Journal of Network Intelligence
Taiwan Ubiquitous Information
 ISSN 2414-8105 (Online)
Volume **10**, Number **3**, August **2025**

Density Peak Clustering Algorithm for Nearest Neighbor Density Estimation and Weighted Similarity

Niu Zhao, Hai-Hua Xie, Li Lv*, Yuan-Min Li, Si-Wei Peng, Xiao-Hong Cha

School of Information Engineering
Nanchang Institute of Technology, Nanchang 330099, China
Nanchang Key Laboratory of IoT Perception and Collaborative Computing for Smart City
zhaoniu23@163.com, Xie_haihua@nit.edu.cn, lvli623@163.com,
1419980075@qq.com, 1344753517@qq.com, 469915604@qq.com

*Corresponding author: Li Lv

Received July 5, 2024, revised December 6, 2024, accepted May 8, 2025.

Abstract. *The density peak clustering algorithm often struggles to accurately identify the cluster centers of sparse clusters when dealing with data exhibiting uneven density distributions, and tends to mistakenly assign samples from sparse clusters to denser ones. To overcome this challenge, we propose the density peaks clustering algorithm for nearest neighbor density estimation and weighted similarity (DPC-NEW). DPC-NEW integrates interactions among all samples during the selection of density peaks, redefines local density by incorporating a density relaxation factor and nearest neighbor information, and reduces the influence of density differences on peak selection, thereby ensuring the correct identification of cluster centers in sparse clusters. In the sample allocation step, an improved similarity strategy is utilized to make samples within the same cluster more tightly associated, thereby enhancing the accuracy of assigning sparse cluster samples. We compare DPC-NEW with five other clustering algorithms on both artificial and UCI datasets to validate its performance. The results demonstrate that DPC-NEW avoids misclassifying the centers of clusters with large density differences, reduces errors in allocating sparse cluster samples and has better overall performance.*
Keywords: Density peaks clustering; Nearest neighbor information; Similarity of samples; Data of uneven density; Cluster analysis

1. **Introduction.** As an essential data mining [1] method, clustering can analyze the data to find the hidden information in it and has been successfully applied across numerous domains, for instance text mining, pattern recognition, market research,and medical research [2–6]. Currently, typical clustering algorithms comprise the K-means [7] and K-medoids [8] based on partitioning [9], the Chameleon [10] based on hierarchy [11], the STING [12] based on lattice [13], the EM [14] based on model [15] and the density-based [16] clustering algorithm. As a renowned density-based clustering algorithm, DBSCAN [17] correctly identifies class clusters of varying shapes when the radius and minimum number of sample parameters are correctly chosen, but it is susceptible to the parameters.

Rodrigues et al. [18] introduced a novel density-based clustering method known as the Density Peaks Clustering algorithm(DPC), which is efficient, simple, capable of identifying class clusters of arbitrary dimensions and shapes, and more straightforward to determine parameters than DBSCAN. However, DPC also has some shortcomings: 1) When selecting

the density peaks, the local density definition in DPC focuses solely on distribution of the samples locally, and if the difference in the density of the class clusters is significant, it can not be accurately expressed, and it is easy to select the center of the class clusters incorrectly. 2) When allocating the rest of the samples, the allocation strategy of DPC tends to assign samples to dense regions with large local densities so that some samples that belong to the sparse regions are incorrectly allocated.

In response to the limitations of local density definition, Du et al. [19] proposed the DPC-KNN, which introduces KNN to redefine local density, thereby accounting for the dataset's local distribution. In literature [20], Lv et al. proposed the minimum second-order K-nearest neighbor concept. They used it to redefine the local density to highlight the difference in density between samples in the center of class clusters and those in the center of non-class clusters. Du et al. [21] introduced the FNDPC. The algorithm introduces a fuzzy neighborhood and computes how samples contribute to local density using Euclidean distance, which improves the robustness of the DPC while maintaining its high efficiency. In the literature [22], Zhao et al. combine K-nearest neighbors and geodesic distance to define local density, which makes the cluster centers of manifold data easier to distinguish. In the literature [23], Zhang et al. combine representative value with K-nearest neighbor density to construct a renewed local density, which renders this density more hierarchical and effectively mitigates the influence of density variations among class clusters on peaks selection.

To address the deficiencies in the design of the allocation strategy of DPC, Zhuo et al. [24] proposed the HCFS, which merges the clusters with high similarity and connectivity to increase the inter-cluster differences, thus effectively identifying the data with uneven density. In the literature [25], Lv et al. solve clustering problems of manifold data easily by choosing geodesic distance as the similarity measure. Xu et al. [26] proposed the FDPC. FDPC firstly obtains some micro-clusters by using the allocation strategy of DPC, then calculates the feedback values among micro-clusters by using the support vectors, and finally merges the micro-clusters recursively based on the feedback values, which improves the clustering accuracy of clusters with multiple density peaks. In the literature [27], Zhao et al. used the mutual proximity between samples to assign the remaining samples to improve the poor clustering accuracy due to the density difference. In literature [28], Ding et al. used the similarity index value to assign connectivity to the samples, which improved the allocation accuracy of streaming data. Guo et al. [29] proposed the DPC-CE, which introduces graph-based strategies to compute the connectivity between the centers of the class clusters and computes the similarity between samples using both connectivity and Euclidean distance, which attenuates the effects of non-sphericality and density inhomogeneity on the performance of the algorithm.

The above algorithms efficiently enhance the clustering performance of DPC. However, they overlook the internal structural attributes of the data, thus failing to achieve satisfactory clustering results for datasets with significant density disparity. Because the density peaks selected by DPC tend to be clustered in dense regions with larger local densities, the density peaks in sparse zone regions tend to be subsumed by the dense regions; at the same time, the allocation strategy of DPC tends to assign samples to densely populated clusters with higher local density, which results in sparse clusters with lower density being incorrectly assigned, thereby reducing clustering accuracy.

Consequently, this article proposes the density peaks clustering algorithm for nearest neighbor density estimation and weighted similarity (DPC-NEW). The innovations of DPC-NEW include: 1) proposing a density estimation strategy for nearest neighbor information, this approach integrates the mutual influence of all samples to balance the local density across sparse and dense regions, thereby accurately selecting density peaks

in these areas. 2) designing a new allocation strategy that introduces natural nearest neighbors and shared nearest neighbors to define the similarity between samples, so that this similarity aligns more proximally with the actual distribution of the samples, thereby enhancing the accuracy of sample allocation in sparse regions.

2. **DPC.** The DPC chooses the sample with both greater local density and relative distance as cluster centers. It then sequentially distributes the rest of the samples to the class clusters whose densities are higher than its own and where the nearest samples are located. The DPC offers two methodologies to compute the local densities ρ_i of the samples. Among them, Formula (1) is the truncated kernel method for datasets of considerable scale; Formula (2) is the Gaussian kernel method utilized for small-scale datasets. The details are as follows:

$$\rho_i = \sum_{j=1} \chi(d_{ij} - d_c), \chi(x) = \begin{cases} 1 & , \quad x < 0 \\ 0 & , \quad x \geq 0 \end{cases} \tag{1}$$

$$\rho_i = \sum \exp\left(-\frac{d_{ij}^2}{d_c^2}\right) \tag{2}$$

where d_{ij} represents the Euclidean distance between samples, d_c represents the truncation distance, which requires human configuration. In Formula (1), ρ_i signifies the number of sample points within the truncation range of the x_i neighborhood, i.e., the count of points within a circle centered at x_i with a radius of d_c. In Equation (2), ρ_i denotes the aggregate of the Gaussian distances between x_i and the other sample points.

The relative distance of DPC is described as follows:

$$\delta_i = \min_{j:\rho_j > \rho_i} (d_{ij}) \tag{3}$$

$$\delta_i = \max_{i \neq j} (\delta_j) \tag{4}$$

Equation (3) signifies the relative distance from sample x_i to its closest high-density sample x_j. In cases where the sample exhibits maximal density, Equation (4) is employed for computing the relative distance.

The DPC typically selects the sample at the upper-right area of the decision chart as the cluster center, distinguished by higher local density and a greater relative distance. However, the decision diagram is determined by human observation rather than quantitative analysis when confirming the class cluster centers, which is more subjective. To objectively and accurately select cluster centers, the decision value γ_i is defined, and the point with a more considerable γ_i -value is taken as the class cluster center. Its calculation Formula is:

$$\gamma_i = \rho_i \cdot \delta_i \tag{5}$$

after determining the peaks density, distribute the rest of the samples to clusters with higher densities than theirs and closest to them, thereby completing the clustering process.

3. **DPC-NEW.** This paper proposes the DPC-NEW, aiming to address the issue that the DPC can't adequately tackle the density-uneven dataset. This algorithm redefines the local density and proposes the weighted similarity allocation strategy.

3.1. **Nearest neighbor information density estimation strategy.** According to Equations (1) and (2) in Section 1, it is evident that the DPC does not have a unified method for local density calculation, requiring the selection of different functions based on the dataset size. The density of the DPC is closely related to the samples within the cutoff distance, and the cluster center is easy to gather in the high-density area. The DPC algorithm typically selects data points with higher density and relative isolation as cluster centers in dense regions, whereas in sparse regions, it may not select any cluster centers. For this reason, the DPC-NEW employs nearest-neighbor information to redefine the local density and proposes the nearest-neighbor information density estimation methodology. The method, unlike conventional kernel functions, takes into account the mutual influences among all samples, unifies the calculation method for local density, and reduces the dependence on d_c.

- K-nearest neighbor (KNN) [30]. For any sample point x_i in the dataset D, the k nearest samples form the k-nearest neighborhood of sample x_i, typically referred to as $KNN(x_i)$. The expression for $KNN(x_i)$ is:

$$KNN\left(x_i\right) = \{x_j \in D | d_{ij} \leq d_{ik}\} \tag{6}$$

- Inter-sample affiliation $\mu\left(x_i, x_j\right)$. Define the affiliation $\mu\left(x_i, x_j\right)$ of x_i sample with x_j as:

$$\mu(x_i, x_j) = \begin{cases} \exp(-d_{ij}) + \sigma & , \quad x_j \in KNN(x_i) \\ \frac{\exp(-d_{ij})}{\exp(-d_{ij})+\sigma} & , \quad x_j \notin KNN(x_i) \end{cases} \tag{7}$$

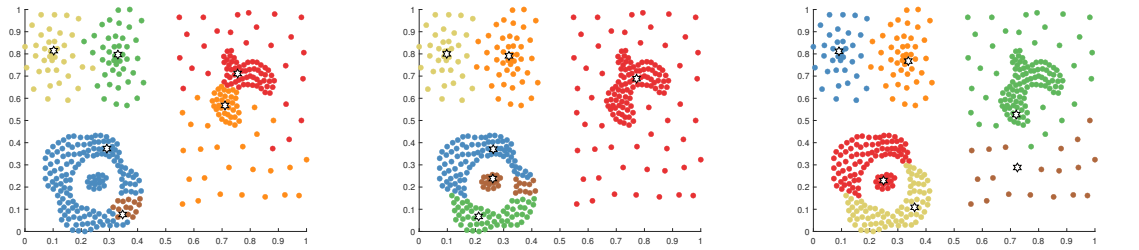
  where σ is the density relaxation factor, usually $\sigma \in [1, 4]$, in this paper, the best results are obtained when σ is 3.
- Local density of nearest-neighbor information ρ_i. The sum of the affiliation of sample x_i with the remaining samples x_j is defined as the local density ρ_i of x_i, with the expression:

$$\rho_i = \sum_{j=1}^{N} \mu_{ij} \tag{8}$$

The local density designed by the algorithm consider both the global distribution of the samples and the neighborhood information of the samples, which makes it more probable that samples of the same k-nearest neighbor set belong to the same cluster. It strengthens the density contribution weight of the k-nearest neighbor samples while ensuring that the density contribution of non-k nearest neighbor points is not ignored. The comprehensively consideration of the mutual influence of all samples and the effect of the density relaxation factor so that the sparse region samples have a greater local density, decreasing the interference of density discrepancies on the choice of the density peaks, to correctly select the density peaks of the sparse region.

In order to verify that the nearest-neighbor density estimation strategy makes it easier to correctly select cluster centers in datasets with significant differences in density, comparative experiments are conducted on the Compound dataset; the outcomes are displayed in Figure 1, in which the 'hexagram' denotes the center of the clusters. The Compound dataset consists of six class clusters with diverse densities and shapes. Among them, the two clusters above the left are sparser; the two clusters at the lower left are denser; and the suitable sparse and dense clusters are intertwined.

Figure 1 (a) and (b) are the cluster centers found using the local density of the cut-off kernel and Gaussian kernel in DPC, respectively. Figure 1 (c) is the cluster center found using the nearest neighbor information density estimation strategy. In Figure 1 (a) and (b), it's evident that the traditional DPC doesn't manage to locate cluster centers in the



(a) cut-off kernel (b) Gaussian kernel (c) nearest neighbor information

FIGURE 1. The density peaks found by different local densities on the Compound dataset

sparse region on the right. This is because the local density of DPC is chiefly dictated by the samples within the truncation distance, resulting in the cluster centers that should belong to the sparse region appearing in the dense region. Figure 1 (c) shows that DPC-NEW can correctly select the clustering center in the right sparse region. This is because the density estimation strategy of neighbor information uses the interaction between the density relaxation factor and all samples, which enlarges the density of sparse clustering samples and increases the probability of sparse region samples becoming clustering centers so that the density peaks can be selected in the right sparse region. Therefore, when handling datasets characterized by uneven density distribution, the neighbor information density estimation strategy proposed by the algorithm enhances the recognition ability of clustering centers.

3.2. **Allocation strategy for weighted similarity.** During the sample allocation process, the DPC assigns the remaining samples nearby. This chain allocation method is less fault-tolerant and prone to cascading effects of allocation faults. About datasets with uneven density, due to the preference of DPC to assign certain samples to locally denser clusters, some samples originally in the category of sparse clusters are misallocated to dense clusters, leading to evident allocation errors and making the clustering effect unsuitable. For this reason, the DPC-NEW proposes a new allocation strategy. This strategy introduces the nearest neighbor concept to calculate the similarity between samples and uses this to allocate the remaining samples, reducing the influence of density disparities on sample allocation.

- Shared nearest neighbor [31]. x is a sample point, D is a dataset, $x_i, x_j \in D$. Call the intersection of $KNN(x_i)$ and $KNN(x_j)$ the collection of shared nearest neighbors of x_i and x_j, denoted as $SNN(x_i, x_j)$. The expression is shown as:

$$SNN(x_i, x_j) = KNN(x_i) \ \cap \ KNN(x_j) \tag{9}$$

- Natural nearest neighbor [31]. If sample x_j belongs to $KNN(x_i)$ while sample x_i belongs to $KNN(x_j)$, then samples x_i and x_j are considered to be natural nearest neighbor points to each other, denoted as $NNN(x_i, x_j)$. The expression is shown as:

$$NNN(x_i, x_j) = \begin{cases} 1 & , \quad x_i \in KNN(x_j) \ and \ x_j \in KNN(x_i) \\ 0 & , \quad others \end{cases} \tag{10}$$

- Inter-sample similarity $Sim(x_i, x_j)$. For samples x_i and x_j, the similarity is defined as follows:

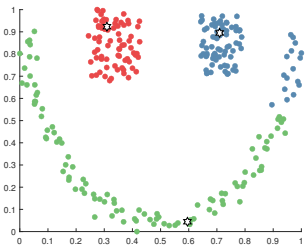
$$Sim(x_i, x_j) = [\ |SNN(x_i, x_j)| \ + NNN(x_i, x_j)\ ]^2 \ \cdot \ A(x_i, x_j) \tag{11}$$

$$A(x_i, x_j) = \frac{\sum\limits_{v\in[KNN(i),i]} \omega_{vj} + \sum\limits_{v\in[KNN(j),j]} \omega_{vi}}{2(k+1)} \tag{12}$$

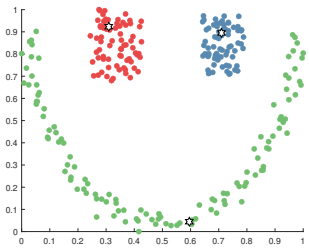
$$\omega(x_i, x_j) = \begin{cases} \mathrm{e}^{-d_{ij}} & , \quad x_j \in KNN(x_i) \\ \frac{1}{4e^{d_{ij}}+1} & , \quad others \end{cases} \tag{13}$$

where $\omega(x_i, x_j)$ is the inter-sample proximity, if x_j is within the k-nearest neighbor set of x_i, then the proximity between x_i and x_j is more significant. Otherwise, the proximity is smaller. Considering the influence of the context within which the samples are situated, the k-neighborhood of x_i and x_j is introduced to compute the inter-sample similarity $A(i, j)$. $|SNN(x_i, x_j)|$ represents the quantity of samples in the set of shared nearest neighbors. The natural nearest neighbors and shared nearest neighbors can reflect the relationships among samples. Therefore, using them to weight $A(i, j)$ forms the final similarity $Sim(x_i, x_j)$ between samples. Obviously, $Sim(x_i, x_j)$ is meaningful when and only when $|SNN(x_i, x_j)|$ or $|NNN(x_i, x_j)|$ is not zero.

Experiments on the Lineblobs dataset, which consists of one manifold cluster and two agglomerated clusters, are conducted to verify that the new allocation strategy can more accurately allocate datasets characterized by large density disparities. The outcomes are shown in Figure 2. Manifold clusters are sparse, and agglomerated clusters are dense.



(a) Allocation strategy for DPC



(b) Allocation strategy for weighted similarity

FIGURE 2. Clustering results of different allocation strategies

Figure 2 presents the clustering results of the two assignment strategies on the Lineblobs dataset. At the top right of Figure 2 (a), some sparse area samples have obvious allocation errors. This is because DPC is more inclined to allocate a particular sample to dense clusters of high density, resulting in chain allocation errors. The allocation strategy proposed in this paper effectively alleviates this situation, as shown in Figure 2 (b). Utilizing similarity to assign the rest of the samples reduces the impact of density differences on sample allocation so that the sparse region samples can be accurately allocated. It can be shown that assignment strategies for weighted similarity is able to efficiently handle datasets with non-uniform density distribution.

The sample allocation process of DPC-NEW algorithm is,in the first place, the inter-sample similarity $Sim(i, j)$ is computed, and the class cluster centers are marked; secondly, from all labeled samples, identify the unlabeled sample with the most considerable $Sim(i, j)$ value and label that sample as the cluster label of the labeled sample; then, the process was repeated until the remaining samples were labeled, or the maximum similarity between all labeled and unlabeled samples was zero; finally, if there are unlabeled samples, they are labeled according to the allocation policy of the DPC.

3.3. **The steps of DPC-NEW.** Input: datasets X, parameter k. Output: clustering results.

- Data preprocessing and normalization;
- Compute the Euclidean distance between samples;
- Compute the relative distance δ_i of the samples based on Equations (3) and (4), and compute the local density ρ_i of the samples based on Equations (6) - (8);
- Compute the decision value γ_i based on Equations (5) and mark the class cluster center;
- Calculate the inter-sample similarity $Sim(i, j)$ based on Equations (9) - (13);
- From all labeled samples, the unlabeled sample with the most giant $Sim(i, j)$ is found, and the unlabeled sample is labeled as the cluster label of the labeled sample;
- If there is a maximum $Sim(i, j)$ of 0, continue to the next step; otherwise, go to the previous step;
- If unlabeled samples are labeled according to the allocation strategy of DPC, the clustering ends.

4. Experimental findings and analysis.

4.1. **Experimental settings.** In this paper, the DPC-NEW is clustered with the DPC [18], DPC-FWSN [32], DPC-CE [29], FNDPC [21] and FKNN-DPC [33] on 18 artificial datasets and eight real datasets to validate its robustness. In order to better reflect the performance of each algorithm, this experiment tunes the parameters of each algorithm. However, due to the different factors of dataset size, dimensions and features, it is usually impossible to determine a parameter value with a general rule to be applied to all kinds of datasets. Therefore, by setting the range of algorithm parameter values and adjusting the parameter values according to the step size, the optimal results are taken after several tests on each dataset. The specific details are: the number of neighbors of DPC-NEW, FKNN-DPC and DPC-FWSN is $k \in [2, 50]$, and the step size is 1; the $\varepsilon \in [0.01, 1]$ of the FNDPC, the step size is 0.01; the $d_c \in [0.1\%, 5\%]$ of DPC, the step size is 0.1%; the DPC-CE has no input parameters and does not need to be tuned. The hardware environment for all experiments in the article is the 64-bit operating system of Windows 10, Intel (R) Core (TM) i5-7200U CPU @ 2.50GHz 2.70GHz, 8G memory, Matlab R2021a.

In this article, three assessment indicators are employed to appraise the clustering outcome: adjusted mutual information (AMI) [34], adjusted rand index (ARI) [34] and Fowlkes-Mallows index (FMI) [35]. AMI is used to measure the similarity between two clustering results, the closer the value is to 1, the more similar the clustering results obtained by the algorithm are to the real category labels. ARI is used to assess the consistency between two sets of clustering results, the closer the value is to 1, the more consistent the clustering results are. FMI measures the quality of the clustering by the precision of the clustering and the recall rate, the closer the value is to 1 the better the clustering results are.

4.2. Experimental findings and analysis of artificial datasets.

4.2.1. *Analysis of experimental results of datasets with uneven density distribution.* The datasets utilized in this section are displayed in Table 1, which are two-dimensional data with significant differences in density, ranging from 266 to 1741 samples, to test the clustering effectiveness of the DPC-NEW on dataset with unevenly distributed density. Table 2 displays the evaluation metrics of six algorithms on the clustering results of the dataset mentioned above. Among them, the optimal result is highlighted in bold font;

the optimal parameters are represented by 'Arg-', and if the value is '-', it means that the algorithm does not need to adjust the parameters. According to Table 2, the DPC-NEW achieves evaluation metrics of 0.98, 0.99 and 0.99 on the Compound dataset, respectively, approaching the optimal value of 1. And the evaluation metric values of DPC-NEW on the Pathbased dataset exceed those of the five comparison algorithms. Additionally, on the Jain, Ls and the remaining six datasets, the metrics of DPC-NEW all reach the optimal score of 1, indicating the best clustering effectiveness. To sum up, the DPC-NEW is superior to other algorithms in dealing with datasets with uneven density.

Table 1. Datasets with uneven density distribution

Datasets	Scales	Dimension	Number of clusters
Compound	399	2	6
Jain	373	2	2
Cth	1016	2	4
Ring	1000	2	2
Pathbased	300	2	3
Cmc	1002	2	3
Ls	1741	2	6
LineBlobs	266	2	3

Table 2. Clustering results of six algorithms on datasets with uneven density distribution

Algorithms	AMI	ARI	FMI	Arg-	AMI	ARI	FMI	Arg-
	Compound				Cth			
DPC-NEW	**0.9831**	**0.9925**	**0.9943**	7	**1**	**1**	**1**	4
DPC-FWSN	0.8664	0.8780	0.9115	16	**1**	**1**	**1**	13
DPC-CE	0.8082	0.6141	0.7060	-	0.8255	0.7158	0.7935	-
FNDPC	0.8516	0.8684	0.9046	0.38	0.8758	0.8327	0.8786	0.45
FKNN-DPC	0.8467	0.8430	0.8884	7	**1**	**1**	**1**	22
DPC	0.7754	0.5910	0.6876	3.8	0.6866	0.5135	0.6473	0.1
	Ring				Jain			
DPC-NEW	**1**	**1**	**1**	5	**1**	**1**	**1**	10
DPC-FWSN	0.2387	0.1627	0.6516	3	**1**	**1**	**1**	4
DPC-CE	**1**	**1**	**1**	-	**1**	**1**	**1**	-
FNDPC	0.5508	0.5651	0.7892	0.47	0.5961	0.7257	0.9051	0.47
FKNN-DPC	**1**	**1**	**1**	6	0.7092	0.8224	0.9359	43
DPC	0.2753	0.2056	0.6595	0.5	0.6183	0.7146	0.8819	0.3
	Pathbased				Cmc			
DPC-NEW	**0.935**	**0.9593**	**0.9728**	20	**1**	**1**	**1**	13
DPC-FWSN	0.5878	0.5517	0.7472	7	**1**	**1**	**1**	6
DPC-CE	0.4462	0.3787	0.6285	-	0.6694	0.7362	0.8352	-
FNDPC	0.5751	0.5067	0.7065	0.01	0.8093	0.8421	0.9027	0.28
FKNN-DPC	0.9305	0.9499	0.9665	9	**1**	**1**	**1**	49
DPC	0.5335	0.5127	0.7322	0.1	0.3857	0.2661	0.5377	5
	Ls				LineBlobs			
DPC-NEW	**1**	**1**	**1**	5	**1**	**1**	**1**	4
DPC-FWSN	**1**	**1**	**1**	15	**1**	**1**	**1**	21
DPC-CE	0.7435	0.6392	0.7415	-	**1**	**1**	**1**	-
FNDPC	0.7564	0.6898	0.7808	0.37	0.7794	0.7179	0.8148	0.11
FKNN-DPC	0.8719	0.8179	0.8735	48	**1**	**1**	**1**	7
DPC	0.7665	0.6894	0.7779	0.9	0.8375	0.8237	0.8842	4.2

Friedman test [36] is a nonparametric statistical approach accustomed to compare whether the mean differences among three or more related samples are significant. Performing a Friedman test on the evaluation metrics of six algorithms can more accurately

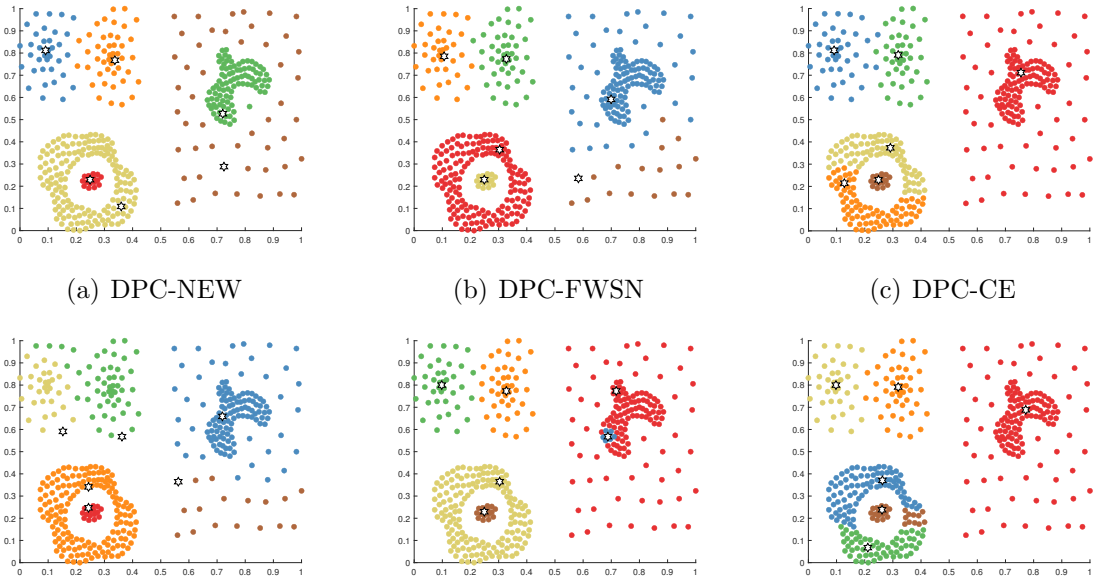
reflect the differences in evaluation metrics among them, with a higher rank mean indicating superior algorithm performance. As depicted in Table 3, the rank-mean values of the clustering evaluation indexes of the DPC-NEW are superior to those of the other algorithms, which indicates that the clustering performance of DPC-NEW significantly surpasses that of the DPC-FWSN, DPC-CE, FNDPC, FKNN-DPC and DPC on datasets with uneven density.

TABLE 3. Friedman values on datasets with uneven density distribution

Algorithms	Mean ranks AMI	Mean ranks ARI	Mean ranks FMI
DPC-NEW	**5.25**	**5.25**	**5.25**
DPC-FWSN	4.38	4.38	4.38
DPC-CE	2.81	2.81	2.81
FNDPC	2.50	2.63	2.63
FKNN-DPC	4.31	4.31	4.31
DPC	1.75	1.63	1.63

In order to more intuitively reflect the clustering effectiveness of the DPC-NEW for datasets with unevenly distributed density, two sets of typical experimental results figures are used to illustrate. In the figure, different clusters are distinguished by color, and the density peaks are represented by a 'hexagonal star'.

Figure 3 displays the clustering results of six algorithms on the Compound dataset, which comprises six clusters with varying shapes. Clearly, the DPC-NEW has the capability to accurately choose the centers of class clusters without any discernible assignment errors, and the clustering results are nearly optimal. Although the DPC-FWSN and FNDPC can also correctly select the class cluster centers, their right-side samples are assigned incorrectly. The FKNN-DPC can accurately cluster the left-side samples but incorrectly identify the centers of the right-side clusters, which reduces the clustering effect. The DPC-CE and DPC misjudge the class cluster centers of the lower left and right sides, resulting in a poor clustering effect.

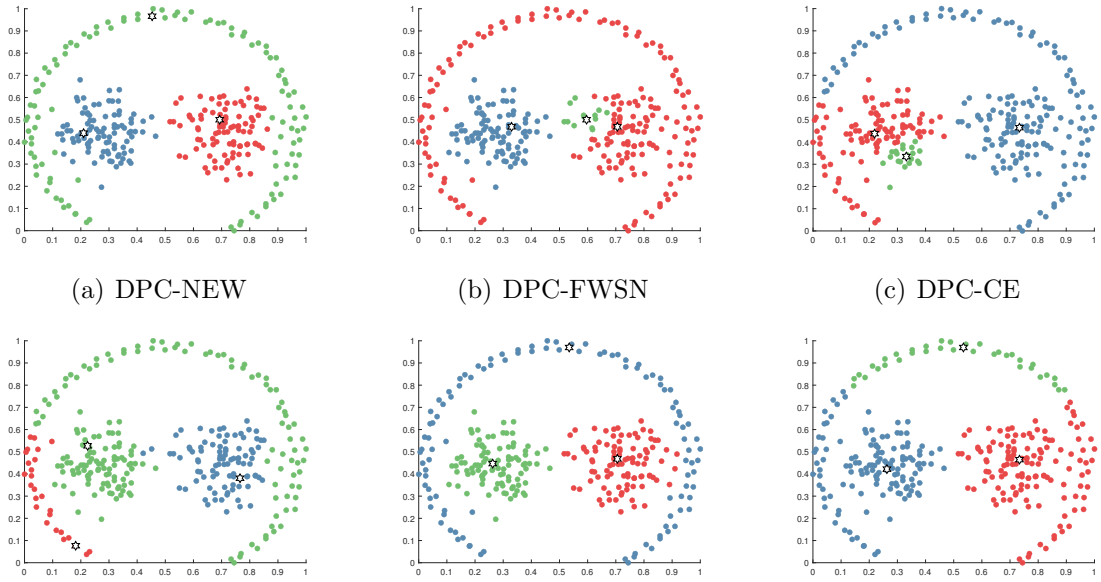


(a) DPC-NEW (b) DPC-FWSN (c) DPC-CE

(d) FNDPC (e) FKNN-DPC (f) DPC

FIGURE 3. Clustering results on Compound datasets

Figure 4 displays the clustering results of the six algorithms for the Pathbased dataset, which consists of one manifold cluster and two agglomerated clusters. Among them, the manifold cluster is sparser, and the agglomerated clusters are denser. There are cases on both the left and right sides where some of the sparse and dense area samples are relatively closer, making cascade errors very easy. However, DPC-NEW and FKNN-DPC have no obvious allocation errors and excellent clustering results. FNDPC and DPC incorrectly allocate more samples from the manifold cluster to the agglomerative cluster, thus having obvious allocation errors. Both DPC-FWSN and DPC-CE select three class cluster centers in the two agglomerative clusters, which makes clustering effects unsatisfactory.



(a) DPC-NEW (b) DPC-FWSN (c) DPC-CE

(d) FNDPC (e) FKNN-DPC (f) DPC

FIGURE 4. Clustering results on Pathbased datasets

4.2.2. *Analysis of experimental results of complex datasets.* To further verify the robustness of the DPC-NEW, this subsection uses complex datasets, as shown in Table 4, to test the above six algorithms. These datasets include streaming datasets such as Db, Circle3 and Spiral, which are predominantly arc-shaped or ring-shaped; datasets D31 and A3, which have larger sample sizes, more class clusters, and overlapping clusters; and datasets 2d-20c, which have more ambiguous distributions. They can better test the generalizability of each algorithm to the problem.

TABLE 4. Complex datasets

Datasets	Scales	Dimension	Number of clusters
Elliptical	500	2	10
Circle3	1897	2	3
R15	600	2	15
Flame	240	2	2
Db	630	2	4
A3	7500	2	50
Aggregation	788	2	7
D31	3100	2	31
Spiral	312	2	3
2d-20c	1517	2	20

Table 5 presents the evaluation metric values for the six algorithms on the complex datasets listed in Table 4. Overall, the DPC-NEW, DPC-FWSN and FKNN-DPC significantly outperform others in clustering these ten datasets. The DPC-NEW achieves the highest evaluation metrics across six datasets, such as Flame; the DPC-FWSN achieves the highest evaluation metrics across three datasets, such as Aggregation; and the FKNN-DPC achieved the highest evaluation metrics only on the D31 and A3 datasets.

Table 5. Clustering results of six algorithms on complex datasets

Algorithms	AMI	ARI	FMI	Arg-	AMI	ARI	FMI	Arg-
	Aggregation				Circle3			
DPC-NEW	**0.9955**	**0.9978**	**0.9983**	8	**1**	**1**	**1**	18
DPC-FWSN	**0.9955**	**0.9978**	**0.9983**	14	0.7780	0.7615	0.8775	41
DPC-CE	0.9922	0.9956	0.9966	-	0.5290	0.2555	0.6279	-
FNDPC	0.9864	0.9913	0.9932	0.02	0.4236	0.2732	0.5863	0.29
FKNN-DPC	0.9905	0.9949	0.9960	20	0.7063	0.6139	0.7790	32
DPC	0.9922	0.9956	0.9966	3.1	0.3596	0.3015	0.6048	0.3
	Db				Flame			
DPC-NEW	**1**	**1**	**1**	9	**1**	**1**	**1**	34
DPC-FWSN	0.6525	0.4942	0.6970	17	**1**	**1**	**1**	44
DPC-CE	0.6758	0.5588	0.7395	-	**1**	**1**	**1**	-
FNDPC	0.6431	0.4412	0.6700	0.74	**1**	**1**	**1**	0.13
FKNN-DPC	0.5107	0.2718	0.5793	19	0.9267	0.9666	0.9845	5
DPC	0.4799	0.3633	0.6067	1	**1**	**1**	**1**	2.7
	R15				D31			
DPC-NEW	**0.9938**	**0.9928**	0.9932	15	0.9631	0.9491	0.9507	45
DPC-FWSN	**0.9938**	**0.9928**	**0.9933**	18	0.9639	0.9480	0.9497	31
DPC-CE	**0.9938**	**0.9928**	0.9932	-	0.9557	0.9364	0.9385	-
FNDPC	**0.9938**	**0.9928**	**0.9933**	0.03	0.9555	0.9364	0.9385	0.04
FKNN-DPC	**0.9938**	**0.9928**	**0.9933**	27	**0.9654**	**0.9523**	**0.9538**	28
DPC	**0.9938**	**0.9928**	0.9932	0.5	0.9568	0.9389	0.9409	1.6
	Elliptical				2d-20c			
DPC-NEW	**0.9955**	**0.9955**	**0.9960**	25	**0.9939**	**0.9941**	**0.9944**	11
DPC-FWSN	0.9333	0.8746	0.8907	14	0.9926	0.9927	0.9931	11
DPC-CE	0.9664	0.9495	0.9544	-	0.9735	0.9660	0.9679	-
FNDPC	0.9790	0.9668	0.9701	0.04	0.9913	0.9916	0.9921	0.06
FKNN-DPC	0.9081	0.8607	0.8795	14	0.9659	0.9353	0.9404	7
DPC	0.9790	0.9668	0.9701	0.8	0.9926	0.9932	0.9935	2.8
	Spiral				A3			
DPC-NEW	**1**	**1**	**1**	2	0.9909	0.9860	0.9863	47
DPC-FWSN	**1**	**1**	**1**	2	0.9819	0.9633	0.9642	43
DPC-CE	**1**	**1**	**1**	-	0.9520	0.8839	0.8888	-
FNDPC	**1**	**1**	**1**	0.07	0.9878	**0.9880**	**0.9882**	0.03
FKNN-DPC	**1**	**1**	**1**	6	**0.9926**	**0.9880**	**0.9882**	22
DPC	**1**	**1**	**1**	1.7	0.9876	0.9814	0.9818	0.3

Table 6. Friedman values on complex datasets

Algorithms	Mean ranks	Mean ranks	Mean ranks
	AMI	ARI	FMI
DPC-NEW	**4.95**	**4.95**	**4.85**
DPC-FWSN	3.90	3.75	3.65
DPC-CE	3.05	2.80	2.90
FNDPC	2.95	3.15	3.25
FKNN-DPC	3.00	2.85	3.05
DPC	3.15	3.50	3.30

Table 6 is the rank-mean of the evaluation indexes of the six algorithms on ten complex datasets. Among them, the rank-mean values of AMI, ARI and FMI of the DPC-NEW

are the largest. That is to say, the DPC-NEW outperforms the DPC-FWSN, DPC-CE, FNDPC, FKNN-DPC and DPC for clustering complex datasets.

4.3. **Experimental findings and analysis of UCI datasets.** To further validate the robustness of the DPC-NEW, this section carries out experiments for six algorithms across eight UCI datasets. Compared to the uneven density distribution and complex datasets mentioned earlier, UCI datasets better test the robustness of the algorithms, as they include data that exists in real-life scenarios. Table 7 presents the basic details of the eight UCI datasets chosen for this study.

Table 7. UCI datasets

Datasets	Scales	Dimension	Number of clusters
Seeds	210	7	3
Wdbc	569	30	2
Iris	150	4	3
Wine	178	13	3
Inonsphere	351	34	2
Ecoli	336	8	8
Dermatology	366	33	6
Libras	360	90	15

Table 8. Clustering results of six algorithms on UCI datasets

Algorithms	AMI	ARI	FMI	Arg-	AMI	ARI	FMI	Arg-
	Seeds				Ecoli			
DPC-NEW	0.7694	**0.8128**	**0.8746**	7	**0.6862**	**0.7698**	0.8316	14
DPC-FWSN	0.7366	0.7871	0.8574	7	0.6637	0.7331	0.8059	6
DPC-CE	0.7144	0.7448	0.8297	-	0.6373	0.7398	0.8190	-
FNDPC	0.7136	0.7545	0.8361	0.07	0.4833	0.5618	0.7178	0.35
FKNN-DPC	**0.7757**	0.8024	0.8682	9	0.5878	0.5894	0.7027	2
DPC	0.7298	0.7670	0.8444	0.7	0.5179	0.4365	0.5693	0.2
	Wine				Iris			
DPC-NEW	**0.8885**	**0.9134**	**0.9425**	24	**0.8831**	**0.9038**	**0.9355**	3
DPC-FWSN	0.8598	0.8837	0.9227	32	**0.8831**	**0.9038**	**0.9355**	32
DPC-CE	0.5841	0.5362	0.6945	-	0.7277	0.6634	0.7824	-
FNDPC	0.7898	0.8025	0.8686	0.26	**0.8831**	**0.9038**	**0.9355**	0.11
FKNN-DPC	0.8481	0.8839	0.9229	8	**0.8831**	**0.9038**	**0.9355**	22
DPC	0.7695	0.7703	0.8474	2.4	**0.8831**	**0.9038**	**0.9355**	3.2
	Dermatology				Inonsphere			
DPC-NEW	0.8538	0.8455	0.8768		**0.3969**	**0.5181**	**0.7958**	25
DPC-FWSN	**0.8610**	**0.8904**	**0.9120**	21	0.3664	0.4746	0.7842	10
DPC-CE	0.8200	0.8112	0.8496	-	0.0704	0.1145	0.5802	-
FNDPC	0.7933	0.8029	0.8441	0.17	0.1630	0.2483	0.6531	0.06
FKNN-DPC	0.8066	0.8361	0.8709	35	0.3485	0.4790	0.7716	8
DPC	0.8308	0.8464	0.8764	1.6	0.1484	0.2306	0.6449	1.8
	Libras				Wdbc			
DPC-NEW	0.5651	**0.3692**	**0.4198**	12	**0.7302**	**0.8307**	**0.9220**	30
DPC-FWSN	0.5342	0.3287	0.4038	14	0.6697	0.7796	0.8993	9
DPC-CE	0.5570	0.3531	0.4192	-	0.3742	0.4355	0.7743	-
FNDPC	0.5494	0.3290	0.3869	0.17	0.6076	0.7305	0.8758	0.05
FKNN-DPC	0.5554	0.3459	0.4044	10	0.6423	0.7613	0.8894	2
DPC	**0.5832**	0.3626	0.4190	0.5	0.6375	0.7548	0.8876	0.7

Table 8 shows the evaluation index values for the clustering results of the six algorithms on the eight UCI datasets. It is evident that the clustering effectiveness of the DPC-NEW surpasses that of the comparison algorithms on four datasets, including Wine; the evaluation indexes on the Seed and Dermatology datasets are second only to the maximum

of the five comparison algorithms; on the Iris dataset, the evaluation indexes of the DPC-NEW and the four comparison algorithms have the same value, all of which surpass those of the DPC-CE. By comprehensively comparing the evaluation indexes of the algorithms, the clustering effect of the DPC-NEW for the UCI dataset is better than that of the other algorithms.

Table 9 shows the rank-means of the evaluation indexes for the six algorithms across the eight UCI datasets. It can be observed that, compared to the other five algorithms, the DPC-NEW achieves the highest rank-mean for each evaluation index. The clustering performance of the DPC-NEW on the UCI dataset significantly outperforms that of the others.

Table 9. Friedman values on UCI datasets

Algorithms	Mean ranks AMI	Mean ranks ARI	Mean ranks FMI
DPC-NEW	**5.38**	**5.50**	**5.63**
DPC-FWSN	4.38	4.00	4.25
DPC-CE	2.13	2.00	2.13
FNDPC	2.13	2.38	2.38
FKNN-DPC	3.75	4.00	3.75
DPC	3.25	3.13	2.88

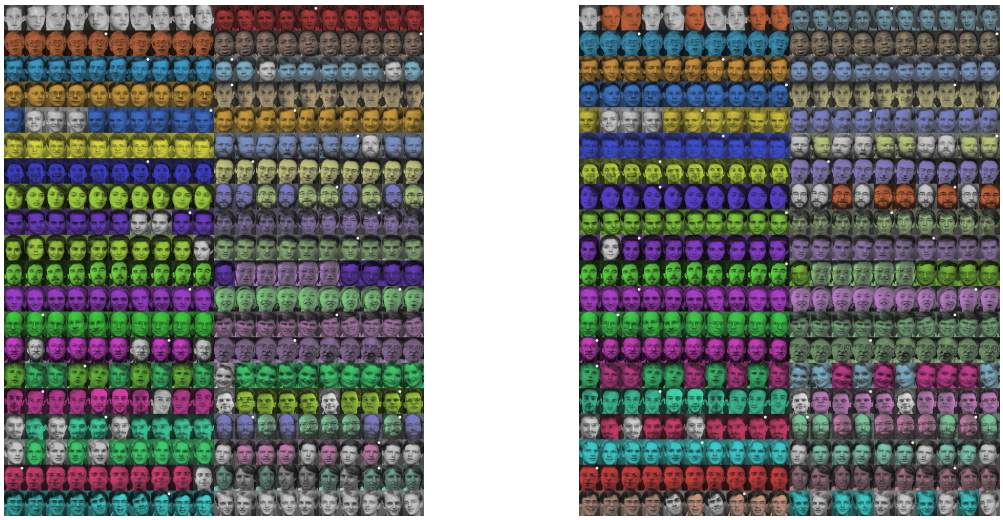
4.4. **Experimental findings and analysis of the face image dataset.** To validate the clustering performance of the DPC-NEW on high-dimensional image datasets, this subsection applies six clustering algorithms to the Olivetti Faces dataset. The Olivetti Faces dataset [37] is a classic image dataset from the Olivetti Research Laboratory in Cambridge, UK. The dataset contains 400 face images taken by 40 subjects under different conditions of time, light and expression. The small white dots indicate the class cluster centers. As shown in Figure 5, every 10 images from left to right is 1 subject, and 40 subjects are numbered from 1 to 40, with 1 to 20 on the left and 21 to 40 on the right. It can be found that in Figure 5(a), subjects 10, 14, 16, 18, 19, 23 and 39 all have 1-5 images not recognized by the DPC, while the DPC-NEW is able to correctly recognize all of these images, as shown in Figure 5(b); subjects 1, 38 and 40 all have 5-10 images not recognized by the DPC, although the DPC-NEW also fails to completely recognize them, but it reduced the recognition error rate so that the number of unrecognized images was reduced by 2-5 images. Combined with Table 10, it can be seen that the clustering performance of the DPC-NEW on the face dataset is better than that of DPC and its improved algorithms.

Table 10. Clustering results of six algorithms on the Olivetti Faces dataset

Algorithms	AMI	ARI	FMI	Arg-
DPC-NEW	**0.8568**	**0.7657**	**0.7725**	5
DPC-FWSN	0.8361	0.7353	0.7436	44
DPC-CE	0.8022	0.6639	0.6333	-
FNDPC	0.7965	0.6652	0.6778	0.44
FKNN-DPC	0.7989	0.6784	0.6885	5
DPC	0.7879	0.6495	0.6657	3.8

5. **Conclusion.** When dealing with data of uneven density distribution, the cluster centers chosen by the DPC tend to be clustered in dense regions with higher densities. Additionally, the allocation strategy of DPC is prone to assigning samples that would

have belonged to sparse clusters into dense clusters. To address these issues, we propose DPC-NEW. This algorithm incorporates both the density relaxation factor and the mutual influence between all samples when selecting cluster centers, enabling it to accurately identify density peaks in sparse clusters. It also uses similarity to assign the remaining samples to improve the accuracy of sample allocation in sparse clusters, which effectively mitigates the 'domino effect'. Experimental results show that DPC-NEW outperforms comparison algorithms on both artificial and UCI datasets. However, it currently requires the manual setting of the parameter k. Future research will focus on enabling the algorithm to select the k value adaptively [38, 39] based on the different datasets. Meanwhile, the introduction of group intelligence is expected to further improve the performance of the clustering algorithm. Specifically, the group intelligence evolution [40–42] pathway is anticipated to enhance the algorithm's self-adaptation. In addition, we will explore the use of deep learning [43,44] to compress data dimensions to reduce the amount of computation, which improving the execution efficiency of the algorithm.



(a) DPC (b) DPC-NEW

FIGURE 5. Clustering results of the two algorithms on the Olivetti Faces dataset

Acknowledgment. This work is supported by the National Natural Science Foundation of China(No.62066030).

REFERENCES

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

[2] Z.-H. Cao, X. Huang, W.-Z. Peng, and Y.-C. Liu, "The topic mining based on word embedding characteristics clustering," *Journal of Jiangxi Normal University(Natural Science Edition)*, vol. 46, no. 5, pp. 468–474, 2022.

[3] M. Gao and G.-Y. Shi, "Ship-handling behavior pattern recognition using ais sub-trajectory clustering analysis based on the t-sne and spectral clustering algorithms," *Ocean Engineering*, vol. 205, p. 106919, 2020.

[4] H.-J. Zhao and C.-Z. He, "Objective cluster analysis in value-based customer segmentation method," in *2009 Second International Workshop on Knowledge Discovery and Data Mining.* IEEE, 2009, pp. 484–487.

[5] Y. Zhao, Z.-Y. Fang, C.-X. Lin, C. Deng, Y.-P. Xu, and H.-D. Li, "Rfcell: A gene selection approach for scrna-seq clustering based on permutation and random forest," *Frontiers in Genetics*, vol. 12, p. 665843, 2021.

[6] H.-H. Shi, S.-W. Zhou, L.-H. Zhong, and Z.-X. Xiao, "The software defect prediction model combining improved sampling algorithm and unsupervised clustering," *Journal of Jiangxi Normal University(Natural Science Edition)*, vol. 48, no. 3, pp. 301–310, 2024.

[7] J. A. Hartigan and M. A. Wong, "Algorithm as 136: A k-means clustering algorithm," *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, vol. 28, no. 1, pp. 100–108, 1979.

[8] H.-S. Park and C.-H. Jun, "A simple and fast algorithm for k-medoids clustering," *Expert Systems with Applications*, vol. 36, no. 2, pp. 3336–3341, 2009.

[9] P. Tavallali, P. Tavallali, and M. Singhal, "K-means tree: an optimal clustering tree for unsupervised learning," *The Journal of Supercomputing*, vol. 77, no. 5, pp. 5239–5266, 2021.

[10] G. Karypis, E.-H. Han, and V. Kumar, "Chameleon: Hierarchical clustering using dynamic modeling," *Computer*, vol. 32, no. 8, pp. 68–75, 1999.

[11] Z.-Y. Chu, W.-F. Wang, B.-Z. Li, W.-C. Jin, S.-Y. Liu, B. Zhang, and Z.-Z. Lin, "An operation health status monitoring algorithm of special transformers based on birch and gaussian cloud methods," *Energy Reports*, vol. 7, pp. 253–260, 2021.

[12] W. Wang, J. Yang, and R. Muntz, "Sting: A statistical information grid approach to spatial data mining," in *Proceedings of the 23rd International Conference on Very Large Data Bases*, 1997, pp. 186–195.

[13] V. Bureva, S. Popov, V. Traneva, and S. Tranev, "Generalized net model of cluster analysis using clique: clustering in quest," *International Journal Bioautomation*, vol. 23, no. 2, p. 131, 2019.

[14] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the Royal Statistical Society: series B (Methodological)*, vol. 39, no. 1, pp. 1–22, 1977.

[15] J. Ren, Z.-L. Wang, M. Xu, F. Fang, and Z.-G. Ding, "An em-based user clustering method in non-orthogonal multiple access," *IEEE Transactions on Communications*, vol. 67, no. 12, pp. 8422–8434, 2019.

[16] Q.-D. Zhu, X.-M. Tang, and A. Elahi, "Application of the novel harmony search optimization algorithm for dbscan clustering," *Expert Systems with Applications*, vol. 178, p. 115054, 2021.

[17] M. Ester, H.-P. Kriegel, J. Sander, and X.-W. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining*, 1996, pp. 226–231.

[18] A. Rodriguez and A. Laio, "Clustering by fast search and find of density peaks," *Science*, vol. 344, no. 6191, pp. 1492–1496, 2014.

[19] M.-J. Du, S.-F. Ding, and H.-J. Jia, "Study on density peaks clustering based on k-nearest neighbors and principal component analysis," *Knowledge-Based Systems*, vol. 99, pp. 135–145, 2016.

[20] L. Lv, M.-Z. Zhu, P. Kang, and L.-Z. Han, "Density peaks clustering with second-order k-nearest neighbors and multi-cluster merging," *Journal of Jilin University (Engineering and Technology Edition)*, vol. 54, no. 5, pp. 1417–1425, 2024.

[21] M.-J. Du, S.-F. Ding, and Y. Xue, "A robust density peaks clustering algorithm using fuzzy neighborhood," *International Journal of Machine Learning and Cybernetics*, vol. 9, no. 7, pp. 1131–1140, 2018.

[22] J. Zhao, G. Wang, L. Lv, and T.-H. Fan, "Density peaks clustering algorithm based on geodesic distance and cosine mutual reverse nearest neighbors for manifold datasets," *Acta Electronica Sinica*, vol. 50, no. 11, pp. 2730–2737, 2022.

[23] Q.-H. Zhang, J.-P. Zhou, Y.-Y. Dai, and G.-Y. Wang, "Density peaks clustering algorithm based on representative points and k-nearest neighbors," *Journal of Software*, vol. 34, no. 12, pp. 5629–5648, 2023.

[24] L.-L. Zhuo, K.-L. Li, B. Liao, H. Li, X.-H. Wei, and K.-Q. Li, "Hcfs: a density peak based clustering algorithm employing a hierarchical strategy," *IEEE Access*, vol. 7, pp. 74 612–74 624, 2019.

[25] L. Lv, J.-Y. Wang, R.-X. Wu, H. Wang, and I. Lee, "Density peaks clustering based on geodetic distance and dynamic neighbourhood," *International Journal of Bio-Inspired Computation*, vol. 17, no. 1, pp. 24–33, 2021.

[26] X. Xu, S.-F. Ding, H. Xu, H.-M. Liao, and Y. Xue, "A feasible density peaks clustering algorithm with a merging strategy," *Soft Computing*, vol. 23, no. 13, pp. 5171–5183, 2019.

[27] J. Zhao, Z.-F. Yao, L. Lü, and T.-H. Fan, "Density peaks clustering based on mutual neighbor degree," *Control and Decision*, vol. 36, no. 3, pp. 543–552, 2021.

[28] Z.-C. Ding and H.-W. Ge, "Density peaks clustering with optimized allocation strategy," *Journal of Frontiers of Computer Science and Technology*, vol. 14, no. 5, pp. 792–802, 2020.

[29] W.-J. Guo, W.-H. Wang, S.-P. Zhao, Y.-L. Niu, Z.-Y. Zhang, and X.-G. Liu, "Density peak clustering with connectivity estimation," *Knowledge-Based Systems*, vol. 243, p. 108501, 2022.

[30] J. Zhao, Q. Ma, R.-B. Xiao, J.-S. Pan, and L.-Z. Han, "Density peaks clustering based on shared nearest neighbor for manifold datasets," *CAAI Transactions on Intelligent Systems*, vol. 18, no. 4, pp. 719–730, 2023.

[31] J. Zhao, L. Chen, R.-X. Wu, B. Zhang, and L.-Z. Han, "Density peaks clustering algorithm with k-nearest neighbors and weighted similarity," *Control Theory & Applications*, vol. 39, no. 12, pp. 2349–2357, 2022.

[32] J. Zhao, G. Wang, J.-S. Pan, T.-H. Fan, and I. Lee, "Density peaks clustering algorithm based on fuzzy and weighted shared neighbor for uneven density datasets," *Pattern Recognition*, vol. 139, p. 109406, 2023.

[33] J.-Y. Xie, H.-C. Gao, W.-X. Xie, X.-H. Liu, and P. W. Grant, "Robust clustering by detecting density peaks and assigning points based on fuzzy weighted k-nearest neighbors," *Information Sciences*, vol. 354, pp. 19–40, 2016.

[34] N.-X. Vinh, J. Epps, and J. Bailey, "Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance," *Journal of Machine Learning Research*, vol. 11, no. 1, pp. 2837–2854, 2010.

[35] E. B. Fowlkes and C. L. Mallows, "A method for comparing two hierarchical clusterings," *Journal of the American Statistical Association*, vol. 78, no. 383, pp. 553–569, 1983.

[36] D. W. Zimmerman and B. D. Zumbo, "Relative power of the wilcoxon test, the friedman test, and repeated-measures anova on ranks," *The Journal of Experimental Education*, vol. 62, no. 1, pp. 75–86, 1993.

[37] F. S. Samaria and A. C. Harter, "Parameterisation of a stochastic model for human face identification," in *Proceedings of 1994 IEEE Workshop on Applications of Computer Vision*. IEEE, 1994, pp. 138–142.

[38] S. Li, R.-X. Wu, P. Kang, and J. Zhao, "The adp-tri-training: tri-training with adaptive editing and probability parameters," *Journal of Jiangxi Normal University(Natural Science Edition)*, vol. 47, no. 5, pp. 490–496, 2023.

[39] J. Zhao, D.-D. Chen, R.-B. Xiao, J. Chen, J.-S. Pan, Z.-H. Cui, and H. Wang, "Multi-objective firefly algorithm with adaptive region division," *Applied Soft Computing*, vol. 147, p. 110796, 2023.

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

[41] J. Zhao, J.-J. Tang, A.-Y. Shi, T.-H. Fan, and L.-Z. Xu, "Improved density peaks clustering based on firefly algorithm," *International Journal of Bio-Inspired Computation*, vol. 15, no. 1, pp. 24–42, 2020.

[42] L. Lv, J. Zhao, J.-Y. Wang, and T.-H. Fan, "Multi-objective firefly algorithm based on compensation factor and elite learning," *Future Generation Computer Systems*, vol. 91, pp. 37–47, 2019.

[43] Y.-B. Liu, W.-L. Kong, X.-R. Liu, Y.-X. Wang, and W.-F. Wang, "The multimodal-based method for tobacco sales forecast," *Journal of Jiangxi Normal University(Natural Science Edition)*, vol. 47, no. 5, pp. 497–505, 2023.

[44] C.-X. Li, J. Zhao, L.-Z. Han, T.-H. Fan, and Z.-Z. Li, "The short-time temperature prediction for multi-channel cnn- bilstm," *Journal of Jiangxi Normal University(Natural Science Edition)*, vol. 47, no. 3, pp. 325–330, 2023.