A Clustering Method Based on Improved Density Estimation and Shared Nearest Neighbors

Ying Guan[1], Yaru Li[1], Bin Li[2], and Yonggang Lu[1(✉)]

[1] School of Information Science and Engineering, Lanzhou University, Lanzhou 730000, Gansu, China
ylu@lzu.edu.cn

[2] Gansu New Vispower Technology Co. Ltd., No. 1689 Yanbei Road, Lanzhou 730000, Gansu, China

Abstract. Density-based clustering methods can detect clusters of arbitrary shapes. Most traditional clustering methods need the number of clusters to be given as a parameter, but this information is usually not available. And some density-based clustering methods cannot estimate local density accurately. When estimating the density of a given point, each neighbor of the point should have different importance. To solve these problems, based on the K-nearest neighbor density estimation and shared nearest neighbors, a new density-based clustering method is proposed, which assigns different weights to k-nearest neighbors of the given point and redefines the local density. In addition, a new clustering process is introduced: the number of shared nearest neighbors between the given point and the higher-density points is calculated first, the cluster that the given point belongs to can be identified, and the remaining points are allocated according to the distance between them and the nearest higher-density point. Using this clustering process, the proposed method can automatically discover the number of clusters. Experimental results on synthetic and real-world datasets show that the proposed method has the best performance compared with K-means, DBSCAN, CSPV, DPC, and SNN-DPC.

Keywords: Clustering · Shared nearest neighbors · Density estimation method · Local density

1 Introduction

Clustering [1–4] is the important process of pattern recognition, machine learning, and other fields. It can be used as an independent tool for data distribution and can be applied in image processing [3–6], data mining [3], intrusion detection [7, 8], and bioinformatics [9]. Without any prior knowledge, clustering methods assign points into different clusters according to their similarity such that points in the same clusters are similar to each other while points in the different clusters have a low similarity. Clustering methods are

D.-S. Huang et al. (Eds.): ICIC 2022, LNAI 13395, pp. 18–31, 2022.
https://doi.org/10.1007/978-3-031-13832-4_2

divided into different categories [10]: density-based clustering method, centroid-based clustering method, model-based clustering method, and grid-based clustering method.

Commonly centroid-based clustering methods include K-means [11] and K-medoids [12]. This type of method performs clustering by judging the distance between the point and the cluster center. Therefore, these methods can only identify spherical or spherical-like clusters and need the number of clusters as a priori [13].

The density-based clustering methods can identify clusters of arbitrary shapes, and it is not sensitive to noise [2]. Common and representative density-based clustering methods include DBSCAN [14], OPTICS [15], DPC [16], etc. DBSCAN defines a density threshold by using a neighborhood radius *Eps* and the minimum number of points *Minpts*. Based on this, it distinguishes core points and noisy points. As an effective extension of the DBSCAN, OPTICS only need to determine the value of *Minpts* and generate an augmented cluster ranking that represents the density-based clustering structure of each point. The DPC proposed by Liao et al. is based on two assumptions: the cluster center is surrounded by neighbors with lower local density, the distance between the cluster center and points with high local density is relatively large. It can effectively identify high-density centers [16].

At present, there are still drawbacks to most density-based clustering methods. DBSCAN-like methods can produce good clustering results but they depend on the distance threshold [17]. To avoid it, ARKNN-DBSCAN [18] and RNN-DBSCAN [19] redefine the local density of points by using the number of reverse nearest neighbors. Jian et al. [20] proposed a clustering center recognition standard based on relative density relationship, which is less affected by density kernel and density difference. IDDC [21] uses the relative density based on K-nearest neighbor to estimate the local density of points. CSPV [22] is a potential-based clustering method, it replaces density with the potential energy calculated by the distribution of all points. The one-step clustering process in some methods may lead to continuity errors, that is, once a point is incorrectly assigned, then more points may be assigned incorrectly [23]. To solve this problem, Yu et al. [24] proposed a method that can assign the non-grouped points to the suitable cluster according to the evidence theory and the information of K-nearest neighbors, improving the accuracy of clustering. Liu et al. [25] proposed a fast density peak clustering algorithm based on shared nearest neighbor(SNN-DPC), which improves the clustering process and reduces the impact of density peak and allocation process on clustering results to a certain extent. However, the location and number of cluster centers still need to be manually selected from the decision graph.

These methods have solved the problems existing in the current methods to a certain extent, but these methods do not consider the importance of points, which may result in inaccurate density calculation. This paper attempts to solve the inaccurate definition of local density and the errors caused by one-step clustering process. Therefore, a new density-based clustering method is proposed. Based on the K-nearest neighbor density estimation [26, 27] and the shared nearest neighbor [25, 28], we redefine the K-nearest neighbor density estimation to calculate the local density, which assigns the different importance for each neighbor of the given point. A new clustering process is proposed: the number of shared nearest neighbors between the given point and the higher-density point is calculated first, the cluster that the given point belongs to can be identified, and

the remaining points are allocated according to the distance between them and the nearest higher-density point. To some extent, it avoids the continuity error caused by directly assigning points to the cluster where the nearest higher-density neighbor is located. After calculating the local density, all points are sorted in the descending order, and then the cluster centers are selected from the points whose density is higher than the given point. Through this process, the method can automatically discover both the cluster center and the number of clusters.

The rest of this paper is organized as follows. Section 2 introduces relevant definitions and the new clustering method we proposed. In Sect. 3, we discuss the experimental results on synthetic datasets and real-world datasets and compare the proposed method with other classical clustering methods according to several evaluation metrics. Section 4 summarizes the paper and discusses future work. Table 1 illustrates the symbols and notations used in this paper.

Table 1. Symbols and notations

x_i	A point in the dataset X
d	The dimension of the feature vector in the dataset
N	The number of points in the dataset
D	The distance matrix of the dataset
K	The number of nearest neighbors
$r_k(i)$	The radius of x_i to its K-th nearest neighbor
K_i	The number of weighted points according to shared nearest neighbors of x_i
k_1	A coefficient to calculate the parameter K
$KNN(i)$	The K-nearest neighbor set of point x_i
$SNN(i, j)$	The numbers of shared nearest neighbors between x_i and x_j
R_i	The radius of x_i to its K-th shared nearest neighbor
V_i	The volume of the high-dimensional sphere with radius R_i
ω	The weight coefficient
$\rho_k(i)$	The estimated density of x_i

2 Method

A new clustering method is proposed according to the new density estimation method and new allocation strategy in the clustering process. And the new density estimation method is based on the K-nearest neighbor density estimation, which is the nonparametric density estimation method proposed by fix and Hodges [26]. K-nearest neighbor density estimation [26, 27] is a well-known and simplest density estimation method which is based on the concept: the density function of a continuity point can be estimated using the number of neighbors observed in a small region near the point. In the dataset

$X[N] = \{x_i\}_{i=1}^{N}$, the estimation of the density function is based on the distance from x_i to its K-th nearest neighbor. For points in different density regions, the neighborhood size determined by the K-nearest neighbors is adaptive, ensuring the resolution of high-density regions and the continuity of low-density regions.

For each $x_i \in R^d$, the estimated density of x_i is:

$$\rho_k(i) = \frac{K}{N * V_d * r_k(i)^d} \tag{1}$$

where V_d is the volume of the unit sphere in R^d, K is the number of neighbors, $r_k(i)$ represent the distance from x_i to the K-th nearest neighbor in the dataset $X[N]$.

2.1 The New Density Estimation Method

Based on the K-nearest neighbor density estimation, the new density estimation method is proposed. According to the number of shared neighbors between the given point and others, the volume of a region containing the K shared nearest neighbors of the given point is used to estimate the local density. Generally, the parameter $K = k_1 \times \sqrt{N}$, k_1 is coefficient. Local density estimation is redefined as:

$$\rho_k(i) = \frac{K_i}{N * V_i} \tag{2}$$

where N is the number of points in the dataset. For point x_i, K_i is the number of weighted points according to shared nearest neighbors, V_i is the volume of the high-dimensional sphere with radius R_i, and all the spheres considered in the experiment are closed Euclidean spheres.

For K-nearest neighbors [27, 29] of each point, it refers that selecting K points according to the distance between points. For points x_i and x_j in the dataset, the K-nearest neighbor sets of x_i and x_j are defined as $KNN(i)$ and $KNN(j)$. Based on K-nearest neighbors, the shared nearest neighbors [25, 28] between x_i and x_j are their common K-nearest neighbor sets, expressed as:

$$SNN(i, j) = KNN(i) \cap KNN(j) \tag{3}$$

That is to say, the matrix SNN represents the numbers of shared nearest neighbors between points.

The points are sorted in descending order according to the number of shared neighbors. Point x_j is the K-th shared nearest neighbor of the given point x_i, and the neighborhood radius R_i of point x_i is defined as:

$$R_i = D[i, j] \tag{4}$$

D is the distance matrix of the dataset, and the distance is Euclidean distance.

Given the radius R_i of point x_i, the volume of its neighborhood can be calculated by:

$$V_i = R_i^d \tag{5}$$

where d is the dimension of the feature vector in the dataset.

In general, for point x_i, the importance of its each neighbor is different, and the contribution to the density estimation of point x_i should be different. In our definition, this contribution is related to the number of shared nearest neighbors between x_i and its each neighbor. According to the number of shared neighbors between points x_i and x_j, the weight coefficient formula is defined to assign different weights to K-nearest neighbors of any point:

$$\omega(i,j) = \frac{|SNN(i,j)|}{K} \tag{6}$$

where $|SNN(i,j)|$ is the number of shared neighbors between point x_i and point x_j.

K_i is redefined by adding the different weights to K-nearest neighbors of point x_i, as shown in Eq. 7:

$$K_i = \sum_{j=1}^{K} \omega(i,j) \tag{7}$$

As the neighbor of x_i, if x_j has the more number of shared nearest neighbors with x_i, that is, the weight of x_j is bigger, x_j has more contribution in calculating the local density of point x_i. Using Eq. 6 and Eq. 7, Eq. 1 can be expressed in the form of Eq. 8.

$$\rho_k(i) = \frac{\sum_{j=1}^{K} \frac{|SNN(i,j)|}{K}}{N * V_i} \tag{8}$$

In summary, when calculating the local density of x_i, different weights are added to K points falling in a neighborhood according to the number of shared neighbors. The more the number of shared nearest neighbors with x_i, the greater contribution to the local density estimation of x_i.

2.2 A New Allocation Strategy in the Clustering Process

The allocation process of some clustering methods has poor fault tolerance. When one point is assigned incorrectly, more subsequent points will be affected, which will have a severe negative impact on the clustering results [23, 24]. Therefore, a new clustering process is proposed to make the allocation more reasonable and avoid the continuity error caused by direct allocation to a certain extent.

In the proposed clustering method, all points are sorted in the descending order according to the local density value. The sorted index is stored in the array $sortedIdx[1 \ldots N]$. Then, in the sorted points queue, points are accessed one by one with the local density value from the highest to the lowest. The first point in the queue has the highest local density and automatically becomes the center of the first cluster. For each subsequent point $sortedIdx[i]$ in the queue, two special points are identified: a point *parent1* is the nearest point to $sortedIdx[i]$ in the visited points, and a point *parent2* is the point that has the most number of shared neighbors with $sortedIdx[i]$. The number of shared nearest neighbors between $sortedIdx[i]$ and *parent2* is compared with $K/2$. If it is at least half of $K/2$, $sortedIdx[i]$ is assigned to the cluster where *parent2* belongs.

Otherwise, the distance between *sortedIdx*[i] and *parent1* is compared with the given distance bandwidth parameter *B*. If the distance is greater than parameter *B*, *sortedIdx*[i] is the center of the new cluster; if not, it is assigned to the cluster where point *parent1* belongs. This process continues until all points are visited and assigned to the proper clusters. The detail of the proposed method is shown in Algorithm 1.

Algorithm 1 The proposed method

Input: distance matrix *D*, number of the neighbors *K*, distance bandwidth *B*

Output: labels of points $root[1 \ldots N]$

1. **for** i=1 to N do
2. calculate the local density according to Eq.1
3. **end for**
4. $root[sortedIdx[1]] \leftarrow sortedIdx[1]$
5. **for** i=2 to N do
6. $P \leftarrow sortedIdx[i]$
7. $parent \leftarrow sortedIdx[1]$
8. $minDist \leftarrow D[P, parent]$
9. $maxSnn \leftarrow SNN[P, parent]$
10. **for** j=2 to i-1 do
11. **if** $D[P, sortedIdx[j]] < minDist$ then
12. $parent1 \leftarrow sortedIdx[j]$
13. $minDist \leftarrow D[P, parent1]$
14. **end if**
15. **if** $SNN[P, sortedIdx[j]] > maxSnn$ then
16. $parent2 \leftarrow sortedIdx[j]$
17. $maxSnn \leftarrow SNN[P, parent2]$
18. **end if**
19. **if** $maxSnn \geq K/2$ then
20. $root[P] \leftarrow root[parent2]$
21. **else if** $minDist < B$ then
22. $root[P] \leftarrow root[parent1]$
23. **else**
24. $root[P] \leftarrow P$
25. **end if**
26. **end if**
27. **end for**
28. **end for**

3 Experiments

In this section, we use classical synthetic datasets and real-world datasets to test the performance of the proposed method. Moreover, we take K-means, DBSCAN, CSPV, DPC, and SNN-DPC as the control group. According to several evaluation metrics, the performance of the proposed method is compared with five classical clustering methods.

3.1 Datasets and Processing

To verify the performance of the proposed method, we select real-world datasets and synthetic datasets with different sizes, dimensions, and the number of clusters. The synthetic datasets include Flame, R15, D31, S2, and A3. The real-world datasets include Iris, Wine, Seeds, Breast, Wireless, Banknote, and Thyroid. The characteristics of datasets used in the experiments are presented in Table 2. The evaluation metrics used in experiments are as followed: Normal Mutual Information (NMI) [30], adjusted Rand index(ARI) [30], and Fowlkes-Mallows index (FMI) [31]. The upper bound is 1, where larger values indicate better clustering results.

Table 2. Characteristics of datasets

Dataset	Points	Dimensions	Clusters	Type
Iris	150	4	3	Real-world
Wine	178	13	3	Real-world
Seeds	210	7	3	Real-world
Breast	699	9	2	Real-world
Wireless	2000	7	4	Real-world
Banknote	1372	4	2	Real-world
Thyroid	215	5	3	Real-world
Flame	240	2	2	Synthetic
R15	600	2	15	Synthetic
D31	3100	2	31	Synthetic
S2	5000	2	15	Synthetic
A3	7500	2	50	Synthetic

3.2 Parameters Selection

We set parameters of each method to ensure the comparison of their best performance. The parameters corresponding to the optimal results of different methods are chosen. The real number of clusters is assigned to K-means, DPC, and SNN-DPC.

The proposed method needs two key parameters: the number of nearest neighbor K and the distance bandwidth B. The selection of parameter B [22] is derived from the distance matrix D:

$$MinD(i) = \underset{j=1,\ldots N, j\neq i}{min}\left(D\left[i,j\right]\right) \tag{9}$$

$$B = \underset{i=1,\ldots N}{max}\left(MinD(i)\right) \tag{10}$$

The parameter K is selected by the formula $K = k_1 * \sqrt{N}$ to determine the relationship between K and N, k_1 is the coefficient. The parameter is related to the size of the dataset and clusters. In the proposed method, k_1 is limited in (0,9] to adapt to different datasets. Figure 1 shows the FMI indices of some representative datasets with different k_1 values. It can be seen that for datasets S2 and R15, the FMI index is not sensitive to k_1 when k_1 is within region (0, 1.5), and for the Wine dataset, the FMI index is not sensitive to k_1 within the whole region.

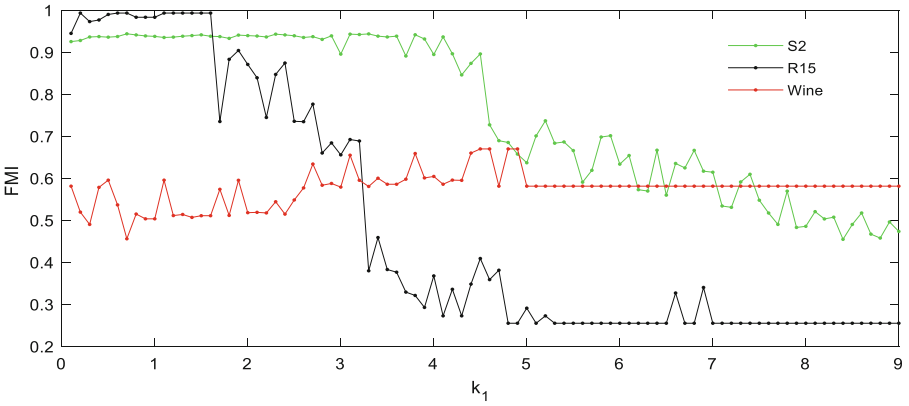

Fig. 1. Results on different datasets with different k_1

3.3 Experimental Results

We conduct comparison experiments on 12 datasets and evaluate the clustering results with different evaluation metrics. In the following experiments, we first verify the effects of the new density estimation method, which is proposed in this paper. Then to test the effectiveness of the automatically discovered number of clusters, the proposed method is compared with the other methods. And the whole proposed method with the other five commonly used methods is compared.

The New Density Estimation Method. Based on the original K-nearest neighbor density estimation [26], the new density estimation method is proposed, which is described in Sect. 2.1. To check if the new method improves the accuracy of the local density calculation, the comparison experiment is conducted between the original method and

the new method. Firstly, the original and the new density estimation method are used to estimate the local density of the points. Secondly, after the local density is calculated and sorted in descending order, the same clustering process is used to assign points, which is proposed by [22]. Finally, the clustering results of datasets are evaluated by different metrics, which are shown in Fig. 2 and Fig. 3.

Compared with the original method, the new method is superior on most real-world datasets but is slightly poor on the Seeds. On the synthetic datasets R15, D31, S2, and A3, the new method has good clustering results, which is not much different from the original method. In summary, the new method shows an advantage over the original density estimation method.

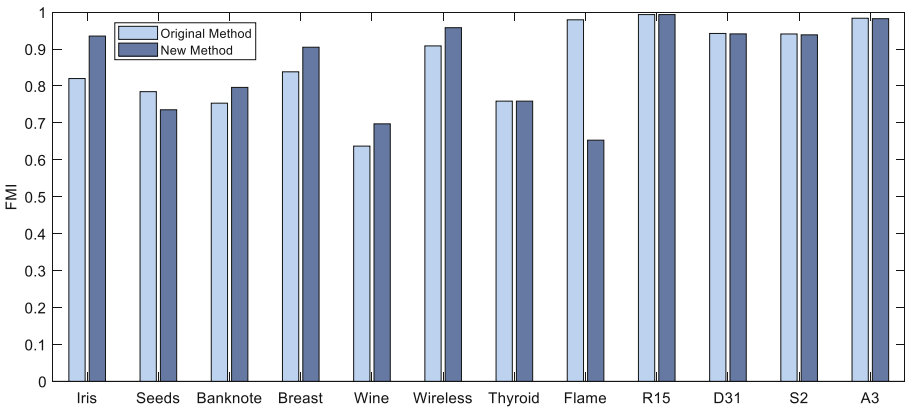

Fig. 2. Comparison of density estimation with the original method on FMI.

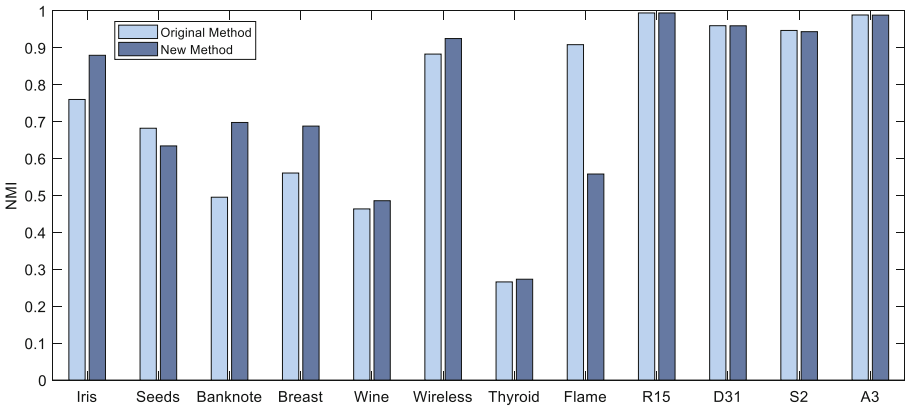

Fig. 3. Comparison of density estimation with the original method on NMI.

The Effectiveness of the Automatically Discovered Number of Clusters. The comparison experiments are conducted among DBSCAN, CSPV, and the proposed method

to verify the validity of the automatically discovered number of clusters. The proposed method is not compared with K-means, DPC, and SNN-DPC because the real number of clusters is used in these methods. The experimental results are shown in Table 3. The accuracy of the discovered number of clusters by DBSCAN, CSPV, and the proposed method are 42%, 42%, and 83% respectively. On 5 synthetic datasets, the proposed method can correctly discover the number of clusters; the proposed method outperforms DBSCAN and CSPV on real-world datasets Iris, Seeds, Breast, and Wireless. In summary, the proposed method is better than DBSCAN and CSPV for automatically discovering the number of clusters.

Table 3. The number of clusters discovered by different methods

Datasets	Real number of clusters	Discovered number of clusters		
		DBSCAN	CSPV	Ours
Iris	3	4	**3**	**3**
Wine	3	1	5	2
Seeds	3	2	7	**3**
Breast	2	**2**	3	**2**
Wireless	4	**4**	1	**4**
Banknote	2	10	6	3
Thyroid	3	6	2	2
Flame	2	**2**	**2**	**2**
R15	15	**15**	18	**15**
D31	31	**31**	**31**	**31**
S2	15	13	17	**15**
A3	50	48	**50**	**50**

Experiments on the Different Datasets. The experiments are conducted on the different datasets, and the experimental results are presented in Table 4. From Table 4, the proposed method has the best clustering results on real-world datasets Iris, Seeds, Breast, Wireless, and Banknote than other clustering methods. On the Wine, the result of the proposed method is the best. For the dataset Thyroid, the proposed method performs better than DPC and SNN-DPC.

Table 4. Comparison of clustering results on datasets with three measures

Method	NMI	ARI	FMI	parm	NMI	ARI	FMI	parm
	Iris				Seeds			
K-means	0.7582	0.7302	0.8208	3	0.6949	0.7166	0.8106	3
DBSCAN	0.7196	0.7063	0.7972	0.4/3	0.5651	0.5975	0.7327	1.7/46
CSPV	0.7355	0.5638	0.7635		0.6452	0.6057	0.7287	
DPC	0.8705	0.8858	0.9234	0.6	0.6744	0.7170	0.8106	0.8
SNN-DPC	0.8851	0.9038	0.9355	5	**0.7566**	0.7549	0.8364	6
Ours	**0.9011**	**0.9222**	**0.9478**	2.1	0.7539	**0.7666**	**0.8441**	1.9
	Banknote				Breast			
K-means	0.0303	0.0485	0.5518	2	0.0111	−0.0128	0.7192	2
DBSCAN	**0.6798**	0.6653	0.8175	1.8/9	0.7304	0.8250	0.9187	4.3/11
CSPV	0.1543	0.0969	0.6457		0.0122	−0.0129	0.7191	
DPC	–	–	–		0.0111	−0.0128	0.7191	1
SNN-DPC	0.6145	0.6309	0.8173	23	0.0915	−0.0516	0.6586	30
Ours	0.6719	**0.7101**	**0.8449**	6.5	**0.7318**	**0.8336**	**0.9257**	8.4
	Wine				Wireless			
K-means	0.4288	0.3711	0.5835	3	0.8904	0.8885	0.9165	4
DBSCAN	–	–	0.5813	0.1/2	0.7891	0.7897	0.8407	8/49
CSPV	0.3830	0.2697	0.5011		–	–	0.4996	
DPC	0.3913	0.4070	0.6069	0.5	0.8674	0.8541	0.8909	0.2
SNN-DPC	**0.4316**	**0.4485**	0.6361	36	0.8997	0.8838	0.9130	33
Ours	0.4307	0.4025	**0.67**	4.5	**0.9308**	**0.9491**	**0.9618**	2.8
	Thyroid				Flame			
K-means	0.4946	0.5791	0.8063	3	0.3989	0.4534	0.7364	2
DBSCAN	**0.5407**	**0.6986**	**0.8731**	3.7/2	0.8097	0.8970	0.9790	1.3/28
CSPV	0.0893	0.0314	0.7318		0.4669	0.4256	0.7207	
DPC	0.2128	0.1815	0.6241	0.1	**1**	**1**	**1**	2.7
SNN-DPC	0.3628	0.4402	0.7596	5	0.8883	0.9337	0.9696	6
Ours	0.3612	0.2665	0.7703	1.6	**1**	**1**	**1**	3.8
	R15				D31			
K-means	**0.9942**	**0.9928**	**0.9932**	15	0.9523	0.9061	0.9029	31
DBSCAN	0.9922	0.9893	0.9900	0.7/29	0.9174	0.8501	0.8551	1.1/48
CSPV	0.9727	0.9451	0.9494		0.9537	0.9278	0.9301	
DPC	**0.9942**	**0.9928**	**0.9932**	0.1	0.9579	0.9370	0.9390	1
SNN-DPC	**0.9942**	**0.9928**	**0.9932**	11	**0.9660**	**0.9509**	**0.9525**	41
Ours	**0.9942**	**0.9928**	**0.9932**	0.6	0.9648	0.9484	0.9501	0.7

(*continued*)

Table 4. (*continued*)

Method	NMI	ARI	FMI	parm	NMI	ARI	FMI	parm
	S2				A3			
K-means	0.9463	0.9378	0.9420	15	0.9760	0.9175	0.9199	50
DBSCAN	0.8866	0.7631	0.7886	4.5/45	0.9454	.0.8693	0.8728	1.8/50
CSPV	0.9313	0.9103	0.9163		0.9860	0.9770	0.9774	
DPC	0.9454	0.9370	0.9412	0.7	0.9880	0.9809	0.9813	0.3
SNN-DPC	0.9402	0.9280	0.9328	35	0.9860	0.9772	0.9776	25
Ours	**0.9481**	**0.9399**	**0.9439**	0.7	**0.9912**	**0.9862**	**0.9865**	0.9

The clustering results of the proposed method for the 5 synthetic datasets are shown in Fig. 4. For 5 synthetic datasets, on the datasets Flame, S2, and A3, the proposed method has the best clustering result, especially on the Flame, the same result as the original data label is obtained. On the dataset D31, the proposed method is slightly poor than the best. The proposed method generates the same results as K-means, DPC, and SNN-DPC on dataset R15, but the proposed method can discover the number of clusters automatically. On the synthetic datasets, the results of the proposed method are similar to SNN-DPC, but slightly better than SNN-DPC. And the proposed method outperforms the other five methods on most real-world datasets.

In summary, the proposed method has more advantages and outperformance than other methods in the effectiveness of clustering results in most cases. These results show that our redefinition of local density and the new clustering process is effective.

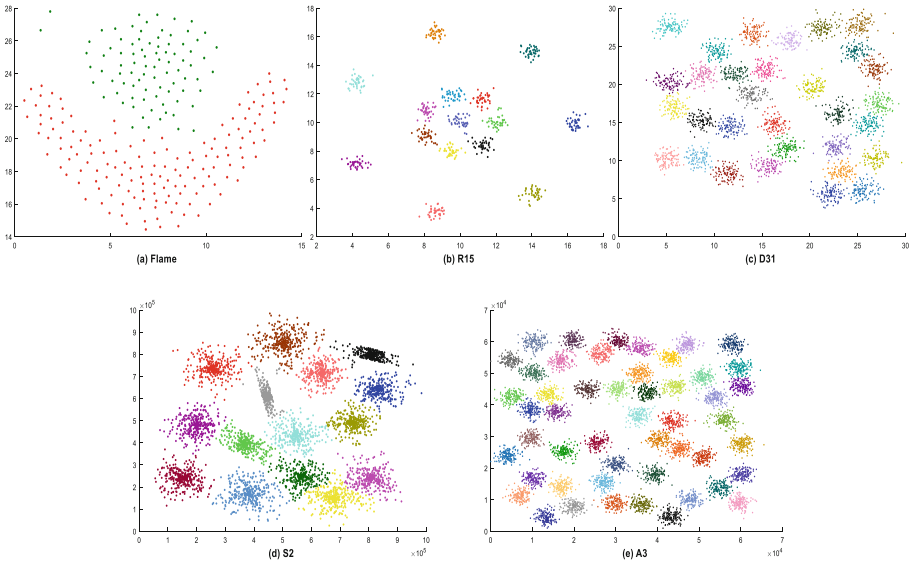

Fig. 4. Clustering results of the proposed method on synthetic datasets

4 Conclusion

In this paper, a new clustering method is proposed according to the K-nearest neighbor density estimation and shared nearest neighbors. When calculating the local density, the number and the different contributions of points in the neighborhood are considered, which improves the accuracy of local density calculation to a certain extent. This paper proves that the proposed method can adapt to most different datasets and using the improved local density estimation can improve the clustering performance. The proposed method has a parameter K, the formula $K = k_1 \times \sqrt{N}$ is used to determine the relationship between K and N, and k_1 is the coefficient. Although k_1 is limited in a reasonable range, k_1 had a considerable influence on the clustering results in some datasets. As a possible direction for future work, we will explore the possibility of reducing the influence of the parameter K on the clustering results.

Acknowledgment. This work was partially supported by the Gansu Provincial Science and Technology Major Special Innovation Consortium Project (Project No. 1), the name of the innovation consortium is Gansu Province Green and Smart Highway Transportation Innovation Consortium, and the project name is Gansu Province Green and Smart Highway Key Technology Research and Demonstration.

References

1. Omran, M., Engelbrecht, A.P., Salman, A.: An overview of clustering methods. Intell. Data Anal. **11**(6), 583–605 (2007)
2. Han, J., Pei, J., Kamber, M.: Data Mining: Concepts and Techniques. Elsevier, Amsterdam (2011)
3. Jain, A.K., Murty, M.N., Flynn, P.J.: Data clustering: a review. ACM Comput. Surv. (CSUR) **31**(3), 264–323 (1999)
4. Zhang, C., Wang, P.: A new method of color image segmentation based on intensity and hue clustering. In: Proceedings 15th International Conference on Pattern Recognition, ICPR-2000, vol. 3, pp. 613–616. IEEE (2000)
5. Reddy, S., Parker, A., Hyman, J., Burke, J., Estrin, D., Hansen, M.: Image browsing, processing, and clustering for participatory sensing: lessons from a dietsense prototype. In: Proceedings of the 4th Workshop on Embedded Networked Sensors, pp. 13–17 (2007)
6. Khan, Z., Ni, J., Fan, X., Shi, P.: An improved k-means clustering algorithm based on an adaptive initial parameter estimation procedure for image segmentation. Int. J. Innov. Comput. Inf. Control **13**(5), 1509–1525 (2017)
7. Portnoy, L.: Intrusion detection with unlabeled data using clustering. Ph.D. thesis, Columbia University (2000)
8. Guan, Y., Ghorbani, A.A., Belacel, N.: Y-means: a clustering method for intrusion detection. In: CCECE 2003-Canadian Conference on Electrical and Computer Engineering. Toward a Caring and Humane Technology (Cat. No. 03CH37436), vol. 2, pp. 1083–1086. IEEE (2003)
9. Frank, E., Hall, M., Trigg, L., Holmes, G., Witten, I.H.: Data mining in bioinformatics using Weka. Bioinformatics **20**(15), 2479–2481 (2004)
10. Rui, X., Wunsch, D.I.: Survey of clustering algorithms. IEEE Trans. Neural Netw. **16**(3), 645–678 (2005)

11. Macqueen, J.: Some methods for classification and analysis of multivariate observations. In: Proceedings of Berkeley Symposium on Mathematical Statistics Probability (1965)
12. Kaufman, L., Rousseeuw, P.J.: Finding Groups in Data: An Introduction to Cluster Analysis. Wiley, Hoboken (2005)
13. Jain, A.K.: Data clustering: 50 years beyond k-means. Pattern Recogn. Lett. **31**(8), 651–666 (2010)
14. Ester, M., Kriegel, H.P., Sander, J., Xu, X.: Density-based spatial clustering of applications with noise. In: International Conference on Knowledge Discovery and Data Mining, vol. 240, p. 6 (1996)
15. Ankerst, M., Breunig, M.M., Kriegel, H.P., Sander, J.: OPTICS: ordering points to identify the clustering structure. In: SIGMOD 1999, Proceedings ACM SIGMOD International Conference on Management of Data, Philadelphia, Pennsylvania, USA, 1–3 June 1999 (1999)
16. Rodriguez, A., Laio, A.: Clustering by fast search and find of density peaks. Science **344**(6191), 1492–1496 (2014)
17. Li, H., Liu, X., Li, T., Gan, R.: A novel density-based clustering algorithm using nearest neighbor graph. Pattern Recogn. **102**, 107206 (2020)
18. Pei, P., Zhang, D., Guo, F.: A density-based clustering algorithm using adaptive parameter k-reverse nearest neighbor. In: 2019 IEEE International Conference on Power, Intelligent Computing and Systems (ICPICS), pp. 455–458. IEEE (2019)
19. Bryant, A., Cios, K.: RNN-DBSCAN: a density-based clustering algorithm using reverse nearest neighbor density estimates. IEEE Trans. Knowl. Data Eng. **30**(6), 1109–1121 (2017)
20. Hou, J., Zhang, A., Qi, N.: Density peak clustering based on relative density relationship. Pattern Recogn. **108**(8), 107554 (2020)
21. Wang, Y., Yang, Y.: Relative density-based clustering algorithm for identifying diverse density clusters effectively. Neural Comput. Appl. **33**(16), 10141–10157 (2021). https://doi.org/10.1007/s00521-021-05777-2
22. Lu, Y., Wan, Y.: Clustering by sorting potential values (CSPV): a novel potential-based clustering method. Pattern Recogn. **45**(9), 3512–3522 (2012)
23. Jiang, J., Chen, Y., Meng, X., Wang, L., Li, K.: A novel density peaks clustering algorithm based on k nearest neighbors for improving assignment process. Phys. A **523**, 702–713 (2019)
24. Yu, H., Chen, L., Yao, J.: A three-way density peak clustering method based on evidence theory. Knowl.-Based Syst. **211**, 106532 (2021)
25. Liu, R., Wang, H., Yu, X.: Shared-nearest-neighbor-based clustering by fast search and find of density peaks. Inf. Sci. **450**, 200–226 (2018)
26. Fukunaga, K., Hostetler, L.: Optimization of k nearest neighbor density estimates. IEEE Trans. Inf. Theory **19**(3), 320–326 (1973)
27. Dasgupta, S., Kpotufe, S.: Optimal rates for k-NN density and mode estimation. In: Advances in Neural Information Processing Systems, vol. 27 (2014)
28. Ertöz, L., Steinbach, M., Kumar, V.: Finding clusters of different sizes, shapes, and densities in noisy, high dimensional data. In: Proceedings of the 2003 SIAM International Conference on Data Mining, pp. 47–58. SIAM (2003)
29. Qaddoura, R., Faris, H., Aljarah, I.: An efficient clustering algorithm based on the k-nearest neighbors with an indexing ratio. Int. J. Mach. Learn. Cybern. **11**(3), 675–714 (2020)
30. Vinh, N.X., Epps, J., Bailey, J.: Information theoretic measures for clusterings comparison: variants, properties, normalization and correction for chance. J. Mach. Learn. Res. **11**, 2837–2854 (2010)
31. Fowlkes, E.B., Mallows, C.L.: A method for comparing two hierarchical clusterings. J. Am. Stat. Assoc. **78**(383), 553–569 (1983)