Enhanced Density Peak Clustering for High-Dimensional Data

Zhongli Wang[1*], Jie Yang[2*], Junyi Guan[1], Chenglong Zhang[1], Xinyan Liang[3], Bingbing Jiang[1], Weiguo Sheng[1†]

[1]Hangzhou Normal University,
[2]University of Technology Sydney,
[3]Shanxi University
wangzhongli1@stu.hznu.edu.cn, jie.yang-1@uts.edu.cn, jonnyguan73@163.com,clzhang123@163.com,
liangxinyan48@163.com, jiangbb@hznu.edu.cn, w.sheng@ieee.org.

Abstract

As a foundational clustering paradigm, Density Peak Clustering (DPC) partitions samples into clusters based on their density peaks, garnering widespread attention. However, traditional DPC methods usually focus on high-density regions, neglecting representative peaks in relatively low-density areas, particularly in datasets with varying densities and multiple peaks. Moreover, existing DPC variants struggle to identify clusters correctly in high-dimensional spaces due to the indistinct distance differences among samples and sparse data distributions. Additionally, existing methods typically adopt a one-step label assignment strategy, making them prone to cascading errors when initial misassignments occur. To address these challenges, we propose an Enhanced Density Peak Clustering (EDPC) method, which creatively incorporates multilayer perceptron (MLP)-based dimensionality reduction and a hierarchical label assignment strategy to significantly improve clustering performance in high-dimensional scenarios. Specifically, we introduce an effective selection condition that combines average densities and density-related distances to generate potential cluster centers, ensuring that peaks across different density regions are considered simultaneously. Furthermore, an MLP, guided by pseudo-labels from sub-clusters, is designed to learn low-dimensional embeddings for high-dimensional data, preserving data locality while enhancing clusterability. Extensive experiments demonstrate the effectiveness and superiority of EDPC against state-of-the-art DPC methods.

Introduction

Clustering, as a fundamental unsupervised learning technique, aims to divide data samples into different clusters based on the intrinsic structures or relationships within data. With the exponential growth of unlabeled samples in practical applications, numerous clustering paradigms have been proposed, such as K-means (Yang et al. 2017), fuzzy clustering, hierarchical clustering, spectral clustering (Hu et al. 2022; Gan et al. 2022), and density-based clustering (Saxena et al. 2017; Cheng et al. 2024). Recently, deep network-based clustering also attracted extensive attention (Xue et al. 2019). Among these clustering techniques, Density Peak Clustering (DPC) (Rodriguez and Laio 2014) focuses on finding appropriate density peaks without prior knowledge, exhibiting better performance than other clustering paradigms in tackling non-spherical cluster problems (Wei et al. 2023). In comparison with other clustering paradigms (e.g., graph-based clustering), DPC can directly achieve final clustering labels without additional post-processing, since it can build a cluster tree for data based on the density-to-distance linkage and then cuts the cluster tree into clusters via a decision graph.

From the perspective of selecting cluster centers, DPC assumes that cluster centers are density peaks with higher densities than their surrounding neighbors and are far from regions of higher density. (Xie et al. 2016) also proposed to identify density peaks as cluster centers, using a density estimation method based on the distance of each sample to its K-nearest neighbors (KNN). (Zhang et al. 2022) applied linear regression and residual analysis to select candidate centers, from which the final cluster centers were determined by using a validity index. Considering that the performance of DPC relies highly on cluster centers, these traditional DPC methods tend to select cluster centers as density peaks from high-density regions, yet overlook representative peaks in relatively low-density regions, leading to suboptimal results. Consequently, it is a critical challenge to guarantee the effectiveness of selected cluster centers especially when there exist varying density and multiple peaks in data distribution.

To address these challenges, various DPC variants have been proposed to improve the selection process of cluster centers. (Liu, Wang, and Yu 2018) redefined densities and density-related distances using shared neighbors to capture data attributes, while (Guan et al. 2023) introduced a graph-based approach to assign non-peak and density peaks labels, accounting for multiple peaks within clusters. However, these improved DPC methods still struggle to accurately identify representative cluster centers in low-density regions. Additionally, most DPC methods assign cluster labels in a single step, making them susceptible to cascading errors if initial label assignments are incorrect. To mitigate this, (Lotfi, Moradi, and Beigy 2020) proposed identifying high-density samples to form cluster backbones, with other samples assigned to their nearest backbones. (Qin

[*]Zhongli Wang and Jie Yang contributed equally to this work.
[†]The corresponding author

et al. 2021) employed the Jaccard coefficient to measure similarities, first assigning high-similarity samples to cluster centers and then allocating remaining samples based on their similarity to already assigned samples. However, high-dimensional data further complicate the cluster center selection, as the distance differences among samples become less distinct, while data distributions tend to be sparse in high-dimensional space (Cao et al. 2023). These factors severely limit the effectiveness of DPC methods and degrade their performance (Yang and Lin 2024). Although (Du, Ding, and Jia 2016) integrated principal component analysis (PCA) into DPC for dimensionality reduction, the extracted features fail to preserve local structures and relationships among original data, misguiding clustering processes (Wu et al. 2022; Zhao et al. 2021; Jiang et al. 2025).

An analysis of DPC methods reveals two factors that limit their performance in high-dimensional tasks. First, these methods primarily focus on high-density regions, often overlooking relatively low-density areas, particularly when data distributions exhibit varying densities and multiple peaks. Second, in high-dimensional spaces, the differences in sample distances become less distinct, resulting in sparse data distributions that obscure the underlying clustering structure, thereby reducing the effectiveness of existing DPC methods. To overcome these inherent limitations, we propose an Enhanced Density Peak Clustering (EDPC) method for high-dimensional data, significantly improving the performance and applicability of DPC in high-dimensional tasks. Specifically, we effectively leverage average densities and density-related distances of samples to select the potential cluster centers that take into account both high and low-density regions simultaneously. Afterward, the pseudo-labels of sub-clusters generated from potential cluster centers can further guide MLP to learn low-dimensional embeddings for high-dimensional data while preserving the local neighbor structures among the original samples. Leveraging the learned low-dimensional embeddings, sub-clusters with well-defined clusterability are identified and hierarchically merged to produce the final clustering labels. The main contributions of this paper are summarized as follows:

- We develop an Enhanced Density Peak Clustering (EDPC) method, which incorporates the hierarchical label assignment process and the MLP-driven dimensionality reduction into a unified framework, seeking the effective partition for high-dimensional data.
- We propose to select potential cluster centers according to the average densities and density-related distances of samples, ensuring that the peaks across different density regions are considered simultaneously.
- A labels-driven MLP dimensionality reduction technique is proposed for high-dimensional data, which effectively leverages pseudo-labels to learn the low-dimensional embedding with the discrimination enhancement, achieving better clustering performance.

Related Works

The DPC Algorithm

Given a dataset $X = \{x_1, x_2, \ldots, x_n \mid x_i \in \mathbb{R}^m\}$, DPC computes the local density ρ_i for each point x_i, along with the density-related distance δ_i to the nearest point with a higher density, as shown in Eq. (1) and (2).

$$\rho_i = \sum_{x_j \in X} \chi(d_{ij} - d_c), \ \chi(z) = \begin{cases} 1 & z < 0 \\ 0 & z \geqslant 0 \end{cases} \tag{1}$$

$$\delta_i = \min_{x_j} (d_{ij}), \text{s.t. } \rho_j > \rho_i \tag{2}$$

Here, d_{ij} represents the Euclidean distance between x_i and x_j, while the "cutoff distance" d_c is typically chosen such that the average number of neighbors corresponds to 1%–2% of the total points in the dataset (Rodriguez and Laio 2014). For the point with the highest density, x_i, the value of δ_i is computed as $\delta_i = \max_{x_j}(d_{ij})$. According to the assumption of DPC regarding cluster centers (i.e., cluster centers are density peaks surrounded by low-density neighbors and located far from regions of higher density), points with the highest γ values ($\gamma = \rho \cdot \delta$) are selected as cluster centers and assigned unique labels. Subsequently, the remaining points inherit the labels of their nearest higher-density neighbors. DPC has been demonstrated to be an effective approach for identifying single-peak clusters and serves as a foundational method for reconstructing complex multi-peak clusters. This is achieved by first identifying single-peak clusters using DPC, followed by merging similar single-peak clusters into multi-peak clusters (Guan et al. 2022).

The SNN-DPC

The SNN-DPC (Liu, Wang, and Yu 2018) replaces the measurement of original density and density-related distance of DPC with the Shared-Nearest-Neighbor-based density and density-related distance, formulated as:

$$\rho_i = \sum_{x_j \in N_k(x_i)} s_{ij} \tag{3}$$

$$\delta_i = \min_{x_j : \rho_j > \rho_i} \left(d_{ij} \left(\sum_{x_z \in N_k(x_i)} d_{iz} + \sum_{x_w \in N_k(x_j)} d_{jw} \right) \right) \tag{4}$$

where $N_k(x_i)$ represents the set of k nearest neighbors of x_i and s_{ij} is the SNN-based similarity between x_i and x_j, defined as:

$$s_{ij} = \begin{cases} \frac{|SN_k(x_i, x_j)|^2}{\sum_{x_z \in SN_k(x_i, x_j)} (d_{iz} + d_{jz})}, & x_i, x_j \in SN_k(x_i, x_j) \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

where $SN_k(x_i, x_j) = N_k(x_i) \cap N_k(x_j)$ is the shared nearest neighbors between x_i and x_j, and $|SN_k|$ indicates the number of shared neighbors in SN_k.

Unlike DPC, SNN-DPC defines density and density-related distance by incorporating local data structure, enabling more effective cluster reconstruction. Based on this way, we can identify numerous potential cluster centers, thereby generating reliable single-peak sub-clusters for subsequent hierarchical cluster merging.

Linkage-based Clustering

Linkage-based Clustering is a classic hierarchical clustering technique, that iteratively merges similar clusters until a single cluster containing all samples is formed, resulting in a cluster tree (i.e., dendrogram) for the dataset. The final clustering is achieved by cutting the dendrogram at the desired number of clusters. In linkage-based clustering, the linkage metric, which estimates the dissimilarity between clusters, is crucial. The classic single-linkage method (Gower and Ross 1969) measures dissimilarity using the shortest distance between clusters, effectively identifying both spherical and complex shapes but is sensitive to noise. In contrast, the average-linkage method (Dasgupta and Long 2005) calculates dissimilarity as the average of all pairwise distances, handling noise and outliers better but tends to favor spherical clusters. To leverage the strengths of both methods, (Zelig, Kariti, and Kaplan 2023) proposed the KMD-linkage to calculate the average of several smallest distances between pairwise clusters to determine their dissimilarity. This approach achieves a balance between robustness to noise and sensitivity to diverse cluster shapes. In this paper, KMD-linkage is employed to accurately quantify the dissimilarity between clusters.

Methodology

In this section, we present our proposed clustering algorithm, which integrates dimensionality reduction and advanced clustering techniques through the following steps. First, pseudo-labels of sub-clusters are leveraged to train a multilayer perceptron (MLP) for dimensionality reduction. Afterward, a selection criterion identifies n_s potential cluster centers based on their SNN-based density and density-related distance. The DPC is then applied to form n_s sub-clusters, each led by one of these potential centers. Finally, these sub-clusters are hierarchically merged into n_c clusters, representing the true number of clusters, based on their dissimilarity. By effectively combining these steps, the proposed algorithm ensures accurate cluster identification and refinement, even in complex data distributions.

Potential Cluster Centers

DPC may face challenges when handling datasets with varying densities and multiple peaks, as such scenarios can lead to incorrect identification of cluster centers (Wang et al. 2023). Intuitively, density peaks with higher densities and greater density-related distances are more likely to be selected as potential cluster centers, whereas points with lower densities are typically identified as boundary points or noises. However, in datasets with varying densities, the low-density points still hold representational significance.

To ensure that all representative density peaks are detected, including those in low-density clusters, the average density $\bar{\rho} = \frac{1}{n}\sum_{i=1}^{n}\rho_i$. The density-related distance $\bar{\delta} = \frac{1}{n}\sum_{i=1}^{n}\delta_i$ is designed to identify potential density peaks that can serve as clusters, where ρ_i and δ_i are SNN-based values computed by Eq.(3) and Eq.(4). Thus, we propose a potential center detection criterion, formulated as:

$$Centers(\lambda, \beta) = \{x_i \in X \mid \rho_i \geqslant \lambda\bar{\rho},\ \delta_i \geqslant \beta\bar{\delta}\} \quad (6)$$

where λ and β are hyperparameters that control the thresholds for selecting potential cluster centers based on the density and density-related distance, respectively. If λ is set as a larger value, the potential cluster centers in low-density regions will be excluded, thereby misguiding subsequent hierarchical clustering. Conversely, a smaller λ could result in an excessive number of samples being identified as potential cluster centers, which can similarly disrupt the clustering process. Similarly, a larger β may prevent certain clusters' density peaks from being selected as potential cluster centers, reducing their representativeness, while a smaller β may lead to more points within high-density clusters being chosen as potential centers, compromising the representativeness of the cluster centers. To balance these factors, we set $Centers(0.5, 1)$ as the thresholds for identifying potential cluster centers in the experiments, such that centers in low-density regions are taken into account, so as to make the overall set of centers more representative. Once these centers are identified, DPC is applied to generate sub-clusters, followed by a hierarchical clustering approach that iteratively merges similar sub-clusters to form final clusters.

Merging of Sub-Clusters

To achieve a balance between robustness against noises and sensitivity to various cluster shapes, the KMD-linkage (Zelig, Kariti, and Kaplan 2023), denoted as D_{KMD}, is utilized to estimate the dissimilarity between cluster C_A and cluster C_B, defined as follows:

$$D_{KMD}(C_A, C_B) = \frac{1}{q}\sum \min_{q}\{d_{ij} : x_i \in C_A, x_j \in C_B\} \quad (7)$$

which means the average value of the top q smallest distances between cluster C_A and C_B, and q is defined as:

$$q = \max\{\lfloor \max(|C_A|, |C_B|)/\phi \rfloor, 1\} \quad (8)$$

where $|C_A|$ represents the number of points within cluster C_A, and $\lfloor \cdot \rfloor$ denotes the floor function. Notably, the value of q is determined by the parameter ϕ. When ϕ becomes large, q decreases and tends toward 1, making D_{KMD} degenerate to single-linkage (i.e., measuring the distance between clusters using the minimum pairwise distance), which is very sensitive to noises and outliers (Jarman 2020). Conversely, when setting ϕ to a smaller value, D_{KMD} approximates average-linkage (i.e., calculating the average of all pairwise distances between clusters), which is unable to handle complex cluster shapes effectively (Charikar, Chatziafratis, and Niazadeh 2019). To this end, we propose adjusting ϕ to determine the appropriate number of pairwise distances for datasets with diverse categories (further analysis is provided in the Experiments). Subsequently, we apply linkage clustering to merge n_s sub-clusters into n_c clusters based on their dissimilarity about the linkage-based metric D_{KMD}.

Dimension Reduction using Sub-Clusters

Existing DPC-related methods face significant challenges in high-dimensional spaces, where the distances between samples tend to become similar, and the distributions are often sparse. This phenomenon hinders the effective identification of representative density peaks, thereby reducing

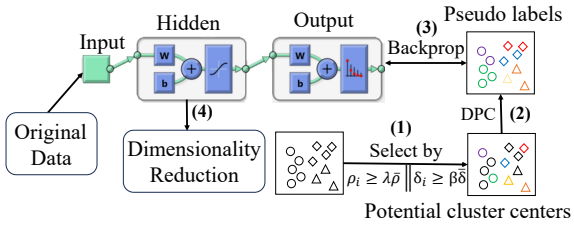

Figure 1: Dimension reduction derived by sub-clusters.

clustering performance. Therefore, we creatively propose to use the pseudo-labels generated by DPC sub-clusters as a guide for dimensionality reduction. Unlike other dimensionality reduction methods, such as PCA, which reduces dimensions based solely on overall variance without any guiding information, leveraging pseudo-labels from sub-clusters for dimensionality reduction effectively utilizes local structural information among data points. This approach extracts low-dimensional representations with enhanced clustering discriminability (Jiang et al. 2024).

Multilayer Perceptron (MLP) The Multilayer Perceptron (MLP) is a fundamental artificial neural network model, which is extensively utilized in machine learning and deep learning. In this study, we employ an MLP with only one hidden layer for dimensionality reduction, offering higher interpretability and lower computational costs than other complex networks. The input layer receives features from data, while the hidden layer, situated between the input and output layers, contains neurons connected by weighted sums and activation functions. The output layer generates the model's final output, typically a classification label.

DPC-derived sub-clusters In EDPC, the $Centers(\lambda, \beta)$ is first used to identify reliable sub-cluster centers by adjusting the hyperparameters $\{\lambda, \beta\}$. We then apply density peak clustering to obtain sub-clusters with these centers. The pseudo-labels of these sub-clusters are treated as ground-truth labels and exploited to train the MLP.

Proposed Framework Figure 1 shows how to utilize the pseudo-labels of sub-clusters to guide MLP for dimensionality reduction. Concretely, in Step (1), the original data is analyzed to identify candidate cluster centers based on the Centers$\{\lambda, \beta\}$ condition, which considers both average density and density-related distance. In Step (2), these candidate centers are organized into sub-clusters using the label assignment strategy of DPC. In Step (3), an MLP is designed with a single hidden layer containing fewer neurons than the original data dimensions. The hidden layer employs ReLU as its activation function, while the output layer utilizes Softmax, with the cross-entropy loss function guiding the training process. The original data is fed into an MLP, where sub-cluster pseudo-labels act as ground-truth labels to supervise the training process, enabling the learning of more discriminative reduced-dimensional representations for the original high-dimensional data. In Step (4), the number of neurons in the hidden layer is set smaller than the original feature dimensions, and the parameters of this layer are utilized

Algorithm 1: EDPC without MLP

Input: $X = \{x_1, x_2, x_3, ..., x_n\}$, the number of clusters n_c;

1: Calculate the point density $\{\rho_1, \rho_2, \rho_3, ..., \rho_n\}$ by Eq.(3) and density-related distance $\{\delta_1, \delta_2, \delta_3, ..., \delta_n\}$ by Eq.(4) for each sample;
2: n_s potential cluster centers $\leftarrow Centers(0.5, 1)$;
3: Obtain n_s sub-clusters with corresponding potential cluster centers via density peak clustering $(n_s \gg n_c)$;
4: **while** $n_s > n_c$ **do**
5: Calculate the distances between sub-clusters by Eq. (7);
6: Merge two clusters with the smallest distance;
7: $n_s \leftarrow n_s - 1$
8: **end while**

Output: Clustering labels.

Algorithm 2: EDPC for High-dimensional Data

Input: $X = \{x_1, x_2, x_3, ..., x_n\}$, the number of clusters n_c;

1: Calculate the point density $\{\rho_1, \rho_2, \rho_3, ..., \rho_n\}$ by Eq.(1) and density-related distance $\{\delta_1, \delta_2, \delta_3, ..., \delta_n\}$ by Eq.(2) for each sample;
2: n_s potential cluster centers $\leftarrow Centers(\lambda, \beta)$;
3: Obtain n_s sub-clusters with corresponding sub-cluster centers via density peak clustering;
4: Train the MLP with the pseudo-labels of sub-clusters;
5: After training, the parameters of the hidden layer are extracted to perform dimensionality reduction;
6: Run Algorithm 1 on the reduced-dimensional data;

Output: Clustering labels.

to transform original data into reduced-dimensional representations. This allows the MLP to leverage fine-grained pseudo-labels, initially generated by EDPC, to guide dimensionality reduction. Unlike traditional approaches that separate dimensionality reduction and clustering (e.g., first applying t-SNE (Van der Maaten and Hinton 2008) or UMAP (McInnes, Healy, and Melville 2018) for dimensionality reduction and then performing DPC), EDPC integrates MLP-based dimensionality reduction and clustering into a unified framework, enabling the two processes to interact and mutually enhance each other in a self-reinforcing manner.

Time Complexity Analysis

Algorithm 1 and Algorithm 2 provide the pseudocode for the EPDC without MLP and the EDPC for high-dimensional data, respectively. Below is the time complexity analysis for these algorithms.

EPDC without MLP: The time complexity of computing the distance matrix is $O(n^2)$, and the complexity for calculating ρ and δ totals $O(n)$. Selecting n_s potential cluster centers requires $O(n)$. Once the potential centers are determined, assigning labels to the remaining points also costs $O(n)$. After obtaining n_s sub-clusters based on potential centers, we need to calculate the distance between each sub-cluster through Eq.(7). We consider the average case, assuming that each sub-cluster is the same size of

Datasets	Instances	Clusters	Features
Dart1	1000	4	2
Donut2	1000	2	2
Cno3	715	3	2
Cno4	863	4	2
Curet1	2000	6	2
Cuboids	1002	4	3
D31	3100	3	2
Jain	373	2	2
Alizadeh	62	3	2093
Armstrong	72	2	1081
Bhatt	203	5	1543
Garber	66	4	4553
Gordon	181	2	1626
Golub1	72	2	1868
Golub2	72	3	1868
Nutt	22	2	1152
Liang	37	3	1411
Laiho	37	2	2202
Pomeroy1	34	2	857
Pomeroy2	42	5	1379
Shipp	77	2	798
West	49	2	1198
UMIST	575	20	10304
MSRC	210	7	512
MSRA	1799	12	256
Ecoil	336	8	343

Table 1: The detailed information of experimental datasets.

$\frac{n}{n_s}$. There are a total of $\frac{n_s(n_s-1)}{2}$ pairs of cluster distances that need to be considered, so the time complexity is $O(\frac{n_s(n_s-1)}{2}\frac{n_s^2}{n_s^2}\log\frac{n_s^2}{n_s^2})$, which is approximately equal to $O(n^2\log n)$. Combining all the above steps, the overall time complexity approximates to $O(n^2\log n)$.

EDPC for High-dimensional Data: The time complexity for generating sub-clusters is $O(n)$, and training the shallow MLP requires $O(nm)$ (where m is the number of dimensions of the data X). The time complexity for applying EDPC without MLP on the reduced-dimensional data is $O(n^2\log n)$. Therefore, the total time complexity of EDPC approximates to $O(n(m+n\log n))$.

Experiments

In this section, extensive experiments on synthetic datasets and real-world high-dimensional datasets are conducted to evaluate the clustering performance of the proposed EDPC.

Experimental Settings

To comprehensively verify the effectiveness and superiority of EDPC, various datasets are employed to assess the clustering performance, including eight synthetic datasets and eighteen real-world datasets with high dimensions whose details are summarized in Table 1. Specifically, the state-of-the-art clustering algorithms are compared, including: (1) Classical density peak clustering **(DPC)** (Rodriguez and Laio 2014); (2) Shared-nearest-neighbor-based density peak clustering **(SNN-DPC)** (Liu, Wang, and Yu 2018); (3) Stable-membership-based multi-peak clustering **(SMMP)** (Guan et al. 2022); (5) Flexible density peak clustering **(DB-DPC)** (Hou et al. 2024); (6) Fast hierarchical clustering of local density peak **(FHC-LDP)** (Guan et al. 2021); (7) Consistent and divergent graph clustering **(CDGC)** (Huang et al. 2022). To ensure fairness, the experiments are implemented based on their default settings, and the number of neighbors (i.e., k) is set to 15 when performing K-nearest neighbors (KNN). For EDPC, the parameter ϕ is fixed at 10. Determining the optimal reduced dimensionality is an NP-hard problem in many fields. To ensure consistency when evaluating dimensionality reduction methods, existing studies often predefine the number of reduced dimensionality, avoiding discrepancies caused by varying dimensions across methods. Following this principle, we fix the reduced dimensionality at 50 for EDPC and other methods in our experiments to ensure fair and consistent comparisons. Two external evaluation metrics are utilized: accuracy (ACC) and adjusted mutual information (AMI). ACC measures the percentage of correctly assigned data points, and AMI quantifies the agreement between the clustering labels and the true labels.

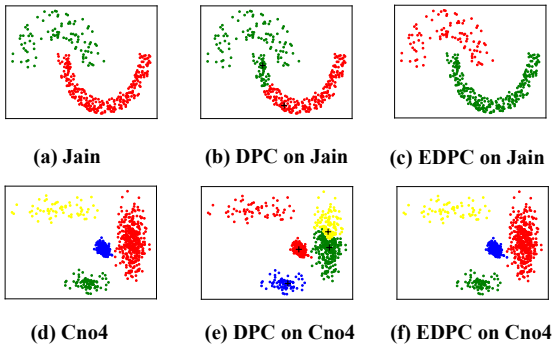

Figure 2: The results on varying density and multiple peaks datasets, in which (a) and (d) depict the original data distributions, (b) and (e) show the results obtained by DPC, (c) and (f) illustrate the results of EDPC.

Experiments on Synthetic Datasets

To validate the clustering effectiveness of EDPC, Figure 2 presents visualization results on datasets with varying densities and multiple peaks (i.e., Jain and Cno4 datasets). We find that DPC struggles to correctly identify cluster centers on these datasets, and it instinctively assigns density peaks to higher-density regions as shown in Figures 2(b) and (e), while neglecting representative lower-density regions, resulting in incorrect center identifications. In contrast, EDPC incorporates a selection condition for generating potential cluster centers, enabling it to consider peaks across different density regions simultaneously and effectively address the challenges posed by varying densities and multiple peaks. To further evaluate performance, we compared EDPC against the DPC and four improved DPC variants (i.e., SNN-DPC, SMMP, KNN-DPC, and DB-DPC) on synthetic datasets. Table 2 reports the results on eight syn-

	ACC scores on synthetic datasets						AMI scores on synthetic datasets					
Datasets\Methods	DPC	SNN-DPC	SMMP	KNN-DPC	DB-DPC	EDPC	DPC	SNN-DPC	SMMP	KNN-DPC	DB-DBC	EDPC
Dart1	0.2500	0.4510	0.6300	0.3210	0.2500	**1.0000**	-0.0014	0.3637	0.7492	0.0611	0.0000	**1.0000**
Donut2	0.6600	0.9950	0.9960	0.6220	0.9410	**0.9970**	0.2210	0.9595	0.9663	0.1536	0.7733	**0.9735**
Cno3	0.8993	0.7804	0.9930	0.8042	0.5692	**0.9944**	0.7925	0.7355	0.9573	0.7168	0.5144	**0.9677**
Cno4	0.7578	0.8737	**0.9988**	**0.9988**	0.7984	**0.9988**	0.7815	0.8384	0.9928	**0.9930**	0.6580	**0.9930**
Curet1	0.8640	0.5960	0.9240	0.7890	**0.9865**	0.9745	0.8951	0.7333	0.9499	0.8800	**0.9787**	0.9599
Cuboids	0.7625	0.7415	**1.0000**	0.7555	0.6517	**1.0000**	0.8254	0.8218	**1.0000**	0.8240	0.7000	**1.0000**
D31	0.9674	0.9690	0.9684	0.9668	0.8761	**0.9694**	0.9548	0.9565	0.9556	0.9539	0.9121	**0.9567**
Jain	0.8606	0.8338	**1.0000**	0.6237	0.9088	**1.0000**	0.5042	0.4578	**1.0000**	0.4578	0.7228	**1.0000**
Average	0.7527	0.7801	0.9388	0.7351	0.7477	**0.9916**	0.6214	0.7333	0.9464	0.6303	0.6574	**0.9814**

Table 2: The ACC and AMI scores on synthetic datasets. The best and second results are in bold and underlined, respectively.

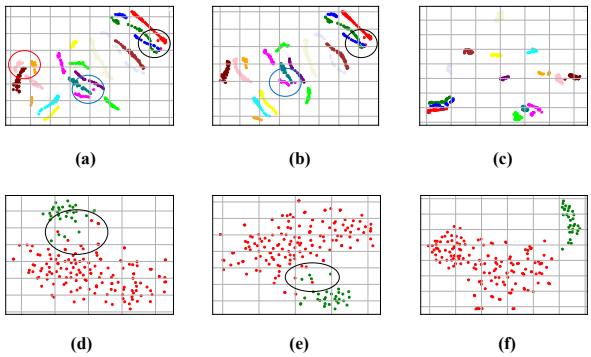

Figure 3: The visualization on the UMIST and Gordon datasets. (a) and (d) show the results of the original space, which are directly utilized by DPC, SNN-DPC, SMMP, FHC-LDP and CDGC. (b) and (e) present the results after dimensionality reduction using PCA, as applied by KNN-DPC. (c) and (f) display the results obtained by leveraging the pseudo-labels of sub-clusters for dimensionality reduction, as implemented in EDPC.

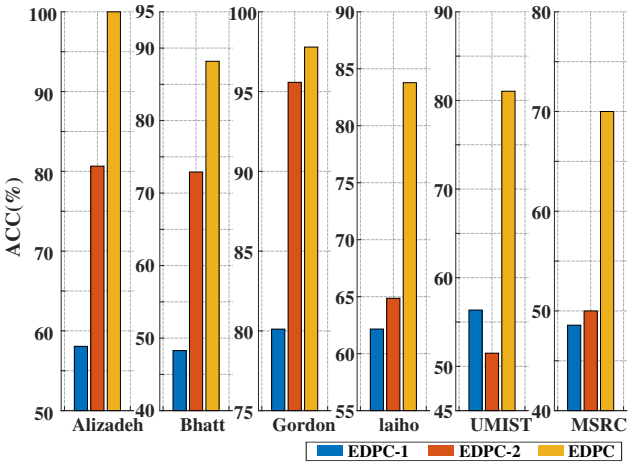

Figure 4: The ACC of EDPC and its variants.

thetic datasets[1], demonstrating that EDPC achieves better or competitive performance over its counterparts on ACC and AMI. In terms of average ACC, EDPC achieves improvements of 23.9%, 21.2%, 5.3%, 25.7%, and 24.4% over DPC, SNN-DPC, SMMP, KNN-DPC, and DB-DPC, respectively. This indicates that the weighted mechanism, which leverages average densities and density-related distances with parameters set as Centers(0.5, 1), enables EDPC to effectively identify representative peaks across regions of varying density, resulting in superior performance.

Experiments on High-dimensional Datasets

Experimental Results To further show the superiority of EDPC in high-dimensional datasets, fourteen gene expression datasets (Shah and Koltun 2017) and four object recognition datasets are utilized, including UMIST (Guan et al. 2023), MSRC (Zhang et al. 2024), MSRA (Wang et al. 2020), Ecoil (Guan et al. 2022). To illustrate how EDPC leverages fine-grained clustering labels to drive neural networks to learn more discriminative representations, we compare EDPC with two additional dimensionality reduction-based DPC variants i.e.,DPC-tSNE (applying DPC after t-SNE dimensionality reduction) and SMMP-tSNE (applying SMMP after t-SNE dimensionality reduction) and 7 state-of-the-art competitors. Table 3 presents the comparison between EDPC and the compared methods, in which EDPC achieves satisfactory performance than existing clustering methods across different high-dimensional datasets. Notably, EDPC not only surpasses multiple improved DPC variants but also outperforms dimensionality reduction-based DPC variants. Specifically, on 18 high-dimensional datasets, EDPC achieves improvements in average ACC of 23.0%, 16.9%, 23.2%, 12.7%, 19.3%, and 19.5% over DPC, SNN-DPC, SMMP, KNN-DPC, DB-DPC, and FHC-LDP, respectively, demonstrating its ability to accurately identify cluster centers from regions with varying densities or multiple peaks. Furthermore, EDPC outperforms DPC variants that combine t-SNE with clustering methods, and it achieves 15.2%, and 14.9% average ACC improvement on 18 high-dimensional datasets over DPC-tSNE and SMMP-tSNE, respectively. Although combining t-SNE with density-based clustering offers a potential alternative, it fails to establish meaningful interactions between clustering labels and dimensionality reduction processes, resulting in information loss and suboptimal clustering performance. By integrating dimensionality reduction and clustering into a uni-

[1]https://github.com/deric/clustering-benchmark

Methods \Datasets	Alizadeh		Bhatt		Garber		Gordon		Golub1		Liang	
	ACC	AMI	ACC	AMI	ACC	AMI	ACC	AMI	ACC	AMI	ACC	AMI
DPC	0.5806	0.5675	0.4828	0.0479	0.5758	0.0116	0.8011	0.2659	0.9028	0.6145	0.5675	0.2940
SNN-DPC	0.8710	0.7278	0.7438	0.3482	0.6518	0.1935	**0.9779**	**0.8072**	0.8333	0.4110	0.5405	0.1445
SMMP	0.5238	-0.0162	0.7537	0.2130	0.6515	0.1248	0.8287	0.0000	0.6528	0.0000	0.5676	0.2416
KNN-DPC	0.7581	0.6341	0.7241	0.3710	0.6515	0.2134	0.9669	0.7101	0.9583	0.7340	0.4865	0.1247
DB-DPC	0.8387	0.6558	0.7734	0.2505	0.6667	0.1603	**0.9779**	0.7748	0.6528	0.000	0.7568	0.0000
FHC-LDP	0.8387	0.6558	0.8128	0.3540	0.6364	0.0695	0.8011	0.0044	0.6806	0.0418	0.5676	0.2416
DPC-tSNE	0.5484	0.5189	0.7192	0.5045	0.6970	**0.3563**	0.9613	0.6637	0.5000	0.1226	0.6486	0.2457
SMMP-tSNE	0.5645	0.4624	0.7291	0.4919	0.6818	0.1746	0.9613	0.6573	0.8611	0.4166	0.7027	0.3020
CDGC	0.9839	0.9089	0.8079	0.5292	0.6818	0.2428	**0.9779**	0.7748	0.9167	0.6314	0.5676	0.2416
EDPC	**1.0000**	**1.0000**	**0.8818**	**0.6619**	**0.7121**	0.3467	**0.9779**	**0.8072**	**0.9722**	**0.8308**	**0.8649**	**0.6131**
Methods \Datasets	Laiho		Pomeroy1		Shipp		UMIST		MSRC		MSRA	
	ACC	AMI	ACC	AMI	ACC	AMI	ACC	AMI	ACC	AMI	ACC	AMI
DPC	0.6216	0.0422	0.5588	0.0858	0.7792	0.2273	0.5635	0.7729	0.4857	0.3822	0.1178	0.0015
SNN-DPC	0.6486	0.0433	0.5294	0.1442	0.7143	0.2300	0.4800	0.6857	0.5143	0.5092	0.4408	0.6045
SMMP	0.7838	0.0000	0.7353	0.0000	0.7532	0.0000	0.3826	0.5412	0.5095	0.4940	0.5442	0.6037
KNN-DPC	0.7297	-0.0067	0.7647	0.1003	0.7403	0.1839	0.5235	0.7129	0.5333	0.5382	0.5386	0.6988
DB-DPC	0.7838	0.000	0.7353	0.0000	0.7352	0.0000	0.3843	0.4748	0.3714	0.2828	0.5643	0.7300
FHC-LDP	0.7297	0.1530	0.5588	-0.0197	0.5065	0.1427	0.6522	0.7945	0.5810	0.5208	0.4691	0.6598
DPC-tSNE	0.5676	-0.0066	0.5588	-0.0202	0.6623	0.1848	0.5583	0.7253	**0.7667**	**0.6922**	**0.6054**	0.7431
SMMP-tSNE	0.5135	0.0218	0.7353	0.0000	0.7143	0.1223	0.5009	0.7020	0.6143	0.6166	0.4953	0.5595
CDGC	0.5135	-0.0157	0.5588	-0.0197	0.7922	0.2518	0.7130	0.8352	0.5857	0.6544	0.5381	0.6876
EDPC	**0.8378**	**0.2077**	**0.8529**	**0.3588**	**0.8442**	**0.3175**	**0.8104**	**0.8972**	0.7000	0.5520	0.5798	**0.8640**
Methods \Datasets	Armstrong		Nutt		Golub2		West		Pomeroy2		Ecoil	
	ACC	AMI	ACC	AMI	ACC	AMI	ACC	AMI	ACC	AMI	ACC	AMI
DPC	0.9028	0.7130	0.5909	0.0040	0.6944	0.4889	0.5510	0.0172	0.7619	**0.6338**	0.4554	0.3009
SNN-DPC	0.8333	0.4110	0.5909	-0.0495	0.7361	0.4988	**0.8776**	0.4886	0.4762	0.2850	0.6310	0.4951
SMMP	0.7222	0.5625	0.6818	0.0000	0.5278	0.0000	0.5102	0.0000	0.2381	0.0000	0.5833	0.5272
KNN-DPC	0.6806	0.5451	**0.7273**	0.0663	**0.9444**	0.7679	**0.8776**	**0.5590**	0.5714	0.4260	0.6696	0.4885
DB-DPC	0.6944	0.4577	0.6818	0.0000	0.5278	0.0000	0.5102	0.0000	0.2381	0.0000	0.7589	0.5478
FHC-LDP	0.7222	0.5625	0.5909	**0.1679**	0.5556	0.2393	0.6939	0.1516	0.6190	0.4597	0.5923	0.5461
DPC-tSNE	0.9444	0.8027	0.5909	-0.0454	0.9167	0.7258	**0.8776**	**0.5590**	0.7619	0.6019	0.5119	0.5591
SMMP-tSNE	0.9444	0.8018	0.6818	0.0000	0.7222	0.2242	0.8367	0.4533	0.6667	0.4280	0.5208	0.5005
CDGC	**0.9722**	**0.8915**	0.5909	0.0452	0.8750	0.6388	**0.8776**	0.5445	0.7381	0.5217	0.6310	0.4951
EDPC	0.9444	0.8413	**0.7273**	0.0663	**0.9444**	**0.7955**	**0.8776**	**0.5590**	**0.7857**	0.6179	**0.8125**	**0.6958**

Table 3: The ACC and AMI on high-dimensional datasets. The best and second results are in bold and underlined, respectively.

fied framework, EDPC effectively strengthens the interaction between these processes, producing more discriminative representations and superior clustering accuracy. Additionally, although spectral clustering has long been regarded as superior to density-based clustering methods for handling high-dimensional data, EDPC achieves 10.0% average ACC improvement over the advanced spectral clustering method (i.e., CDGC) on the 18 high-dimensional datasets, highlighting the effectiveness of leveraging sub-cluster pseudo-labels with MLP for dimensionality reduction.

Visualization To demonstrate the effectiveness of the MLP dimensionality reduction guided by pseudo-labels of sub-clusters, we adopt t-SNE to project the UMIST and Gordon datasets into a two-dimensional space (Van der Maaten and Hinton, 2008). Figure 3 presents the visualization results, where Figures 3(a)-(c) show the results for UMIST (15 clusters selected), and Figures 3(d)-(f) correspond to Gordon. In the original feature space (as shown in Figures 3(a) and (d)), the visualization exhibits significant overlaps between different clusters, failing to effectively separate them. Although PCA-generated reduced features provide better-separated cluster structures (as shown in Figures 3(b) and (e)), they lack compact intra-cluster distributions, as PCA ignores the local structures and relationships among samples. In contrast, our sub-cluster-based dimensionality reduction method (as shown in Figures 3(c) and (f)) effectively separates samples into distinct clusters with more compact intra-cluster structures, thereby facilitating clustering.

Ablation Study

To further evaluate the significance of the components within EDPC, we conducted an ablation study with two EDPC variants: EDPC-1, which assigns cluster labels directly through cluster centers without the sub-cluster merging process, and EDPC-2, which skips the MLP-based dimensionality reduction and directly identifies potential cluster centers and merges sub-clusters in the original feature space. Figure 4 illustrates the ACC performance of EDPC and its variants across different datasets. The following conclusions can be drawn: (1) The performance of EDPC-2 surpasses that of EDPC-1, underscoring the significance of the hierarchical label assignment strategy in EDPC for address-

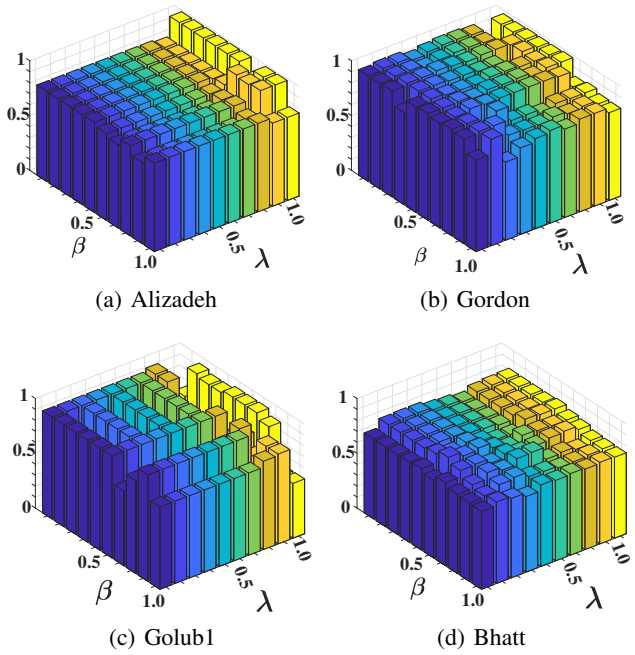

Figure 5: ACC with different λ and β on different datasets.

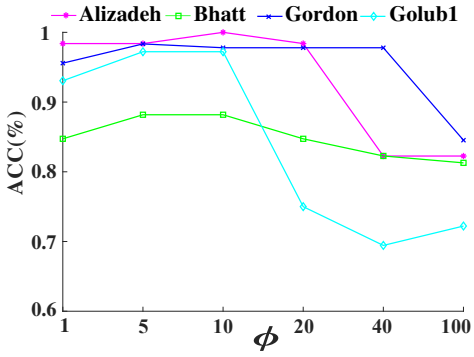

Figure 6: The ACC of EDPC with different values of ϕ.

ing varying density and multi-peak issues. (2) EDPC significantly outperforms EDPC-2, demonstrating that leveraging pseudo-labels from sub-clusters for dimensionality reduction effectively preserves local structural information among data points, thereby facilitating clusterability.

Parameter Sensitivity

The EDPC incorporates two crucial intrinsic parameters, i.e., λ and β, which are pivotal to the selection of potential cluster centers during the MLP dimensionality reduction process. To assess the impact of these parameters on clustering effectiveness, we show clustering accuracy under different λ and β settings in Figure. 5. Specifically, λ and β are tuned from 0.1 to 1.0 in increments of 0.1 to generate diverse sub-clusters. We find that EDPC performs better when both λ and β are greater than 0.5, demonstrating that properly tuning the thresholds based on local density and density-related distance can improve the selection of potential cluster centers and further enhance the performance.

Meanwhile, the value of ϕ influences D_{KMD}, which affects the distance between clusters, ultimately impacting the clustering results. Here, we vary the value of ϕ in the range of $\{1, 5, 10, 20, 40, 100\}$ to analyze the impacts on the performance. Figure. 6 displays the ACC of EDPC with varied ϕ on four datasets, from which we can find EDPC achieves better performance when ϕ is smaller than 20. This indicates that better performance can be obtained when $\phi \in [5, 20]$. For simplicity, we suggest setting $\phi = 10$ for promising results in real applications.

Conclusion

In this paper, an Enhanced Density Peak Clustering (EDPC) is presented to break the dilemma of existing DPC methods in high-dimensional scenarios, which creatively uses the clustering pseudo-labels to guide MLP-based dimensionality reduction, thereby learning low-dimensional representation from high-dimensional data. In this way, EDPC can learn the low-dimensional embedding with the locality preservation and the discrimination enhancement, significantly improving the performance in high-dimensional scenarios. Meanwhile, EDPC can fully leverage the combination of average densities and density-related distances to select the potential cluster centers that properly balance the representative peaks of different density regions. Additionally, EDPC adopts a hierarchical technique to iteratively assign labels to samples, effectively mitigating the possible cascading errors from initial misassignments. Extensive experiments fully validate the effectiveness and superiority of EDPC against existing DPC methods.

Acknowledgments

This work was supported in part by the "Pioneer" and "Leading Goose" R&D Program of Zhejiang Province under Grant 2023C01022, in part by the National Natural Science Foundation of China under Grants 62306171 and 62306282, in part by the Science and Technology Major Project of Shanxi Province under Grant 202201020101006.

References

Cao, Z.; Xie, X.; Sun, F.; and Qian, J. 2023. Consensus cluster structure guided multi-view unsupervised feature selection. *Knowledge-Based Systems*, 271: 110578.

Charikar, M.; Chatziafratis, V.; and Niazadeh, R. 2019. Hierarchical clustering better than average-linkage. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, 2291–2304. SIAM.

Cheng, D.; Huang, J.; Zhang, S.; Xia, S.; Wang, G.; and Xie, J. 2024. K-means clustering with natural density peaks for discovering arbitrary-shaped clusters. *IEEE Transactions on Neural Networks and Learning Systems*, 35(8): 11077 – 11090.

Dasgupta, S.; and Long, P. M. 2005. Performance guarantees for hierarchical clustering. *Journal of Computer and System Sciences*, 70(4): 555–569.

Du, M.; Ding, S.; and Jia, H. 2016. Study on density peaks clustering based on k-nearest neighbors and principal component analysis. *Knowledge-Based Systems*, 99: 135–145.

Gan, J.; Hu, R.; Zhan, M.; Mo, Y.; Wan, Y.; and Zhu, X. 2022. Multi-view Unsupervised Graph Representation Learning. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 2987–2993.

Gower, J. C.; and Ross, G. J. 1969. Minimum spanning trees and single linkage cluster analysis. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 18(1): 54–64.

Guan, J.; Li, S.; He, X.; and Chen, J. 2023. Clustering by fast detection of main density peaks within a peak digraph. *Information Sciences*, 628: 504–521.

Guan, J.; Li, S.; He, X.; Zhu, J.; and Chen, J. 2021. Fast hierarchical clustering of local density peaks via an association degree transfer method. *Neurocomputing*, 455: 401–418.

Guan, J.; Li, S.; He, X.; Zhu, J.; Chen, J.; and Si, P. 2022. SMMP: a stable-membership-based auto-tuning multi-peak clustering algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(5): 6307–6319.

Hou, J.; Lin, H.; Yuan, H.; and Pelillo, M. 2024. Flexible density peak clustering for real-world data. *Pattern Recognition*, 156: 110772.

Hu, R.; Peng, L.; Gan, J.; Shi, X.; and Zhu, X. 2022. Complementary graph representation learning for functional neuroimaging identification. In *Proceedings of the ACM International Conference on Multimedia*, 3385–3393.

Huang, S.; Tsang, I. W.; Xu, Z.; and Lv, J. 2022. Measuring diversity in graph learning: A unified framework for structured multi-view clustering. *IEEE Transactions on Knowledge and Data Engineering*, 34(12): 5869–5883.

Jarman, A. M. 2020. Hierarchical cluster analysis: Comparison of single linkage, complete linkage, average linkage and centroid linkage method. *Georgia Southern University*, 29.

Jiang, B.; Liu, J.; Wang, Z.; Zhang, C.; Yang, J.; Wang, Y.; Sheng, W.; and Ding, W. 2025. Semi-supervised multi-view feature selection with adaptive similarity fusion and learning. *Pattern Recognition*, 159: 111159.

Jiang, B.; Wu, X.; Zhou, X.; Cohn, A. G.; Liu, Y.; Sheng, W.; and Chen, H. 2024. Semi-Supervised Multi-View Feature Selection with Adaptive Graph Learning. *IEEE Transactions on Neural Networks and Learning Systems*, 35(3): 3615–3629.

Liu, R.; Wang, H.; and Yu, X. 2018. Shared-nearest-neighbor-based clustering by fast search and find of density peaks. *Information Sciences*, 450: 200–226.

Lotfi, A.; Moradi, P.; and Beigy, H. 2020. Density peaks clustering based on density backbone and fuzzy neighborhood. *Pattern Recognition*, 107: 107449.

McInnes, L.; Healy, J.; and Melville, J. 2018. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*.

Qin, X.; Han, X.; Chu, J.; Zhang, Y.; Xu, X.; Xie, J.; and Xie, G. 2021. Density peaks clustering based on Jaccard similarity and label propagation. *Cognitive Computation*, 13: 1609–1626.

Rodriguez, A.; and Laio, A. 2014. Clustering by fast search and find of density peaks. *science*, 344(6191): 1492–1496.

Saxena, A.; Prasad, M.; Gupta, A.; Bharill, N.; Patel, O. P.; Tiwari, A.; Er, M. J.; Ding, W.; and Lin, C.-T. 2017. A review of clustering techniques and developments. *Neurocomputing*, 267: 664–681.

Shah, S. A.; and Koltun, V. 2017. Robust continuous clustering. *Proceedings of the National Academy of Sciences*, 114(37): 9814–9819.

Van der Maaten, L.; and Hinton, G. 2008. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(11).

Wang, F.; Zhu, L.; Liang, C.; Li, J.; Chang, X.; and Lu, K. 2020. Robust optimal graph clustering. *Neurocomputing*, 378: 153–165.

Wang, Y.; Qian, J.; Hassan, M.; Zhang, X.; Zhang, T.; Yang, C.; Zhou, X.; and Jia, F. 2023. Density peak clustering algorithms: A review on the decade 2014–2023. *Expert Systems with Applications*, 121860.

Wei, X.; Peng, M.; Huang, H.; and Zhou, Y. 2023. An overview on density peaks clustering. *Neurocomputing*, 126633.

Wu, D.; Dong, X.; Nie, F.; Wang, R.; and Li, X. 2022. An attention-based framework for multi-view clustering on Grassmann manifold. *Pattern Recognition*, 128: 108610.

Xie, J.; Gao, H.; Xie, W.; Liu, X.; and Grant, P. W. 2016. Robust clustering by detecting density peaks and assigning points based on fuzzy weighted K-nearest neighbors. *Information Sciences*, 354: 19–40.

Xue, Z.; Du, J.; Du, D.; and Lyu, S. 2019. Deep low-rank subspace ensemble for multi-view clustering. *Information Sciences*, 482: 210–227.

Yang, J.; and Lin, C.-T. 2024. Enhanced Adjacency-Constrained Hierarchical Clustering Using Fine-Grained Pseudo Labels. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 8(3): 2481–2492.

Yang, J.; Ma, Y.; Zhang, X.; Li, S.; and Zhang, Y. 2017. An initialization method based on hybrid distance for k-means algorithm. *Neural computation*, 29(11): 3094–3117.

Zelig, A.; Kariti, H.; and Kaplan, N. 2023. KMD clustering: robust general-purpose clustering of biological data. *Communications Biology*, 6(1): 1110.

Zhang, C.; Zhu, X.; Wang, Z.; Zhong, Y.; Sheng, W.; Ding, W.; and Jiang, B. 2024. Discriminative Multi-View Fusion via Adaptive Regression. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 8(6): 3821–3833.

Zhang, R.; Miao, Z.; Tian, Y.; and Wang, H. 2022. A novel density peaks clustering algorithm based on Hopkins statistic. *Expert Systems with Applications*, 201: 116892.

Zhao, H.; Hu, Q.; Zhu, P.; Wang, Y.; and Wang, P. 2021. A recursive regularization based feature selection framework for hierarchical classification. *IEEE Transactions on Knowledge and Data Engineering*, 33(7): 2833–2846.