methods are computationally expensive, the eigen decomposition of Laplacian matrix requires a time complexity of $O(n^3)$ and a space complexity of $O(n^2)$, where n is the number of sample points. To alleviate the computational burden of spectral clustering, researchers have adopted anchor points techniques [8] to reduce the size of the similarity matrix. Anchor points are used as representatives of all sample points and can well reflect the distribution characteristics of the samples. Subsequently, they construct a similarity matrix based on the similarity between sample points and anchor points, and perform clustering operations on this matrix. Liu et al. [9] used K-nearest neighbors to generate anchor points, while Liu et al. [10], Nie et al. [11], and Zhang et al. [12] employed the balanced k-means-based hierarchical k-means (BKHK) algorithm. However, these methods struggle to capture the true distribution of high-dimensional data, leading to suboptimal similarity matrix and poor clustering performance.

Granular-ball computing is a new modeling approach in the field of multi-granularity cognitive computing. Xia et al. [13] implemented a coarse-to-fine approach to create granular-balls of various granularity. By utilizing granular-balls of different sizes, they effectively represented and encompassed the sample space. These granular-balls serve as a means of both covering and representing data, providing a precise characterization of the sample space by serving as input elements. Leveraging this concept, they devised innovative and efficient machine learning models, such as SVM [14], K-Means [15] and so on. Despite its application in diverse models, existing granular-ball generation methods fall short in managing high-dimensional data, because they rely on Euclidean distance to assess similarity between sample points.

Using granular-balls as input not only reduces the data volume but also improves computational efficiency, while maintaining accuracy. However, the existing granular-ball computing model [13], [14] uses a supervised metric, purity, to evaluate the quality of a granular-ball. The purity is defined as the proportion of the majority of samples belonging to the same class. If the purity of a granular-ball is larger than the threshold, it means that the granular-ball has high quality and well represents the distribution of local data. In unsupervised clustering, since the data lack labels, inspired by the work in [16], we expect to use pseudo-labels to calculate the purity of granular-balls and employ it to assess whether a granular-ball needs to be divided. To address the failure of Euclidean distance in high-dimensional data, we introduce feature selection in the process of granular-ball division. By integrating feature selection and pseudo-label into the granular-

ball division process, our method effectively addresses the challenges of high-dimensional clustering. Based on pseudo-labels, we define the fake-purity of granular-ball and use it to determine whether a granular-ball should be divided. We perform unsupervised granular-ball division to obtain granular-balls and use the centers of the granular-balls as anchor points, so that it improves the effectiveness of spectral clustering for high-dimensional data. In simple terms, the method uses mutual information for the first stage feature selection, and introduces pseudo-label and the second stage feature selection to aid in the granular-balls division. Based on the anchor points obtained from the granular-balls, a similarity matrix is constructed with all sample points, and finally, spectral clustering is applied to obtain the final clustering results. The key contributions of this paper lie in the following aspects:

- 1) We utilize weighted K-Means to assign pseudo-labels to each sample point and introduce the notion of “fake-purity” to evaluate the granular-balls’ quality based on these pseudo-labels. Consequently, we propose a novel granular-ball division technique.

- 2) We integrate feature selection and pseudo-labels into the granular-ball division process to ensure the final granular-balls better fit the distribution of high-dimensional data. The first stage feature selection uses mutual information between features and pseudo-labels, filtering out irrelevant features. In the granular-balls division, we further select independent features based on feature importance for each granular-ball.

- 3) Experiments on 9 real datasets show that compared to the current newest method LLSRFS, our approach improves clustering accuracy by 10.74% and speeds up the running time by 74.04%.

II. RELATED WORK

A. Feature Selection

Unsupervised feature selection encompasses three categories: filter methods [17], wrapper methods [18], and embedded methods [19]. Filter methods operate independently of models, aiding large-scale data preprocessing but may overlook feature interdependencies, leading to redundancy [20]. Wrapper methods integrate feature selection with data mining, enhancing task performance but demanding high computation, potentially weakening generalization. Embedded methods seamlessly incorporate selection in model training, offering efficiency and excelling in large feature spaces with tight timelines.

However, hybrid feature selection is currently the most popular. This approach amalgamates the strengths of diverse feature selection methodologies [21], leading to the development of numerous hybrid feature selection techniques over time. In the work of Xu et al. [22], a hybrid approach to feature selection tailored for high-dimensional gene expression data was introduced. This algorithm initially employs spectral clustering to eliminate redundant features, subsequently allocating the remaining features evenly following a strategic indexing approach. Subsequently, a population-driven evolutionary multi-objective genetic algorithm is leveraged to explore the

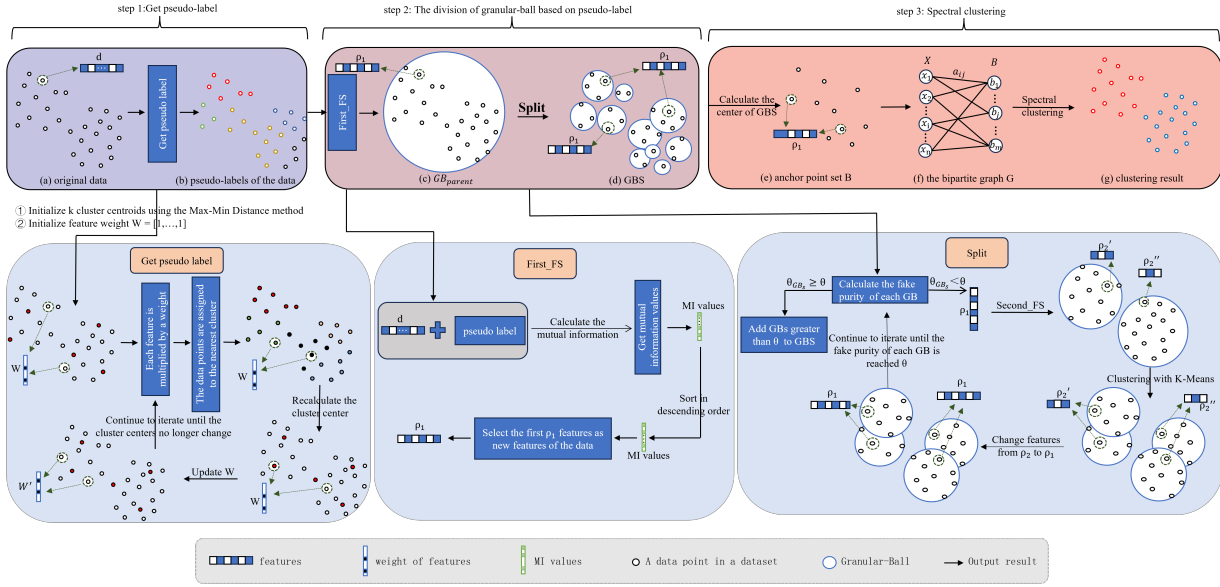


Fig. 1. Overall framework of the PLGB-FSC.

filtered feature subsets, evaluating candidate subsets through both intra-cluster and inter-cluster criteria. Alhenawi et al. [23] devised a hybrid strategy for microarray data analysis, fusing an ensemble filtering approach with an enhanced Intelligent Water Drops (IWD) algorithm functioning as a wrapper. In each iteration of the enhanced IWD, one of three local search heuristics, tabu search, a novel local search mechanism, or hill climbing, is integrated. The original IWD algorithm is augmented with a correlation coefficient filter as a guiding principle for selecting the subsequent node, fostering improved feature selection.

B. Granular-Ball Computing

Human cognition exhibits a marked preference for large-scale processing [24], with the brain naturally inclined to progress from broad to specific, coarse to fine, contingent upon the context. Capitalizing on this cognitive trait, Xia et al. [13] proposed granular-ball computing. This methodology entails approximating data distributions using various granular-balls: larger balls signify coarse granularity, while smaller ones denote fine granularity. Given a dataset X , we define GB_j ($j = 1, 2, \dots, m$) as a granular-ball generated from X , with m representing the total number of granular-balls. Each granular-ball GB_j comprises n_j data points, denoted as $GB_j = \{x_i \mid i = 1, 2, \dots, n_j\}$. In normal form, the center point c_j of GB_j is calculated as the mean of all data points x_i and the radius r_j of GB_j is the average distance from all data points to its center c_j .

Researchers have conducted extensive research on granular-ball computing. In the unsupervised learning domain, Cheng et al. [25] innovated by introducing granular-ball for efficient, high-quality anchor-based manifold learning. Xie et al. [26] introduced a weighted granular-ball clustering method with local iterations for parameter-free weight calculation and high-

dimensional data. In the context of fuzzy stream clustering, Xie et al. [27] introduced a granular-ball structure with fuzziness for efficient, accurate data stream clustering, mitigating efficiency issues and cluster overlap due to concept drift. In the supervised learning domain, Yang et al. [28] introduced an advanced granular-ball generation method, DBGBC, which is based on DBSCAN and significantly improves the quality of the granular-balls. In the field of Natural Language Processing (NLP), Wang et al. [29] proposed a novel GBRAIN framework that leverages dynamic granular-ball clustering and coarse-grained representation learning to combat label noise. In terms of improvements to granular-ball, Xia et al. [30] introduced a method that replaces K-Means with division to expedite the generation of granular-balls, achieving comparable accuracy to existing methods while doing so. Current granular-ball methods are primarily utilized for large-scale datasets. However, in high-dimensional data, numerous irrelevant and redundant features can interfere, rendering Euclidean distance ineffective. Consequently, these methods are not ideal for high-dimensional scenarios. To address this, feature selection is integrated into the granular-ball division process. Besides, to evaluate the quality of granular-balls in an unsupervised way, we introduce pseudo-label and use it to compute fake-purity.

C. Spectral Clustering

Spectral clustering is a technique that uses graph theory and linear algebra for data clustering. It stands out in its ability to handle complex, non-convex shapes and high-dimensional datasets. The main idea of spectral clustering is to represent the relationships between sample points by constructing the Laplacian Matrix of a graph, and then use its eigenvalues and eigenvectors for clustering.

Recent improvements in spectral clustering have been quite significant. Huang et al. [31] used mixed representatives and

k-nearest neighbors for sparse affinity submatrices, achieving fast clustering via spectral cuts. Liu et al. [9] introduced granular-ball computing for manifold learning and proposed a fast spectral embedding clustering algorithm. Xie et al. [32] proposed a granular-ball-based spectral clustering that optimizes similarity matrix construction for large datasets, reducing time and memory usage while maintaining clustering accuracy. Zhang et al. [12] presented an efficient ensemble clustering approach that leveraged K-nearest neighbors, anchor graphs, and SVD to reduce runtime on large datasets. Bai et al. [33] introduced a self-constrained spectral clustering algorithm that learns results and constraints simultaneously. Nie et al. [34] proposed Direct Spectral Clustering (DSC), which optimizes the clustering process by learning a weighted indicator matrix and structured similarity matrix. However, these techniques fail to accurately represent the true distribution of high-dimensional data, resulting in a suboptimal similarity matrix and poor clustering performance.

III. OUR PROPOSED METHOD

In this section, we introduce a granular-ball division method for high-dimensional data and propose an efficient clustering algorithm based on it. The method first uses weighted K-Means on features to obtain pseudo-labels [16] for the dataset. These pseudo-labels are used to eliminate irrelevant features and assess the quality of granular-balls during the division process, ensuring high-quality anchor points. Specifically, we select features by calculating the mutual information between each feature (d features) and the pseudo-labels to obtain ρ_1 features ($\rho_1 < d$). For each granular-ball, we use standard deviation [35] and pearson correlation coefficients [36] to select mutually independent features ρ_2 ($\rho_2 < \rho_1$) for division, and evaluate the quality of granular-balls using pseudo-labels to determine whether further division is needed. Then, we obtain m anchor points based on the centers of granular-balls and construct a similarity matrix between all sample points and the anchor points. Finally, an efficient spectral clustering algorithm is used to obtain the final clustering results. The overall framework is shown in Fig. 1.

Algorithm 1 Get Pseudo-labels

Input: Dataset $X \in \mathbb{R}^{n \times d}$, the number of clusters k , and feature weight $W^0 = [1, \dots, 1] \in \mathbb{R}^d$;
Output: Pseudo-labels $\hat{Y} \in \mathbb{R}^{n \times 1}$;
1: Initialize cluster centers C^0 using Max-Min Distance.
2: Assign the data to the nearest cluster center C^0 to obtain the partition matrix U^0 .
3: $t = 0$;
4: **while** True **do**
5: Fix $\hat{C} = C^t$ and $\hat{W} = W^t$, update U^t ;
6: Fix $\hat{U} = U^{(t+1)}$ and $\hat{W} = W^t$, update C^t ;
7: Fix $\hat{U} = U^{(t+1)}$ and $\hat{C} = C^{(t+1)}$, update W^t ;
8: **if** $P(\hat{U}, \hat{C}, W^t) == P(\hat{U}, \hat{C}, W^{(t+1)})$ or $t > 20$ **then**
9: **break**
10: **end if**
11: $t = t + 1$;
12: **end while**
13: **return** \hat{Y} ;

A. Pseudo-label Based on Weighted K-Means

Given a dataset $X = \{x_1, x_2, \dots, x_n\} \in \mathbb{R}^{n \times d}$, where x_i presents an individual data sample, with n samples and d features. The K-Means clustering algorithm encounters challenges in dealing with high-dimensional data, potentially compromising clustering accuracy. Therefore, to enhance the precision of pseudo-labels, we introduce a weighted K-Means approach that can yield better clustering results. The specific algorithmic process of getting pseudo-label is outlined in Algorithm 1. The objective function is as follows:

$$P(U, C, W) = \min \sum_{q=1}^k \sum_{i=1}^n \sum_{j=1}^d u_{i,q} w_j^\beta d(x_{i,j}, c_{q,j}), \quad (1)$$

$$\text{s.t.} \quad \sum_{q=1}^k u_{i,q} = 1, \quad 1 \leq i \leq n.$$

where

- U serves as an $n \times k$ partitioning matrix, with each element $u_{i,q}$ being a binary indicator. When $u_{i,q} = 1$, it signifies that the data point i belongs to cluster q . The constraint condition indicates that each sample only belongs to one cluster.
- $C = \{C_1, C_2, \dots, C_k\}$ represents the central points of the k distinct clusters.
- $d(x_{i,j}, c_{q,j})$ denotes the distance or dissimilarity between object i and the center of cluster q for the j -th variable, which is computed using the squared Euclidean distance. Thus,

$$d(x_{i,j}, c_{q,j}) = (x_{i,j} - c_{q,j})^2. \quad (2)$$

- β is determined according to commonly used empirical rules. In this paper, β is set to 3 or 10.
- $W = [w_1, w_2, \dots, w_d]$ represents the weights for the d variables, where w_j denotes the weight of the j -th feature. Thus,

$$\hat{w}_j = \begin{cases} 0 & \text{if } D_j = 0 \\ \frac{1}{\sum_{t=1}^h \left(\frac{D_j}{D_t}\right)^{\frac{1}{\beta-1}}} & \text{if } D_j \neq 0, \end{cases} \quad (3)$$

$$\text{s.t.} \quad \sum_{j=1}^d w_j = 1, \quad 0 \leq w_j \leq 1.$$

where $D_j = \sum_{q=1}^k \sum_{i=1}^n u_{i,q} (x_{i,j} - c_{q,j})^2$ represents the sum of the distances from each sample point in the cluster to the cluster center, and h is the number of features where $D_j \neq 0$. Initially, W is set to 1 for all components.

Below, we will prove how W is obtained.

Proof.

Given fixed U and C , the objective function for updating W becomes

$$P(\hat{U}, \hat{C}, W) = \sum_{j=1}^d w_j^\beta \sum_{q=1}^k \sum_{i=1}^n (\hat{u}_{i,q}) d(x_{i,j}, c_{q,j}) = \sum_{j=1}^d w_j^\beta D_j. \quad (4)$$

If $D_j = 0$, then $w_j = 0$. For $D_j \neq 0$, we incorporate the constraint $\sum_{j=1}^d w_j = 1$ into the objective function using Lagrange multipliers. Let α be the multiplier, and $\phi(W, \alpha)$ be the Lagrangian function:

$$\phi(W, \alpha) = \sum_{j=1}^d w_j^\beta D_j + \alpha \left(\sum_{j=1}^d w_j - 1 \right). \quad (5)$$

To minimize $\phi(W, \alpha)$, the gradients must vanish. Thus, we need to compute the partial derivatives with respect to W and α :

$$\frac{\partial \phi(\hat{W}, \hat{\alpha})}{\partial \hat{w}} = \beta \hat{w}_j^{\beta-1} D_j + \hat{\alpha} = 0 \quad \text{for } 1 \leq j \leq h, \quad (6)$$

$$\frac{\partial \phi(\hat{W}, \hat{\alpha})}{\partial \hat{\alpha}} = \sum_{j=1}^d \hat{w}_j - 1 = 0. \quad (7)$$

According to Eq. (6), we obtain

$$\hat{w}_j = \left(\frac{-\hat{\alpha}}{\beta D_j} \right)^{\frac{1}{\beta-1}} \quad \text{for } 1 \leq j \leq h. \quad (8)$$

Substitute Eq. (8) into Eq. (7):

$$\sum_{t=1}^h \left(\frac{-\hat{\alpha}}{\beta D_t} \right)^{\frac{1}{\beta-1}} = 1. \quad (9)$$

Then, solving for $\hat{\alpha}$, we obtain:

$$(-\hat{\alpha})^{-\frac{1}{\beta-1}} = \frac{1}{\sum_{t=1}^h \left(\frac{1}{\beta D_t} \right)^{\frac{1}{\beta-1}}}. \quad (10)$$

Substituting Eq. (10) into Eq. (8) yields:

$$\hat{w}_j = \frac{1}{\sum_{t=1}^h \left(\frac{D_j}{D_t} \right)^{\frac{1}{\beta-1}}}. \quad (11)$$

B. Feature Selection in the First Stage

Mutual Information (MI) [37] is used to measure the correlation or the degree of information sharing between two random variables. However, high-dimensional data contain irrelevant features. To reduce interference during granular-ball division and similarity matrix construction, the mutual information values of each feature are obtained according to Def. 1, sorted in descending order, and the top ρ_1 features are selected to eliminate the interference of irrelevant features.

Definition 1 (MI between Pseudo-labels and Features): Given a feature set $D = [D_1, D_2, \dots, D_d]$ and pseudo-labels \tilde{Y} , the mutual information $MI(D_i; \tilde{Y})$ between each feature D_i and the pseudo-labels \tilde{Y} is calculated as follows:

$$MI(D_i; \tilde{Y}) = \sum_{\tilde{d} \in D_i} \sum_{\tilde{y} \in \tilde{Y}} p(\tilde{d}, \tilde{y}) \log_2 \left(\frac{p(\tilde{d}, \tilde{y})}{p(\tilde{d})p(\tilde{y})} \right), \quad (12)$$

where $MI(D_i; \tilde{Y}) \in \mathbb{R}^{d \times 1}$, $p(\tilde{d}, \tilde{y})$ serves as the joint probability distribution, capturing the likelihood of both $D_i = \tilde{d}$ and $\tilde{Y} = \tilde{y}$ occurring concurrently. Meanwhile, $p(\tilde{d})$ and $p(\tilde{y})$ represent the marginal distributions of D_i and \tilde{Y} , quantifying the probabilities of D_i and \tilde{Y} attaining specific values \tilde{d} and \tilde{y} .

A larger $MI(D_i; \tilde{Y})$ score indicates a stronger relevance of the i -th feature to the pseudo-labels, thereby facilitating the selection of features characterized by significant mutual information.

C. Feature Selection in the Second Stage

In this section, we delve into the second stage feature selection tailored for granular-ball division. This refinement stage stems from the feature redundancy and the various degrees of feature importance and redundancy within individual granular-balls. Notably, features exhibit distinct behaviors across different granular-balls, with some displaying high discriminative capabilities in certain granular-balls while appearing redundant in others. Consequently, it becomes imperative to undertake an individualized feature selection process for each granular-ball, ensuring that only the most pertinent features contribute to the division process. Therefore, we select different features when dividing different granular-balls, which can solve the high-dimensional problem of existing granular-ball generating methods.

By conducting feature selection specifically for each granular-ball, we can precisely pinpoint and retain the most discriminative features that effectively differentiate samples within particular subset. This approach minimizes redundant information, thereby bolstering the algorithm's efficiency and accuracy. It ensures that the feature selection process is targeted and effective, yielding a feature set that not only diminishes redundancy but also exhibits superior discriminative capabilities across diverse granular-balls.

Definition 2 (The Standard Deviation of a Feature): The standard deviation dis is used to measure the degree of deviation of data points from the mean in a dataset. The calculation is as follows:

$$u = \frac{1}{s} \sum_{i=1}^s x_i, \quad (13)$$

$$dis = \sqrt{\frac{1}{s} \sum_{i=1}^s (x_i - u)^2}, \quad (14)$$

where $dis \in \mathbb{R}^{1 \times \rho_1}$, s represents the number of sample points in a given granular-ball, and u represents the mean of all sample points in a granular-ball.

The larger the dis , the higher the discernibility of the feature. A feature which can effectively distinguish different

classes has greater variability, so its standard deviation is larger. This dispersion signifies its effectiveness in distinguishing classes, pivotal for classification tasks. Hence, this paper suggests assessing a feature's class differentiation ability via its standard deviation. A higher deviation implies greater distinction between class values, fostering superior classification outcomes.

The pearson correlation coefficient assesses the degree of association between two variables. The lesser the magnitude of its absolute value, the weaker the correlation between the two variables is.

Definition 3 (Feature Independence Based on Pearson Correlation Coefficient): The feature independence is defined as the reciprocal of the sum of the absolute values of the pearson correlation coefficients. For the feature with the highest discernibility, its independence is defined as the reciprocal of the minimum sum of the absolute values of the pearson correlation coefficients. The calculation is as follows:

$$ind_i = \begin{cases} \frac{1}{\min(\sum_{b=1}^{\rho_1} |r_{D_a, D_b}|)} & \text{if } \max_{a=1, \dots, \rho_1} \{dis_a\} \\ \frac{1}{\sum_{b=1}^{\rho_1} |r_{D_a, D_b}|} & \text{otherwise} \end{cases}, \quad (15)$$

$$r_{D_a, D_b} = \frac{\sum_{i=1}^n (x_{i,a} - \bar{D}_a)(x_{i,b} - \bar{D}_b)}{\sqrt{\sum_{i=1}^n (x_{i,a} - \bar{D}_a)^2 \sum_{i=1}^n (x_{i,b} - \bar{D}_b)^2}}, \quad (16)$$

where $a, b = 1, \dots, \rho_1$, $i = 1, \dots, s$ and $ind \in \mathbb{R}^{\rho_1 \times 1}$. D_b represents the b -th feature, and ρ_1 ($\rho_1 < d$) denotes the number of features selected after applying mutual information for the first feature selection.

Definition 4 (The Significance of a Feature): The significance of features D , denoted as $score \in \mathbb{R}^{\rho_1 \times 1}$, is defined as the product of feature discernibility and feature independence. The calculation is as follows:

$$score = dis^T \times ind, \quad (17)$$

where $score_i$ represents the score of the i -th feature, and a larger $score_i$ indicates that the feature D_i is more important. Because the standard deviation reflects the dispersion of a feature, while the pearson correlation coefficient measures the linear correlation between features. By combining the product of both, we can effectively filter out features that are both redundant and have little variation, thus further reducing the negative impact of redundant features on distance calculations.

Before dividing the granular-ball, we first calculate the feature discernibility (Def. 2) and feature independence (Def. 3) for all sample points within the granular-ball. By multiplying the computed standard deviations and pearson correlation coefficients for each feature, we obtain the final feature importance score (Def. 4). A higher $score_i$ indicates that the feature is more important. Features are then ranked from highest to lowest based on their scores, and the top ρ_2 features are selected. The granular-ball is then divided into two sub-granular-balls using 2-means clustering based on these ρ_2 features. The rationale for performing the second stage feature

selection before dividing the granular-balls will be explained in the ablation study in Section IV.F.

D. Granular-ball Division Based on Pseudo-label

The acquisition of anchor points typically relies on either random sampling or K-Means clustering. However, despite their simplicity and efficiency, both methods fall short in adequately describing the dataset's distribution, resulting in anchor points that lack representativeness. In light of this, Xia et al. [13] introduced granular-balls for data adaptiveness, assuming similar distributions cluster into the granular-ball. In this paper, we combine pseudo-labels and feature selection with granular-balls to generate high-quality anchor points. The process of granular-ball division is illustrated in Fig. 2.

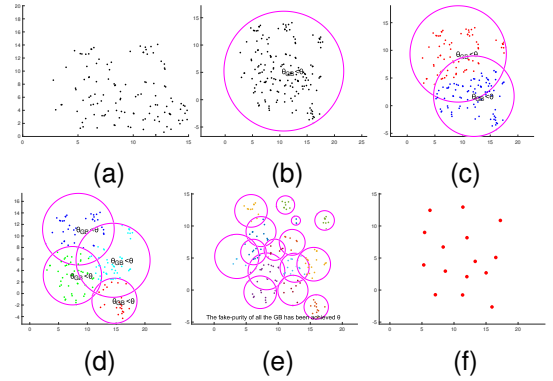


Fig. 2. Granular-ball division process. (a) The initial dataset. (b) The entire dataset is initially considered a single granular-ball. (c) Using the second feature selection, 2-means splits the granular-ball into two. (d) Following the second iteration, the granular-balls undergo refinement, resulting in four distinct balls. (e) The final stage partitions the data into m granular-balls. (f) The centers of the granular-balls (anchors).

Definition 5 (Fake-Purity of Granular-ball): Given a granular-ball GB where the data is divided into t clusters (p_1, \dots, p_t , $1 \leq t \leq k$), the fake-purity of the granular-ball is defined as the ratio of the number of data points in the largest cluster to the total number of data points in the granular-ball. The calculation is as follows:

$$F - purity(GB) = \frac{\max(|p_i|)}{\sum_{i=1}^k |p_i|}, \quad (18)$$

where p_i represents the number of sample points in the i -th cluster.

Definition 6 (Anchor Point Set): Given a granular-ball GB , the center point of the granular-ball GB is called an anchor point. The anchor point set B is the collection of these center points of granular-balls. For the f -th granular-ball GB_f , its center is denoted as $center_f$. The calculation is as follows:

$$center_f = \frac{1}{s} \sum_{i=1}^s x_i, \quad x_i \in GB_f, \quad (19)$$

$$B = \{center_1, \dots, center_m\}. \quad (20)$$

The entire dataset initializes as a single granular-ball GB . We set a fake-purity threshold θ to determine whether a

granular-ball should be divided. If the fake-purity of the granular-ball exceeds the threshold θ , no division occurs; else, 2-means is applied for further division. Algorithm 2 outlines the process to obtain the anchor point set B .

Clustering high-dimensional data involves selecting relevant features via mutual information, but redundancy may persist. To enhance granular-balls quality, we apply the second stage feature selection (Section III.C) to eliminate redundant features before dividing the parent granular-ball GB_{parent} into GB_{child1} and GB_{child2} using 2-means on ρ_2 features. Post-division, child granular-balls revert to ρ_1 features since each granular-ball may have different important features. Each division is done according to this rule. Division of child granular-balls hinges on their fake-purity (Def. 5). If fake-purity reaches the threshold θ , division halts; else, 2-means proceeds. This iterative process terminates when all granular-balls' fake-purity meets the threshold θ .

Algorithm 2 Granular-ball Division

Input: Dataset $X \in \mathbb{R}^{n \times d}$, Pseudo-labels $\tilde{Y} \in \mathbb{R}^{n \times 1}$;
Output: Anchor point set $B \in \mathbb{R}^{m \times \rho_1}$;
1: Select the top ρ_1 features index $IND1$ with the highest MI values from d features, where MI is calculated by Eq. (12);
2: $X' = X[:, IND1]$;
3: Initialize $X' \in \mathbb{R}^{n \times \rho_1}$ as a granular-ball GB ;
4: $GBS = \{GB\}$;
5: **while** True **do**
6: $OLD_GB_NUM = \text{length}(GBS)$;
7: **for** each GB_i in GBS **do**
8: **if** $F - \text{purity}(GB_i) > \theta$ **then**
9: continue;
10: **else**
11: Select the top ρ_2 features index $IND2$ with the highest $score$ values from the ρ_1 features of X' ;
12: $GB'_i = GB_i[:, IND2]$;
13: Use 2-means to divide GB'_i into $Child1$ and $Child2$;
14: $GB_{child1} = GB'_i[Child1, :]$;
15: $GB_{child2} = GB'_i[Child2, :]$;
16: $GBS = GBS \cup GB_{child1} \cup GB_{child2}$;
17: $GBS = GBS - GB_i$; // removing GB_i
18: **end if**
19: **end for**
20: $NEW_GB_NUM = \text{length}(GBS)$;
21: **if** $OLD_GB_NUM == NEW_GB_NUM$ **then**
22: break;
23: **end if**
24: **end while**
25: Calculate the center of each GB in GBS as the anchor point set B according to Eq. (19) - (20);
26: **return** B ;

After obtaining m granular-balls, we then use Def. 6 to designate the center point of each granular-ball as an anchor point, thereby obtaining the anchor point set B .

A granular-ball's quality is gauged by its fake-purity. High fake-purity signifies a granular-ball dominated by samples of a single class, indicating its goodness. Low fake-purity, conversely, reflects uneven sample distribution, indicating a poor granular-ball. Thus, for low fake-purity granular-balls, we employ recursion to further divide them until achieving sufficient fake-purity. However, it's important to note that setting

the fake-purity threshold too high can result in smaller and more numerous granular-balls. These overly small granular-balls may fail to capture the underlying distribution structure of the data, which may adversely affect subsequent clustering efficiency.

E. Spectral Clustering

a) *Similarity Matrix:* We construct the similarity matrix between anchor points and sample points. When computing the similarity between anchor points and sample points, we only use their ρ_1 features selected from the first stage feature selection, instead of the original d features, because these features are relevant to the cluster label and precisely capture the relationships between points.

Based on the granular-balls, m anchor points are obtained, which form the anchor point set $B \in \mathbb{R}^{m \times \rho_1}$. The sample points are in the set $X' \in \mathbb{R}^{n \times \rho_1}$. The similarity matrix represents the relationships between each sample and its nearest anchor points. It is constructed as follows.

Step 1: Use KNN to find the nearest ξ anchor points for each sample point, which facilitates the construction of a bipartite graph connecting sample points and anchor points.

Step 2: Use the Gaussian kernel function to construct the similarity matrix $A \in \mathbb{R}^{n \times m}$, where $A = \{a_{ij}\}_{n \times m}$ is defined as:

$$a_{ij} = \begin{cases} \exp\left(-\frac{\|x_i - b_j\|_2^2}{2\sigma^2}\right) & \text{if } b_j \in N_\xi(x_i) \\ 0 & \text{otherwise} \end{cases}, \quad (21)$$

where $\sigma = \frac{1}{n\xi} \sum_{i=1}^n \sum_{j=1}^\xi \|x_i - b_j\|_2^2$, and $N_\xi(x_i)$ denotes the set of the ξ nearest anchor points to x_i .

b) *Bipartite Graph Partitioning:* The similarity matrix A reflects the relationship between sample points and anchor points. By utilizing a bipartite graph structure, transfer cut [38] can be employed to effectively partition the graph, resulting in the final clustering outcome.

Given a dataset X , an anchor set B , and a similarity matrix A , let the bipartite graph be $G = (X, B, A)$. The complete adjacency matrix of the bipartite graph can be represented as $E = \begin{bmatrix} A^T & A \end{bmatrix}$, where $E \in \mathbb{R}^{(n+m) \times (n+m)}$. The graph G can then be partitioned by solving the following eigenvalue problem:

$$Lu = \alpha Du, \quad (22)$$

where $L = D - E$ is the Laplacian matrix, and $D \in \mathbb{R}^{(n+m) \times (n+m)}$ is the degree matrix corresponding to E . To reduce the time and space complexity during eigenvalue decomposition, we utilize the transfer cut method. This method transforms the problem of $n + m$ nodes in graph G into a smaller graph G_s with m nodes, where the complete adjacency matrix becomes $E_s = A^T D^{-1} A$, and $E_s \in \mathbb{R}^{m \times m}$. The eigenvalue problem in the above Eq. (22) is thus transformed into:

$$L_s v = \beta D_s v, \quad (23)$$

where $L_s = D_s - E_s$, and $D_s \in \mathbb{R}^{m \times m}$ is the degree matrix corresponding to E_s .

Finally we can perform eigen decomposition on the Laplacian matrix L_s by considering the relationship between α and β as well as u and v as described in [38]. Following this meticulous preparation, we integrate K-Means clustering into the process, harnessing its power to group the data points within the reduced-dimensional manifold embedding.

The overall algorithm is outlined in Algorithm 3.

Algorithm 3 PLGB-FSC

Input: Dataset $X \in \mathbb{R}^{n \times d}$, the number of clusters k , feature weight $W^0 = [1, \dots, 1] \in \mathbb{R}^d$, and the number of nearest neighbors ξ ;

Output: Clustering label;

- 1: $\tilde{Y} = \text{Get Pseudo-labels}(X, k, W^0)$;
 - 2: Anchor point set $B = \text{Granular-ball Division}(X, \tilde{Y})$;
 - 3: Calculate ξ nearest neighbors using KNN for sample points X and anchor point set B ;
 - 4: Calculate similarity matrix A according to Eq. (21);
 - 5: Utilize transfer cut based on the bipartite graph structure to obtain low-dimensional embedding Y ($Y \in \mathbb{R}^{n \times k}$);
 - 6: Execute K-Means clustering on the low-dimensional embedding Y to categorize the data, yielding the final clustering;
-

F. Time Complexity

Given a dataset $X \in \mathbb{R}^{n \times d}$, where n is the number of samples, d is the number of features, and k is the number of clusters. This algorithm mainly consists of three steps: 1) pseudo-label generation; 2) granular-ball division based on feature selection; 3) spectral clustering. Next, we analyze the time complexity of PLGB-FSC in detail.

In step 1, we initialize k centroids using the Max-Min Distance method with a time complexity of $O(dnk^2)$. Each iteration involves assigning samples to centroids $O(dnk)$, updating centroids $O(dn)$, and adjusting feature weights $O(dnk + d^2)$. Considering the number of iterations t is set to a constant 20, therefore, the time complexity of step 1 is $O(dnk^2)$. In step 2, the first stage feature selection has a time complexity of $O(dn)$. In each iteration of granular-ball division, we first conduct the second stage feature selection with $O(\rho_1^2 n)$ time complexity and then use 2-means to divide the granular-ball with $O(\rho_2 n)$ time complexity. Since the number of iterations in granular-ball division process is $\log m$, granular-ball division has a time complexity of $O((\rho_1^2 n + \rho_2 n) \log m)$. Thus, the time complexity of step 2 is $O(dn + \rho_1^2 n \log m)$. In step 3, the time complexity for utilizing k-nearest neighbors is $O(\rho_1 n \log m)$, constructing the similarity matrix A requires $O(n\xi\rho_1)$ time complexity, eigen decomposition has a time complexity of $O(m^3)$, performing K-Means clustering on Y entails $O(nk^2)$ time complexity. The time complexity of step 3 is $O(\rho_1 n \log m + m^3 + nk^2)$. The overall time complexity is thus streamlined to $O(dnk^2 + \rho_1^2 n \log m + m^3)$.

IV. EXPERIMENT

In this section, we evaluate PLGB-FSC by comparing it with baselines and state-of-the-art techniques on real-world datasets. Experiments were conducted in MATLAB R2018b

on a system with an Intel i5-13500H CPU, 16GB RAM, and Windows 11. Clustering performance was assessed using NMI (Normalized Mutual Information) [39], ACC (Accuracy) [40], and F-measure [41].

A. Dataset

Our experiments were conducted on nine real-world datasets with various sizes and dimensions, including four face datasets: COIL20¹, ORL¹, warpPIE10P¹, and Yale¹; three biological datasets: GLIOMA¹, ALLAML¹, and TOX_171¹; a hand written image data: USPS¹ and one cancer dataset: Suncancer [42]. Detailed information about the datasets is provided in Table I.

TABLE I
DATASETS

Dataset	Object	Dimensions	Classes
USPS	7291	256	10
COIL20	1440	1024	20
ORL	400	1024	40
Yale	165	1024	15
warpPIE10P	210	2420	10
GLIOMA	50	4434	4
TOX_171	171	5748	4
ALLAML	72	7129	2
Suncancer	174	7909	2

B. Comparison Methods

In this experiment, we compared a traditional unsupervised clustering algorithm, two weighted clustering algorithms, and state-of-the-art feature selection clustering algorithms (These comparison algorithms inherently include the clustering process).

(1) baseline-SC [43]: The traditional spectral clustering that constructs k-nearest neighbor graph and uses the eigenvectors of the normalized Laplacian matrix to perform clustering.

(2) W-KMeans [44]: A K-Means clustering algorithm that can automatically compute feature weights.

(3) WGB [26]: A novel weighted granular-ball structure that dynamically optimizes feature weights during granular-ball division.

(4) GB-USC [25]: A spectral clustering algorithm that introduces granular-ball to obtain anchor points, where K-Means divides the granular-ball if its sample count exceeds threshold p .

(5) GLUFC [45]: An unsupervised feature selection algorithm which combines graph learning and the $l_{2,0}$ -norm sparsity constraint.

(6) FGOC [46]: A feature selection algorithm that is based on feature grouping and orthogonal constraints.

(7) SFESA [7]: A feature selection algorithm that builds an adaptive anchor nearest neighbor graph and approximates the projected data to the low-dimensional structure via a regularization term, with the $l_{2,0}$ -norm constraining the projection matrix for enhanced sparsity.

¹<https://jundongl.github.io/scikit-feature/datasets.html>

TABLE II
AVERAGE PERFORMANCE (IN TERMS OF NMI, ACC, AND F-MEASURE) FOR REAL DATASETS (%)
(THE OPTIMAL RESULTS ARE HIGHLIGHTED IN BOLD AND THE SUBOPTIMAL RESULTS ARE UNDERLINED)

NMI	USPS	COIL20	ORL	Yale	warpPIE10P	GLIOMA	TOX_171	ALLAML	Sucancer
SC	<u>80.29</u>	50.11	24.60	22.07	9.86	11.13	3.14	1.34	2.27
W-KMeans	59.77	68.11	73.90	47.13	29.72	59.43	26.4	18.94	3.78
WGB	17.76	68.93	75.57	44.20	37.39	28.52	25.23	13.18	19.72
GB-USC	78.41	<u>84.73</u>	74.63	44.98	26.36	50.31	16.16	15.47	23.17
RC-PCA-SC	65.79	73.31	79.95	40.03	61.08	19.99	8.15	1.66	1.17
GLUFC	63.11	80.86	75.14	50.16	34.38	52.36	22.87	12.98	12.20
FGOC	71.27	74.58	68.02	40.46	<u>59.91</u>	19.80	12.52	18.36	9.97
SFESA	55.57	69.36	70.56	45.71	36.78	12.74	28.60	14.43	12.48
SPCAFS	55.31	63.25	67.57	45.25	49.18	16.00	28.14	<u>22.61</u>	10.73
LLSRFS	-	81.79	<u>79.47</u>	57.96	43.77	<u>59.08</u>	30.15	17.13	<u>22.47</u>
PLGB-FSC	86.35	93.58	82.29	<u>57.79</u>	<u>59.91</u>	57.99	<u>30.02</u>	28.93	23.17
ACC	USPS	COIL20	ORL	Yale	warpPIE10P	GLIOMA	TOX_171	ALLAML	Sucancer
SC	66.47	27.15	12.60	18.18	17.71	36.00	29.82	51.39	53.45
W-KMeans	61.84	53.88	48.75	38.18	28.09	56.00	50.29	<u>76.38</u>	50.00
WGB	16.38	53.01	54.98	35.45	34.62	49.1	47.39	73.61	67.75
GB-USC	70.44	<u>76.45</u>	54.05	38.06	26.19	55.60	43.63	73.61	52.30
RC-PCA-SC	69.26	58.13	<u>63.75</u>	35.16	<u>54.07</u>	46.00	36.84	63.89	52.87
GLUFC	68.76	71.46	54.75	43.03	30.95	58.00	47.95	72.22	64.94
FGOC	<u>75.47</u>	62.90	45.38	33.58	52.33	47.40	41.17	75.14	<u>67.70</u>
SFESA	59.70	53.47	48.85	38.67	33.05	42.80	49.24	70.56	67.24
SPCAFS	59.21	50.74	45.60	39.15	42.76	45.60	48.19	64.27	61.84
LLSRFS	-	74.03	62.40	50.67	37.05	<u>68.00</u>	53.68	75.00	67.24
PLGB-FSC	83.24	88.68	65.75	<u>49.09</u>	54.29	74.00	<u>50.88</u>	81.94	67.24
F-measure	USPS	COIL20	ORL	Yale	warpPIE10P	GLIOMA	TOX_171	ALLAML	Sucancer
SC	75.24	32.40	16.02	20.34	19.74	41.88	37.70	60.24	66.50
W-KMeans	65.32	57.59	53.12	44.21	34.23	66.49	50.14	<u>76.60</u>	66.16
WGB	19.23	58.11	57.97	39.16	39.67	48.15	50.94	<u>73.39</u>	64.04
GB-USC	76.81	<u>78.61</u>	57.69	42.83	28.99	61.86	47.53	74.07	63.36
RC-PCA-SC	71.58	61.53	64.94	38.62	<u>62.47</u>	50.37	44.51	68.98	<u>66.61</u>
GLUFC	71.07	73.90	59.50	44.93	48.18	58.31	47.47	72.63	62.68
FGOC	73.28	64.86	49.99	36.24	56.54	52.01	46.18	75.14	66.83
SFESA	63.11	58.82	52.51	42.90	37.06	49.77	<u>52.94</u>	70.91	63.72
SPCAFS	62.54	52.98	48.82	42.45	48.03	51.44	<u>50.41</u>	67.92	60.28
LLSRFS	-	75.60	<u>66.01</u>	57.13	44.96	69.08	52.75	75.37	63.36
PLGB-FSC	87.15	89.96	68.71	57.95	63.15	73.55	55.09	82.02	63.37

(8) SPCAFS [6]: An unsupervised feature selection algorithm in which PCA is formulated as a reconstruction error minimization with $l_{2,p}$ -norm regularization to sparsify the projection matrix, used to select discriminative features.

(9) LLSRFS [47]: An unsupervised feature selection algorithm in which an exponential weighting mechanism is introduced to guide feature distribution and explore the data structure in the feature subspace.

(10) RC-PCA-SC [48]: A spectral clustering algorithm that introduces an elementary cost and storage reduction method for spectral clustering and principal component analysis.

C. Parameters Setting

The number of features for the comparison algorithms is searched from the grid $\{20, 40, 60, \dots, 200\}$. All other parameter configurations adhere strictly to the original paper's specifications. In the PLGB-FSC algorithm, the number of iterations t to get the pseudo-labels is set to 20. During the first stage feature selection, the number of features ρ_1 retained for each dataset is as follows: USPS (256), COIL20 (800), ORL (450), warpPIE10P (400), Yale (400), GLIOMA (400), Sucancer (400), ALLAML (2000), and TOX_171 (1000). For the second stage feature selection, the number of retained features ρ_2 is searched from the grid $\{50, 60, 70, \dots, 200\}$.

The selection of ρ_1 and ρ_2 is based on the parameter sensitivity analysis shown in Fig. 6. When constructing the similarity matrix, ξ is set to 5 to determine the nearest neighbor anchor points. The fake-purity threshold θ is searched from the grid $\{0.70, 0.75, 0.80, 0.85, 0.90, 0.95\}$.

D. Experimental Results and Time Analysis

The experimental outcomes for all methods across the nine benchmark datasets are summarized in Table II, with the top-performing results emphasized in bold and the second-best underlined. The results are denoted by “-” if the algorithm cannot obtain results due to memory limitations. The PLGB-FSC method significantly outperforms the baseline-SC, showcasing substantial improvements across all datasets. Specifically, PLGB-FSC achieves average NMI, ACC, and F-measure scores of 57.78%, 68.35%, and 71.22%, surpassing the state-of-the-art high-dimensional data clustering algorithm, LLSRFS, which reports averages of 48.98%, 61.01%, and 63.03% in these metrics. This translates to improvements of 8.80%, 7.34%, and 8.19% for PLGB-FSC. The PLGB-FSC algorithm exhibits a marginal deficit in NMI and ACC compared to LLSRFS on the Yale and TOX_171 datasets. Similarly, on the Sucancer dataset, its ACC and F-measure fall slightly short of the best results by 0.51% and 3.46%, respectively. On the

TABLE III
THE AVERAGE RUNNING TIME OF 5 RUNS FOR EACH ALGORITHM ON REAL DATASETS (S)
(THE OPTIMAL RESULTS ARE HIGHLIGHTED IN BOLD AND THE SUBOPTIMAL RESULTS ARE UNDERLINED)

Time	USPS	COIL20	ORL	Yale	warpPIE10P	GLIOMA	TOX_171	ALLAML	Sucancer
SC	<u>1.80</u>	<u>0.61</u>	1.23	0.7	0.3	<u>0.07</u>	<u>0.23</u>	<u>0.05</u>	<u>0.19</u>
W-KMeans	1.55	1.56	0.67	0.17	0.52	0.93	1.82	2.44	3.04
WGB	186.72	52.03	11.75	3.14	33.06	20.17	136.74	64.44	175.41
GB-USC	19.05	0.38	0.07	0.02	0.05	0.02	0.09	0.04	0.17
RC-PCA-SC	6.62	0.90	<u>0.33</u>	<u>0.15</u>	<u>0.20</u>	0.21	0.33	0.24	0.43
GLUFC	61.75	4.42	1.70	1.88	13.07	41.26	88.07	143.44	208.07
FGOC	52901	49186	192.85	63	1018	568.87	1350.72	3741.2	12024
SFESA	16.29	1.79	0.85	0.32	5.71	42.12	94.13	189.95	254.14
SPCAFS	4.94	21.67	49.19	58.84	35.48	3257.55	1804.58	2321.28	19585.12
LLSRFS	-	230.62	17.03	0.77	13.63	17.87	43.20	66.07	50.26
PLGB-FSC	88.95	8.58	1.47	0.67	2.04	3.63	6.15	7.66	9.18

GLIOMA dataset, the sole setback is a 1.09% decrease in NMI relative to LLSRFS. Nevertheless, PLGB-FSC shines brightly on most datasets, notably surpassing LLSRFS with a 6% and 6.94% increase in ACC on the GLIOMA and ALLAML datasets, respectively. This exhaustive experimental validation underscores the potency and efficacy of the proposed method.

As evident from Table III, the PLGB-FSC method demonstrates superior speed compared to WGB, FGOC, SPCAFS, and LLSRFS across all datasets. Notably, on the Sucancer dataset, PLGB-FSC operates 211.46 times quicker than SPCAFS. Furthermore, it outperforms GLUFC and SFESA in terms of runtime on the majority of datasets. While SC, W-KMeans, GB-USC, and RC-PCA-SC may run faster, their accuracy lags significantly behind PLGB-FSC. Across all datasets, our algorithm boasts an average runtime of just 14.26 seconds, significantly outpacing GLUFC (62.63s), FGOC (13449.52s), SFESA (67.25s), SPCAFS (3015.41s), and LLSRFS (54.93s). In fact, compared to the fastest competitor LLSRFS, our average runtime improvement is a remarkable 74.04%. As the feature dimensionality escalates, our algorithm's performance in terms of speed becomes even more pronounced.

E. Experiment on Real Hyperspectral Image Dataset

To validate the clustering performance of the proposed algorithm, we conducted experiments using the Pavia University² hyperspectral image dataset. This dataset, captured by the ROSIS-03 system in 2003, covers part of Pavia, Italy. It contains 115 spectral bands in the 0.43-0.86 μm range with 1.3-meter spatial resolution. After removing 12 noisy bands, 103 bands remain for use. The dataset consists of 610 \times 340 pixels (2,207,400 total), but only 42,776 pixels represent land cover types, categorized into 9 groups, such as Trees, Asphalt, and Bricks. A sample image and ground truth are shown in Fig. 3 (a).

Since the PaviaU dataset is of size 610 \times 340 \times 103, most of the comparison algorithms experience memory issues when running on the entire dataset. Therefore, we followed the approach from the comparison algorithm GB-USC, randomly selecting 5000 data samples for running, and assigning the

labels of the remaining samples to the labels of the nearest point among the 5000 samples. Despite selecting only 5000 samples, some comparison algorithms still encounter memory issues and cannot provide results. Therefore, We compared the algorithms SC, W-KMeans, WGB, GB-USC, GLUFC, and SFESA, which successfully provided results.

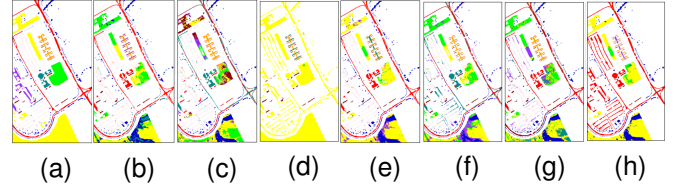


Fig. 3. Comparison of clustering results in PaviaU dataset. (a) Ground truth. (b) SC. (c) W-KMeans. (d) WGB. (e) GB-USC. (f) GLUFC. (g) SFESA. (h) PLGB-FSC.

TABLE IV
THE CLUSTERING RESULTS OF ALGORITHMS FOR THE PAVIAU DATASET
(THE OPTIMAL RESULTS ARE HIGHLIGHTED IN BOLD)

	NMI	ACC	F-measure	Time
SC	57.15	53.99	56.73	83.15
W-KMeans	50.84	47.52	51.95	3.03
WGB	39.58	41.23	35.93	52.43
GB-USC	59.85	58.37	60.92	1.50
GLUFC	53.34	50.20	55.00	88.49
SFESA	52.93	49.91	54.50	71.97
PLGB-FSC	62.22	61.78	62.10	12.37

Fig. 3 and Table IV showcase the experimental results, with the top-performing outcomes highlighted in bold. Our algorithm PLGB-FSC demonstrates a superior clustering performance compared to other algorithms. Specifically, PLGB-FSC surpasses the second-best results by 2.37% in NMI, 3.41% in ACC, and 1.18% in F-measure. Furthermore, it exhibits an advantage in terms of time consumption, making it an efficient and effective solution.

F. Ablation Experiment

In this section, we delve into assessing the influence of various components on clustering performance: the first and second stages of feature selection, and the utilization of granular-balls. To do so, we created four ablated models by

²<https://www.ehu.es/ccwintco/index.php/HyperspectralRemoteSensingScenes>

TABLE V
RESULTS OF ABLATION STUDY ON 9 DATASETS(%)
(THE OPTIMAL RESULTS ARE HIGHLIGHTED IN BOLD)

	Methods	USPS	COIL20	ORL	Yale	warpPIE10P	GLIOMA	TOX_171	ALLAML	Sucancer
NMI	without first_FS	84.75	92.73	82.21	55.36	47.78	54.73	26.25	23.49	23.02
	without second_FS	81.78	84.69	77.66	52.27	57.74	53.69	19.63	12.04	23.17
	without all_FS	83.22	87.03	79.80	52.86	35.99	50.77	12.38	10.81	20.67
	without anchor points	85.99	90.37	81.95	52.10	54.71	53.27	17.61	22.48	20.92
	generate anchor points with W-KM	74.30	91.05	79.21	54.68	56.50	53.13	24.01	15.41	23.17
	PLGB-FSC	86.35	93.58	82.29	55.37	58.44	56.74	30.02	28.93	23.17
ACC	without first_FS	82.57	84.04	65.15	47.76	38.95	63.20	49.71	77.78	67.13
	without second_FS	69.84	75.15	57.90	44.97	50.67	62.00	46.55	70.56	67.24
	without all_FS	72.24	78.25	60.85	46.30	32.57	62.00	40.35	69.17	66.67
	without anchor points	82.29	75.08	65.25	44.85	50.95	66.00	43.86	79.17	66.90
	generate anchor points with W-KM	74.09	85.49	61.75	46.42	51.33	70.00	49.47	72.22	67.24
	PLGB-FSC	83.24	88.68	65.75	48.73	53.05	74.00	50.88	81.94	67.36
F-measure	without first_FS	86.47	86.80	68.22	53.83	46.26	68.86	49.18	77.40	63.37
	without second_FS	77.27	78.32	61.26	49.98	61.37	64.01	48.83	71.12	63.37
	without all_FS	79.67	81.15	64.14	52.48	37.88	62.76	45.45	69.03	63.19
	without anchor points	87.07	81.43	68.07	51.14	63.31	66.85	49.08	79.06	63.26
	generate anchor points with W-KM	77.08	86.85	64.49	52.83	61.83	70.28	50.12	72.63	63.37
	PLGB-FSC	87.15	89.96	68.71	55.31	62.49	73.76	55.09	82.02	63.37

TABLE VI
THE RESULTS OF ANCHOR POINTS OBTAINED USING DIFFERENT METHODS(%)
(THE OPTIMAL RESULTS ARE HIGHLIGHTED IN BOLD AND THE SUBOPTIMAL RESULTS ARE UNDERLINED)

	Methods	USPS	COIL20	ORL	Yale	warpPIE10P	GLIOMA	TOX_171	ALLAML	Sucancer
NMI	farthest-point_SM	81.49	86.60	73.24	51.19	49.29	51.31	13.26	20.94	23.17
	WGB_DM_SM	83.44	88.27	77.24	51.63	24.22	48.16	25.04	12.16	<u>22.47</u>
	PLGB-FSC	<u>86.35</u>	<u>93.58</u>	<u>82.29</u>	<u>55.37</u>	<u>58.44</u>	<u>56.74</u>	30.02	<u>28.93</u>	23.17
	real_label	88.15	94.03	83.52	59.29	60.82	59.38	<u>28.47</u>	42.70	23.17
ACC	farthest-point_SM	78.19	80.00	55.00	44.24	44.76	60.00	44.44	77.78	<u>67.24</u>
	WGB_DM_SM	79.05	77.59	58.00	44.85	25.71	57.20	46.78	72.22	<u>67.24</u>
	PLGB-FSC	<u>83.24</u>	<u>88.68</u>	<u>65.75</u>	<u>48.73</u>	53.05	<u>74.00</u>	50.88	81.94	67.36
	real_label	83.42	89.10	69.50	49.70	<u>50.48</u>	76.00	<u>50.29</u>	87.50	67.36
F-measure	farthest-point_SM	82.41	81.14	57.62	47.71	52.28	59.78	45.15	77.96	63.37
	WGB_DM_SM	80.34	80.87	60.85	49.44	30.08	58.41	49.30	72.45	63.37
	PLGB-FSC	<u>87.15</u>	<u>89.96</u>	<u>68.71</u>	<u>55.31</u>	62.49	<u>73.76</u>	55.09	<u>82.02</u>	63.37
	real_label	87.60	90.29	71.57	57.78	<u>56.59</u>	75.57	<u>54.15</u>	87.29	63.37

excluding these elements: one without the first stage feature selection (“without first_FS”), another without the second stage (“without second_FS”), a third omitting both stages (“without all_FS”), and a fourth discarding the granular-balls approach (“without anchor points”). Furthermore, we utilize the W-KMeans algorithm to generate granular-balls, designated as “generate anchor points with W-KM”. As seen in Table V, the optimal outcomes are emphasized in bold.

Omitting the first stage feature selection reveals that irrelevant features hinder the construction of the similarity matrix, adversely affecting performance. Similarly, excluding the second stage selection highlights the importance of removing redundant features, as they negatively impact the granular-ball division. When all feature selection stages are absent, accuracy notably declines, underscoring their significance. In the absence of granular-balls (“without anchor points”), noisy data obscures the data’s intrinsic structure. Altering the granular-ball generation approach (“generate anchor points with W-KM”), we observed that its performance did not exceed our proposed method. This is primarily attributed to the fact that W-KMeans focuses on capturing the overall

data distribution from a global standpoint, resulting in the neglect of local features and intricate details within the dataset. Consequently, the anchors produced lack adequate typicality and representativeness.

Hence, our model consistently surpasses the five ablation variants across datasets, validating the synergy of its components for clustering. Notably, the “without anchor points” model achieves the highest F-measure on warpPIE10P, but this is only 0.82% higher than PLGB-FSC, due to the sensitivity of F-measure to Precision and Recall. It is worth noting that the granular-ball we proposed and the granular-ball division based on pseudo-labels and feature selection have a positive impact on the experimental results.

G. Comparison of Different Granular-ball Division Methods

To demonstrate that the anchor points generated by the granular-balls in our algorithm are superior. We replaced the anchor points generation method in PLGB-FSC with the methods from [25] and [26], referred to as the “farthest-point splitting method” and “weighted GB based on DM splitting method”, abbreviated as “farthest-point_SM” and “WGB_DM_SM”, respectively. Additionally, we replaced the

pseudo-labels in the PLGB-FSC algorithm with real labels, referred to as “real_label”.

As shown in Table VI, the optimal results are highlighted in bold and the suboptimal results are underlined. The “farthest-point_SM” and “WGB_DM_SM” methods match our algorithm’s accuracy solely on the Sucancer dataset, displaying notable variations across other datasets. This indicates that the anchor points generation methods from [25] and [26] are not as effective as the method used in PLGB-FSC. This also confirms that incorporating the second stage feature selection and using fake-purity to assess the quality of granular-balls is effective and significantly improves performance. The average NMI, ACC, and F-score for “real_label” are 59.95%, 69.26%, and 71.58%, for PLGB-FSC are 57.78%, 68.35%, and 71.22%. Compared to PLGB-FSC, “real_label” exhibit a 2.17% higher NMI, a 0.91% higher ACC, and a 0.36% higher F-score. This implies that although the results achieved using pseudo-labels are slightly lower than those using real labels, the differences are minimal and fall within an acceptable range. This further underscores the rationality, feasibility, and effectiveness of using pseudo-labels. More importantly, in scenarios where real labels are scarce or unattainable, our method demonstrates broader applicability and emphasizes its practical value, showcasing exceptional performance in practical applications.

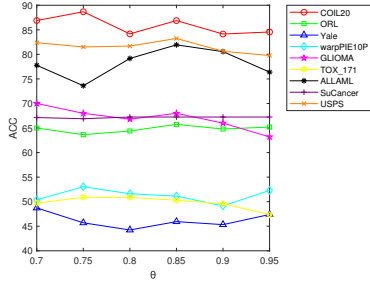


Fig. 4. ACC results on eight datasets with different θ when ρ_1 and ρ_2 is fixed.

H. Parameter Sensitivity Analysis

Fig. 4 illustrates the various influence of distinct thresholds θ on the outcomes, with ρ_1 and ρ_2 held constant. Among the datasets, ALLAML exhibits the most significant variability, where the performance gap between the best and worst cases amounts to 8.33%. Conversely, the remaining datasets experience fluctuations no greater than 4.51%. Notably, the results for Sucancer remain strikingly consistent. Moreover, the optimal value of θ varies across datasets, as exemplified by 0.70 for Yale, 0.75 for COIL20, and 0.85 for ORL.

Fig. 5 demonstrates the sensitivity to the choice of selected features, ρ_1 and ρ_2 , with the threshold θ held constant. Note that the number of ρ_2 must be smaller than ρ_1 , resulting in some NaN values in (a). TOX_171 and ALLAML exhibit pronounced variations. Specifically, TOX_171 is primarily influenced by ρ_1 , though changes beyond 1000 features are minimal. For ALLAML, ρ_1 has a more significant impact

than ρ_2 , with ρ_2 peaking around 380 when ρ_1 is fixed. The remaining datasets show lesser sensitivity to both ρ_1 and ρ_2 .

In summary, the experimental outcomes reveal that parameters θ , ρ_1 and ρ_2 exert various levels of impact on the performance across different datasets. Specifically, the ALLAML and TOX_171 datasets demonstrate the highest sensitivity to parameter adjustments, showcasing marked fluctuations. Conversely, the remaining datasets exhibit relative stability. Consequently, PLGB-FSC typically demonstrates parameter insensitivity for the majority of datasets, with only a minority potentially requiring parameter optimization.

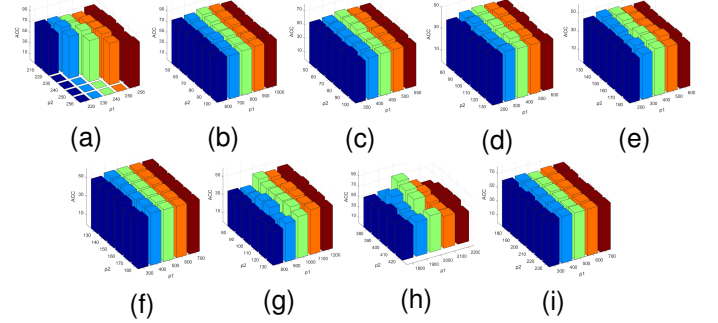


Fig. 5. Parameter sensitivity demonstration on eight public datasets when $\theta = 0.85$. The x-axis represents ρ_1 , the y-axis represents ρ_2 , and the z-axis represents ACC. (a) USPS. (b) COIL20. (c) ORL. (d) Yale. (e) warpPIE10P. (f) GLIOMA. (g) TOX_171. (h) ALLAML. (i) Sucancer.

V. CONCLUSION

This paper introduces a fast spectral embedding clustering algorithm PLGB-FSC based on feature selection and pseudo-labeled granular-balls for high-dimensional data unsupervised clustering. It integrates feature selection with pseudo-labeled granular-balls. PLGB-FSC first assigns pseudo-labels using feature-weighted K-Means and selects relevant features based on mutual information. A second round of feature selection utilizes standard deviation and pearson correlation coefficient to aid granular-ball division, with fake-purity assessing their quality. Anchor points from these granular-balls form a similarity matrix with all samples, facilitating spectral clustering for final results. Comprehensive experiments on 9 benchmarks and a hyperspectral dataset validate PLGB-FSC’s efficacy and superiority over 10 other algorithms, demonstrating faster performance and superior clustering outcomes due to its innovative feature selection and granular-ball approach.

However, our approach entails a number of parameters, with two of them optimized through a grid search. To tackle these hurdles, we aim to explore adaptive granular-ball division strategies in the future. This will eliminate the dependency on a predetermined threshold θ during the division phase, thereby enhancing the model’s versatility across varied datasets. As for the quantity of features selected, we are committed to devising solutions to streamline this aspect, paving the way for promising avenues of future research.

REFERENCES

- [1] L. Ou-Yang, X.-F. Zhang, and H. Yan, "Sparse regularized low-rank tensor regression with applications in genomic data analysis," *Pattern Recognition*, vol. 107, p. 107516, 2020.
- [2] G. A. Khan, J. Hu, T. Li, B. Diallo, and S. Du, "Multi-view subspace clustering for learning joint representation via low-rank sparse representation," *Applied Intelligence*, vol. 53, no. 19, pp. 22 511–22 530, 2023.
- [3] Y. Li, L. Hu, and W. Gao, "Robust sparse and low-redundancy multi-label feature selection with dynamic local and global structure preservation," *Pattern Recognition*, vol. 134, p. 109120, 2023.
- [4] P. Huang, Z. Kong, M. Xie, and X. Yang, "Robust unsupervised feature selection via data relationship learning," *Pattern Recognition*, vol. 142, p. 109676, 2023.
- [5] X. Li, H. Zhang, R. Zhang, Y. Liu, and F. Nie, "Generalized uncorrelated regression with adaptive graph for unsupervised feature selection," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 5, pp. 1587–1595, 2018.
- [6] Z. Li, F. Nie, J. Bian, D. Wu, and X. Li, "Sparse pca via $l_{2,p}$ -norm regularization for unsupervised feature selection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 4, pp. 5322–5328, 2021.
- [7] J. Wang, H. Wang, F. Nie, and X. Li, "Sparse feature selection via fast embedding spectral analysis," *Pattern Recognition*, vol. 139, p. 109472, 2023.
- [8] Q. Qiang, B. Zhang, F. Wang, and F. Nie, "Fast multi-view discrete clustering with anchor graphs," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 11, 2021, pp. 9360–9367.
- [9] S. Liu, D. Cheng, and J. Xie, "Granular-ball-based fast spectral embedding clustering algorithm for large-scale data," in *Proceedings of the 2024 16th International Conference on Machine Learning and Computing*, 2024, pp. 16–20.
- [10] C. Liu, F. Nie, R. Wang, and X. Li, "Scalable fuzzy clustering with anchor graph," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 8, pp. 8503–8514, 2022.
- [11] F. Nie, J. Xue, W. Yu, and X. Li, "Fast clustering with anchor guidance," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.
- [12] R. Zhang, S. Hang, Z. Sun, F. Nie, R. Wang, and X. Li, "Anchor-based fast spectral ensemble clustering," *Information Fusion*, vol. 113, p. 102587, 2025.
- [13] S. Xia, Y. Liu, X. Ding, G. Wang, H. Yu, and Y. Luo, "Granular ball computing classifiers for efficient, scalable and robust learning," *Information Sciences*, vol. 483, pp. 136–152, 2019.
- [14] S. Xia, X. Lian, G. Wang, X. Gao, Q. Hu, and Y. Shao, "Granular-ball fuzzy set and its implement in svm," *IEEE Transactions on Knowledge and Data Engineering*, 2024.
- [15] S. Xia, D. Peng, D. Meng, C. Zhang, G. Wang, E. Giem, W. Wei, and Z. Chen, "Ball k -means: Fast adaptive clustering with no bounds," *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 1, pp. 87–99, 2020.
- [16] D.-H. Lee *et al.*, "Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks," in *Workshop on Challenges in Representation Learning, ICML*, vol. 3, no. 2. Atlanta, 2013, p. 896.
- [17] J. Ircio, A. Lojo, U. Mori, and J. A. Lozano, "Mutual information based feature subset selection in multivariate time series classification," *Pattern Recognition*, vol. 108, p. 107525, 2020.
- [18] M. G. Altarabichi, S. Nowaczyk, S. Pashami, and P. S. Mashhadi, "Surrogate-assisted genetic algorithm for wrapper feature selection," in *2021 IEEE Congress on Evolutionary Computation (CEC)*. IEEE, 2021, pp. 776–785.
- [19] J.-S. Wu, M.-X. Song, W. Min, J.-H. Lai, and W.-S. Zheng, "Joint adaptive manifold and embedding learning for unsupervised feature selection," *Pattern Recognition*, vol. 112, p. 107742, 2021.
- [20] Z. Zhao, L. Wang, and H. Liu, "Efficient spectral feature selection with minimum redundancy," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 24, no. 1, 2010, pp. 673–678.
- [21] N. Almugren and H. Alshamlan, "A survey on hybrid feature selection methods in microarray gene expression data for cancer classification," *IEEE Access*, vol. 7, pp. 78 533–78 548, 2019.
- [22] Z. Xu, F. Yang, C. Tang, H. Wang, S. Wang, J. Sun, and Y. Zhang, "Fg-hfs: A feature filter and group evolution hybrid feature selection algorithm for high-dimensional gene expression data," *Expert Systems with Applications*, vol. 245, p. 123069, 2024.
- [23] E. Alhenawi, R. Al-Sayyed, A. Hudaib, and S. Mirjalili, "Improved intelligent water drop-based hybrid feature selection method for microarray data processing," *Computational Biology and Chemistry*, vol. 103, p. 107809, 2023.
- [24] L. Chen, "Topological structure in visual perception," *Science*, vol. 218, no. 4573, pp. 699–700, 1982.
- [25] D. Cheng, S. Liu, S. Xia, and G. Wang, "Granular-ball computing-based manifold clustering algorithms for ultra-scalable data," *Expert Systems with Applications*, vol. 247, p. 123313, 2024.
- [26] J. Xie, C. Hua, S. Xia, Y. Cheng, G. Wang, and X. Gao, "W-gbc: An adaptive weighted clustering method based on granular-ball structure," in *2024 IEEE 40th International Conference on Data Engineering (ICDE)*. IEEE, 2024, pp. 914–925.
- [27] J. Xie, M. Dai, S. Xia, J. Zhang, G. Wang, and X. Gao, "An efficient fuzzy stream clustering method based on granular-ball structure," in *2024 IEEE 40th International Conference on Data Engineering (ICDE)*. IEEE, 2024, pp. 901–913.
- [28] J. Yang, Z. Liu, S. Xia, G. Wang, Q. Zhang, S. Li, and T. Xu, "3wc-gbnrs++: A novel three-way classifier with granular-ball neighborhood rough sets based on uncertainty," *IEEE Transactions on Fuzzy Systems*, 2024.
- [29] Z. Wang, T. Zhang, S. Xia, L. Lin, and G. Wang, "Gbrain: Combating textual label noise by granular-ball based robust training," in *Proceedings of the 2024 International Conference on Multimedia Retrieval*, 2024, pp. 357–365.
- [30] S. Xia, X. Dai, G. Wang, X. Gao, and E. Giem, "An efficient and adaptive granular-ball generation method in classification problem," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 35, no. 4, pp. 5319–5331, 2022.
- [31] D. Huang, C.-D. Wang, J.-S. Wu, J.-H. Lai, and C.-K. Kwok, "Ultra-scalable spectral clustering and ensemble clustering," *IEEE Transactions on Knowledge and Data Engineering*, vol. 32, no. 6, pp. 1212–1226, 2019.
- [32] J. Xie, W. Kong, S. Xia, G. Wang, and X. Gao, "An efficient spectral clustering algorithm based on granular-ball," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 9, pp. 9743–9753, 2023.
- [33] L. Bai, J. Liang, and Y. Zhao, "Self-constrained spectral clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 4, pp. 5126–5138, 2022.
- [34] F. Nie, C. Liu, R. Wang, and X. Li, "A novel and effective method to directly solve spectral clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.
- [35] K. Pearson, "Contributions to the mathematical theory of evolution," *Philosophical Transactions of the Royal Society of London. A*, vol. 185, pp. 71–110, 1894.
- [36] I. Cohen, Y. Huang, J. Chen, J. Benesty, J. Benesty, J. Chen, Y. Huang, and I. Cohen, "Pearson correlation coefficient," *Noise Reduction in Speech Processing*, pp. 1–4, 2009.
- [37] S. Lall, D. Sinha, A. Ghosh, D. Sengupta, and S. Bandyopadhyay, "Stable feature selection using copula based mutual information," *Pattern Recognition*, vol. 112, p. 107697, 2021.
- [38] Z. Li, X.-M. Wu, and S.-F. Chang, "Segmentation using superpixels: A bipartite graph partitioning approach," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 789–796.
- [39] S. Romano, J. Bailey, V. Nguyen, and K. Verspoor, "Standardized mutual information for clustering comparisons: one step further in adjustment for chance," in *International Conference on Machine Learning*. PMLR, 2014, pp. 1143–1151.
- [40] M. Wu and B. Schölkopf, "A local learning approach for clustering," *Advances in Neural Information Processing Systems*, vol. 19, 2006.
- [41] A. Borji, M.-M. Cheng, H. Jiang, and J. Li, "Salient object detection: A benchmark," *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp. 5706–5722, 2015.
- [42] M. R. Yousefi, J. Hua, C. Sima, and E. R. Dougherty, "Reporting bias when using real data sets to analyze classification performance," *Bioinformatics*, vol. 26, no. 1, pp. 68–76, 2010.
- [43] A. Ng, M. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," *Advances in Neural Information Processing Systems*, vol. 14, 2001.
- [44] J. Z. Huang, M. K. Ng, H. Rong, and Z. Li, "Automated variable weighting in k -means type clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 5, pp. 657–668, 2005.

- [45] P. Zhu, X. Hou, K. Tang, Y. Liu, Y.-P. Zhao, and Z. Wang, “Unsupervised feature selection through combining graph learning and $l_{2,0}$ -norm constraint,” *Information Sciences*, vol. 622, pp. 68–82, 2023.
- [46] A. Yuan, J. Huang, C. Wei, W. Zhang, N. Zhang, and M. You, “Unsupervised feature selection via feature-grouping and orthogonal constraint,” in *2022 26th International Conference on Pattern Recognition (ICPR)*. IEEE, 2022, pp. 720–726.
- [47] C. Wang, J. Wang, Z. Gu, J.-M. Wei, and J. Liu, “Unsupervised feature selection by learning exponential weights,” *Pattern Recognition*, vol. 148, p. 110183, 2024.
- [48] R. Couillet, F. Chatelain, and N. Le Bihan, “Two-way kernel matrix puncturing: towards resource-efficient pca and spectral clustering,” in *International Conference on Machine Learning*. PMLR, 2021, pp. 2156–2165.