J Inf Process Syst, Vol.19, No.6, pp.778~790, December 2023
https://doi.org/10.3745/JIPS.04.0294

ISSN 1976-913X (Print)
ISSN 2092-805X (Electronic)

JOURNAL OF INFORMATION PROCESSING SYSTEMS JIPS

A Density Peak Clustering Algorithm Based on Information Bottleneck

Yongli Liu*, Congcong Zhao, and Hao Chao

Abstract

Although density peak clustering can often easily yield excellent results, there is still room for improvement when dealing with complex, high-dimensional datasets. One of the main limitations of this algorithm is its reliance on geometric distance as the sole similarity measurement. To address this limitation, we draw inspiration from the information bottleneck theory, and propose a novel density peak clustering algorithm that incorporates this theory as a similarity measure. Specifically, our algorithm utilizes the joint probability distribution between data objects and feature information, and employs the loss of mutual information as the measurement standard. This approach not only eliminates the potential for subjective error in selecting similarity method, but also enhances performance on datasets with multiple centers and high dimensionality. To evaluate the effectiveness of our algorithm, we conducted experiments using ten carefully selected datasets and compared the results with three other algorithms. The experimental results demonstrate that our information bottleneck-based density peaks clustering (IBDPC) algorithm consistently achieves high levels of accuracy, highlighting its potential as a valuable tool for data clustering tasks.

Keywords

Density Peak Clustering, Information Bottleneck, Multicenter Clustering

1. Introduction

Considering the big data age, we have entered a new era of big data development, facing a new technological revolution and a critical juncture in big data generation, clustering techniques have become increasingly important in a wide range of applications, such as text mining, search engine result classification and image segmentation [1-4]. Over time, many sophisticated clustering techniques have been developed. However, when dealing with some complex datasets, these algorithms often encounter certain problems, including sensitivity to initial values, struggles with handling high-dimensional data, and requiring manual intervention to determine the number of clusters, etc.

To address the aforementioned limitations, Rodriguez and Laio [5] introduced a renowned clustering algorithm called DPC (clustering by fast search and find of density peaks). This algorithm not only rapidly identifies density peak points and assigns samples but also handles data of arbitrary shapes and large scales. Its simplicity and efficiency have garnered significant attention since its inception. In the

Manuscript received August 4, 2022; first revision October 14, 2022; second revision December 26, 2022; accepted January 8, 2023.
* **Corresponding Author:** Yongli Liu (yongli.buaa@gmail.com)
School of Computer Science and Technology, Henan Polytechnic University, Jiaozuo, Henan, China (yongli.buaa@gmail.com, 1812115222@qq.com, chaohao@hpu.edu.cn)

DPC approach, when calculating local density and relative distance, the similarity measurement relies solely on the geometric distance between objects. Consequently, when working on multi-center and high-dimensional datasets, selecting an appropriate similarity measurement based on the shape or size of the datasets is very crucial. Consequently, numerous enhanced concepts for local density calculation have emerged based on DPC [6]. Du et al. [7] incorporated the concept of k-nearest neighbor (KNN) to enhance the robustness and standardized the local density calculation by considering the local distribution of objects. Their algorithm mitigated the impact of cutoff distance dc on clustering results. Liu et al. [8] adopted the concept of nearest neighbor and proposed a new clustering algorithm. The calculation method of local density has been redefined, and relative distance and designing a two-step non central allocation scheme, this algorithm showed some advantages over multi-center and irregular datasets. Ding et al. [9] proposed a dissimilarity-measurement-optimization based density peak clustering (DPC) algorithm. This algorithm considers the distribution around nodes and re-measures the similarity of nodes with probability blocks, thereby changing the singleness of using geometric distance in calculating local density.

The preceding algorithms adopted the "whole-to-local" approach during calculating the local density, narrowing the selection range but incorporating richer information, thus escalating complexity. Additionally, they introduced the concept of KNN to improve the clustering accuracy, and ultimately relied on Euclidean distance. Evidently, the selection of similarity measurement is very crucial. However, there is a scarcity of criteria for selecting measurement methods, leading to subjectivity and potentially compromising clustering accuracy. Aiming to avoid this subjective error, Slonim and Tishby [10] introduced the information bottleneck (IB) theory into the agglomerative clustering, and proposed the AIB algorithm. After that, to expedite clustering, Slonim et al. [11] further proposed an IB-based sequential clustering algorithm, named SIB. The IB takes the information loss in grouping objects as the basis of similarity, omits the selection of similarity measures, and therefore avoids the subjectivity mentioned above. Liu and Wan [12] were inspired by the IB theory and proposed an IB based incremental fuzzy clustering method, which can deal with the clustering problem of high-dimensional and large-scale data. Hu et al. [13] designed an IB theory-based clustering algorithm, which shows good clustering performance for high-dimensional data. These algorithms measure the similarity between objects through mutual information loss, and all of the experimental results showed that clustering based on IB cannot only show higher clustering accuracy than traditional clustering algorithms [14], but also be suitable for processing high-dimensional objects.

To eliminate the arbitrariness in selecting similarity measures and enhance clustering performance on datasets with multiple centers and higher dimensions, we employed the IB theory as our similarity measure. Based on this idea, we proposed an IB-based density peaks clustering algorithm, named IBDPC. In summary, our work mainly includes the following aspects:

- IBDPC inherits the simplicity and efficiency of DPC.
- The incorporation of IB theory mitigates the subjectivity of similarity measure, thereby potentially enhancing clustering accuracy for high-dimensional and multi-center datasets.
- The experiments on ten datasets verify the effectiveness of IBDPC.

The rest of our work in this paper is arranged as follows. In Section 2, we introduced knowledge about DPC and IB theory. In Section 3, our algorithm IBDPC is described in detail. Section 4 includes the contents of our experiments. In Section 5, a brief summary of the paper's work is given.

2. Related Work

2.1 DPC

The DPC algorithm operates under two fundamental presumptions: (1) objects with high local density are candidates for cluster centers, and their nearest neighbors have relatively low local density and (2) the separation between cluster centers is considerable. By invoking the previous assumptions, the DPC algorithm mainly performs clustering through the following steps:

1) Constructing a decision diagram, and then finding the clustering centers, where the abscissa is the value of local density (ρ) of each sample, and the ordinate is the minimum distance (δ) from one sample to other samples with greater density;

2) Assigning the membership of the remaining points by rendering.

In DPC, the following formula is used to get the local density ρ_i for each sample point x_i:

$$\rho_i = \sum_j X(d_{ij} - d_c), X(x) = \begin{cases} 1, x < 0 \\ 0, x \geq 0 \end{cases}, \tag{1}$$

where d_{ij} is a variable measuring the distance between the object x_j and x_i, and d_c is a distance threshold. This calculation formula obtains the number of objects whose distance to the object x_i is less than the threshold d_c, that is, the local density value of the object x_i. The density value increases as there are more objects surrounding the object x_i.

For small datasets, local density ρ_i can be calculated by employing the Gaussian function as follows:

$$\rho_i = \sum_j exp(-\frac{d_{ij}^2}{d_c^2}). \tag{2}$$

The δ_i value of object x_i is generally calculated as the minimum distance from this object to all other objects whose densities are greater. And it is calculated as:

$$\delta_i = \min_{j:\rho_j > \rho_i} (d(x_i, x_j)). \tag{3}$$

The δ_i of the object with the highest density in the dataset is the maximum distance from the object to other objects in the dataset, which is defined as:

$$\delta_i = \max_j (d(x_i, x_j)). \tag{4}$$

After calculating the values of ρ_i and δ_i, and constructing the decision diagram, the objects with large density value and relative distance value will be chosen as clustering centers, which are usually located in the upper right corner of the decision diagram. Initially, to locate the cluster center and assign cluster labels, we sort the objects in descending order with the local density values. We then assign the remaining objects to the cluster where the nearest neighbor object is located, and obtain the final clustering results.

2.2 Information Bottleneck

IB theory, a technique rooted in information theory, finds its origins in Shannon's rate distortion theory. Introduced by Tishby et al. [14] in 1999, this theory examines the pertinent information of Y contained in source X, and defines the distortion through the mutual information between them. If the value of

mutual information is greater, the distortion will be smaller. Therefore, one particular application of rate distortion theory is the information bottleneck theory.

The integration of IB theory into clustering algorithms has gained traction in recent years. Within this framework, clustering results can be regarded as a representation of the dataset to be processed, and the clustering process tries to make the clustering results compress the information of the original data as much as possible, and retain its relevant feature information as much as possible. Therefore, clustering is regarded as the bottleneck between data and feature information. By invoking the principles of IB, an iterative process is formed to gradually maintain a balance between the compressed representation of data and the preservation of related information.

When compared to traditional clustering algorithms, the strength of IB lies in its ability to circumvent the definition of a distance function, instead leveraging information theory within the clustering process. Consequently, it can avoid the subjective error derived from the random selection of measurement methods. In IB theory, according to the shared likelihood distribution between the objects and the feature information, the information loss generated in the clustering process is used as the measurement standard for clustering, and the objects and related feature information are mainly considered.

3. Information Bottleneck based Density Peak Clustering

Based on IB theory, we propose the IBDPC approach. When calculating the distance from and density of the area, IBDPC introduces the IB theory. Due to the good performance of the IB theory in high-dimensional data clustering, it aids IBDPC algorithm in achieving higher accuracy when tackling complex datasets.

The IB addresses measurement distortion by introducing a related variable Y that characterizes the nature of X. Given random variable X and correlated variable Y adhering to the joint distribution of probability P(X,Y), we aim to make X as concise as possible under the condition of trying to preserve the maximum amount of information about Y. A higher compression ratio T for X results in lower mutual information I(T,Y). In addition, to ensure less mutual information loss during the compression process, the amount of compression will be a "bottleneck" of mutual loss.

Considering the sample space and feature space (represented by X and Y, respectively) for classification, the core principle of IBDPC algorithm is to identify a classification approach that incurs the minimal loss of mutual information between samples and features. Mutual information can be used to calculate the amount of information about another random variable contained in one random variable. Assuming that X and Y represent input random variables and output random variables in the information processing system respectively, let $p(x, y)$ represent the joint probability distribution, the edge probabilities be $p(x)$ and $p(y)$ respectively, and then the mutual information [10] can be measured as:

$$I(x; y) = \sum_x \sum_y p(x, y)\, log \frac{p(x,y)}{p(x)p(y)}. \tag{5}$$

In IBDPC, local density ρ_i is defined as:

$$\rho_i = \sum_j exp(-\frac{I_{ij}^2}{d_c^2}). \tag{6}$$

Let c_x and c_y represent two clusters, and I_{ij} stand for the loss of information experienced by both x_i and x_j. The value of I_{ij} can be calculated based on information theory using Eq. (7).

$$I_{ij} = \frac{|x_i|}{N} x_i \, log \frac{x_i}{t} + \frac{|x_j|}{N} x_j \, log \frac{x_j}{t} \tag{7}$$

where $t = \frac{|x_i| * x_i}{|x_i \cup x_j|} + \frac{|x_j| * x_j}{|x_i \cup x_j|}$, and $|\cdot|$ is a function for calculating the number of points in the cluster.

This formula can be used to determine the value of δ_i:

$$\delta_i = \min_{j:\rho_j > \rho_i} (I(x_i, x_j)). \tag{8}$$

The value can be defined as follows for the object with the highest density:

$$\delta_i = \max_j (I(x_i, x_j)). \tag{9}$$

Our IBDPC utilizes the aforementioned information to construct a decision diagram, effectively segregating objects and yielding the final clustering results.

Below are the step-by-step proceedings of our IBDPC algorithm:

Step 1: Compute the mutual information loss among pair-wise objects, where the I_{ij} distance is calculated according to Eq. (3).

Step 2: Based on the predefined cutoff distance, determine the local density ρ_i of item x_i.

Step 3: For object x_i, calculate the value of δ_i.

Step 4: Construct the decision graph with the values of ρ_i and δ_i.

Step 5: Identify the objects in the decision diagram's rightmost area as center points of the cluster.

Step 6: Allocate of remaining points. Arrange the remaining points in descending order based on their densities, and then arrange each object into a cluster that contains its closest neighbor with the highest density.

4. Experiments

To assess the clustering effectiveness of IBDPC, we selected ten datasets for empirical evaluation, with detailed information presented in Tables 1 and 2. In our experiment, we compared our IBDPC with DPC, DPC-KNN, and SNN-DPC (shared-nearest-neighbor-based clustering by fast search and find of density peaks) in terms of clustering accuracy. Specifically, the DPC-KNN algorithm enhances the local density computation approach by integrating the KNN concept into DPC. The quick density peak of the shared nearest neighbor serves as the foundation for the SNN-DPC algorithm. These two algorithms utilize the KNN idea to change the local density calculation method of DPC algorithm, however, easily affected by the k-value in KNN.

In our study, we employed three evaluation metrics to gauge the quality of our clustering results: adjusted mutual information (AMI), adjusted rand index (ARI) and Fowlkes-Mallows index (FMI). Notably, the maximum value for all these three metrics is 1, indicating optical clustering performance. Based on the mutual information between the artificially manufactured cluster vector and the actual cluster vector, AMI calculates similarity. The range of values for ARI, which is used to assess the

consistency of the distribution of the two data groups, is [-1, 1]. The FMI metric measures how well the clustering solution and the dataset's actual classification label match each other.

Table 1. Synthetic datasets

Dataset	#Clusters	#Samples	#Features	Description
Flame	2	240	2	This dataset has two clusters with different sizes and shapes.
Aggregation	7	788	2	This dataset has seven clusters with different shapes.
R15	15	600	2	This dataset has 15 two-dimensional Gaussian distribution clusters.
2circles	2	1,000	2	Multi-center dataset, 2 rings
Fourlines	4	512	2	Multi-center dataset, 4 straight lines

Table 2. Real-world datasets

Dataset	#Clusters	#Samples	#Features	Description
Iris	3	150	4	The dataset contains three categories, each representing an iris plant.
Wine	3	178	13	Wine component content
Glass	6	214	10	6 kinds of glass defined by oxide content
20NewsGroups	10	200	500	20newsgroups corpus, which uses 500 features to describe the information of data samples.
Webdata	10	314	2,000	UW-CAN corpus contains 10 categories.

UW-CAN is a collection of 314 web documents manually collected and labeled from the various University of Waterloo and Canadian websites.

4.1 Results from Experiments using Synthetic Datasets

The first part of the experiments was conducted on five synthetic datasets, including two typical multi-center datasets, 2circles and Fourlines. The results of these five two-dimensional datasets are presented in a visual format. Clustering results on these five artificial datasets are shown as Figs. 1–5 respectively, where different colors represent different clusters.

The analysis of clustering results employing several techniques on the Flame dataset is shown in Fig. 1. These experimental findings demonstrated that all algorithms can accurately cluster the dataset.

The clustering outcomes for the Aggregation dataset are displayed in Fig. 2. Each algorithm functions effectively, can precisely pinpoint the cluster center, and allocate remaining sample points reasonably, which shows that each algorithm performs well on this dataset.

The clustering performance of these four techniques on the R15 dataset is shown in Fig. 3. The clustering results of every algorithm are precise, with the cluster center being correctly identified and the remaining sample points allocated successfully.

The clustering results on 2circles dataset are illustrated as Fig. 4. This dataset is a typical multi-center dataset, featuring multiple density peak points within a class has, which are eligible as centers. Clustering results on this dataset show that the clustering centers obtained by each algorithm are different. The clustering centers of the two rings are distributed on different rings. On the whole, the distribution of clustering centers is correct, but only IBDPC and SNN-DPC algorithms can achieve perfect clustering results. For DPC and DPC-KNN algorithms, although the centers generated are correct, both of them

calculate local density based on the Euclidean distance for each object, and then adopt the DPC sample allocation strategy. Once the cluster allocation of sample points with large density is wrong, it may lead to misallocation of nearby objects.

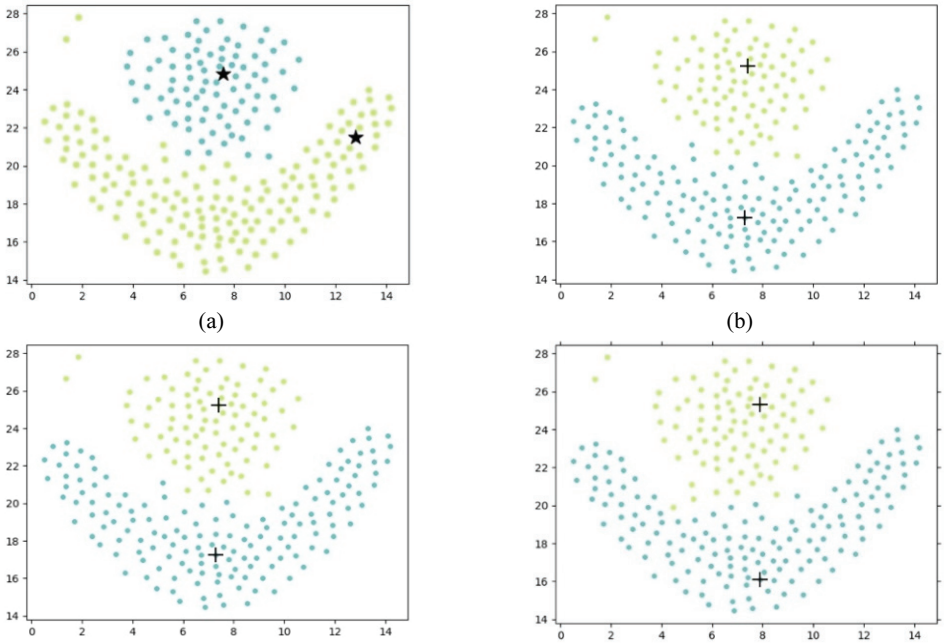


Fig. 1. Clustering results on Flame dataset: (a) IBDPC, (b) DPC, (c) DPC-KNN, and (d) SNN-DPC.

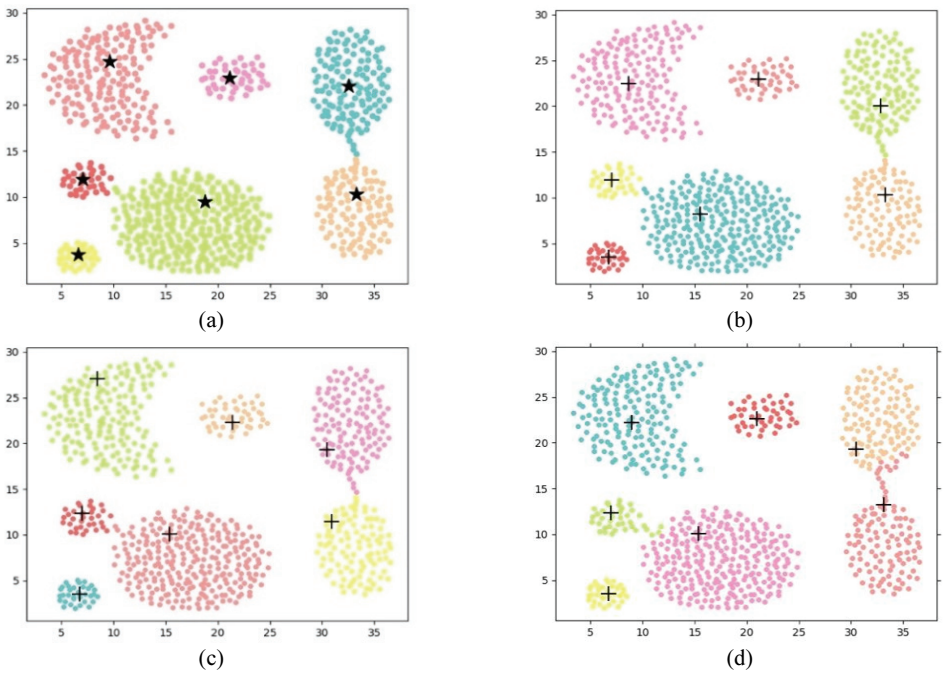


Fig. 2. Clustering results on Aggregation dataset: (a) IBDPC, (b) DPC, (c) DPC-KNN, and (d) SNN-DPC.

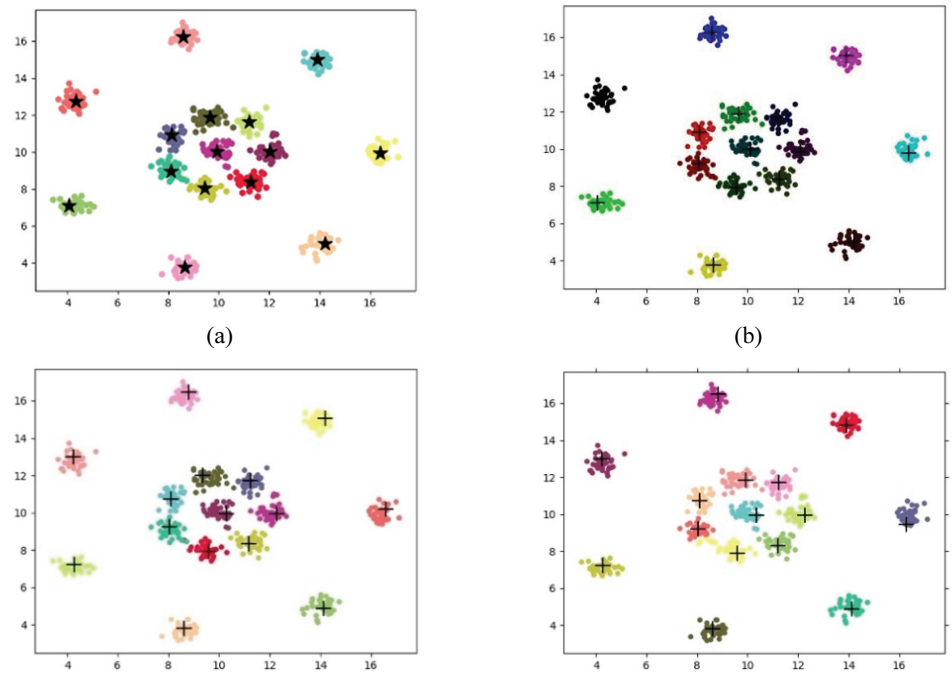


Fig. 3. Clustering results on R15 dataset: (a) IBDPC, (b) DPC, (c) DPC-KNN, and (d) SNN-DPC.

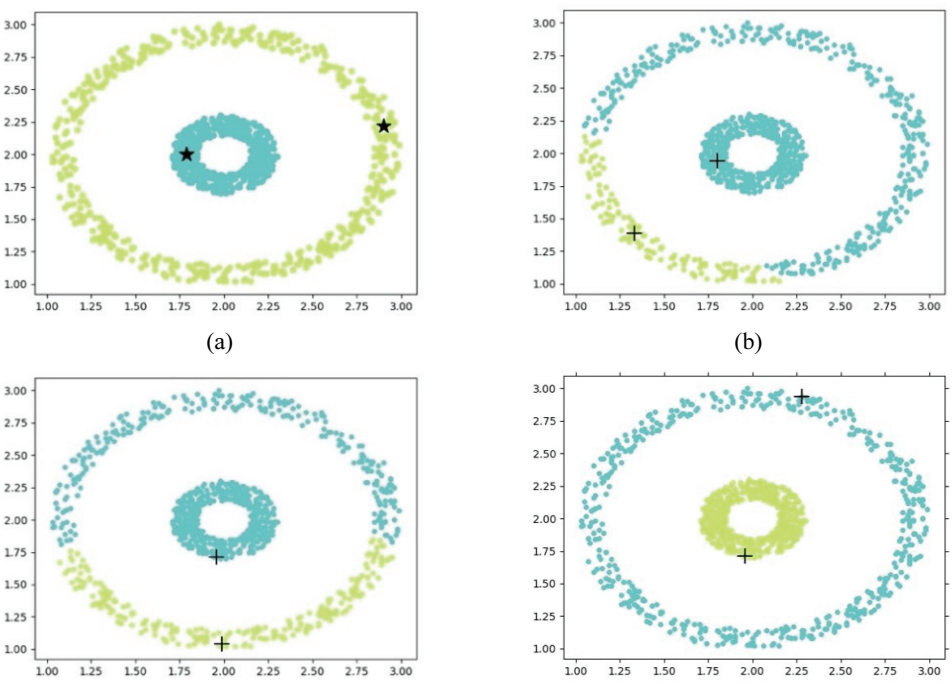


Fig. 4. Clustering results on 2circles dataset: (a) IBDPC, (b) DPC, (c) DPC-KNN, and (d) SNN-DPC.

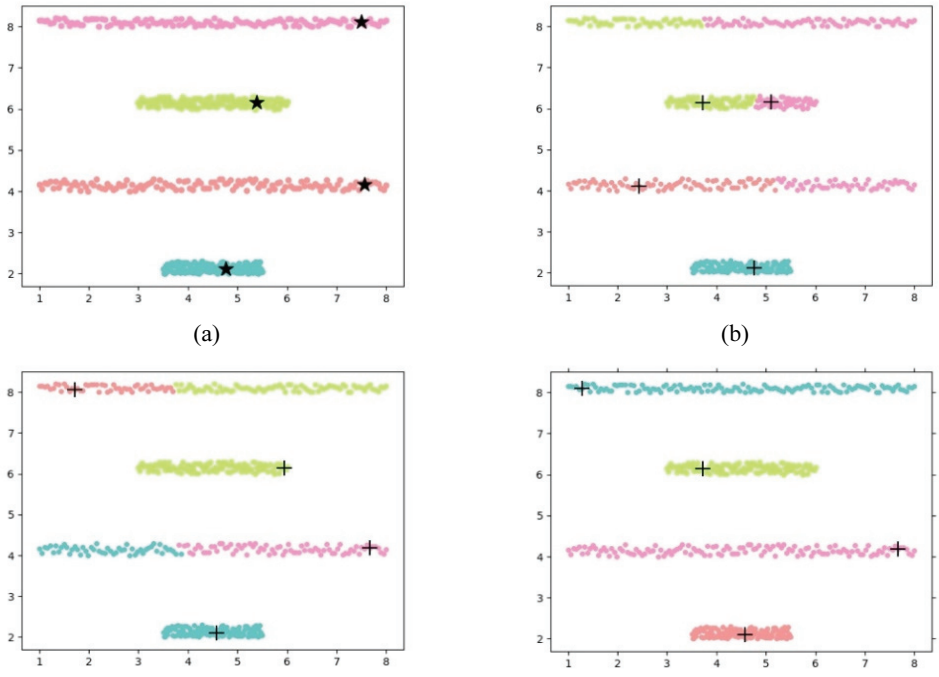


Fig. 5. Clustering results on Fourlines dataset: (a) IBDPC, (b) DPC, (c) DPC-KNN, and (d) SNN-DPC.

Fig. 5 shows the clustering results on the multi-center Fourlines dataset. In this group of clustering results, only IBDPC and SNN-DPC algorithms yield accurate results. Different from results on the multi-center 2circles dataset, DPC algorithm fails to generate the correct cluster centers, leading to subsequent errors in sample point allocation. The DPC-KNN algorithm obtains four correct cluster centers. However, due to the calculation method of similarity measurement, the high-density sample points may be wrongly assigned to a cluster center, resulting in the error of subsequent nearest neighbor sample points. And therefore, it also failed to produce the correct results.

In conclusion, all algorithms in our experiments can achieve relatively satisfactory results on the three datasets with uniform density distribution: Flame, Aggregation, and R15. However, on two multi-center datasets, 2circles and Fourlines, only results generated by IBDPC and SNN-DPC algorithms are satisfactory, while the outcomes of the other two algorithms are less than ideal. Further analysis showed that, although DPC-KNN introduces the KNN idea to unify the calculation of local density rather than the parameter d_c, it still uses an elementary geometric distance measurement to compute the value of local density, without the surrounding environment of data distribution. IBDPC algorithm retains the core idea of DPC, but measures dissimilarity using the IB theory, making it effective for processing multi-center datasets.

4.2 Results from Experiments using Real-World Data

In the second section, we compare the accuracy of these four algorithms using five real-world datasets, including high-dimensional datasets and traditional small sample datasets. The outcomes are shown in Table 3–5.

Table 3. Comparison in terms of AMI synthetic datasets

Dataset	DPC	DPC-KNN	SNNDPC	IBDPC
Iris	0.8032	0.8032	0.8203	**0.8689**
Wine	0.4131	0.3965	0.4169	**0.4318**
Glass	0.6075	**0.6091**	0.5945	0.5361
20NewsGroups	0.2904	0.0018	0.3309	**0.3398**
Webdata	0.7490	0.1955	0.6735	**0.7902**

The ideal outcomes are highlighted in bold.

Table 4. Comparison in terms of ARI

Dataset	DPC	DPC-KNN	SNNDPC	IBDPC
Iris	0.7592	0.7592	0.8186	**0.8858**
Wine	0.3715	0.2926	0.3981	**0.4554**
Glass	**0.3977**	0.3811	0.3863	0.3851
20NewsGroups	0.1181	0.0009	0.1828	**0.1877**
Webdata	0.5172	0.0399	0.3573	**0.6153**

The ideal outcomes are highlighted in bold.

Table 5. Comparison in terms of FMI

Dataset	DPC	DPC-KNN	SNNDPC	IBDPC
Iris	0.8407	0.8407	0.879	**0.9234**
Wine	0.5834	0.6192	0.5999	**0.6480**
Glass	0.5335	0.5233	0.5278	**0.5290**
20NewsGroups	0.2724	0.2963	0.3417	**0.3126**
Webdata	0.6102	0.3492	0.5127	**0.6747**

The ideal outcomes are highlighted in bold.

In Tables 3–5, on the two classical datasets, Iris and Wine, IBDPC algorithm achieved higher clustering accuracy than DPC-KNN and SNN-DPC algorithms who both use KNN idea to calculate local density, which shows that IBDPC algorithm can achieve better clustering results after using the IB measure instead of traditional Euclidean distance. On the Glass dataset, while the AMI and ARI metrics for the IBDPC algorithm are marginally lower than those of the competing algorithms, its FMI index is comparatively strong, which indicates that the clustering performances of each algorithm are relatively similar. The IBDPC algorithm performs best in the three clustering indexes on the datasets 20NewsGroups and Webdata with higher dimensions. Especially on the Webdata dataset, although the dimension is the highest in our experiments, the clustering accuracy is improved significantly. The value of AMI is 0.7902, ARI is 0.6153, and FMI is 0.6747, which are 5.5%, 18.99%, and 10.57% higher than that of DPC algorithm.

On the classic real-world datasets, the performance of each algorithm is relatively close. On the two datasets, 20NewsGroups and Webdata, with relatively high dimensions, the clustering accuracy of IBDPC clustering algorithm is greatly improved. In summary, IBDPC algorithm can handle datasets with diverse internal architectures, and its overall performance is commendable.

Based on the clustering results of above two parts of experiments, both the original DPC algorithm and the DPC-KNN algorithm successfully finds correct cluster centers on multi-center datasets such as 2circles. By analyzing the clustering strategy of these two algorithms, we know that:

1) While DPC employs the cutoff distance approach to identify the local density of sample points, DPC-KNN algorithm optimizes the method by using the idea of KNN to find the local density of sample points using knowledge about their surroundings.
2) These two algorithms are consistent during allocating subsequent sample points, that is, they find the nearest neighbors for allocation according to the local density of the sample points from high to low.
3) These two approaches are comparable in terms of how local density and relative distance are calculated.

While DPC-KNN algorithm calculates local density and relative distance in the local range of sample points, both of which are determined by Euclidean distance, DPC algorithm is based on overall local density and relative distance. For example, on the 2circles dataset, even though DPC and DPC-KNN can find the correct cluster centers through the decision graph, they still cannot generate satisfactory clustering results in the subsequent allocation. To improve on this problem, we calculate the local density information of the sample points using the IB theory, and we apply mutual information loss to determine how similar the sample points are. We unified the measurement method, and verified the feasibility of our algorithm through experiments, and also performed well on high-dimensional real datasets. In light of this comprehensive analysis, it is evident that IBDPC algorithm has the ability to process datasets with different internal structures, delivering relatively satisfactory results overall.

Table 6. Sample t-test on Iris

	Levene's test for equality of variances		t-test for equality of means						
	F	Sig.	t	df	Sig. (2-tailed)	Mean difference	SE difference	95% CI difference	
								Lower	Upper
(0 and 1)									
Equal variances assumed	18.474	0	-29.365	102	0.000	-1.719	0.0585	-1.835	-1.603
Equal variances not assumed	-	-	-30.003	82.452	0.000	-1.719	0.0573	-1.833	-1.605
(0 and 2)									
Equal variances assumed	12.136	0.001	-18.558	94	0.000	-1.017	0.0548	-1.126	-0.908
Equal variances not assumed	-	-	-18.195	72.484	0.000	-1.017	0.0559	-1.126	-0.906
(1 and 2)									
Equal variances assumed	0.719	0.398	-9.987	98	0.000	-0.702	0.0703	-0.841	-0.562
Equal variances not assumed	-	-	-10.078	97.763	0.000	-0.702	0.0696	-0.840	-0.564

Table 7. Sample t-test on Flame

	Levene's test for equality of variances		t-test for equality of means						
	F	Sig.	t	df	Sig. (2-tailed)	Mean difference	SE difference	95% CI difference	
								Lower	Upper
(0 and 1)									
Equal variances assumed	7.325	0.007	-6.231	238	0.000	-2.042	0.3277	-2.687	-1.396
Equal variances not assumed	-	-	-6.905	108.799	0.000	-2.042	0.2957	-2.628	-1.456

4.3 T-Test Analysis

For the t-test analysis, we focused on the clustering results obtained by IBDPC on two datasets, Flame and Iris. On the Iris dataset, which has three clusters, the clustering accuracy is examined by t-test, as shown in Table 6. The F-statistic value for this test is 18.474, with a corresponding P-value of 0.000. Since there is a clearly difference between the two population variances, we direct our attention to the second row of t-test results. The T-statistic value is -30.003, with a matching 2-tailed P-value of 0.000, below the 0.05 threshold, indicating that there is a significant difference between Group 0 and Group 1. Likewise, Group 0 and Group 2, as well as Group 1 and Group 2, exhibit substantial differences, respectively.

Table 7 illustrates the t-test results on the Flame dataset, which is comprised of two clusters. In this instance, the F-statistic value is 7.325, with a corresponding P-value of 0.007 and a variance significance below 0.05, which indicates that the homogeneity of variance is not met. In the final row of the t-test results in Table 7, the 2-tailed probability P-value is 0.000, below the 0.05 threshold, and the T-statistic value is -6.905. These findings underscore a considerable difference between Group 0 and Group 1.

This section, based on our t-test analysis, we can confirm the superior clustering quality of IBDPC.

5. Conclusion

In this research, we have refined the DPC algorithm, leading to the introduction of IBDPC, an algorithm tailored for density peak clustering that leverages IB theory. IBDPC retains such advantages of DPC algorithm as quickly finding density peak points and efficiently allocating samples. Meanwhile, it introduces IB theory to calculate the similarity between samples, redefining the local density calculation, and unify the similarity measurement method. The IBDPC improves the clustering accuracy on complex datasets, especially on high-dimensional and multi-center ones. To evaluate the clustering quality of IBDPC, 10 datasets were selected for our comparative experiments. Experimental results showed that IBDPC algorithm could outperform the competitors and achieve satisfactory clustering results.

References

[1] D. Crowther, S. Kim, J. Lee, J. Lim, and S. Loewen, “Methodological synthesis of cluster analysis in second language research,” *Language Learning*, vol. 71, no. 1, pp. 99-130, 2021. https://doi.org/10.1111/lang.12428

[2] R. Cohn and E. Holm, “Unsupervised machine learning via transfer learning and k-means clustering to classify materials image data,” *Integrating Materials and Manufacturing Innovation*, vol. 10, no. 2, pp. 231-244, 2021. https://doi.org/10.1007/s40192-021-00205-8

[3] A. S. Ramos, C. H. Fontes, A. M. Ferreira, C. C. Baccili, K. N. da Silva, V. Gomes, and G. J. A. de Melo, “Somatic cell count in buffalo milk using fuzzy clustering and image processing techniques,” *Journal of Dairy Research*, vol. 88, no. 1, pp. 69-72, 2021. https://doi.org/10.1017/S0022029921000042

[4] P. Bhattacharjee and P. Mitra, “A survey of density based clustering algorithms,” *Frontiers of Computer Science*, vol. 15, article no. 151308, 2021. https://doi.org/10.1007/s11704-019-9059-3

[5] A. Rodriguez and A. Laio, “Clustering by fast search and find of density peaks,” *Science*, vol. 344, no. 6191, pp. 1492-1496, 2014. https://doi.org/10.1126/science.1242072

[6] Y. Wang, D. Wang, W. Pang, C. Miao, A. H. Tan, and Y. Zhou, "A systematic density-based clustering method using anchor points," *Neurocomputing*, vol. 400, pp. 352-370, 2020. https://doi.org/10.1016/j.neucom.2020.02.119

[7] M. Du, S. Ding, and H. Jia, "Study on density peaks clustering based on k-nearest neighbors and principal component analysis," *Knowledge-Based Systems*, vol. 99, pp. 135-145, 2016. https://doi.org/10.1016/j.knosys.2016.02.001

[8] R. Liu, H. Wang, and X. Yu, "Shared-nearest-neighbor-based clustering by fast search and find of density peaks," *Information Sciences*, vol. 450, pp. 200-226, 2018. https://doi.org/10.1016/j.ins.2018.03.031

[9] S. F. Ding, X. Xu, and Y. R. Wang, "Optimized density peaks clustering algorithm based on dissimilarity measure," *Journal of Software*, vol. 31, no. 11, pp. 3321-3333, 2020. https://doi.org/10.13328/j.cnki.jos.005813

[10] N. Slonim and N. Tishby, "Agglomerative information bottleneck," *Advances in Neural Information Processing Systems*, vol. 12, pp. 617-623, 1999.

[11] N. Slonim, N. Friedman, and N. Tishby, "Unsupervised document classification using sequential information maximization," in *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Tampere, Finland, 2002, pp. 129-136. https://doi.org/10.1145/564376.564401

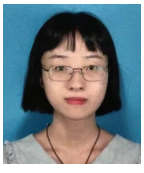
[12] Y. Liu and X. Wan, "Information bottleneck based incremental fuzzy clustering for large biomedical data," *Journal of Biomedical Informatics*, vol. 62, pp. 48-58, 2016. https://doi.org/10.1016/j.jbi.2016.05.009

[13] S. Hu, R. Wang, and Y. Ye, "Interactive information bottleneck for high-dimensional co-occurrence data clustering," *Applied Soft Computing*, vol. 111, article no. 107837, 2021. https://doi.org/10.1016/j.asoc.2021.107837

[14] N. Tishby, F. C. Pereira, and W. Bialek, "The information bottleneck method," in *Proceedings of the 37th Annual Allerton Conference on Communication, Control and Computing*, Monticello, IL, USA, 1999, pp. 368-377.

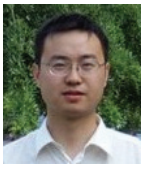
Yongli Liu **https://orcid.org/0000-0002-0540-865X**

He received his doctoral degree in engineering from Beihang University in 2010. He is currently a professor at Henan Polytechnic University. His main research fields are information retrieval, data mining and big data.



Congcong Zhao **https://orcid.org/0000-0001-6696-9795**

She is a postgraduate student in the school of computer science and technology of Henan Polytechnic University. Her main research interests are data mining and big data.



Hao Chao **https://orcid.org/0000-0001-6700-9446**

He obtained his Ph.D. degree from Institute of Automation, Chinese Academy of Sciences in 2012. He is currently an associate professor in Henan Polytechnic University. His current research fields include data mining and pattern recognition.