Hindawi
Complexity
Volume 2018, Article ID 2032461, 8 pages
https://doi.org/10.1155/2018/2032461

Research Article

A Novel Hierarchical Clustering Algorithm Based on Density Peaks for Complex Datasets

Rong Zhou,[1,2] **Yong Zhang**,[1] **Shengzhong Feng**,[1] **and Nurbol Luktarhan**[3]

[1]*Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China*
[2]*University of Chinese Academy of Sciences, China*
[3]*Information Science and Engineering College, Xinjiang University, Urumqi, China*

Correspondence should be addressed to Yong Zhang; zhangyong@siat.ac.cn

Received 13 March 2018; Accepted 5 June 2018; Published 18 July 2018

Academic Editor: Shyam Kamal

Clustering aims to differentiate objects from different groups (clusters) by similarities or distances between pairs of objects. Numerous clustering algorithms have been proposed to investigate what factors constitute a cluster and how to efficiently find them. The clustering by fast search and find of density peak algorithm is proposed to intuitively determine cluster centers and assign points to corresponding partitions for complex datasets. This method incorporates simple structure due to the noniterative logic and less few parameters; however, the guidelines for parameter selection and center determination are not explicit. To tackle these problems, we propose an improved hierarchical clustering method HCDP aiming to represent the complex structure of the dataset. A k-nearest neighbor strategy is integrated to compute the local density of each point, avoiding to select the nonnecessary global parameter d_c and enables cluster smoothing and condensing. In addition, a new clustering evaluation approach is also introduced to extract a "flat" and "optimal" partition solution from the structure by adaptively computing the clustering stability. The proposed approach is conducted on some applications with complex datasets, where the results demonstrate that the novel method outperforms its counterparts to a large extent.

1. Introduction

Clustering is a process of partitioning data objects into subsets. Each subset is a cluster whose objects are similar to each other while dissimilar to objects in other clusters. For decades, numerous clustering algorithms have been proposed and widely used in many fields, including business intelligence, image pattern recognition, web search, computational biology. The clustering by fast search and find of density peak (CDP) algorithm is proposed by Rodriguez and Laio in Science [1]. It is based on the assumption that the density peaks are candidates for cluster centers, and cluster centers are far away from each other. After determining the cluster centers, the remaining objects will be directly assigned to the nearest clustering center. Compared to most clustering algorithms, the CDP algorithm does not require to design an objective function for iterative optimization, and it can find clusters in spite of their shapes. However, the CDP algorithm has following shortcomings:

(i) There is no explicit criterion for the selection of the key parameter d_c, which is the threshold of scanning radius for density calculation, and it greatly affects the clustering results. The authors claimed that when the average density of data objects is 1 – 2% of the datasets, one can get good results, without giving an explicit method of determining the optimal d_c to achieve the best clustering effect.

(ii) Initial cluster centers are selected interactively but not automatically, whereas it is quite difficult to get a correct selection of some datasets.

To overcome the above issues, this paper proposes a new algorithm where

(1) k-nearest neighbors method is introduced to estimate local density so that the deficiencies of CDP in computing the local density of an object can be avoided;

(2) instead of directly clustering, a hierarchical clustering method is used to generate a complete clustering structure;

(3) the task of extracting a set of significant clusters is formulated as an optimization problem and a control algorithm that finds the globally optimal solution to this problem is proposed.

The rest of this paper is organized as follows. Related works are introduced in Section 2. Section 3 describes the proposed method in detail. In Section 4, experimental results are presented and discussed. Conclusions and future work are stated in Section 4.

2. Related Works

As a novel and efficient algorithm, the clustering using density peak algorithm is brought into sharp focus. However, there are still some shortcomings that cannot be ignored. In this section, we will first review CDP briefly and then introduce a hierarchical clustering method to represent the clustering structures of datasets.

2.1. Cluster Using Density Peaks. The clustering using density peak (CDP) algorithm is on the basis of the assumption that cluster centers are characterized by a higher density than their neighbors and by a relatively large distance from points with higher density. By calculating these two quantities of each data object, CDP builds a decision graph for users to pick up cluster centers and exclude outliers.

Formally speaking, let D denote a dataset of n objects, for each data object i, its local density ρ_i is defined by (1), and its nearest distance δ_i from points of higher density is defined by (2).

$$\rho_i \sum_j \chi(d_{ij} - d_c), \tag{1}$$

$$\delta_i = \min_{j:\rho_j > \rho_j} d_{ij}, \tag{2}$$

where d_{ij} is the Euclidean distance between points i and j, d_c is a cutoff distance specified by the users, and $\chi(t) = 1$ if $t < 0$ and otherwise $\chi(t) = 0$. For the point with the highest density, it takes $\delta_i = \max_j(d_{ij})$.

For small datasets, the algorithm turns to the exponential kernel for density calculation, as described as follows:

$$\rho_i \sum_j \exp\left(-\frac{d_{ij}^2}{d_c^2}\right). \tag{3}$$

The cluster centers are then recognized as points for which the values of both δ_i and ρ_i are anomalously large. After the cluster centers are determined, the algorithm assigns each remaining objects to the same cluster as its nearest neighbor of higher density.

The problem is there is no objective metric to decide whether the dataset is large or small, so users might face a dilemma in selecting methods for computing density indicators. Moreover, clustering results may vary greatly according to the selection of d_c. In [1], the authors suggested to choosing d_c so that the average number of neighbors is around 1-2% of the total number of points in the dataset. However, it is easy to find out that the suggested choice is not always applicable when the size of dataset changes.

Some subsequent researchers attempt to solve this problem. Mehmood et al. [2] introduced heat diffusion method [3] to estimate point density and used the time parameter of heat diffusion to efficiently create clusters. This method is similar to the kernel density estimation, and the bandwidth parameter is determined according to [3]. Chen and He [4] calculated the field intensity and distance of every data point and fit them by a regression analysis. The cluster centers are determined by a residual analysis. Reference [5] computed local density of each point using its k-nearest neighbors instead of d_c in a kernel density estimation. Reference [6] also adopted a new local density metric using k-nearest neighbors in a kernel density estimation too. Reference [7] analyzed the local density metrics using k-nearest neighbors and tried to discriminate points belonging to different clusters more accurately.

Besides considering a single cutoff distance d_c (or bandwidth), some studies also analyzed the case of multiple densities. In [8], a cover map procedure was applied iteratively with a decreasing locally adaptive window to build a multidimensional density map, which allows cluster center selection. Wang and Xu [9] proposed an adaptive peak detection with nonparametric multivariate kernel density estimation. The algorithm treats the dataset as a multivariate normal distribution, and the bandwidth matrix $H \in \mathbb{R}^{d\times d}$ of kernel density estimation is correlated to the dataset's dimensions d. By narrowing down the possible ranges of H and the number of clusters k, the optimal values are chosen from all possible combinations of H and k. Mehmood et al. [10] considered density regions instead of cluster centers. They used CDP to find local clusters and merge them using the concept of shared density regions.

The other flaw of the algorithm is the vague, dull, and unclear choice of cluster centers. Some works [4, 11, 12] tried to identify the number of clusters by finding out singular points from the indicator curves; the others focus on merging microclusters [8, 10, 13] or splitting clusters [14] until some stopping conditions are satisfied. In this paper, we extract the optimal result from the hierarchical clustering structure, which will be introduced in the following section.

2.2. Hierarchical Clustering. The cluster centers are interactively determined in [1] rather than using explicit criteria. Some existing works [4, 11, 12] determined the number of clusters k first, then they sorted the data points based on

both indicators ρ_i and δ_i in descending order and chose the first k-largest points as cluster centers. Instead of directly calculating k, hierarchical methods try to depict the clustering structure of the dataset. For an easier interpretation of the structure, some automatic techniques are provided to extract a "flat" solution.

Hierarchical clustering (HC) methods represent data objects in a hierarchy or "tree" structure of clusters [15]. They are effective in detecting true clustering structures of datasets. Many works of HC have been published in recent years. They can be categorized into two classes: one uses distance-based methods and the other adopts density-based methods.

For distance-based HC methods, the core idea is to agglomerate or divide clusters according to the distance between two clusters, where each cluster contains a set of data points. The most common measure of cluster distance calculates the closest pair of points belonging to different clusters. It can be regarded as a nearest-neighbor clustering algorithm. Moreover, if there is a threshold to terminate the clustering process, it is called single-link method [16]. The merging process repeats until all the points eventually form one cluster. Similar approaches such as average-link or complete-link are also widely used [17]. If we take data objects as nodes of a graph, with distance-weighting edges, the clustering algorithm that uses the minimum distance measure is called a minimal spanning tree (MST) algorithm [15, 18]. Usually, a single, global threshold cutting through the hierarchical cluster representation can give a "flat" partition of the data, which users are most interested in.

To the contrast, density-based HC methods received less attention. The main idea of these methods is to investigate the reachable distance of all data points and form hierarchical structures using the concept of "density connectivity." The most famous OPTICS algorithm [19] is able to represent a density-based clustering structure of the dataset. Although a postprocessing procedure to extract a simplified clustering result was proposed, the procedure did not become as popular as OPTICS itself since it has heavy reliance to the reachability plot and is sensitive to the choice of a critical parameter that cannot be determined easily. Gupta et al. [20] proposed a density-shaving strategy applying to the hierarchical structure referring to the work [21] to achieve cluster extraction. Campello et al. [22] presented HDBSCAN as an improvement over OPTICS. By defining cluster stability, they turned the cluster extraction problem to an optimization problem of maximizing the overall stability of the set of clusters extracted from the HDBSCAN hierarchy.

3. The Proposed Method

In this section, we describe the proposed method in detail. Generally speaking, the proposed method consists of three steps: local density calculation, hierarchy representation, and optimal cluster extraction. Local density calculation is conducted based on the k-nearest neighbors. Hierarchical clustering method is then applied to depict the cluster structure. Finally, by introducing a concept of cluster stability, we propose an algorithm to solve the extraction problem from a cluster hierarchy.

3.1. Local Density Estimation Using k-Nearest Neighbors. Most of the extended works for CDP attempt to improve density estimation from the perspective of kernel method assuming that the data points are in the same or different Gaussian distribution(s). These works did not solve the problem of choosing d_c, on the contrast, they just turn the problem of selecting a suitable d_c into the problem of determining a suitable bandwidth of the Gaussian distribution kernel. Unlike these works, we incline to assess an object's local density using the information of its neighbors. k-nearest neighbor (kNN) has been shown to be a powerful technique for density estimation [23], clustering [24–26]. The goal of this approach is to find k-nearest neighbors (kNN) of each data object in the dataset. By using this approach, no assumptions on the distribution of the dataset are required, which means that the dataset can have arbitrary shapes and different density peaks.

To determine the density of a data object, we consider k-nearest neighbor distance of the object. We call this k-nearest density and it is formally defined as follows:

Definition 1 (k-nearest density). Given a dataset D and $\mathrm{dist}(\cdot)$ be a distance function on points in D, for $k \in \mathbb{R}$, point i's k-nearest density is defined as

$$\rho_i = \max_{j \in kNN(i)} \mathrm{dist}(i, j), \tag{4}$$

where $kNN(i) = \{j \in D \mid \mathrm{dist}(i, j) \leq \mathrm{dist}(i, NN_k(i))\}$, and $NN_k(i)$ is the kth nearest point to i according to $\mathrm{dist}(\cdot)$.

In general, the distance $\mathrm{dist}(\cdot)$ can be any distance measured. The most common choice is standard Euclidean distance. The most naive implementation of kNN search involves the brute-force computation of distances between all pairs of data points in the dataset, which scales as $O(n^2)$, where n is the size of the dataset. In order to avoid the computation inefficiencies, spatial indexing structures such as KD tree [27] and ball tree [28] can be applied, leading to $O(n \log n)$ computation cost.

3.2. Hierarchical Clustering Based on Density Peaks. In this section, we propose to construct a hierarchy structure to represent the original dataset. The cluster hierarchy based on density peaks enables to represent the fact that each level corresponds to objects' distance from their nearest neighbor with higher density.

First of all, for each data object i in the dataset, we compute its local density ρ_i according to (4), and its distance δ_i from points with higher density, which is shown below.

$$\delta_i = \min_{j:\rho_j > \rho_j} \mathrm{dist}(i, j). \tag{5}$$

Then, we treat the data objects as nodes in a graph. For each node i (except the node of maximal density), we connect

it with its nearest neighbor of higher density by an edge weight δ_i. It is obvious that we finally get a tree, whose root is the point with maximal density, and each other node's density is lower than its ancestors and higher than its descendants. Comparing to the MST method, this tree efficiently integrates information on not only distance but also density.

We can sort the edges and iteratively remove them from the tree in decreasing order of weights. After each edge cutting, the tree might be split, shrunk, or even disappeared, as defined below:

Definition 2 (tree split, shrink, and disappear). For a tree T, remove edge(s) with weight w.

(1) If the number of children in the subtree is less than a given threshold, all child nodes in the subtree will be regarded as "noise" and this subtree disappears.

(2) If there is only one subtree with "nonnoise" nodes, we say that T is shrunk.

(3) If there are more than one subtrees with "nonnoise" nodes, we say that T is split.

Algorithm 1 shows the main steps of our HCDP algorithm, which requires 2 input parameters k and th. It produces a clustering tree that contains all partitions obtainable by CDP in a nested way. This "HCDP hierarchy" can be implemented in $O(n \log n)$ time. Applying this algorithm transforms the clustering problem into a subtree partition problem.

3.3. Clustering Extraction and Evaluation. As mentioned above, the "HCDP hierarchy" can present the cluster structure of the dataset; however, interpreting the structure into a more understandable result, that is, extracting a "suitable" partition from the hierarchy to demonstrate the "focal" clusters, remains a problem. HCDP contains all possible CDP solutions with respect to given parameters of k and th. When decreasing the value of edge weight w, more and more edges are cut, and clusters get split or shrunk until they disappear. Obviously, the significant clusters "last" longer than the insignificant ones. To capture these significant clusters, we hope to evaluate the quality of the generated clusters instead of simply providing a single, global threshold.

For the sake of simplicity, we consider a modified version of cluster stability from [22]. It is based on Hartigan's model [29] and try to construct a tree of nested clusters by varying the threshold of density level. The power of Hartigan's model is mainly relying on the following aspects:

(1) It allows the concept of noise to be modeled as those objects lying in sparse regions of the data space.

(2) It allows clusters of varied shapes to be modeled as the connected components of the density level sets. Such components are not restricted to the region of a single density peak; they can possibly represent the union of multiple density peaks.

(3) It allows one to model the presence of nested clusters of varied densities in data, through the hierarchical relationships described by the density-contour tree.

Though the original definition is applied to density-based clustering algorithms such as DBSCAN, there is similar property between clustering by density peaks and clustering by density, that is, they both search dense regions based on density connectivity. So the original definition can also be employed after modification in clustering algorithms based on density peaks.

Given a cluster C_i, we define its stability as

$$S(C_i) = \sum_{x_j \in C_i} \left(\frac{1}{\delta_{\text{exclude}}(x_j, C_i)} - \frac{1}{\delta_{\text{emerge}}(C_i)} \right), \quad (6)$$

where δ_{exclude} is the maximal weight of which removing edges excludes point x_j from the cluster, and δ_{emerge} is the maximal weight by removing edges of which cluster C_i emerges (gets separated from the previous cluster).

Let $\{C_1, C_2, \ldots, C_\kappa\}$ be the collection of nonoverlapping clusters extracted from the hierarchy, and let $S(C_i)$ denote the stability value of each cluster. We can treat the extraction problem as an optimization problem with the objective of maximizing the sum of stabilities of the clusters:

$$\max \ S = \sum_{i=1}^{\kappa} S(C_i). \quad (7)$$

To solve (7), we start from the edge of the highest weight. Every time we cut an edge from the tree, we determine the current number of clusters and calculate their stability. Algorithm 2 gives the pseudocode for finding the optimal solution to (7). Here, we use k for both neighborhood calculation and "noise" threshold as a classic smoothing factor whose effect can be well understood referencing [19, 20, 30, 31].

4. Experiments, Results, and Discussion

In this section, we conduct experiments to assess the effectiveness of the proposed method. To demonstrate that HCDP is effective to clusters with both convex and nonconvex shapes, we benchmarked the algorithm on some 2-dimensional datasets for easy visualization. Artificial datasets 1, 2, and 3 are selected. Dataset 1 is from [32], which consists of clusters with both convex and nonconvex shapes in a hierarchical structure; dataset 2 [33] consists of 3 nonconvex shape clusters; dataset 3 is a synthetic dataset consisting of 2 isotropic Gaussian blobs and 2 interleaving half-circles. By choosing an appropriate $k \in [3, 7]$, the intuitive visualization of hierarchical structure and clustering results is shown in Figure 1.

Require: dataset D, neighborhood k, child threshold th.
1: Compute k-nearest density ρ_i and distance δ_i for each point $x_i \in D$
2: Generate a tree T by connecting from one point to its nearest point with higher density, and assign the whole tree as a single cluster.
3: Sort all the edges of the tree with respect to the weights in descending order.
4: **repeat**
5: Remove the highest edge(s) in T (in case of same weights, edges must be cut simultaneously) to get subtrees $T_{sub} = \{t_1, \dots, t_\kappa\}, \kappa \geq 1$
6: **for** $t_i \in T_{sub}$ **do**
7: **if** children of $t_i < th$ **then**
8: All children nodes in this subtree are assigned as "noise".
9: **else**
10: assign a new cluster to subtree t_i
11: **end if**
12: **end for**
13: **until** some stopping condition is satisfied.

Algorithm 1: HCDP main steps

Require: dataset D, parameter k
1: Generate a tree T of D using Algorithm 1 and assign the whole tree as a single cluster.
2: Sort all the edges of the tree with respect to the weights in descending order E_s.
3: $T_{prev} = \{T\}$
4: $S(T) = 0.0$
5: **for** $e \in E_s$ do
6: $T_{next} = \{\}$,
7: Remove e and split previous subtrees $T_{prev} = \{t_1, t_2 \dots\}$ into new subtrees $\{t_{11}, t_{12}, \dots, t_{21}, t_{22}, \dots\}$, where t_{ij} denotes subtrees split from previous tree t_i
8: **for** $t_i \in T_{prev}$ do
9: **if then** $S(t_i) < S(t_{i1}) + S(t_{i2}) + \cdots$
10: $T_{next} \leftarrow \{t_{i1}, t_{i2}, \dots\}$
11: **else**
12: $T_{next} \leftarrow \{t_i\}$
13: **end if**
14: **end for**
15: $T_{prev} = T_{next}$
16: **end for**

Algorithm 2: HCDP extraction

We also considered datasets from the UCI Machine Learning Repository [34–37] and compare HCDP with DBSCAN and CDP algorithms. For the clustering results, we evaluated them using normalized mutual information (NMI) score, which can be information theoretically interpreted. It is defined as below:

$$\mathrm{NMI}(\mathbb{C}, \mathbb{L}) = 2 \times \frac{I(\mathbb{C}, \mathbb{L})}{H(\mathbb{C}) + H(\mathbb{L})}, \tag{8}$$

where $\mathbb{C} = \{C_1, C_2, \dots, C_\kappa\}$ is the set of clusters and $\mathbb{L} = \{L_1, L_2, \dots, L_\iota\}$ is the set of known labels.

I is mutual information:

$$I(\mathbb{C}, \mathbb{L}) = \sum_i \sum_j P(C_i \cap L_j) \log \frac{P(C_i \cap L_j)}{P(C_i) P(L_j)}, \tag{9}$$

where $P(C_i)$, $P(L_j)$, and $P(C_i \cap L_j)$ are the probabilities of an object being in cluster C_i, labeled L_j, and in the intersection of C_i and L_j.

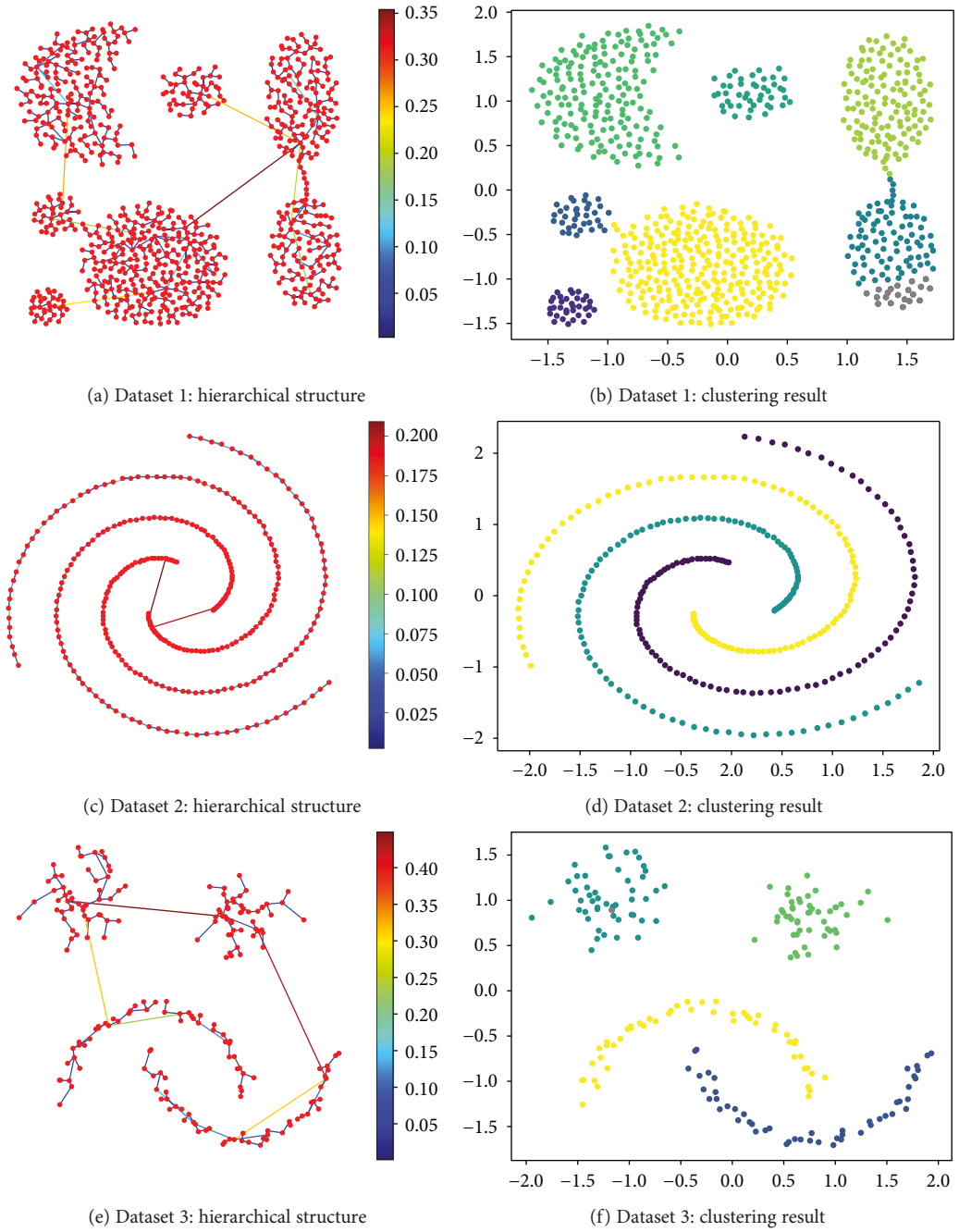
H is entropy defined as

$$H(\mathbb{C}) = -\sum_i P(C_i) \log P(C_i), \tag{10}$$

and

$$H(\mathbb{L}) = -\sum_j P(L_j) \log P(L_j). \tag{11}$$

The NMI score ranges from 0 (no mutual information) to 1 (perfect correlation). The NMI scores of the clustering results in the experiment are shown in Table 1.



(a) Dataset 1: hierarchical structure

(b) Dataset 1: clustering result

(c) Dataset 2: hierarchical structure

(d) Dataset 2: clustering result

(e) Dataset 3: hierarchical structure

(f) Dataset 3: clustering result

Figure 1: Hierarchical structures and clustering results for easy visualization.

Table 1: Experiment results on UCI datasets.

	HCDP	CDP	DBSCAN
Iris	**0.9595**	0.9159	0.8851
Wine	**0.9611**	0.9366	0.7469
Seeds	**0.8736**	0.8564	0.8975
WDBC	0.8702	**0.8764**	0.7300

5. Conclusion

This paper presented a hierarchical clustering method and introduced clustering stability which enables HCDP to extract an optimal clustering result. We used k-nearest neighbors to calculate the local density of data objects and construct clustering hierarchy according to the concept of density peaks. Clustering stability was computed to evaluate and extract "suitable" partitions from the hierarchy. Our experiments have shown that our methods are robust and accurate compared to the original density peak clustering algorithm and DBSCAN algorithm.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research is supported by the National Natural Science Foundation of China (NSFC) (nos. 61433012 and U1435215) and Shenzhen Basic Research Grant JCYJ20160229195940462.

References

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

[2] R. Mehmood, G. Zhang, R. Bie, H. Dawood, and H. Ahmad, "Clustering by fast search and find of density peaks via heat diffusion," *Neurocomputing*, vol. 208, pp. 210–217, 2016.

[3] Z. I. Botev, J. F. Grotowski, and D. P. Kroese, "Kernel density estimation via diffusion," *The Annals of Statistics*, vol. 38, no. 5, pp. 2916–2957, 2010.

[4] J.-Y. Chen and H.-H. He, "A fast density-based data stream clustering algorithm with cluster centers self-determined for mixed data," *Information Sciences*, vol. 345, pp. 271–293, 2016.

[5] J. Xie, H. Gao, W. Xie, X. 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.

[6] 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.

[7] L. Yaohui, M. Zhengming, and Y. Fang, "Adaptive density peak clustering based on K-nearest neighbors with aggregating strategy," *Knowledge-Based Systems*, vol. 133, pp. 208–220, 2017.

[8] V. Courjault-Radé, L. D'Estampes, and S. Puechmorel, *Improved Density Peak Clustering for Large Datasets*, 2016, working paper or preprint.

[9] X.-F. Wang and Y. Xu, "Fast clustering using adaptive density peak detection," *Statistical Methods in Medical Research*, vol. 26, no. 6, pp. 2800–2811, 2015.

[10] R. Mehmood, S. El-Ashram, R. Bie, H. Dawood, and A. Kos, "Clustering by fast search and merge of local density peaks for gene expression microarray data," *Scientific Reports*, vol. 7, article 45602, 2017.

[11] C. Jinyin, L. Xiang, Z. Haibing, and B. Xintong, "A novel cluster center fast determination clustering algorithm," *Applied Soft Computing*, vol. 57, pp. 539–555, 2017.

[12] R. Zhou, S. Zhang, C. Chen et al., "A distance and density-based clustering algorithm using automatic peak detection," in *2016 IEEE International Conference on Smart Cloud (SmartCloud)*, pp. 176–183, New York, NY, USA, November 2016.

[13] Z. Liang and P. Chen, "Delta-density based clustering with a divide-and-conquer strategy: 3DC clustering," *Pattern Recognition Letters*, vol. 73, pp. 52–59, 2016.

[14] J. Xu, G. Wang, and W. Deng, "DenPEHC: density peak based efficient hierarchical clustering," *Information Sciences*, vol. 373, pp. 200–218, 2016.

[15] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*, Elsevier Inc., Third Edition edition, 2012.

[16] R. Sibson, "SLINK: an optimally efficient algorithm for the single-link cluster method," *The Computer Journal*, vol. 16, no. 1, pp. 30–34, 1973.

[17] A. K. Jain and R. C. Dubes, *Algorithms for Clustering Data*, Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1988.

[18] C. T. Zahn, "Graph-theoretical methods for detecting and describing gestalt clusters," *IEEE Transactions on Computers*, vol. C-20, no. 1, pp. 68–86, 1971.

[19] M. Ankerst, M. M. Breunig, H.-P. Kriegel, and J. Sander, "OPTICS: ordering points to identify the clustering structure," in *Proceedings of the 1999 ACM SIGMOD international conference on Management of data - SIGMOD '99*, pp. 49–60, New York, NY, USA, June 1999.

[20] G. Gupta, A. Liu, and J. Ghosh, "Automated hierarchical density shaving: a robust automated clustering and visualization framework for large biological data sets," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 7, no. 2, pp. 223–237, 2010.

[21] M. Herbin, N. Bonnet, and P. Vautrot, "Estimation of the number of clusters and influence zones," *Pattern Recognition Letters*, vol. 22, no. 14, pp. 1557–1568, 2001.

[22] R. J. G. B. Campello, D. Moulavi, and J. Sander, "Density-based clustering based on hierarchical density estimates," in *Advances in Knowledge Discovery and Data Mining*, Lecture Notes in Computer Science, J. Pei, V. S. Tseng, L. Cao, H. Motoda, and G. Xu, Eds., pp. 160–172, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

[23] D. O. Loftsgaarden and C. P. Quesenberry, "A nonparametric estimate of a multivariate density function," *The Annals of Mathematical Statistics*, vol. 36, no. 3, pp. 1049–1051, 1965.

[24] E. Aksehirli, B. Goethals, E. Müller, and J. Vreeken, "Cartification: a neighborhood preserving transformation for mining high dimensional data," in *2013 IEEE 13th International Conference on Data Mining*, pp. 937–942, Dallas, TX, USA, December 2013.

[25] R. A. Jarvis and E. A. Patrick, "Clustering using a similarity measure based on shared near neighbors," *IEEE Transactions on Computers*, vol. C-22, no. 11, pp. 1025–1034, 1973.

[26] J. Schneider and M. Vlachos, "Fast parameterless density-based clustering via random projections," in *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management - CIKM '13*, pp. 861–866, New York, NY, USA, October-November 2013.

[27] J. L. Bentley, "Multidimensional binary search trees used for associative searching," *Communications of the ACM*, vol. 18, no. 9, pp. 509–517, 1975.

[28] S. M. Omohundro, *Five Balltree Construction Algorithms*, International Computer Science Institute Berkeley, 1989.

[29] J. A. Hartigan, "Estimation of a convex density contour in two dimensions," *Journal of the American Statistical Association*, vol. 82, no. 397, pp. 267–270, 1987.

[30] T. Pei, A. Jasra, D. J. Hand, A.-X. Zhu, and C. Zhou, "Decode: a new method for discovering clusters of different densities in spatial data," *Data Mining and Knowledge Discovery*, vol. 18, no. 3, pp. 337–369, 2009.

[31] W. Stuetzle and R. Nugent, "A generalized single linkage method for estimating the cluster tree of a density," *Journal of Computational and Graphical Statistics*, vol. 19, no. 2, pp. 397–418, 2010.

[32] A. Gionis, H. Mannila, and P. Tsaparas, "Clustering aggregation," *ACM Transactions on Knowledge Discovery from Data*, vol. 1, no. 1, p. 4, 2007.

[33] H. Chang and D.-Y. Yeung, "Robust path-based spectral clustering," *Pattern Recognition*, vol. 41, no. 1, pp. 191–203, 2008.

[34] M. Charytanowicz, J. Niewczas, P. Kulczycki, P. A. Kowalski, S. Łukasik, and S. Żak, "Complete gradient clustering algorithm for features analysis of X-ray images," in *Information Technologies in Biomedicine*, E. Piętka and J. Kawa, Eds., vol. 69 of Advances in Intelligent and Soft Computing, pp. 15–24, Springer, Berlin, Heidelberg, 2010.

[35] G. Gates, "The reduced nearest neighbor rule (corresp.)," *IEEE Transactions on Information Theory*, vol. 18, no. 3, pp. 431–433, 1972.

[36] B. Vandeginste, "PARVUS: an extendable package of programs for data exploration, classification and correlation, M. Forina, R. Leardi, C. Armanino and S. Lanteri, Elsevier, Amsterdam, 1988, Price: US $645 ISBN 0-444-43012-1," *Journal of Chemometrics*, vol. 4, no. 2, pp. 191–193, 1990.

[37] W. H. Wolberg, W. N. Street, and O. L. Mangasarian, "Machine learning techniques to diagnose breast cancer from image-processed nuclear features of fine needle aspirates," *Cancer Letters*, vol. 77, no. 2-3, pp. 163–171, 1994.