Clustering of Multiple Density Peaks

Borui Cai[1(✉)], Guangyan Huang[1], Yong Xiang[1], Jing He[2], Guang-Li Huang[3], Ke Deng[3], and Xiangmin Zhou[3]

[1] School of Information Technology, Deakin University, Melbourne, Australia
{bcai,guangyan.huang,yong.xiang}@deakin.edu.au
[2] Data Science Research Institute, Swinburne University of Technology, Melbourne, Australia
lotusjing@gmail.com
[3] School of Science, RMIT University, Melbourne, Australia
guangli.huang@student.rmit.edu.au
{ke.deng,xiangmin.zhou}@rmit.edu.au

Abstract. Density-based clustering, such as Density Peak Clustering (DPC) and DBSCAN, can find clusters with arbitrary shapes and have wide applications such as image processing, spatial data mining and text mining. In DBSCAN, a core point has density greater than a threshold, and can spread its cluster ID to its neighbours. However, the core points selected by one cut/threshold are too coarse to segment fine clusters that are sensitive to densities. DPC resolves this problem by finding a data point with the peak density as centre to develop a fine cluster. Unfortunately, a DPC cluster that comprises only one centre may be too fine to form a natural cluster. In this paper, we provide a novel clustering of multiple density peaks (MDPC) to find clusters with arbitrary number of regional centres with local peak densities through extending DPC. In MDPC, we generate fine seed clusters containing single density peaks, and form clusters with multiple density peaks by merging those clusters that are close to each other and have similar density distributions. Comprehensive experiments have been conducted on both synthetic and real-world datasets to demonstrate the accuracy and effectiveness of MDPC compared with DPC, DBSCAN and other base-line clustering algorithms.

Keywords: clustering, density peaks, cluster merge

1 Introduction

Clustering can discover the relationship of points by grouping similar points into the same cluster; this capability makes it attractive in many data mining tasks. K-means finds the best k centres to minimize the overall distance between the points and their centres [13]. Affinity Propagation (AP) [7] finds the best point to represent the whole cluster. However, both of them are not effective in finding non-spherical clusters.

Mean-shift [9] is designed to find non-spherical clusters, but it highly relies on the significance of density gradients among data points. Density-based clustering methods, such as Density Peak Clustering (DPC) [14] and DBSCAN [6], use critical data points to form clusters. DBSCAN finds natural shape clusters by finding core

data points which spread cluster IDs to their neighbours; and a core point is a data point that has density greater than a threshold. However, core data points that are selected based on one cut/threshold in DBSCAN are too coarse to segment fine clusters that are sensitive to density, as shown in the third dataset in Fig.5(c), where we can only accurately find the top left clusters (two sparse clusters) or the bottom left clusters (two dense clusters), but not both. Also, in practice it is hard to find the optimal values of DBSCAN's two parameters.

DPC [14] resolves the problem of DBSCAN by finding a data point with the peak density as a centre to develop a fine cluster, and it only needs one parameter d_c, the cut-off distance. DPC converts n-dimensional features into two features: *density* and *delta* (the distance to the point which spreads cluster ID to it), chooses one point with the peak density as the centre for each cluster, and assigns the rest points to the relative cluster centres. DPC can find clusters with more fine densities than DBSCAN. This advantage makes DPC a potential solution to many data mining tasks [15, 17]. Unfortunately, a DPC cluster that comprises only one centre (density peak) may be too fine to form a natural cluster, since this is too strict for a natural cluster that comprises multiple regional centres with local peak densities. DPC is unable to achieve a satisfying result due to two problems. First, it is difficult to choose correct cluster centres from candidate density peaks (with "anomalously" large *density* and *delta*); for example, in Fig.1(c), although we know there are five cluster centres, it is challenging to pick them from those candidate density peaks (large red dots) in Fig.1(c') and requires exhaustively searching. Second, the assumption in DPC that each cluster only comprises one centre is not always true thus DPC cannot find natural shape clusters accurately; three examples are shown in Fig.5(b).

Hierarchical clustering inspires us to merge DPC clusters and form the correct clusters that comprise multiple density peaks. We initially generate seed clusters by DPC and merge seed clusters according to their distances. Thus, a good cluster distance is the key for the accuracy. We have tested four existing cluster distances ("single", "average", "complete" and Hausdorff linkage [1]), but their accuracy is not good in merging DPC seed clusters, as shown in Fig.4.

In this paper, we propose a novel Multiple Density Peaks Clustering (MDPC) method to flexibly find clusters with arbitrary number of regional centres that have local peak densities (as they exist in the real world) through extending DPC. In MDPC, we generate fine seed clusters, which are simpler than natural shape clusters since each seed cluster has only one density peak, and discover a natural shape cluster with multiple density peaks by merging those seed clusters that are close to each other and have similar density distributions. We conduct comprehensive experiments on both synthetic and real-world datasets to demonstrate the accuracy and effectiveness of MDPC. Therefore, this paper has three contributions:

- We provide a novel MDPC method, which improves DPC in two aspects: to find natural shape clusters with arbitrary number of local density peaks and to form seed clusters by automatic selection of cluster centres.
- We define a new measurement of the distance between two seed clusters, based on which seed clusters are merged more accurately in MDPC than four counterpart cluster distances ("single", "average", "complete" and Hausdorf [1]).

– We conduct experiments on both synthetic and real-world datasets to prove that MDPC achieves more accurate results than DPC, DBSCAN and other baseline clustering algorithms.

The rest of this paper is organized as follows. Section 2 presents the preliminary knowledge and problem definition. Section 3 details the proposed MDPC method. Section 4 conducts experimental studies to evaluate MDPC. Finally, Section 5 concludes the paper.

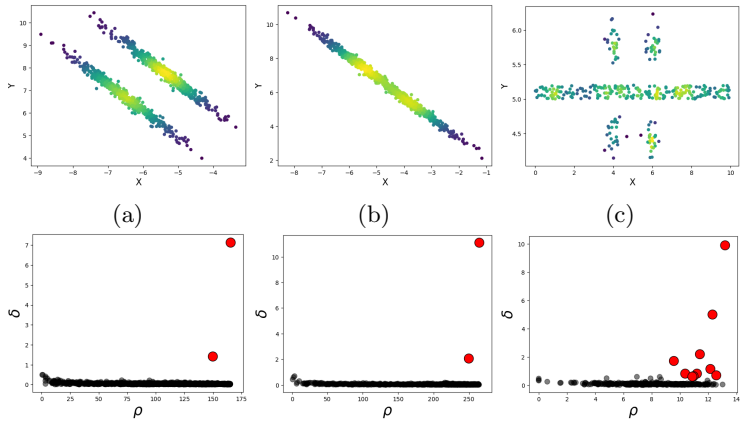


Fig. 1: (a)-(c) are three datasets with color indicates density, and (a')-(c') are corresponding decision graphs with large red dots as candidate cluster centers.

2 Preliminary Knowledge and Problem Definition

In this section, we present the DPC algorithm, point out its two problems and discuss the state of the art methods that approach these problems.

2.1 The Algorithm of DPC

As we mentioned in Section 1, DPC improves DBSCAN to find more fine clusters with only one density peak as a centre. That is, a DBSCAN cluster with n local density peaks is possible to be split into n DPC clusters. To help understand this, we introduce the main idea of DPC as follows. In DPC, the first step is to determine cluster centres using a 2-dimensional "decision graph' as follows. The n-dimensional feature space of a point is mapped into the 2-dimensional feature space: ρ and δ. The density ρ_i of data point i is given by:

$$\rho_i = \Sigma_j \chi\left(d_{ij} - d_c\right), \tag{1}$$

where $\chi(x) = 1$ if $x < 0$, and $\chi(x) = 0$ otherwise. d_c is a threshold distance (normalized by the largest distance in this paper) and d_{ij} is the Euclidean distance

between data points i and j. Then, these local density peaks are those points that have the greatest ρ in its d_c region. Delta δ_i is the minimum distance between point i (with density ρ_i) and any other point, j, with higher density ρ_j:

$$\delta_i = \min_{j:\rho_j > \rho_i} (d_{ij}) . \tag{2}$$

DPC assigns the largest distance in dataset to the δ of the point with the highest density,

In DPC, the two-dimensional features (ρ and δ) form a "decision graph"; an example is shown in Fig.1(a'), which is transferred from raw data as shown in Fig.1(a). User chooses one point that has "anomalously" large ρ and δ in decision graph [14] as the cluster centre for each cluster manually or using some heuristic ways. Then, the non-centre points are sorted in a descending order by ρ and sequentially assigned the same cluster ID as their nearest point with a higher density. All data points obtain cluster IDs before DPC finds the border points whose neighbourhood cross more than one clusters.

2.2 The Problems and Related Approaches

In this paper, we will discover natural clusters that have arbitrary number of regional centres with local peak densities by extending DPC to achieve a better accuracy. Therefore, we will tackle the two problems of DPC to satisfy this paper's goal: (1) to search for proper centres if the dataset has clusters that comprise multiple local density peaks and (2) to discover natural clusters that have multiple regional centres with local peak densities.

If clusters contain multiple local density peaks, how to properly choose a centre for each cluster from decision graph can be difficult in DPC. As demonstrated in the dataset containing two single centre clusters as shown in Fig.1(a) it is intuitive to pick up two large red dots as centres in its decision graph (Fig.1(a')). However, the cluster with two regional centres with local peak densities in Fig.1(b) generates nearly identical decision graph (Fig.1(b')), thus the cluster may incorrectly be divided into two in DPC. Meanwhile in some cases, even though the number of clusters is given, it is still difficult to distinguish centres since those candidates are close to each other in the decision graph. One example is shown in Fig.1(c), where the large cluster in the middle contains several local density peaks. Although we know there are five clusters, it is challenging to select five centres for five clusters from the candidate large red dots in Fig.1(c') and requires exhaustively searching. A recursive dividing DPC (3DC) is proposed in [12], but it still uses the heuristic in [14] to pick two greatest $\rho_i \cdot \delta_i$ cluster centres, which is not true in the clusters with multiple density peaks. By applying graph kernel to data vectors before inferring DPC, the candidate centres are differentiated from the common points in [18], however, it still needs to choose cluster centres manually.

Even if we luckily select the right cluster centres, the non-centre local density peaks still can severely decrease the clustering accuracy. The cluster ID propagation rule of DPC is that one non-centre point obtains a cluster ID from its nearest point that has a higher density. If the nearest point with a higher density of one non-centre local density peak belongs to another cluster, this local density peak will

wrongly be assigned to that cluster, and this mistake is propagated to other data points to form incorrect clusters, as shown in the three datasets in Fig.5(b). In the first dataset of Fig.5(b), the ring shape cluster is incorrectly divided into three parts and the left and right parts are assigned to the other two spherical clusters, respectively; in the second dataset, the bottom arch-shape cluster is also divided into two with the top part group with the top arch; in the third dataset, the dot shape cluster is wrongly grouped with the ring shape cluster surrounding it. To resolve the second problem of DPC, a two-step DPC is developed in [19] using the notion of core-reachable of DBSCAN. However, the merging strategy is neither well explained nor proved with convincing experiments.

3 The MDPC Approach

In this section, we present our MDPC approach that effectively finds clusters with multiple local density peaks as regional centres by two steps:

- Find seed clusters. We discover the seed clusters that have single density peaks by allowing automatic selection of cluster centres.
- Merge seed clusters. We define a new distance between seed clusters and hierarchically merge these seed clusters using this distance.

3.1 Find Seed Clusters

Different from DPC, natural shape clusters are regarded as the combination of several seed clusters which contain single density peaks in the scope of MDPC. That is, all of the density peak points are centres of seed clusters. Accordingly, we define a cluster centre/density peak as follows:

$$c : \rho_c \geq \rho_j, dist_{ij} \leq d_c, \forall j \in D. \tag{3}$$

According to Eq. (2), the cluster centres who have peak densities in their d_c radius have δ larger than d_c . Accordingly, we write the list of cluster centres, C, as:

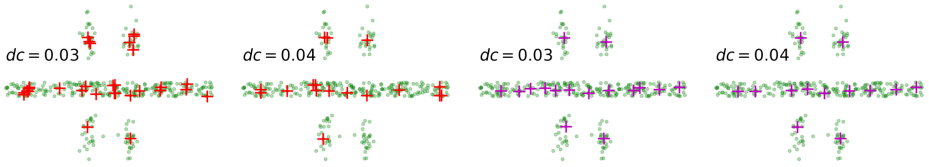
$$C = \{c|\delta_c > d_c, c \in D\}. \tag{4}$$

From the list of cluster centres, we find relevant seed clusters, which comprises only one density peak and satisfies the characteristic of a DPC cluster (that is, data points in a cluster coarsely following a density descending order from centre to border).

In our implementation, the DPC's original definition of density in Eq. (1) performs not good: it either finds the unnecessary centres in a small region or incorrectly labels the non-centralized points as the cluster centres, because it regards each neighbour in d_c region of one point the same weight without considering the distance to the centre. As a result, we modify the fuzzy density defined in [5] as:

$$\rho_i = \Sigma_j \left(1 - \frac{dist_{ij}^2}{d_c^2}\right), dist_{ij} \leq d_c, \forall j \in D. \tag{5}$$

In Fig.2, the cluster centres found by fuzzy density metric (Eq. (5)) are more significant than those found by DPC density (Eq. (1)). When d_c is fixed, cluster centres (cross) found by the fuzzy density (Fig.2(c)-(d)) are sparser than those found by the DPC density (Fig.2(a)-(b)). Meanwhile, the fuzzy density is also more robust to d_c; the number of the cluster centres in Fig.2(a) significantly decreases when d_c increases to 0.04, and the centre of the bottom right cluster in Fig.2(b) is lost, while the cluster centres found using fuzzy density keep stable when increasing d_c from 0.03 (Fig.2(c)) to 0.04 (Fig.2(d)). The pseudocode of finding seed clusters



(a) (b) (c) (d)

Fig. 2: Centres (colored cross) discovered by original density ((a)-(b)), and fuzzy density ((c)-(d)), with d_c as the cut-off distance.

is shown in Algorithm 1. We calculate ρ and δ of each data point at lines 1-7 and discover those cluster centres at line 6. Then we find seed cluster centres and assign cluster IDs to the non-centre data points at lines 8-13.

Algorithm 1 Find Seed Clusters (n_i: nearest point with a higher density)

Input: Dataset D, Cutoff distance d_c
1: **for** each $i \in D$ **do**
2: Calculate ρ_i Eq. (5)
3: **end for**
4: **for** each $i \in D$ **do**
5: Calculate δ_i by Eq. (2), get n_i
6: Add i into C if $\delta_i > d_c$
7: **end for**
8: Sort D in descending order of ρ
9: **for** each $i \in D$ **do**
10: **if** i *in* C **then** $clusterID_i$ = new id
11: **else** $clusterID_i = clusterID_{n_i}$
12: **end if**
13: **end for**
Output: points $clusterID$

3.2 Merge Seed Clusters

Borrowing the idea from hierarchical clustering, we iteratively merge those seed clusters. In our MDPC, merging is conducted only on two clusters which have border regions (mutually have border points whose neighbourhood cross the two clusters). We need to define a new distance metric to achieve an effective merging since the wide-used linkage distances are too coarse for these fine seed clusters. First, two rules are defined to determine whether the seed clusters should be merged based on the characteristic of these seed clusters:

- Rule 1: Two clusters should be merged together if they have comparable scales and spatially close border points, as exhibited in Fig.1(b).
- Rule 2: One small seed cluster should be absorbed by a larger cluster, if its scale is similar to the border region of the larger cluster and have spatially close border regions.

A seed cluster in MDPC has a coarsely monotonic density distribution in a descending order from its centre to border. Based on this characteristic and the above two rules, we define a new cluster distance by both their density distributions and their spatial distance. As a density-based metric, our cluster distance should define the comparable scale in Rule 1 by measuring their difference of density distributions. If the density is monotonic, this difference can be defined on two aspects: the average density and the density-descending rate. Given two seed clusters A and B with average density ρ^A_{avg} and ρ^B_{avg}, the distance between the average densities of A and B is:

$$d_{merge} = \frac{\|\rho^A_{avg} - \rho^B_{avg}\|}{max\left(\rho^A_{avg}, \rho^B_{avg}\right)}. \tag{6}$$

Similarly, in Rule 2, we define the cluster distance to be the difference of the density distributions of the smaller cluster and the border region of the larger one. Assuming A absorbs B ($\rho^A_{avg} > \rho^B_{avg}$), and the border region of A, $border^A$ is defined as $\{i|dist(i,j) < d_c, i \in A, j \in B\}$. Thus ρb^A_{avg}, the average density of $border^A$, is calculated as the average density of points in $border^A$ together with their neighbours. In this way, the distance between ρb^A_{avg} and ρ^B_{avg} is defined as:

$$d_{absorb} = \frac{\|\rho b^A_{avg} - \rho^B_{avg}\|}{max\left(\rho b^A_{avg}, \rho^B_{avg}\right)}. \tag{7}$$

We normalize the above distances and enable them to be combined together and get the overall measurement as $d_{density}$:

$$d_{density} = min\left(d_{merge}, d_{absorb}\right). \tag{8}$$

The distance of the density descending rate and the spatial closeness of the cluster borders are measured by d_{border}, with the density of cluster centre denoted as ρ_c and the density descending rate from the cluster centre to point i as $r_i = \frac{\rho_c}{\rho_i}$.

$$d_{border} = \inf_{i \in border^A, j \in border^B} \frac{d_{ij}}{2d_c} \times \left(r^A_i + r^B_j\right) \|r^A_i - r^B_j\|. \tag{9}$$

We calculate d_{border} to satisfy (1) A and B is spatial close if one point in B falls in the d_c region of A's centre; (2) when one point of B falls in the d_c region of A's border point, note that the border has smaller density and thus their distance must be proportionally smaller to compensate the density difference. Also, the larger the density of a pair of border points (one from A and the other from B), the more likely these two border points are changed into inner points by merging the two clusters. We explain how the new metric can measure the difference of the density descending rates and clusters' spatial closeness as follows. If two clusters are not close to each other, that means the distance of the border points pair d_{ij} is

great then makes d_{border} great. A greater value of $\|r_i^A - r_j^B\|$ means their density descending rates are more different. In addition, we prefer to merge those clusters whose border points have greater densities (i.e., the value of $\left(r_i^A + r_j^B\right)$ is small). Combining the above density distance $S_{density}$ with border distance S_{border} the total distance of two seed clusters is:

$$d_{AB} = d_{border} \times exp\left(d_{density}\right), \tag{10}$$

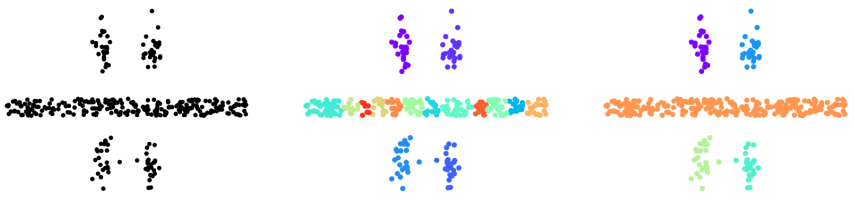
where d_{AB} is non-negative and symmetric, and exponential (exp) is used to ensure that d_{AB} is sensitive to small density variation and prevent merging of two clusters if their densities are globally distinctive even though they are spatially close.

Using the d_{AB} in Eq. (10) to measure the distance of two seed clusters, we develop a hierarchical merging algorithm (Algorithm 2) to form the final clusters with arbitrary numbers of density peaks. We calculate the distances of pairs of seed clusters and sort them in an ascending order at lines 1-7. Then, we merge those with distances under a threshold and re-assign new cluster IDs to all the data points at lines 8-15. Results of each iteration are recorded into a result list at line 17.

Algorithm 2 Hierarchical Merging of Seed Clusters (*dists*: cluster distance list)

Input: Seed clusters C_{seed}
1: **for** seed clusters pair $a, b \in C_{seed}$ **do**
2: **if** a, b has border points **then**
3: calculate d^{ab} with (10)
4: add d^{ab} into *dists*
5: **end if**
6: **end for**
7: Sort *dists* in ascending order;
8: **for** each $d \in dists$ **do**
9: Init. $clusterID_{merge} = \mathbf{0}$;
10: **for** seed cluster pair $a, b \in C_{seed}$ **do**
11: **if** $d^{ij} \leq d$ **then** merge a b
12: **end if**
13: **end for**
14: $clusterID_{merge}$ = merged IDs;
15: add $clusterID_{merge}$ into L;
16: **end for**
Output: Points cluster IDs list L

We use an example dataset [4] in Fig.3(a) to explain how MDPC works. Fig.3(b) are the generated seed clusters; the horizontal cluster in the middle comprises twelve seed clusters (i.e., twelve density peaks). Fig.3(c) shows that the twelve seed clusters are correctly merged into one cluster.



(a) (b) (c)

Fig. 3: (a) is the original dataset, (b) and (c) shows the seed clusters found in step one and the final result in step two, respectively.

3.3 Algorithm Complexity

We denote the number of neighbours of each point as N_{dc} regarding to d_c, the size of border point as N_b, and N_s as the number of seed cluster pairs that mutually have border points, in practice, N_{dc}, N_b and N_s are far less than the dataset size, N. The time complexity of finding seed clusters (Algorithm 1) and border points is $O(2N \log N + N \times N_{dc} + N)$. The time complexity of calculating the average densities of all seed clusters is $O(N)$, and the complexity of calculating all average border densities is $O(N_b)$. $O(N_s + N_s \log N_s)$ is spent to compute and sort distances of seed cluster pairs, and $O(N_s)$ is spent on hierarchical merging. Thus, the overall time complexity of merging seed clusters (Algorithm 2) is $O(N + N_b + 2N_s + N_s \log N_s)$.

4 Experiments

In this section, we use various datasets including both synthetic datasets (four shape datasets and two density datasets) and one real-world dataset to evaluate the accuracy of our MDPC. Accuracy is measured using Normalized Mutual Information (NMI) [16]. All algorithms, including both the proposed MDPC and the counterpart methods (DPC, DBSCAN, K-means, AP and mean-shift), are compared using the best performance under the optimal values of their parameters. We use the heuristic $\rho \times \delta$ in [14] to select the optimal cluster centres for DPC and we automatically search the best NMI of our MDPC from its output list. All the algorithms are implemented in Python 3.2 (with packages scikit-learn, numpy) and experiments are run on Windows 10 with 3.4GHz CPU and 16GB RAM.

4.1 Synthetic Datasets

We use six different synthetic datasets with true labels. Four shape datasets with natural shape clusters (shape datasets) are Pathbased [3], Jain [11], Flame [8] and Spiral [3]. Two datasets with uneven density-pattern clusters (density datasets) are Compound [20] and Aggregation [10]. We evaluate the proposed cluster distance by comparing with four counterpart cluster distances ("single", "average", "complete" and Hausdorff [1]). We set the same experimental environment by replacing our proposed cluster distance with each of the four in MDPC. In Fig.4, we can see that our proposed cluster distance achieves the best accuracy compared to the other four distances in all of the six datasets. The best performance improvement of the proposed distance for MDPC is in Flame, where the accuracy of the proposed method wins the best of the other four distances (Single) by 1.000 to 0.521.

We compare our MDPC with two other density-based methods (DPC and DBSCAN) using three challenging datasets containing natural shape clusters with multiple local density peaks (Pathbased, Jain and Compound) (Fig.5). We can see from the first row that MDPC achieves the best accuracy in the Pathbased dataset, where both DPC and DBSCAN incorrectly split the ring shape cluster into three and two parts, respectively. In Jain dataset as shown in the second row, MDPC correctly separates the two arch-shaped clusters, while DPC incorrectly divides the bottom arch-shaped cluster into two clusters and wrongly assigns the upper

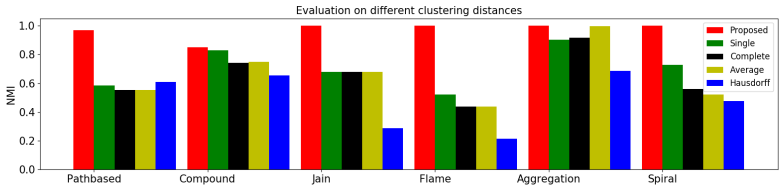
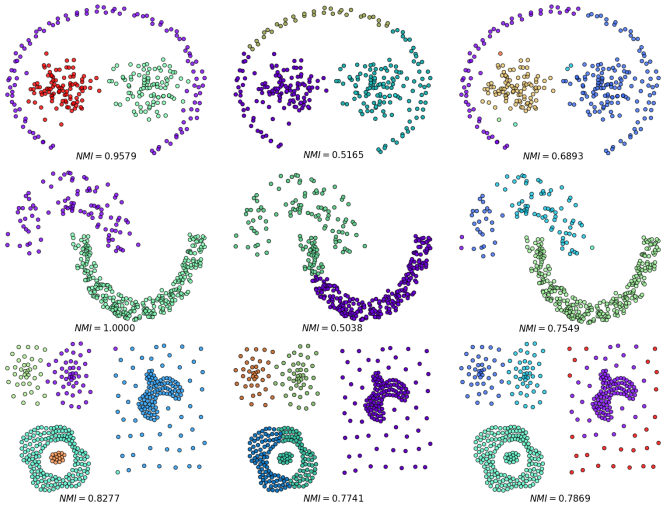


Fig. 4: Evaluation of the cluster distance of MDPC with four cluster distances.

cluster to the other arch-shaped cluster and DBSCAN wrongly splits the top arch-shaped cluster into two parts. In Compound dataset at the last row, only MDPC successfully finds both the two clusters (a ring and a dot the middle of the ring) at the bottom left and two spatial close round-shaped clusters at the top left, while DBSCAN only can correctly identify the two dense clusters at the bottom left or the two close sparse round-shaped clusters but not both since its one cut/threshold strategy cannot adapt itself to the clusters with different density patterns.



(a) MDPC (b) DPC (c) DBSCAN

Fig. 5: (a)-(c) are the clustering results of MDPC, DPC and DBSCAN, respectively.

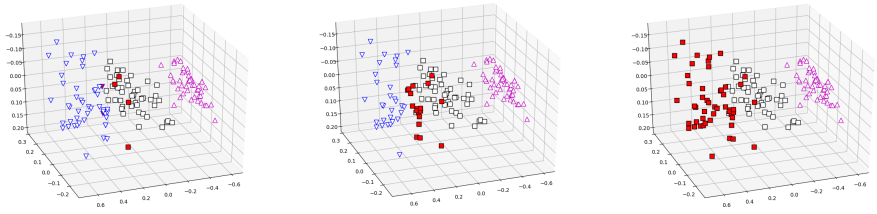
We also evaluate the scalability of MDPC by comparing with both density-based clustering methods (DPC and DBSCAN) and other popular ones (K-means, Affinity Propagation (AP) and mean-shift) using all of the six datasets (Table 1). The overall trend is that the three density-based clustering methods, MDPC, DPC and DBSCAN, perform better than the three non-density-based clustering methods. We can see from Table 5 that MDPC excels both DPC and DBSCAN in all of the six datasets, while DPC performs better than DBSCAN in Flame and Aggregation and DBSCAN performs better than DPC in Pathbased, Jain and Compound. This validates that our MDPC is more adaptive to density sensitive datasets than too coarse clustering in DBSCAN and too fine clustering in DPC.

Table 1: NMI on six synthetic datasets, with the best in bold and negative as '-'.

Type	Dataset	MDPC	DPC	DBSCAN	K-means	AP	mean-shift
Shape	Pathbased	**0.9579**	0.5165	0.6893	0.5102	0.3530	0.5431
	Jain	**1.0000**	0.5038	0.7549	0.3362	0.2073	0.3282
	Flame	**1.0000**	**1.0000**	0.8654	0.4478	0.3011	0.4442
	Spiral	**1.0000**	**1.0000**	**1.0000**	-	0.3142	0.2767
Density	Compound	**0.8277**	0.7482	0.7869	0.7421	0.5289	0.8110
	Aggregation	**1.0000**	**1.0000**	0.9787	0.8376	0.5035	0.8983

4.2 Real Datasets

We use the iris dataset (with four features and three labels) from the real-world dataset UCI [2] to evaluate the accuracy of the three density-based clustering methods (MDPC, DPC and DBSAN). We use PCA to map the original four features into three features for visualization. From the result in Fig.6, we can see that the number of mislabelled items (marked in red color) by MDPC in Fig.6(a) is significantly less than both DPC in Fig.6(b) and DBSCAN in Fig.6(c) where DBSCAN wrongly groups the three categories into only two clusters.



(a) MDPC (b) DPC (c) DBSCAN

Fig. 6: Clustering results for the iris dataset, (a)-(c) correspond to MDPC, DPC and DBSCAN. Incorrectly clustered points are in red color.

5 Conclusion

In this paper, we provide the MDPC clustering method for shape and density sensitive datasets. MDPC overcomes the two problems of DPC by automatically selecting cluster centres and finding natural shape clusters with multiple local density peaks. Extensive experiments based on both synthetic and real-world datasets have demonstrated that our MDPC is a more adaptive clustering method for density sensitive datasets, compared with too coarse clustering in DBSCAN and too fine clustering in DPC, and thus achieves the best accuracy and effectiveness.

Acknowledgement. This work was partially supported by Australia Research Council (ARC) DECRA Project (DE140100387).

References

1. Basalto, N., Bellotti, R., De Carlo, F., Facchi, P., Pantaleo, E., Pascazio, S.: Hausdorff clustering. Phys. Rev. E 78(4), 046112 (2008)
2. Blake, C.L., Merz, C.J.: Uci repository of machine learning databases [http://www. ics. uci. edu/˜ mlearn/mlrepository. html]. irvine, ca: University of california. Department of Information and Computer Science 55 (1998)
3. Chang, H., Yeung, D.Y.: Robust path-based spectral clustering. Pattern Recogn. 41(1), 191–203 (2008)
4. Cho, M., MuLee, K.: Authority-shift clustering: Hierarchical clustering by authority seeking on graphs. In: CVPR. pp. 3193–3200. IEEE (2010)
5. Du, M., Ding, S., Xue, Y.: A robust density peaks clustering algorithm using fuzzy neighborhood. Int. J. Mach. Learn. Cyb. (2017), https://doi.org/10.1007/s13042-017-0636-1
6. Ester, M., Kriegel, H.P., Sander, J., Xu, X., et al.: A density-based algorithm for discovering clusters in large spatial databases with noise. In: SIGKDD. pp. 226–231 (1996)
7. Frey, B.J., Dueck, D.: Clustering by passing messages between data points. Science 315(5814), 972–976 (2007)
8. Fu, L., Medico, E.: Flame, a novel fuzzy clustering method for the analysis of dna microarray data. BMC Bioinformatics 8(1), 3 (2007)
9. Fukunaga, K., Hostetler, L.: The estimation of the gradient of a density function, with applications in pattern recognition. IEEE Trans. Inf. Theory 21(1), 32–40 (1975)
10. Gionis, A., Mannila, H., Tsaparas, P.: Clustering aggregation. ACM Trans. Knowl. Discov. Data 1(1) (2007)
11. Jain, A.K., Law, M.H.: Data clustering: A user's dilemma. Pattern Recognition and Machine Intelligence 3776, 1–10 (2005)
12. Liang, Z., Chen, P.: Delta-density based clustering with a divide-and-conquer strategy: 3dc clustering. Pattern Recogn. Lett. 73, 52–59 (2016)
13. Ray, S., Turi, R.H.: Determination of number of clusters in k-means clustering and application in colour image segmentation. In: ICAPRDT. pp. 137–143. Calcutta, India (1999)
14. Rodriguez, A., Laio, A.: Clustering by fast search and find of density peaks. Science 344(6191), 1492–1496 (2014)
15. Shi, Y., Chen, Z., Qi, Z., Meng, F., Cui, L.: A novel clustering-based image segmentation via density peaks algorithm with mid-level feature. Neural Comput. Appl. 28(1), 29–39 (2017)
16. Strehl, A., Ghosh, J.: Cluster ensembles—a knowledge reuse framework for combining multiple partitions. J. Mach. Learn. Res. 3(Dec), 583–617 (2002)
17. Wang, P., Xu, B., Xu, J., Tian, G., Liu, C.L., Hao, H.: Semantic expansion using word embedding clustering and convolutional neural network for improving short text classification. Neurocomputing 174, 806–814 (2016)
18. Yang, H., Zhao, D., Cao, L., Sun, F.: A precise and robust clustering approach using homophilic degrees of graph kernel. In: PAKDD. pp. 257–270. Springer (2016)
19. Yaohui, L., Zhengming, M., Fang, Y.: Adaptive density peak clustering based on k-nearest neighbors with aggregating strategy. Knowl.-Based Syst. 133, 208–220 (2017)
20. Zahn, C.T.: Graph-theoretical methods for detecting and describing gestalt clusters. IEEE Trans. Comput. 100(1), 68–86 (1971)