Density Gain-Rate Peaks for Spectral Clustering

JIEXING LIU¹ AND CHENGUI ZHAO²

¹Department of Computer Science, The University of Manchester, Manchester M13 9PL, U.K.
²School of Information, Yunnan University of Finance and Economics, Kunming 650221, China

Corresponding author: Chenggui Zhao (zhaochenggui@126.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61562089.

Abstract
Clustering has been troubled by varying shapes of sample distributions, such as line and spiral shapes. Spectral clustering and density peak clustering are two feasible techniques to address this problem, and have attracted much attention from academic community. However, spectral clustering still cannot well handle some shapes of sample distributions in the space of extracted features, and density peak clustering encounters performance problems because it cannot mine the local structures of data and well deal with non-uniform distributions. In order to solve above problems, we propose the density gain-rate peak clustering (DGPC), a new type of density peak clustering method, and then embed it in spectral clustering for performance promotion. Firstly, in order to well handle non-uniform sample distributions, we propose density gain-rate for density peak clustering. Density gain-rate is based on the assumption that the density of a clustering center will be higher with the reduce of the radius. Even under non-uniform distributions, the cluster center in low density region will still have a significant density gain-rate thus can be detected. We combine density gain-rate in density peak clustering to construct DGPC method. Then in the framework of spectral clustering, we use our new density peak clustering to cluster the samples by their extracted features from a similarity graph of these samples, such as the neighbor-based similarity graph or the self-expressiveness similarity graph. Compared with the previous spectral clustering and density peak clustering, our method leads to better clustering performances on varying shapes of sample distributions. The experiment measures the performances of our clustering method and existing clustering methods by NMI and ACC on seven real-world datasets to illustrate the effectiveness of our method.

Index Terms
Density clustering, spectral clustering, density gain-rate, non-spherical, non-uniform.

I. INTRODUCTION
Clustering is widely used to extract potentially useful information in unsupervised learning environment. It aims to group samples in such a way that the samples within the same group exhibit a higher similarity than ones in different groups [14], [15], [30]. Traditional clustering methods, such as k-means and k-medoids [24], [28], just well cluster the samples with spherical distributions [36]. They often fail to deal with nonspherical sample distributions.

Spectral clustering (SC) is a significant way to improve performance of the traditional clustering methods on varying shapes of sample distributions [1]. A group of SC methods have been proposed in recent years [11], [38], [40]. Firstly they use a similarity graph to extract features of the samples, then use k-means to cluster these samples in the space of the extracted features [4]. In certain cases, the clustered samples with nonspherical distributions will become spherical distributions by mining the local structures of these samples in feature extraction, thus k-means can well cluster these samples. However in real-world datasets, samples with nonspherical distributions still cannot change to spherical distributions by the feature extraction [26], [47].

Density peak clustering (DPC) emerges as another essential way to handle the problem of varying shapes of sample distributions [36]. DPC can recognize the clusters regardless of their shapes of sample distributions, as it clusters the samples by their densities, [7], [12]. Recently, a variety of approaches based on DPC method also succeed in recognizing clusters under varying shapes of sample distributions. Jiang et al. proposed GDPC method [23], this method uses gravitation theory and nearby distance to identify centers of the clusters and outliers more accurately than DPC with an alternative decision graph. Du et al. [10] contributed a density peaks clustering based on k-nearest neighbors and PCA. This method provides another option...
for the local density computation by introducing the idea of $k$ nearest neighbors (KNN) into DPC. Xu et al. proposed Denpehc method. It directly generates clusters on each possible clustering layer and uses a grid granulation framework to cluster large-scale and high-dimensional datasets [44]. Xie et al. proposed FKNN-DPC method which computes the local density of the sample relative to its $K$-nearest neighbors for any size dataset independent of the cutoff distance, and then assigns the remaining points to the most probable clusters by the fuzzy weighted K-nearest neighbors [43]. Wu et al. proposed DBG method, applying only distances between a much smaller number of grid nodes to cluster the samples rapidly Instead of calculating Euclidean distances between all the samples in dataset. [42].

Unfortunately, algorithms based on DPC, namely, density peak clustering and its variants, encounter the problem of performance. To explain this, DPC cannot well capture the local structures of the data under non-uniform distributions. Some researchers propose DP-SC [26]. This method uses DPC to cluster the samples after feature extraction in SC. Though DP-SC can better mine the local structure of samples than DPC, it still cannot well handle the non-uniform sample distributions in the space of extracted features.

In order to solve above problem, we propose density gain-rate peak clustering (DGPC) which is enlightened from information gain-ratio, and we embed DGPC in spectral clustering then [35]. It is based on the assumption that the density of a clustering center will be higher with the reduce of the radius. Firstly, for each sample, DGPC computes its density gain-rate which can present the rate of gain in its density with the reduce of the radius. Thus even under the non-uniform distributions, the cluster center of a low density region will have a significant density gain-rate like the cluster center in high density region. Then DGPC clusters the samples in data with following three assumptions listed below.

- A cluster center should have a higher density gain-rate than that of its neighbors.
- There should be a long distance between a cluster center and any point with a higher density gain-rate than that of this center.
- Each sample point, besides the cluster centers, should be in the same cluster as its nearest neighbors with the higher density gain-rates in the sample points.

Based on DGPC, we propose our two new DGPC-SC method which embed DGPC into SC with a neighbor-based similarity graph and a self-expressiveness similarity graph. Compared with the previous SC methods and DPC methods, our DGPC-SC method can get the better cluster of the samples under nonspherical and non-uniform distributions.

Fig. 1 as example illustrates the advantages of using DGPC to cluster non-uniform samples by extracted features in SC. Fig. 1 (a) shows the clustering unbalance-spiral dataset where sample distributions are spiral shapes. The different colors of the samples in Fig. 1 (a) indicate the different classes. The red samples in this dataset are much more than blue ones, which simulates the case of non-uniform samples distribution. The clustering results of this dataset with different clustering methods ($k$-means, DPC, SC, DP-SC, DGPC-SC) are shown in Fig. 1 (b) - (f). For making a fair comparison, the SC, DP-SC and DGPC-SC use the same similarity graph (e.g., $K$-NN graph, which is agile to build and well-known) to extract features. As we can see from Fig. 1 (b) - (f),

![FIGURE 1. The clustering results of different methods on unbalance-spiral dataset.](image)
k-means hardly cluster this dataset well due to the fact that it fail to handle nonspherical sample distributions effectively. DPC encounters problem on the non-uniform samples, so even most of the sample distributions obey its assumptions, it still cannot well cluster this dataset. SC and DP-SC can get better clustering results than k-means and DPC, but the performances of them are still not excellent enough. DGPC-SC generates an improved clustering performance on this dataset.

The key contributions of this paper are summarized as follows. Firstly, we propose density gain-rate to mine the structure of data under non-uniform distributions. Secondly, we combine density gain-rate into DPC method to construct our DGPC method. Finally, based on DGPC we propose a new SC method, DGPC-SC.

The rest of this paper is as follows: Section 2 reviews DPC and two SC methods including Ncut-SC and \(\ell_2\)-SC. In Section 3, we present our DGPC. We also propose two new SC methods, Ncut-DGPC and \(\ell_2\)-DGPC, by embedding DGPC into SC with the neighbor-based similarity graph and the self-expressiveness similarity graph, respectively. In Section 4, we connect our DGPC with related works including DPC and DENCLUE. Section 5 shows the experimental results of our clustering methods. We conclude this paper in Section 6.

II. PRELIMINARIES
Our work is based on DPC and two SC methods, therefore we review them in this section. For convenience, we list the notations in Table 1.

| Notation | Representation |
|----------|----------------|
| \(n\)    | The number of the samples |
| \(d\)    | The number of features |
| \(A^T\)  | The transpose of matrix A |
| \(a_i\)  | The \(i\)th row of matrix A |
| \(a_{i,j}\) | The \(i\)th column of matrix A |
| \(a_{*,j}\) | The element in the \(i\)th row of the \(j\)th column of matrix A |
| \(C_j\)  | The \(j\)th cluster of all the samples |
| \([*]\)  | The ceiling function |
| \(1\)    | The column vector with all the elements is 1 |

A. DPC METHODS
DPC is able to recognize clusters regardless of their shapes [36]. It is based on the following three assumptions.

- A cluster center should have a higher local density than that of its neighbors.
- There should be a long distance between a cluster center and any point with a higher local density than that of this center.
- Each sample point, except the cluster centers, should be in the same cluster as its nearest neighbor with the higher local densities in the sample points.

DPC method firstly defines two variables, \(\rho_i\) and \(\delta_i\), for each \(x_i\). \(\rho_i\) is the local density of \(x_i\) and defined as

\[
\rho_i = \sum_j \chi(d_{i,j} - d_c),
\]

where \(\chi(*)\) is the sign function defined as

\[
\chi(*) = \begin{cases} 
1 & \text{if } d \leq 0, \\
0 & \text{otherwise}. 
\end{cases}
\]

\(d_c\) is a cutoff distance and \(d_{i,j}\) is the Euclidean distance between \(x_i\) and \(x_j\). A commonly used strategy to tune the \(d_c\) is as follow [36]:

\[
d_c = d_{[n^2 \times p/100]},
\]

where \(p \in [0, 100]\), \(d_{[n^2 \times p/100]} \in \{d_1, d_2, \ldots, d_n\}\) indicate the Euclidean distances between all pairs of samples and are sorted in ascending order.

\(\delta_i\) is the minimum value of the Euclidean distances between \(x_i\) and all the other points with the higher local densities than \(x_i\).

\[
\delta_i = \min_{j, \rho_j > \rho_i} \{d_{i,j}\} \text{ if } \exists j, \text{ s.t. } \rho_j > \rho_i.
\]

According to DPC, each of the cluster centers should be with a high \(\rho_i\) and a large \(\delta_i\). Based on this guideline, there are two ways to select the cluster centers. One is that we can use a decision graph to manually select the cluster centers. This decision graph has \(\rho\) and \(\delta\) as its axes and provides a visualization of what sample points can be selected as cluster centers. The other is that we can select those points whose \(\gamma_i\) are large as the cluster centers [36], [43], where

\[
\gamma_i = \rho_i \times \delta_i.
\]

After the cluster centers are selected, DPC method assigns each sample point \(x_i\), except the selected cluster centers, to the same cluster of its nearest neighbor in the sample points with the higher local densities than that of \(x_i\).

Recently, a variety of approaches based on DPC method also succeed in recognizing clusters under varying shapes of sample distributions. Jiang et al. have proposed GDPC method, this method uses gravitation theory and nearby distance to identify centers of the clusters and outliers more accurately than DPC with an alternative decision graph [23]. Xu et al. proposed Denpehc method. It directly generates clusters on each possible clustering layer and uses a grid granulation framework to cluster large-scale and high-dimensional datasets [44]. Xie et al. proposed FKNN-DPC method which computes the local density of the sample relative to its \(K\)-nearest neighbors for any size dataset independent of the cutoff distance, and then assigns the remaining points to the most probable clusters by the fuzzy weighted \(K\)-nearest neighbors [43]. Wu et al. proposed DBG method, applying only distances between a much smaller number of grid nodes to cluster the samples rapidly Instead of calculating Euclidean distances between all the samples in dataset [42].

B. SC METHODS
In an SC framework, one extracts features by a similarity graph of the samples, and then uses k-means to cluster.
the samples in the space of the extracted features [6], [9]. Generally speaking, two kinds of similarity graphs are often used in SC methods: the neighbor-based and the self-expressiveness [3], [11], [33].

Ncut-SC is a representative SC method, which is based on the neighbor-based similarity graph [38]. Given a sample matrix \( X = [x_1; x_2; \ldots; x_n] \), this method firstly constructs a neighbor-based similarity graph associated with a weighted adjacency matrix \( W \). Each element \( w_{ij} \) in \( W \) is defined as follows [34].

\[
w_{ij} = \begin{cases} 
    e^{-\frac{d_{ij}^2}{\sigma^2}} & x_i \in N_j \text{ or } x_j \in N_i, \\
    0 & \text{otherwise,}
\end{cases} 
\]  

(6)

where \( N_i \) is the set of the nearest neighbors of \( x_i \) and \( \sigma \) is a free parameter. Then the features that well represent the manifold of this graph can be extracted by the following optimization problem:

\[
y = \arg \min_{y^Ty=1} \frac{1}{2} \sum_{i,j} (y_i - y_j)^2 w_{ij},
\]

(7)

where \( D \) is a diagonal matrix with the \( i \)th diagonal element decided by \( d_{ii} = \sum_j w_{ij} \). The optimal solution \( y \) to this problem is the normalized eigenvector corresponding to the smallest nonzero eigenvalue of \( D^{-1}L \) with the normalization of \( y^Ty = 1 \), where \( L = D - W \) is a Laplacian matrix [38].

Although, this eigenvector can well represent a part of manifold structures in the above graph, we cannot get the ideal cluster performance by only this eigenvector. As the suboptimal solutions to (7), which correspond to a few small eigenvalues, some of the next normalized eigenvectors also contain useful partitioning information. thus Ncut-SC also selects all above normalized eigenvectors as the extracted features.

As another crucial SC method, \( \ell_2 \)-SC is based on the self-expressiveness similarity graph [33]. This method firstly constructs a regression coefficient matrix \( G \) where each column vector \( g_i \) in \( G \) is decided by solving optimization problem:

\[
\min_{g_i} \frac{1}{2} ||x_i - g_iX||^2 + \lambda ||g_i||^2 \quad \text{s.t.} \quad g_i e_i^T = 0,
\]

(8)

where \( \lambda \) is a free parameter and \( e_i \) is the standard orthogonal basis of \( R^n \). The optimal solution of the above formulation is given by

\[
g_i = [x_iX^T - e_i(x_iQ_e)^T]P,
\]

(9)

where \( X \) is the sample matrix, \( P = (XX^T + \lambda I) \) and \( Q = X^TP \).

After the samples are projected into the linear space, \( G \) handles the errors by performing a hard thresholding operator \( H_k(\ast) \) over \( g_i \), where \( H_k(\ast) \) keeps the \( k \) largest elements in \( g_i \) and zeroizes the others [33]. Then \( \ell_2 \)-SC computes the weighted adjacency matrix \( W = G^T + G \) and uses this matrix to compute the Laplacian matrix \( L \), like Ncut-SC. Finally, this method selects the normalized eigenvectors corresponding to a few small nonzero eigenvalues of \( L \) as the extracted features.

The above SC methods can be combined in a one-after-the-other way. To take advantages of multiple clustering methods, another strategy is to exploit the ensemble clustering technique [41]. Recently, there are many effective ensemble clustering methods have been proposed. Huang et.al proposed a novel ensemble clustering approach named robust ensemble clustering using probability trajectories. The method is developed from sparse graph representation and probability trajectory analysis. The researchers grew the elite neighbor selection strategy for the identification of uncertain links using locally adaptive thresholds, and they establish a sparse graph with a small number of probably reliable links [18]. Furthermore, Huang et.al developed locally weighted ensemble clustering based on ensemble-driven cluster uncertainty estimation and local weighting strategy. They considered the cluster labels in the entire ensemble by an entropic criterion for the estimation of the uncertainty of each cluster [19]. Besides, Huang et.al presented enhanced ensemble clustering via fast propagation of cluster-wise similarities. The experts firstly built a cluster similarity graph which would be utilized to generate a new cluster-wise similarity matrix. Then, enhanced co-association matrix would be achieved by mapping the former matrix from the cluster-level to the object-level [20].

III. DGPC AND DGPC-SC

We firstly propose DGPC method to better deal with non-uniform sample distribution. Then, we construct our two DGPC-SC methods through embedding DGPC into SC with the neighbor-based similarity graph and the self-expressiveness similarity graph, respectively. Compared with previous SC methods, our two DGPC-SC algorithms can better cluster the samples under non-spherical and non-uniform distributions.

A. DENSITY GAIN-RATE PEAK CLUSTERING

In order to solve the problem of non-uniform distribution for DPC, we define the gain rate of density. It is based on the assumption that with the reduce of the radius, the density of a clustering center will be higher. Firstly, we define the density of \( x_i \) as

\[
\rho_i = \frac{\sum_{j} x_i (d_{ij} - d_c)}{d_c^2} - \rho_i
d_i\delta
\]

(10)

Based on this renewed density, we define the density gain of \( x_i \), \( G_i \), from a large radius \( d_q \) to a small radius \( d_l \) as

\[
G_i = \rho_i^{d_l} - \rho_i^{d_q}
\]

(11)

where \( d_l < d_q \). Then the gain rate of density of \( x_i \) is define as

\[
GR_i = \frac{G_i}{\rho_i^{d_q}}
\]

(12)
In DGPC, we select those sample points whose $\gamma'_i$ are large as the cluster centers, where

$$
\gamma'_i = GR(i) \times \delta'_i, \quad (13)
$$

where

$$
\delta'_i = \begin{cases} 
\min\{GR_j : GR_j > GR_i\} & \text{if } \exists j, \ s.t. \ GR_j > GR_i, \\
\max\{d_{i,j}\} & \text{otherwise.}
\end{cases} \quad (14)
$$

Then, we assign each sample point $x_i$, except the selected cluster centers, to the same cluster of its nearest neighbor with a higher density gain-rate than that of $x_i$. We list the main algorithmic steps of DGPC in Algorithm 1.

There is an example to show the advantage of our density gain-ratio on a synthetic dataset in Fig 2. As we can see, the two clusters in this dataset have non-uniform distributions. DPC cannot accurately cluster the samples in these two clusters. However, DGPC can cluster these samples accurately with the assistance of density gain-rate. Like DPC, DGPC also can discover the outliers in data. The difference is that DPC regards the samples with low densities as the outliers, but DGPC regards the samples with low GRs as the outliers. The result of discovering the outliers in our above synthetic dataset of DGPC is shown in Fig. 3. The black points are the outliers discovered by DGPC. From the clustering result, we can see that DGPC can well discover the outliers in data.

Although the time complexity of DGPC is $O(n^2d)$ which is same with time complexity of DPC, the time complexity of DGPC-SC highly depends on the corresponding SC method. Unfortunately, because of the time complexity, SC methods including our DGPC-SC are hard to deal with big data.
of distance, an effective kernel KROD is used to measure the ods, Ncut [38] and ℓ and nonspherical sample distributions in this representation, considering that DPC cannot well deal with non-uniform sam-
ple distributions perfectly. SC-DPC method uses DPC to cluster the samples in this representation, while
Most of SC methods use non-uniform samples distributions in this representation. However, there still are nonspherical sample distributions and
low-dimensional representation for the original data. How-
As we can see from Fig. 4 (a) and Fig. 4 (c), even in the space of the extracted features, there are still many sample under non-spherical and non-uniform distributions. Ncut fail to handle these distributions excellently, and to recognize the cluster with the smallest number of the samples (the cluster marked by color brown). But Neut-DGPC not only can better cluster the these samples. From this example and the example of the above subsection, the reason, behind Neut-DGPC can get a better cluster result, is that it uses DGPC to cluster the samples by the extracted features.

B. DGPC-SC METHODS

In SC, the spectral embedding can serve as a better and low-dimensional representation for the original data. However, there still are nonspherical sample distributions and non-uniform samples distributions in this representation. Most of SC methods use k-means to cluster the samples in this representation, while k-means owns no ability of solving nonspherical sample distributions perfectly. SC-DPC method uses DPC to cluster the samples in this representation, considering that DPC cannot well deal with non-uniform samples distributions, we use DGPC to deal with non-uniform and nonspherical sample distributions in this representation, which leads to the better clustering result.

Based on our framework, we improve two typical SC methods, Ncut [38] and ℓ2-SC [33], then get our two DGPC-SC methods: Ncut-DGPC and ℓ2-DGPC. Furthermore, instead of distance, an effective kernel KROD is used to measure the differences between the samples in our methods [22]. The two new algorithms are present in Algorithm 2 and Algorithm 3.

Fig. 4 is an example to illustrate the advantages of using DGPC in SC. Fig. 4 (a) and (c) show Lung dataset with the respective representations of the Euclidean distance and KROD in the space of the extracted features. Fig. 4 (b) and (d) show the respective clustering results of Ncut and Ncut-DGPC.

IV. CONNECTED TO RELATED WORKS

In this section, we theoretically analyze the DGPC with DPC and DENCLUE to show the advantages of DGPC [17], [36]. In DENCLUE the density gradient of x with respect to a sample set D = \{x_1, x_2, \ldots, x_t\} is defined as

$$
\nabla f^D_B(x) = \sum_{i=1}^{k} (x_i - x) f_B(x, x_i)
$$

(15)

where \(f_B(x, x_i)\) is the basic influence function of sample \(x_i\) on \(x\). Generally speaking \(f_B(x, x) > 0\). Then DENCLUE computes density-attractor of the sample. A sample \(x\) is density attracted to a \(x^*\), if and only if \(\exists k, d(x^k, x^*) \leq \epsilon \) with

$$
x^0 = 0, x^i = x^{i-1} + \delta \frac{\nabla f^D_B(x^{i-1})}{||\nabla f^D_B(x^{i-1})||}
$$

(16)

where \(\epsilon\) and \(\delta\) are two parameters. The calculation stops at \(k\), if \(f^D_B(x^{k+i}) < f^D_B(x^k)\) and takes \(x^* = x^k\) as a new density-attractor. Finally DENCLUE clusters the samples under non-spherical distributions by the following definition.
**Definition 1:** A arbitrary-shape of the set density-attractors $X$ is a subset $C \in D$, 1. $\forall x \in C$, $\exists x^*$ such that $x$ is density-attracted to $x^*$ in $X$. 2. $\forall x_i, x_j \in X$: $\exists$ a path $P$ from $x_i$ to $x_j$ with $\forall p \in P : f_B^D(p) > \zeta$ However, it is hard to tune the parameters $\epsilon, \delta$ and $\zeta$ for DENCLUE. DPC can be seen as a special case of DENCLUE in the clustering process without tuning these parameters.

**Proposition 1:** DPC is a special case of DENCLUE in the clustering process. In this case $D = \{y\}$, where $y$ is the nearest neighbor of $x$ with the higher local density.

It is easy to see, in this case, the density gradient of $x$ will be same with $(y - x)$. Thus $x$ will be clustered with $y$ without tuning any parameters. The disadvantage of DPC is not robust for non-uniform sample distributions. The samples in low density regions will be easily clustered to the high density regions. DGPC improves this problem with only one more parameter, the larger radius $d_q$. We also prove that DPC is a special case of DGPC.

**Proposition 2:** DPC is a special case of DGPC when $d_q > \max\{d_1, d_2, \ldots, d_n\}$.

**Proof:** When $d_q > \max\{d_1, d_2, \ldots, d_n\}$, the density of each sample $x_i$, $\rho_{d_i}$, is equal to $N$. In this case, the density gain of each $x_i$ can be computed as

$$G_i = \rho_{d_i}^t - \rho_{d_i}^c = \rho_{d_i}^t - N$$

Thus the density gain-rate of $x_i$ is

$$GR_i = \frac{G_i}{\rho_i^{d_q}} = \frac{\rho_{d_i}^t}{N} - 1$$

The $GR_i$ above has linear relationship with $\rho_{d_i}^t$. Therefore, in the case of $d_q > \max\{d_1, d_2, \ldots, d_n\}$, DPC will has the same clustering results as DGPC after selecting clustering centers.

In practice, DPC selects cluster centers by the large $\gamma_i$ in (5). For any two non-uniform distribution clusters $C_1$ and $C_2$, if $\exists x_i \in C_1$ makes $\rho_i \times \delta_i > \rho_{c_2} \times \delta_{c_2}$. (19) where $c_2$ is the center of $C_2$. DPC will incorrectly select the cluster center of $C_2$, leading to the inaccurate clustering result, as shown in Fig. (2) (a). In this process, DPC directly compares weighted $\rho_i$ and weighted $\rho_{c_2}$, thus ignores the local structure of $c_2$. Our DGPC mines the local structure that even in sparse cluster, with the reduce of the radius, the density of a clustering center will be higher by GR, thus get the correct clustering result, as shown in Fig. (2) (b).

As described above, DGPC is a extend of DPC and can better mine the cluster in the local structure under the non-uniform sample distributions. Compared with DENCLUE, DGPC needs not to tune the complex parameters.

**V. EXPERIMENT AND ANALYSIS**

In this section, we introduce the experimental settings in Subsection 4.1. Experimental results of different clustering methods are shown in Subsection 4.2. Finally, some visual results of different methods on ORL dataset are shown in Subsection 4.3 [37].
A. EXPERIMENTAL SETTINGS

In our experiment, we use eight public datasets to test the performance of our clustering methods. These datasets include Iris, Wine, Isolet [2], Lung [39], ORL [37], Jaffe [27], UMist [13], COIL20 [29], and Mnist-test [8]. The experimental results of Lung dataset are shown in the examples of Section 3 to illustrate the motivation of this work. The experimental results of the other datasets are shown in this section. All of the above datasets are described in Table 2.

**TABLE 2. Data description.**

| DID | Dataset | # Instances | # Features | # Classes |
|-----|---------|-------------|------------|-----------|
| 1   | Lung    | 203         | 3312       | 5         |
| 2   | Iris    | 150         | 4          | 3         |
| 3   | Wine    | 178         | 13         | 3         |
| 4   | Isolet  | 1560        | 617        | 26        |
| 5   | ORL     | 400         | 10304      | 40        |
| 6   | Jaffe   | 213         | 65536      | 10        |
| 7   | UMist   | 575         | 10304      | 20        |
| 8   | COIL20  | 1440        | 1024       | 20        |

To verify the effectiveness of our clustering methods, we compare ours with eleven clustering methods listed below.

1) **k-means.** It works by computing centres of the different clusters iteratively [28].
2) **NMF-LP.** It extracts features of samples by non-negative matrix factorization then uses k-means to cluster these samples [5].
3) **N-Cuts.** It extracts features of samples by non-negative matrix factorization then uses k-means to cluster these samples [5].
4) **SCC.** It is a famous subspace SC method with ℓ₁-minimization and can well handle samples near the intersections of subspaces [11].
5) **DPC.** It works by combining the sparse samples into the dense samples [36].
6) **GDPC.** It uses gravitation theory to improve DPC [23].
7) **PK-DPC.** It improves DPC based on K-nearest neighbors and PCA [10].
8) **DP-SC.** It uses DPC to cluster the samples with extracted features in SC [26].
9) **U-SENC.** It works by interpreting the sparse sub-matrix as a bipartite graph then transfer cut is utilized to efficiently partition the graph [21].
10) **CLR.** It constrains Laplacian Rank method to learn a graph and clusters samples on this graph with ℓ₁-norm or ℓ₂-norm constraint [31].
11) **ℓ₂-SC.** It uses ℓ₂-graph to extract features in SC [33].

We use two clustering evaluation metrics: clustering accuracy (ACC) and normalized mutual information (NMI) to measure the clustering performance. Denote the clustering results with \( B = \{b_1, b_2, \ldots, b_n\} \) and the truth label of \( X \) with \( A = \{a_1, a_2, \ldots, a_n\} \) respectively. ACC is defined as [46]

\[
ACC(B, A) = \frac{1}{n} \sum_{i=1}^{n} E(a_i, map(b_i)).
\]  

(20)

If \( a == b \), \( E(a, b) = 1 \); otherwise, \( E(a, b) = 0 \). \( map(b_i) \) is the best mapping function that permutes clustering labels to match the given truth labels using the Kuhn-Munkres algorithm [25], [32]. The larger ACC leads to a better clustering result. The NMI is defined as [16]

\[
NMI(B, A) = \frac{MI(B, A)}{\sqrt{(H(B)H(A))}},
\]  

(21)

where \( H(B) \) and \( H(A) \) are the entropies of distribution \( B \) and \( A \). \( MI(B, A) \) is the mutual information metric of \( B \) and \( A \). The larger NMI is, the better clustering result is. For the original features from a dataset and the extracted features of our graphs, we use ℓ₂-norm to normalize them since it empirically improves the clustering performances [33], [45].

B. PERFORMANCE COMPARISON

Firstly, we compare our SC methods with eleven clustering methods for demonstrating the advantages of our SC methods. There are some hyper-parameters that need to be set in our methods. \( k \) for k-nearest neighbors are set from \{4, 5, 10, 15, 20, 25\}, \( p \) is set from \{2, 2.5, 3\}, \( \lambda \) is set from \{0.05, 0.5, 5, 50, 500\}, \( d_1 \) is set by \( t \) and \( d_2 \) is set by \( 2 \times t \).

Furthermore, we combine the Gaussian kernel into the process of computing density peaks, like the original DPC [36]. In Table 3, we show the NMIs of the different clustering methods. Then we show the ACCs in Table 4. From Table 3 and Table 4, we can see that, in most of the cases, our clustering methods outperform the previous methods. Interestingly, we achieve perfect result (NMI = 1 and ACC = 1) on Jaffe dataset, which means that all samples in the same class are clustered into the same group. For SC methods, we can see that compared with k-means or DPC, DGPC can get the better clustering results for SC. Then, in order to demonstrate the advantages of DGPC, we compare DPC, GDPC and DGPC in Fig. 5 and Fig. 6. For a fair comparison,
we just tune $p$, which is the parameter for DPC GDPC and DGPC. As we can see, DGPC performs better than GDPC and DPC.

C. THE VISUAL RESULTS
As did in many other DPC-based methods [36], [43], we show the visual clustering results on ORL-20 dataset, which is composed of the 200 images of the first 20 subjects in ORL dataset.

Fig. 7 shows the visual results of the different methods on ORL-20. The different back ground colors index the different clusters. As we can see, DGPC performs much better than DPC. Ncut-DGPC and $\ell_2$-DGPC get the best clustering
results in Fig. 7. Especially, $\ell_2$-DGPC clusters the samples of ORL-20 with a mere three mistakes.

VI. CONCLUSION
We firstly propose DGPC. It uses density gain-rate to find the cluster centers in low density regions. Compared with the previous DPC methods, DGPC can better handle non-uniform samples in non-spherical distributions. Then, we improve two typical SC methods, Neut and $\ell_2$-SC, then get our two DGPC-SC methods: Ncut-DGPC and $\ell_2$-DGPC. Compared with the previous clustering methods, new methods achieve the better performances on real-world datasets.

In the future work, we can explore how to automatically recognize the cluster number and the cluster centers by DGPC in SC methods. Furthermore, as mentioned above, because of the time complexity of DGPC, improving the efficiency of the SC methods can be another future work.

ACKNOWLEDGMENT
Jiexing Liu thanks Professor Chenggui Zhao for guidance and advice in scientific research during January 2020 to December 2020, when Jiexing Liu is pursuing the master’s degree with The University of Manchester, and staying in China due to COVID-19 Pandemic.

REFERENCES
[1] E. Arias-Castro, G. Lerman, and T. Zhang, “Spectral clustering based on local PCA,” J. Mach. Learn. Res., vol. 18, no. 9, pp. 1–57, 2017.
[2] A. Asuncion and D. Newman, “UCI machine learning repository,” School Inf. Comput. Sci., Univ. California Irvine, Irvine, CA, USA, Tech. Rep., 2007.
[3] M. Belkin and P. Niyogi, “Laplacian eigenmaps and spectral techniques for embedding and clustering,” in Proc. Adv. Neural Inf. Process. Syst., 2002, pp. 585–591.
[4] Y. Bengio, J.-F. Paiement, P. Vincent, O. Delalleau, N. L. Roux, and M. Ouimet, “Out-of-sample extensions for LLE, isomap, MDS, eigenmaps, and spectral clustering,” in Proc. Adv. Neural Inf. Process. Syst., 2004, pp. 177–184.
[5] D. Cai, X. He, X. Wang, H. Bao, and J. Han, “Locality preserving non-negative matrix factorization,” in Proc. IJCAI, vol. 9, 2009, pp. 1010–1015.
[6] W.-Y. Chen, Y. Song, H. Bai, C.-J. Lin, and E. Y. Chang, “Parallel spectral clustering in distributed systems,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 33, no. 3, pp. 568–586, Mar. 2011.
[7] Y. Chen, X. Hu, W. Fan, L. Shen, Z. Zhang, X. Liu, J. Du, H. Li, Y. Chen, and H. Li, “Fast density peak clustering for large scale data based on kNN,” Knowl-Based Syst., vol. 187, Jan. 2020, Art. no. 104824.
[8] L. Deng, “The MNIST database of handwritten digit images for machine learning research [best of the Web],” IEEE Signal Process. Mag., vol. 29, no. 6, pp. 141–142, Nov. 2012.
[9] C. Ding, X. He, and H. D. Simon, “On the equivalence of nonnegative matrix factorization and spectral clustering,” in Proc. SIAM Int. Conf. Data Mining, Apr. 2005, pp. 606–610.
[10] M. Du, S. Ding, and H. Jia, “Study on density peaks clustering based on K-nearest neighbors and principal component analysis,” Knowl-Based Syst., vol. 99, pp. 135–145, May 2016.
[11] E. Elhamifar and R. Vidal, “Sparse subspace clustering: Algorithm, theory, and applications,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 35, no. 11, pp. 2765–2781, Nov. 2013.
[12] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, “A density-based algorithm for discovering clusters in large spatial databases with noise,” in Proc. KDD, vol. 96, 1996, pp. 226–231.
[13] D. B. Graham and N. M. Allinson, “Characterising virtual eigensignatures for general purpose face recognition,” in Face Recognition. Berlin, Germany: Springer, 1998, pp. 446–456.
G. Sun, Y. Cong, Q. Wang, J. Li, and Y. Fu, “Lifelong spectral clustering,” in Proc. AAAI, 2020, pp. 5867–5874.

S. Vega-Pons and J. Ruiz-Shulcloper, “A survey of clustering ensemble algorithms,” Int. J. Pattern Recognit. Artif. Intell., vol. 25, no. 3, pp. 337–372, May 2011.

B. Wu and B. M. Wilamowski, “A fast density and grid based clustering method for data with arbitrary shapes and noise,” IEEE Trans. Ind. Informat., vol. 13, no. 4, pp. 1620–1628, Aug. 2017.

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,” Inf. Sci., vol. 354, pp. 19–40, Aug. 2016.

J. Xu, G. Wang, and W. Deng, “DenPEHC: Density peak based efficient hierarchical clustering,” Inf. Sci., vol. 373, pp. 200–218, Dec. 2016.

J. Yang, D. Parikh, and D. Batra, “Joint unsupervised learning of deep representations and image clusters,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 5147–5156.

Y. Yang, D. Xu, F. Nie, S. Yan, and Y. Zhuang, “Image clustering using local discriminant models and global integration,” IEEE Trans. Image Process., vol. 19, no. 10, pp. 2761–2773, Oct. 2010.

W. Zhang, X. Wang, D. Zhao, and X. Tang, “Graph degree linkage: Agglomerative clustering on a directed graph,” in Computer Vision—ECCV 2012. Berlin, Germany: Springer, 2012, pp. 428–441.

---

JIEXING LIU received the bachelor’s degree in computer science from the Beijing University of Technology, He is currently pursuing the master’s degree with the Department of Computer Science, The University of Manchester, Manchester, U.K. His current research interests include machine learning and data science.

CHENGGUI ZHAO was born in Qujing, Yunnan, China, in 1974. He received the bachelor’s degree in mathematics and the master’s degree in computer science from Yunnan Normal University, China, in 1999 and 2003, respectively, and the Ph.D. degree in computer science from the South China University of Technology, China, in 2007. From September 2016 to September 2017, he worked as a Visiting Academic Researcher with the Department of Electrical and Computer Engineering, University of California at Santa Barbara. From July 2018 to August 2018, he also experienced a short-term academic visiting with the Wireless Networking and System Laboratory, Department of Electrical and Computer Engineering, State University of New York at Stony Brook. He is currently a Full Professor with the School of Information, Yunnan University of Finance and Economics, China. His research interests include communication network and computer system architecture, studied via the application of algebra, graph, and probability theories. He is also involved in the areas of data and social network analysis.

---

* * *