Automatic Clustering via Outward Statistical Testing on Density Metrics

Guangtao Wang, Qinbao Song

Abstract—Clustering is one of the research hotspots in the field of data mining and has extensive applications in practice. Recently, Rodriguez and Laio [1] published a clustering algorithm on Science that identifies the clustering centers in an intuitive way and clusters objects efficiently and effectively. However, the algorithm is sensitive to a preassigned parameter and suffers from the identification of the "ideal" number of clusters. To overcome these shortages, this paper proposes a new clustering algorithm that can detect the clustering centers automatically via statistical testing. Specifically, the proposed algorithm first defines a new metric to measure the density of an object that is more robust to the preassigned parameter, further generates a metric to evaluate the centrality of each object. Afterwards, it identifies the objects with extremely large centrality metrics as the clustering centers via an outward statistical testing method. Finally, it groups the remaining objects into clusters containing their nearest neighbors with higher density. Extensive experiments are conducted over different kinds of clustering data sets to evaluate the performance of the proposed algorithm and compare with the algorithm in Science. The results show the effectiveness and robustness of the proposed algorithm.

Index Terms—Clustering, Clustering Center Identification, Long-tailed Distribution, Outward Statistical Testing.

1 Introduction

Clustering is an important technique of exploratory data mining, which divides a set of objects (instances or patterns) into several groups (also called clusters) in such a way that objects in same group are more similar with each other in some sense than with the objects in other groups. It has been widely used in different disciplines and applications, such as machine learning, pattern recognition [2], data compression [3], image segmentation [4], [5], time series analysis [6], [7], information retrieval [8], [9], spatial data analysis [10], [11], [12] and biomedical research [13]. Moreover, as data's variety and scale increase rapidly, and the prior knowledge (e.g., category or class label) about the data is usually limited, clustering has been a challenging task.

In this context, a number of clustering algorithms have been proposed based on different clustering mechanisms [14], [15], [16], [17], [18], such as i) the connectivity based clustering assumes that the objects close to each other are more possible to be in the same cluster than the objects far away from each other; this kind of clustering algorithms usually organizes the objects as a hierarchical structure but does not produce a unique partition, and still needs users to preassign a distance threshold to generate appropriate clusters. The representative algorithms include Single-Link [19] and Complete-Link [20]. ii) The centroid based clustering represents each cluster as a central vector (or named clustering center), and the objects are assigned to the nearest clustering center, the famous examples are k-Means and its variants such as k-Medoids [21] and k-Means++ [22], where k denotes the number of clusters preassigned by user. The requirement of the parameter k specified in advance is considered as one of the critical drawbacks of this kind of algorithms. Meanwhile, it is usually not able to detect the non-spherical clusters. iii) The distribution-based clustering assumes that the objects in a given cluster are most likely to be derived from the same distribution. The most famous example is EM (Expectation maximization) algorithm [23] which employs a fixed number of Gaussian distributions to approach the distribution of the objects. However, for most real world data sets, the real distribution of the objects is usually difficult to define in advance and cannot be concisely defined as Gaussian distribution. Moreover, this kind of clustering algorithms still needs to preassign the number of clusters (or different distributions). iv) The density based clustering defines the clusters as areas with higher density, and can detect the clusters in any arbitrary shape. The most popular example of density-based clustering is DBSCAN [24] in which only the objects whose density is greater than the given thresholds are connected together to form a cluster. However, the proper threshold setting varies with different data sets, there is still no effective method to preassign these thresholds. v) The spectral clustering based algorithm does not make assumptions on the forms of the clusters; it utilizes the spectrum (i.e., eigenvalues) of the similarity matrix of the data to map the data into a lower-dimensional space in which the objects can be easily clustered by traditional clustering techniques [18], [25], [26]. Comparing to the traditional algorithms, such as k-Means and single-linkage, this kind of clustering algorithm is useful in non-convex boundaries and performs empirically very well [27]. And the first few eigenvalues can be used to determine the number of clusters and reduce the dimension of data. Yet, it has stated in [28] these first eigenvectors cannot successfully cluster objects that contain structures with different sizes and densities.

Recently, Rodriguez and Laio [1] proposed a novel Clustering algorithm (denoted as RLClu for convenience in this paper) that integrates the merits of the above mentioned algorithms. First, similar to the connectivity and centroid based clustering, RLClu is only based on the distance (or similarity) between objects. Secondly, as the density based clustering, it defines the clustering centers as the objects with maximum local density, and can detect the non-

- *G. Wang and Q. Song are with the Department of Computer Science & Technology, Xi'an Jiaotong University, Xi'an, Shaanxi, 710049, China.*

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2016.2535209, IEEE Transactions on Knowledge and Data Engineering

spherical clusters. Moreover, in contrast with the other well-known clustering algorithms (e.g., k-means, EM) where an objective function needs to be optimized iteratively, RLClu assigns the clustering label for each object in a single step.

The algorithm RLClu first defines two metrics (local density and minimum density-based distance) for each object based on the distances among objects. Then, it constructs a two-dimensional plot (named as decision graph in RLClu) with these two metrics, and identifies the objects with both greater local density and minimum density-based distance as clustering centers via the decision graph. Finally, each of the remaining objects is assigned into the cluster including its nearest neighbor with a higher local density. However, there is still room for improving RLClu. Firstly, the local density plays a critical role in RLClu but is sensitive to a preassigned parameter, cutoff distance, when the data set is small. Secondly, for clustering center identification, it still needs users to preassign two minimum thresholds of the local density and the minimum density-based distance. Different threshold settings would result in different clustering results. The proper setting of these thresholds will vary with different clustering data sets. Consequently, as the other existing representative clustering algorithms (e.g., k-means, EM and DBSCAN), RLClu is also sensitive to some preassigned parameters and suffers from the parameter setting problem.

In order to address the shortages in RLClu, we propose a new clustering algorithm STClu (Statistical Test based Clustering) [1] in this paper. At first, we define a new metric to evaluate the local density of each object, which shows better performance in distinguishing different objects than the metric used in RLClu and is not so sensitive to the preassigned parameter. Then, we employ an outward statistical test method to identify the clustering centers automatically on a centrality metric constructed based on the new local density and new minimum density-based distance. The experimental results on the synthetic and real world data sets show the proposed algorithm is more effective and robust than RLClu. In a nutshell, the proposed algorithm STClu obtains the object representation in a low-dimensional (specifically two dimensional) space in which the objects can be easily clustered. This idea is quite similar with that of spectral clustering in which the spectrum of the similarity matrix of the data is used for dimension reduction and the reduced space is not necessarily two-dimensional.

The rest of this paper is organized as follows. Section 2 reviews the related work of clustering. Section 3 presents the details of our clustering algorithm STClu. Section 4 gives the experimental results comparing our clustering algorithm to RLClu. Section 5 concludes our work.

2 Related Work

Traditionally, many researchers have proposed a number of clustering algorithms to divide objects into different categories on the basis of their similarity. Yet, there is still no unified definition of a cluster [1] since that we could get different clusters with different clustering mechanisms. For centroid based clustering (such as k-Means), the objects are always grouped into the nearest clustering center. So this kind of algorithm works well on the data set with spherical clusters but is not able to detect the non-spherical clusters. The spectral clustering based algorithms first make use the spectrum of the similarity matrix to reduce the dimension of data, then perform clustering on the reduced data by traditional clustering algorithms (e.g., k-Means). The distribution based clustering algorithm aims at reproducing the data with a set of predefined probability distribution functions; its performance depends on the number of distribution functions and the quality of these functions to approximate the implied distributions. The density based clustering algorithm usually can be used to identify the clusters in arbitrary shape. It defines clusters as connected dense regions in the data space. The well-known density based clustering algorithm is DBSCAN [24] which can not only detect non-spherical clusters but also discard the noise in the data set.

Although the above algorithms can be used to explore the structures implied in a given data set, one challenge for these algorithms is that they need some proper parameter settings in advance. Otherwise, they might fail to find the true structures. Such as, the number of clusters and the initial clustering centers for k-Means, the number of clusters for EM [23] and spectral clustering [18], the radius of epsilon-range-queries and the minimum number of objects required in an epsilon-range-query for DBSCAN [24], etc. That is, the performance of these clustering algorithms depends on the parameter settings. Nevertheless, the proper settings will vary with the clustering data sets. In order to overcome the parameter setting problem, researchers have attempted to resort to some automatic (or parameter-free) clustering algorithms. These algorithms can automatically search for the proper parameters in a specific way or do not require users to specify the parameters in advance. Such as, for the problem of determining the "ideal" number of clusters which has been discussed for a while [29], [30] and is attracting ever growing interest recently [31], [32], [33], the researchers put forward different kinds methods including information-theoretic based [34], structure complexity based [32] and recently quantization error based [33], the eigengap heuristic based for determining the number of clusters for spectral clustering [18], [35]. Meanwhile, some of these automatic clustering algorithms view the process of clustering as an optimization problem, and utilize different optimization strategies to search the optimal (or sub-optimal) partitions. In practice, the commonly-used optimization strategy is stochastic search, such as evolutionary algorithms (EA) [36], [37] and Simulated Annealing (SA) Algorithm [38] or their improvements [39], [40]. However, the performance of these search methods is related to choice of the fitness or energy function and proper parameter setting for optimization. For instance, the probabilities of crossover and mutation, the size of population for Generic Algorithm (GA), and the state space, the candidate generator procedure, the acceptance probability function, and the annealing schedule temperature, and initial temperature for Simulated Annealing (SA). Therefore, these algorithms are hypocritical parameter-free and still suffer from the parameter setting problem.

The clustering algorithm proposed by Rodriguez and Laio [1] gives an alternative approach which can detect the clustering centers from irregular shapes of clusters in an intuitive way. They construct a two-dimensional decision graph with two metrics (i.e., local density and minimum

[1] The corresponding software can be obtained via https://cn.mathworks.com/matlabcentral/fileexchange/54893-automatic-clustering-via-statistical-testing.

density-based distance), and the points located in the top right corner of this graph are more possible to be the clustering centers (See details in Section 3.1). Once the clustering centers have been found, each one of the rest objects is grouped into the same cluster as its nearest neighbor with a higher density. This is completed in a single step and quite effective compared with other clustering algorithms (e.g., k-Means and EM) where an objective function needs to be optimized iteratively [23], [41].

However, for different data sets, the decision graphs are different as well. The local density used for decision graph construction is sensitive to a preassigned parameter (named cutoff distance) especially for small data sets. Moreover, although the algorithm can map the clustering centers into the top right corner of the decision graph, it still needs users to pick up proper number of objects from the decision graph artificially or set proper thresholds to determine the exact number of clustering centers in advance. There is no any straightforward method to handle the threshold setting problem (either for local density or the decision graph). Consequently, RLClu also suffers from the problem of how to determine the "ideal" number of clusters.

In this paper, we propose a novel clustering algorithm in which we first redefine the metrics of local density and minimum density-based distance with good robustness; then, instead of identifying the clustering centers by observing the decision graph artificially in RLClu, we detect the clustering centers by an outward statistical test method automatically on the basis of the redefined metrics. Extensive experiments demonstrate the effectiveness of the proposed algorithm.

3 Outward Statistical Testing Based Clustering Algorithm

In this section, we first review the original clustering algorithm RLClu proposed in [1], and then discuss the shortages in RLClu yet to be resolved. Furthermore, we propose an outward statistical testing based clustering algorithm to relieve these shortages.

3.1 Review of the Clustering Algorithm RLClu

The clustering algorithm RLClu is proposed based on the assumption of "Cluster centers usually have a higher local density and a relative larger distance from objects with higher local densities" [1]. It consists of three steps: metric extraction, clustering center identification, and object clustering.

1) **Metric extraction**: For each of the n objects $\{O_1, O_2, \cdots, O_n\}$ being clustered, RLClu defines two metrics ρ and δ to evaluate the local density of the given object and the minimum density-based distance between the given object and the other objects.

2) **Clustering center identification**: RLClu constructs a two-dimensional point (ρ_i, δ_i) for each object and maps all these objects into a two-dimensional space, where the two-dimensional plot is referred to as a decision graph. In the decision graph, only points which are far away from both of the ρ-axis and δ-axis are identified as the clustering centers, i.e., the objects with both high ρ_i and δ_i. RLClu defines two minimum thresholds of ρ_{min} and δ_{min} to identify the clustering centers.

3) **Object clustering**: This part is straightforward once the clustering centers are picked up. That is, for all the objects except for the clustering centers, each one is assigned to a cluster which contains its nearest neighbor with higher local density ρ.

According to the brief introduction of RLClu, we can get that the metrics ρ and δ play important roles in RLClu. In order to further understand RLClu and analyze its drawbacks, we briefly introduce the metrics ρ and δ in advance.

In RLClu, the local density of a given object O_i is defined by Definition 1.

Definition 1. Local density ρ.

$$\rho_i = \sum_{j=1}^{n} \Delta(d_{i,j} - d_c). \tag{1}$$

Where $d_{i,j}$ denotes the distance between objects O_i and O_j. The distance can be Euclidean distance or any measure which can evaluate the difference between two objects, d_c is the cutoff distance preassigned by users. And $\Delta(x) = 1$ if $x < 0$ and $\Delta(x) = 0$ otherwise. From Definition 1, we can get that the local density of object O_i is the number of objects appearing in the hypersphere whose center is O_i and radius is d_c, i.e., the number of neighbors with distance to O_i being smaller than the cutoff distance d_c.

Based on the local density ρ, the minimum density-based distance δ_i of O_i to any other object with higher density is defined as follows.

Definition 2. Minimum density-based distance δ.

$$\delta_i = \min_{j \neq i \wedge \rho_i < \rho_j} (d_{i,j}) \tag{2}$$

According to Definition 2, for a given object O_i, we can get a distance δ_i which is the minimum distance between O_i and any other object with higher local density. It is noted that, for the object O_i with the highest local density ρ_i, its δ_i is defined as $\max_{j=1}^{n}(d_{i,j})$. Definition 2 tells us that for the objects with local or global maximum density, their δ_i is much larger than their typical nearest neighbor distance. Moreover, the clustering centers usually have local or global maximum density. Thus, the cluster centers can be recognized as the objects with anomalously large δ_i.

However, in the case of a data set contains outliers, it is not enough to detect the clustering centers only with δ_i. This is because the outliers are usually far from other objects, and have large δ_i as well. Fortunately, considering that the outliers usually have quite small local density ρ_i, so we can detect the outliers on the basis of both ρ and δ. This is the reason why RLClu constructs a two-dimensional decision graph for clustering center detection.

3.2 Shortages of The RLClu Algorithm

1) *RLClu is sensitive to the parameter cutoff distance d_c.*

Definition 1 shows that the values of the local density ρ are all natural numbers, i.e., $\rho_i \in \mathscr{N}$. This means for the n objects in a given data set D, the density ρ_i of each object O_i is one of the elements in $\{1, 2, \cdots, n\}$. The value of ρ_i depends on the cutoff distance d_c. A smaller d_c usually results in a smaller ρ_i. For a given cutoff distance d_c, only a part of these n integer values will be the density ρ_i (e.g., see the distribution of ρ in Fig. 2). According to the pigeonhole principle [42] that if x items are put into y containers, with

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2016.2535209, IEEE Transactions on Knowledge and Data Engineering

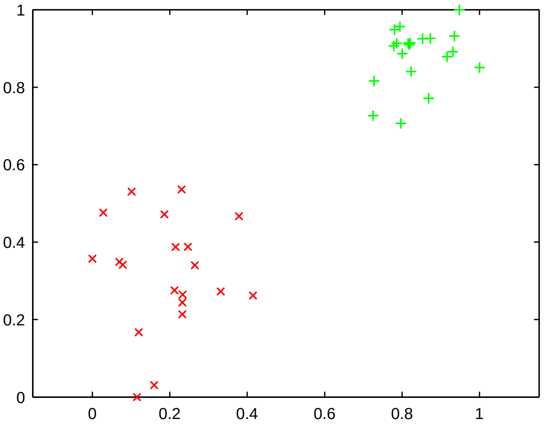


Fig. 1. A synthetic data set in two dimensions with two clusters

$x > y$, then at least one container must contain more than one item, there would be multiple objects with the same ρ_i in this case. So it will be difficult to distinguish different objects by ρ_i.

Furthermore, according to Definition 2, the minimum density-based distance δ_i of an object is evaluated based on ρ_i. When there are multiple objects whose ρ_i is identical and local maximum, δ might be unstable and misleads the construction of decision graph. This would result in that the algorithm RLClu fails to detect the correct clustering centers. Fig. 2 shows an example of RLClu in detecting the clustering centers on the synthetic data set in Fig. 1 with two clustering centers. From Fig. 2, we know that RLClu is quite sensitive to the cutoff distance d_c since that it detects different clustering centers under different d_c, and even gets wrong clustering centers for some d_c.

2) *The decision graph is not sufficient to identify the clustering centers and there still needs to predetermine the number of clusters*.

According to Definitions 1 and 2, the clustering centers will be of both larger density ρ and minimum density-based distance δ. In the decision graph drawn with ρ and δ, it is usually confusing to compare two points where one with greater ρ and smaller δ and the other with smaller ρ but greater δ. In order to overcome this problem, RLClu introduces a two minimum thresholds of local density ρ_{min} and minimum density-based distance δ_{min} to detect the clustering centers. The object O_i satisfying both $\rho_i > \rho_{min}$ and $\delta_i > \delta_{min}$ is identified as the clustering center. However, for different data sets, these two thresholds ρ_{min} and δ_{min} will be different. RLClu still does not give any quantitative method to answer how to pick up these two thresholds. That is, there is still no effective method to predetermine how many objects should be chosen as the clustering centers. This is also one of the challenging works in other clustering algorithms, such as k-means and EM.

3.3 The Proposed Clustering Algorithm

In this section, we propose a novel clustering algorithm aiming at overcoming the shortages of RLClu in Section 3.2. In the proposed algorithm, we first put forward a new metric $\hat{\rho}$ to measure the density of an object, which shows better performance in terms of the ability to distinguish different objects and is more robust to the preassigned parameter than the local density ρ in RLClu. Meanwhile, on the basis of this new density metric, we redefine the minimum density-based distance of an object as a new version $\hat{\delta}$. With these two new metrics $r\hat{h}o$ and $\hat{\delta}$, we weigh the possibility of an object being a clustering center by a new centrality metric $\hat{\gamma}$ which is the product of $\hat{\rho}$ and $\hat{\delta}$ due to the fact that the clustering centers usually have both of higher density (measured by $\hat{\rho}$) and larger distance from each other (measured by $\hat{\delta}$). The objects with extremely large $ga\hat{m}ma$ are recognized as the clustering centers. With this in mind, afterwards, by analyzing the distribution of this product metric $\hat{\gamma}$, we transform the problem of clustering center identification into a problem of extreme-value detection from a long-tailed distribution, and employ an outward statistical testing method to identify the clustering centers automatically. Finally, we accomplish the clustering process by assigning proper clustering labels to the remaining objects based on these identified clustering centers.

3.3.1 *Improved Metrics to Evaluate the Centrality of Objects*

In this section, we first define a new metric to evaluate the local density of an object which is named K-density $\hat{\rho}$ (See Definition 3), and further demonstrate the new metric is more robust in clustering center detection than ρ in RLClu.

Definition 3. K-density $\hat{\rho}$.

$$\hat{\rho}_i = \frac{K}{\sum\limits_{j=1}^{K} d_{i,j}}. \tag{3}$$

Where $\{d_{i,j} | 1 \le j \le K\}$ denotes the set of distances between object O_i and its K nearest neighbors.

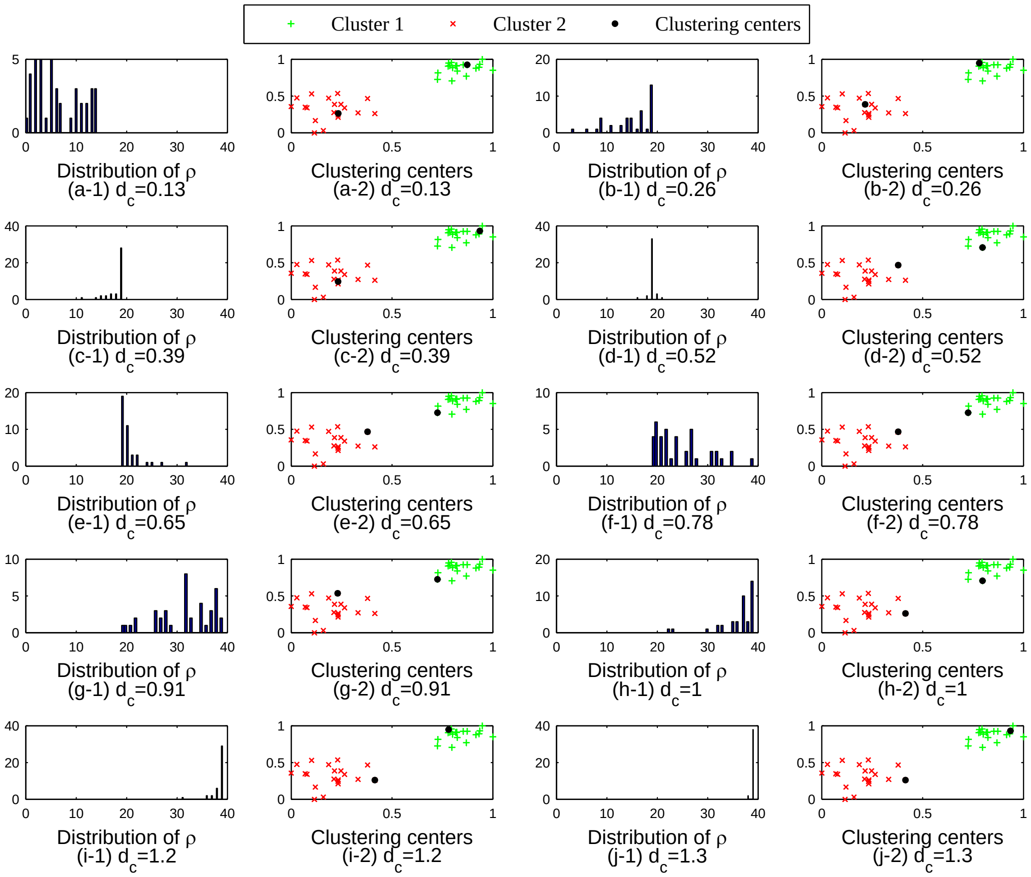
As we know, clustering centers usually lie in the center of a dense region. This means that, the sum of the distances between a clustering center and its K nearest neighbors is usually smaller than the sum of the distances between other non-clustering centers and their K-neighbors. According to Definition 3, clustering centers usually have higher $\hat{\rho}_i$. So it is reasonable to employ the metric K-density $\hat{\rho}$ to evaluate the density of a given object. Meanwhile, Definition 3 implies that the value of $\hat{\rho} \in \Re$ will be a non-negative continuous value rather than an integer value of ρ. The non-negative continuous values are more conductive to catch up the density differences of different objects than the integer values. So it is easier for K-density metric $\hat{\rho}$ to distinguish different objects than the local density ρ in RLClu.

Fig. 3 shows the distribution of $\hat{\rho}$ and clustering centers identified with $\hat{\rho}$ on the simple synthetic data set in Fig. 1. From this figure, we know that i) the distribution of $\hat{\rho}$ corresponds to a wider range than ρ and is easier to distinguish different objects; ii) for different number of nearest neighbors used in K-density, the clustering centers identified by $\hat{\rho}$ are steady. This indicates that the new density metric $\hat{\rho}$ is more robust to the parameter K (i.e., number of nearest neighbors) and shows better performance in clustering center detection.

Similar to the minimum density-based distance in Definition 2 of RLClu, based on K-density $\hat{\rho}$, we can get a new minimum density-based distance $\hat{\delta}$ of an object as follow.

Definition 4. New minimum density-based distance $\hat{\delta}$.

$$\hat{\delta}_i = \min_{j \neq i \wedge \hat{\rho}_i < \hat{\rho}_j} (d_{i,j}) \tag{4}$$



Distribution of ρ (a-1) d_c=0.13; Clustering centers (a-2) d_c=0.13; Distribution of ρ (b-1) d_c=0.26; Clustering centers (b-2) d_c=0.26

Distribution of ρ (c-1) d_c=0.39; Clustering centers (c-2) d_c=0.39; Distribution of ρ (d-1) d_c=0.52; Clustering centers (d-2) d_c=0.52

Distribution of ρ (e-1) d_c=0.65; Clustering centers (e-2) d_c=0.65; Distribution of ρ (f-1) d_c=0.78; Clustering centers (f-2) d_c=0.78

Distribution of ρ (g-1) d_c=0.91; Clustering centers (g-2) d_c=0.91; Distribution of ρ (h-1) d_c=1; Clustering centers (h-2) d_c=1

Distribution of ρ (i-1) d_c=1.2; Clustering centers (i-2) d_c=1.2; Distribution of ρ (j-1) d_c=1.3; Clustering centers (j-2) d_c=1.3

Fig. 2. Clustering center identification using RLClu with ρ on the synthetic data set in Fig. 1. Where for each sub-figure of the distribution of ρ, the horizontal axis shows the value of ρ, and the vertical axis shows the count of the corresponding ρ.

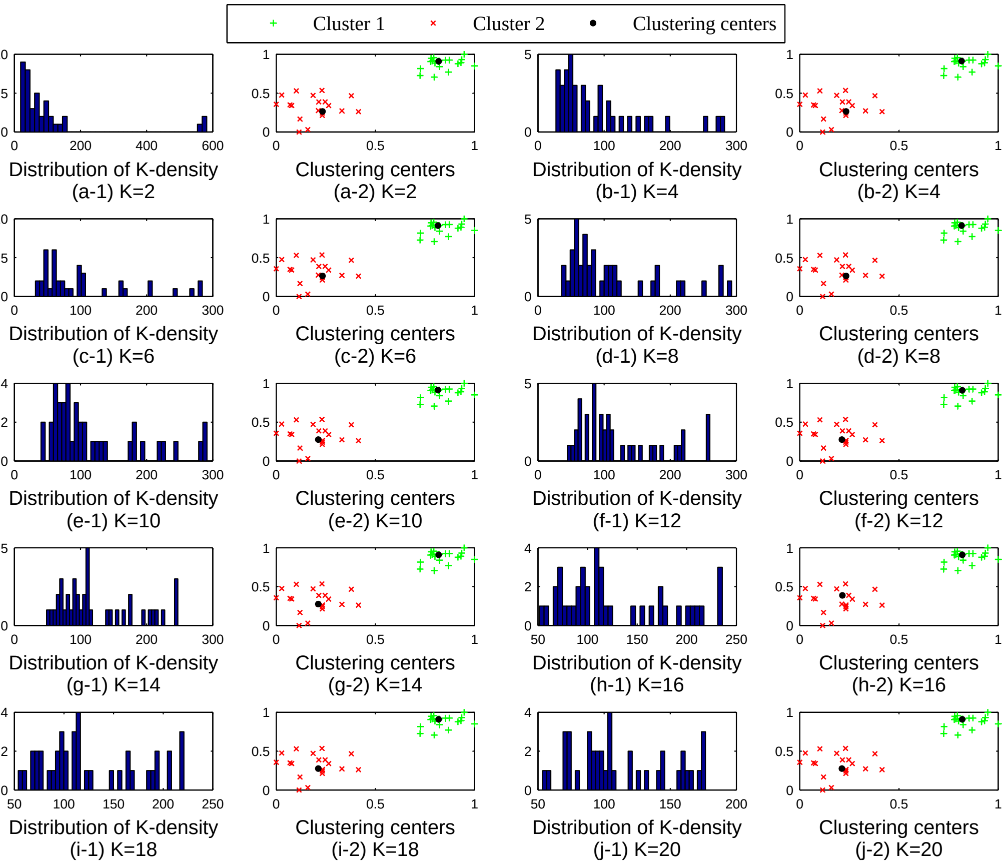
Definition 4 tells us that only for the objects with local or global maximum K-density, their $\hat{\delta}_i$ is much larger than their nearest neighbor distances. Meanwhile, the clustering centers usually have local or global maximum density, and further can be recognized as the objects with anomalously large $\hat{\delta}_i$.

According to the definition of $\hat{\delta}$ and property of the underlying structure of the clusters, it is rational to infer that the new minimum density-based distance $\hat{\delta}$ follows a long-tailed distribution (or heavy-tailed distribution). This is due to the fact that i) $\hat{\delta}$ is non-negative, and ii) $\hat{\delta}$ of the clustering center is usually relatively large and the number of clustering centers is usually relatively smaller than the number of other objects. i.e., for a given set of objects being clustered, the corresponding values of $\hat{\delta}$ of most objects are usually small and only a few are quite large. So it is intuitional that the distribution of $\hat{\delta}$ will be long-tailed. In contrast, the minimum density-based distance δ in RLClu would not necessarily follow the long-tailed distribution because of that it is also sensitive to the parameter cutoff distance d_c. According to Definition 2 of δ, δ_i of an object O_i is the minimum distance between O_i and any other object with higher local density. As mentioned in Section 3.2, the local density ρ (See Definition 1) will not distinguish different objects under an improper setting of d_c; that is, there exist multiple objects with the same local density. It is more possible for quite a few objects with higher local density are used for calculation of δ, see Fig. 2 (j-1) with d_c = 1.3 for an example. Therefore, the distribution of δ will be same of the distribution of distances between most objects and these quite a few objects. In this case, for the spherical clusters, distribution of δ (i.e., the distribution of the distances in each clusters) is more likely be symmetrical instead of heavy-tailed.

However, it is noted that only having the new minimum density-based distance $\hat{\delta}$ is still insufficient to identify the clustering centers since the outliers also have larger $\hat{\rho}$ according to Definition 4. Fortunately, the clustering centers usually have both large $\hat{\rho}$ and $\hat{\delta}$. Therefore, a new metric $\hat{\gamma}$ which is the product of K-density $\hat{\rho}$ and new minimum density-based distance $\hat{\delta}$ is introduced. And the objects with extreme maximum $\hat{\gamma}$ are identified as the clustering centers. Different from the clustering algorithm RLClu which needs the preassigned thresholds to identify the proper number of clusters, by analyzing the distribution of $\hat{\gamma}$, we propose an outward statistical testing method to identify the clustering centers automatically. The details of clustering center identification are introduced in Section 3.3.2.

3.3.2 Clustering Center Identification

In this section, we first demonstrate that $\hat{\gamma}$ follows the long-tailed distribution. Afterward, we translate the clustering center identification problem into an extreme maximum value detection problem from a long-tailed distribution, and present a statistical test technique based clustering center identification method over long-tailed distributions.



Distribution of K-density (a-1) K=2
Clustering centers (a-2) K=2
Distribution of K-density (b-1) K=4
Clustering centers (b-2) K=4
Distribution of K-density (c-1) K=6
Clustering centers (c-2) K=6
Distribution of K-density (d-1) K=8
Clustering centers (d-2) K=8
Distribution of K-density (e-1) K=10
Clustering centers (e-2) K=10
Distribution of K-density (f-1) K=12
Clustering centers (f-2) K=12
Distribution of K-density (g-1) K=14
Clustering centers (g-2) K=14
Distribution of K-density (h-1) K=16
Clustering centers (h-2) K=16
Distribution of K-density (i-1) K=18
Clustering centers (i-2) K=18
Distribution of K-density (j-1) K=20
Clustering centers (j-2) K=20

Fig. 3. Clustering center identification with $\hat{\rho}$ on the synthetic data set in Fig. 1. Where for each sub-figure of the distribution of K-density, the horizontal axis shows the value of K-density, and the vertical axis shows the count of the corresponding K-density.

Since the long-tailed distribution will play an important role in identifying clustering centers, we first give the formal definition of long-tailed family of distributions $\mathcal{L}$ and further a theorem about $\mathcal{L}$.

Definition 5. Long-tailed family of distributions ($\mathcal{L}$). Let X be a random variable following a given non-negative distribution F and x be an observation of X, F is a long-tailed distribution (i.e., $F \in \mathcal{L}$) if and only if

$$\lim_{x \to +\infty} \frac{F(x-l)}{F(x)} = 1, \forall (l > 0). \tag{5}$$

Suppose random variable X is the new minimum density-based distance $\hat{\delta}$ of a set of objects, i.e., $x = \hat{\delta}$, according to Definition 4 of $\hat{\delta}$, only clustering centers (or outliers) have greater $\hat{\delta}$ and just a few objects could be clustering centers (or outliers), so both of $F(x)$ and $F(x-l)$ will approach 1 when x becomes $+\infty$, further $F(x-l) \leq F(x)$ always is true. According to the squeeze theorem [43], Eq. 5 will hold. That is, the new minimum density-based distance $\hat{\delta}$ follows the long-tailed distribution in Definition 5.

Meanwhile, since only a few objects can be clustering centers, for most other objects, the greater $\hat{\rho}$ does not mean either greater or smaller $\hat{\delta}$, so there is usually no significant interaction between $\hat{\rho}$ and $\hat{\delta}$. Therefore, it is reasonable to assume that $\hat{\delta}$ and $\hat{\rho}$ are independent with each other in some way. Keeping this in mind, considering the following theorem of the long-tailed family of distributions $\mathcal{L}$ [44], [45], it is reasonable to say that $\hat{\gamma} = \hat{\rho} \times \hat{\delta}$ will also follow the long-tailed distribution.

Theorem 1. Suppose that there are two random variables X and Y being independent with each other, F and G denote the distributions of X and Y, respectively, and H denotes the distribution of the product of X and Y, if $(F \in \mathcal{L}_c) \wedge (P(Y > 0) > 0)$, then $H \in \mathcal{L}$.

Where $\mathcal{L}_c$ is defined as $\{F : F \in \mathcal{L} \wedge F \text{ is continuous}\}$.

In this paper, the metric $\hat{\gamma} = \hat{\rho} \times \hat{\delta}$ is proposed to weigh the possibility of an object being a clustering center. From the above analyses of $\hat{\rho}$ and $\hat{\delta}$, we know : i) $\hat{\delta}$ follows the long-tailed distribution and is also a continuous distribution, and ii) all the values of $\hat{\rho}$ are positive (i.e., $P(\hat{\rho} > 0)$ is natural positive but the metric ρ in RLClu might be zero), thus, $\hat{\gamma}$ follows the long-tailed distribution.

The $\hat{\rho}$ and $\hat{\delta}$ of clustering centers usually are large, so their $\hat{\gamma}$ is also big. Enlightened by the idea of decision graph in RLClu, if we represent the $\hat{\rho}$ and $\hat{\delta}$ of each object in a two dimensional plot, clustering centers are usually quite far away from other objects, and these clustering centers can be viewed as the outliers in the-two dimensional plot. Therefore, in this paper, we define the clustering centers as the objects corresponding to the outliers of $\hat{\gamma}$ in a long-tailed distribution, and further handle the problem of clustering center identification via detecting multiple outliers in the long-tailed distribution. Next, we give the formal definition of clustering centers.

Suppose that there are n objects being clustered, $\{\hat{\gamma}_i, 1 \leq i \leq n\}$ records the metrics to evaluate the centrality of each object, the identification of clustering centers is to determine whether the top m ($m \leq n-1$) metrics have been generated by a long-tailed distribution F associated with $\hat{\gamma}$. If the m extreme order statistics have not been generated by F, we refer them as "outliers", and the corresponding objects are identified as the clustering centers. This paper employs a statistical test procedure to identify the outliers in a long-tailed distribution.

For this purpose, borrowing the idea in [46], we first construct a set of ordered statistics $X_{1,n} \geq X_{2,n} \geq \cdots \geq X_{n,n}$, where $X_{1,n}$ is the 1st maximum value in $\{\hat{\gamma}_i, 1 \leq i \leq n\}$, $X_{2,n}$ is the 2nd maximum, and so on. Then, we explore the sequence of null hypotheses $\{H_{0,k}\ , 1 \leq k \leq m\}$, where the null hypothesis $H_{0,k}$ assumes that the kth statistic $X_{k,n}$ belongs to the long-tailed distribution F, it is not an outlier and the corresponding object is not a clustering center which is defined as follows.

Definition 6. Clustering center. A given object is a clustering center if and only if the null hypothesis with respect to its $\hat{\gamma}$ is rejected, i.e., the corresponding metric $\hat{\gamma}$ is an outlier in a long-tailed distribution.

In order to identify the outliers, we need to find an effective statistic to test the sequence of hypotheses $\{\mathtt{H}_{0,k}$, $1 \leq k \leq m\}$. Considering that i) the power function can be used to describe the distribution of the product of local density ρ and the minimum density-based distance δ under a proper setting of cutoff distance d_c [1], and ii) Rodriguez and Laio also stated that in the region with both large ρ and δ, the distribution will be strikingly different from the power law, and the high values of the product would be more likely to be outliers [1]. Meanwhile, comparing to ρ and δ, the new metrics $\hat{\rho}$ and $\hat{\delta}$ proposed in this paper enhanced them in robustness while still follows the idea in [1] that the objects with both of larger ρ and δ are more possible to be the clustering centers. So, it is rational to assume that the tails of the long-tailed distribution (e.g., $\hat{\gamma} = \hat{\rho} \times \hat{\delta}$) which decay as power functions. Specifically, the cumulative density function of F can be defined as follows, for some $\lambda > 0$ and sufficiently large x,

$$F(x) = 1 - L_0(x) \cdot x^{-\lambda}, \tag{6}$$

where L_0 is a slowly varying function and for sufficiently large x, L_0 behaves almost like a constant, and the parameter λ denotes the tail index.

According to the idea of detecting outlier in a long-distribution described by a power function [46], the ratios $R_t = X_{t,n}/X_{t+1,n}$ ($1 \leq t \leq n-1$) are constructed as the statistic to explore the null hypotheses. If the hypothesis $H_{0,k}$ is rejected by R_k, all the objects corresponding to hypotheses $H_{0,1}, H_{0,2}, \cdots, H_{0,k}$ are identified as the clustering centers. Once achieving the statistics R_t, the most important step of the statistical test is to find the proper critical value to judge whether R_t is statistically significant to reject the null hypothesis. Based on the outward testing for multiple outlier identification in [46] which defines the critical value for R_t, we can give the following proposition for clustering center identification.

Proposition 1. (Outward testing for clustering center identification) For a given set of null hypotheses $H_{0,1}, H_{0,2}, \cdots, H_{0,m}$ and the corresponding statistics $R_1, R_2, \cdots, R_m$, initially testing $H_{0,m}$ using R_m, if $H_{0,m}$ is not rejected, then test the null hypothesis $H_{0,m-1}$ using R_{m-1}. Continue this outward testing until a null hypothesis is rejected or all the m hypotheses are processed. If $H_{0,k}$ is the first hypothesis being rejected, the objects with respect to the statistics $R_1, R_2, \cdots, R_k$ are identified as the clustering centers. For a given level of significance α (5%), the individual critical value for kth ($1 \leq k \leq m$) hypothesis can be set to

$$r_k = [1 - (1-\alpha)^{1/m}]^{-1/(\lambda \cdot k)} \tag{7}$$

The proof of the critical value of Eq. 7 can be found in [46]. With Proposition 1, it is easy to compare the statistic R_k with r_k to identify the clustering centers. If $R_k > r_k$, we can reject the null hypothesis $H_{0,k}$ and get the clustering centers.

In Eq. 7, the tail index λ needed to be estimated with the following modified Hill-type estimator suggested in [46].

$$\hat{\lambda}(\kappa) = [\frac{m}{\kappa-m+1}\ln X_{m+1,n} - \frac{\kappa}{\kappa-m+1}\ln X_{\kappa+1,n} + \frac{1}{\kappa-m+1}\sum_{i=m+1}^{\kappa}\ln X_{i,n}]^{-1}, \tag{8}$$

where n is the number of objects being clustered and κ is the largest index of element in X used to estimate λ ($m < \kappa < n$). That is, the elements indexed between m and κ in X are used to estimate λ. In order to ensure the statistical significance of the estimated λ, the value of $\kappa - m$ should be large enough. Meanwhile, in order to guarantee that the m statistics contain the clustering centers, the value of m should be greater than the ideal number of clustering centers which is unknown in advance but is usually quite small compared with n. For these reasons, and in order to identify the clustering centers automatically, $m = \lceil 0.1n \rceil$ and $\kappa = \lceil 0.95n \rceil$ generally to make the number of objects used for statistical test is large enough.

3.3.3 Statistical Test based Clustering Algorithm

In this section, we present the proposed algorithm STClu (Statistical Test based Clustering), which is implemented on the basis of the metrics defined in Section 3.3.1 and the clustering center identification method introduced in Section 3.3.2. Algorithm 1 shows the pseudo-code description of STClu.

The pseudo-code consists of three parts: i) metric extraction; ii) clustering center identification; and iii) object clustering. In the first part (lines 1-5), by calculating the K-density $\hat{\rho}_i$ and new minimum distance $\hat{\delta}_i$ for each object O_i, STClu gets a set of metrics GamaSet = $\{\hat{\gamma}_1, \hat{\gamma}_2, \cdots, \hat{\gamma}_n\}$ to evaluate the centrality of each object. The second part (lines 6-17) employs the outward statistical testing method presented in Proposition 1 to identify k clustering centers. First, by sorting the metrics of GamaSet in descending order, STClu generates a set of ordered statistics X and further constructs a set of statistics R for statistical testing. Then, starting at the mth hypothesis $H_{0,m}$, STClu identifies the first hypothesis $H_{0,k}$ rejected by comparing the statistic R_i with the estimated critical value r_i. Finally, the number of clustering centers is set as k and the objects corresponding to the first k hypotheses $\{H_{0,1}, H_{0,2}, \cdots, H_{0,k}\}$ are detected as the clustering centers $\{c_1, c_2, \cdots, c_k\}$. In the third part (lines 18-24), for each object being not the clustering centers, STClu clusters it into the group containing its nearest neighbor with higher K-density. After that, $CLU = \{Clu_i, 1 \leq i \leq k\}$ records the k clusters found by STClu, where each cluster Clu_i ($1 \leq i \leq k$) is non-empty and contains at least one object (e.g., clustering center c_i) and each object belongs to exactly one cluster. Therefore, the proposed algorithm STClu is a kind of partitional clustering.

In algorithm STClu, the number of nearest neighbors K is associated with the calculation of K-density, it is set to be a default value $\lceil\sqrt{n}\rceil$ which is usually adequate in most of the situations according to the sensitivity analysis of K on the performance of STClu in Section 4.3.

```
Inputs : $O \leftarrow \{O_1, O_2, \cdots, O_n\}$: A set of $n$ objects
         $K$: the number of nearest neighbors in $K$-density $\hat{\rho}$;
Output: $CLU$;//A set of clusters
RhoSet ← $\phi$, DeltaSet ← $\phi$, NNSet ← $\phi$, GamaSet ← $\phi$;
//Part 1: Metric extraction
distanceMatrix ← DistanceFunction($O$);//Calculate distance
RhoSet ← $F_{\hat{\rho}}$(distanceMatrix, $k$);//Calculate $\hat{\rho}$
[DeltaSet, NNSet] ← $F_{\hat{\delta}}$(distanceMatrix, RhoSet);//Calculate $\hat{\delta}$ and
  identify the nearest neighbor for each object
GamaSet ← RhoSet · DeltaSet;//$\hat{\gamma} = \hat{\rho} \cdot \hat{\delta}$
//Part 2: Clustering center identification
$X$ ← sort(GamaSet, "descend");//Sort GamaSet in descending
  order to get a set of ordered statistics $X$
$R \leftarrow \{R_i \leftarrow X_{i,n}/X_{i+1,n}\}$ ($1 \leq i \leq n-1$);
$m \leftarrow \lceil 0.1n \rceil, k \leftarrow 0$;//Start at the $m$th hypothesis
//Identify the number of clusters $k$ by Outward
  statistical testing
while $m > 2$ do
    Calculate the critical value $r_m$ according to Eq. 7;
    if $R_m > r_m$ then
        $k \leftarrow m$;
        break;
    end
    $m \leftarrow m - 1$;
end
Identify the objects corresponding to $\{R_1, R_2, \cdots, R_k\}$ as the clustering
centers $\{c_1, c_2, \cdots, c_k\}$, and label $c_i$ as $i$;
//Part 3: Object clustering
for $i \leftarrow 1$ to $n$ do
    if $O_i$ is unlabeled then
        Mark $O_i$ the label of its nearest neighbor with higher $\hat{\rho}$
        according to NNSet;
    end
end
$CLU \leftarrow \{Clu_i, 1 \leq i \leq k\}$, where $Clu_i$ denotes the set of objects with
label $i$;
return;
```

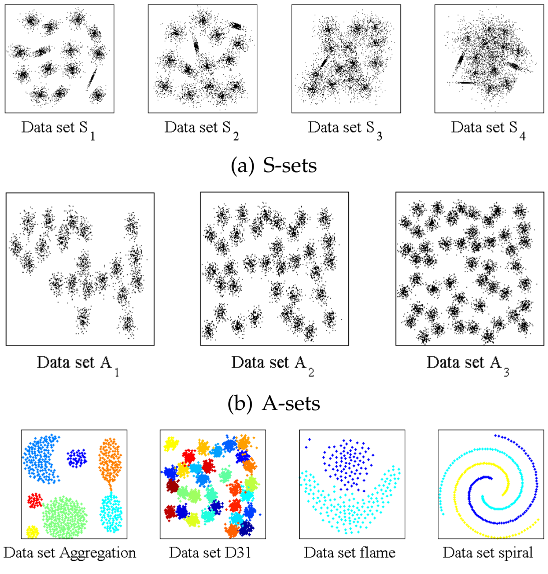
Algorithm 1: Outward statistical testing based clustering algorithm STClu

Time complexity analysis: In the first part, the distances among n objects should be computed in advance. Let $dist()$ be the function to compute the distance between two objects, and $O(dist())$ be the time complexity of this function[2], the time complexity of the distance collection will be $O(n^2 \cdot O(dist()))$. This time consumption can not be ignored by any clustering algorithm. After the distance collection, the time complexity is $O(n \cdot K)$ for K-density $\hat{\rho}$ calculation (note that K is smaller than n), $O(n \cdot \log(n))$ for $\hat{\delta}$ calculation and the nearest neighbor identification of each object, and $O(n)$ for GamaSet collection. Therefore, for the first part, the time complexity is $O(n^2 \cdot O(dist())) + O(n \cdot K) + O(n \cdot \log(n)) + O(n) = O(n^2 \cdot O(dist()))$. In the second part of clustering center identification, the time complexity of the order statistic X calculation depends on sorting the GamaSet and is $O(n \cdot \log(n))$. And the time complexity is $O(n)$ for the statistic R calculation and $O(m)$ (note that m is the number of hypotheses being tested) for outward statistical testing. Thus, the time complexity of the second part is $O(n \cdot \log(n)) + O(n) + O(m) = O(n \cdot \log(n))$. The third part is straightforward, that is, by scanning the unlabeled objects once, we can get the clustering results, and time complexity is $O(n)$. This step is very effective and quite different with some well-known algorithms (such as k-means and EM) which need multiple iterations to optimize the given object function and get the final clustering results.

In summary, the time complexity of the proposed algorithm STClu is $O(n^2 \cdot O(dist)) + O(n \cdot \log(n)) + O(n) = O(n^2 \cdot O(dist()))$. That is, the efficiency of the proposed algorithm depends on that of the distance calculation. Since the distance calculation is an inevitable step for all clustering algorithms, if we ignore the part of distance calculation, the time complexity of the first part will be $O(n \cdot K)$. Compared to the algorithm RLClu proposed in [1], the differences between STClu and RLClu focus on i) the calculation of local density and ii) the clustering center identification. The computation of these two parts does not play a critical role in STClu. Therefore, the time complexity of STClu is as same as that of RLClu which is quite effective. But STClu can detect the clustering centers in a more effective way than RLClu.

4 Experimental Study

In this section, we experimentally evaluate the performance of the proposed clustering algorithm with representative clustering data. At first, we introduce the benchmark clustering data sets in Section 4.1, and then present the experimental results and analyses in Section 4.2. Finally, we conduct the sensitive analysis of STClu in Section 4.3.



(a) S-sets

(b) A-sets

(c) Shape sets

Fig. 4. Benchmark Data Sets

4.1 Benchmark Data Sets

Five groups of representative clustering data sets (e.g, including low and high dimensional data sets, synthetic and real world data sets) are employed as the benchmark to assess the performance of the proposed algorithm. These data sets are available on http://cs.joensuu.fi/sipu/datasets/ and http://people.sissa.it/~laio/Research/Res_clustering.

[2] For different distance functions, their time complexities are usually different as well.

php. The correct clustering centers of the data sets are known in advance. The details of these data sets are introduced as follows.

1) **S-sets**: two-dimensional data sets with 5000 objects and 15 Gaussian clusters with four different degree of clustering overlapping [47]. See Fig. 4(a) for details. This kind of data can be used to evaluate the robustness of the proposed algorithm. The degree of the clustering overlapping increases from data set S_1 to S_4. The greater the degree of overlapping, the more difficult to distinguish different clusters.

2) **A-sets**: two-dimensional data sets with varying number of clusters (20, 35 and 50 for A1, A2 and A3), and there are 150 objects per cluster [48]. See Fig. 4(b) for details. This kind of data can be used to evaluate the scalability of the proposed algorithm in detecting different numbers of clusters.

3) **Shape sets**: two-dimensional data sets (named Aggregations, D31, flame and Spiral) represent some difficult clustering objects because they contain clusters of arbitrary shape, proximity, orientation and varying densities [49], [50], [51], [52]. The number of objects in these four data sets is 788 for Aggregations, 3100 for D31, 240 for flame and 312 for Spiral, respectively. See Fig. 4(c) for details. This kind of data can be used to evaluate the proposed algorithm in detecting the clustering centers in complex clustering data sets. Meanwhile, the four data sets in Fig. 4(c) have also been used to assess the algorithm RLClu.

4) **High-dimensional data sets**: six high-dimensional data sets with 1000 objects and 16 Gaussian clusters in different dimensions [53]. The dimension of these six data sets is 32, 64, 128, 256, 512 and 1024, respectively. Each data set with dimension x is named "Dim$x$". This kind of data can be used to assess the performance of the clustering algorithms when the dimension of the data increases.

5) **Real world data sets**: the Face detection database including 400 figures with 40 people. This data set proposes a serious challenge to the algorithm RLClu since the real number of clusters is comparable with the number of objects in each cluster (10 different pictures for each people).

In order to evaluate whether the new metrics and clustering center identification method in STClu work well in improving the performance of clustering, we compared STClu with the clustering algorithm RLClu on these data sets. The program of RLClu can be found on http://www.sciencemag.org/content/suppl/2014/06/25/344.6191.1492.DC1.html. It is noted that, in order to demonstrate the effectiveness of the proposed algorithm comparing to RLClu in terms of clustering center identification, we also draw the decision graph of each data set in our experiments. However, limited to the length of the paper, we represent the decision graphs in the Supplementary materials.

4.2 Experimental Results And Analysis

In this section, we give the experimental results of the proposed algorithm STClu on the four different kinds of clustering data sets. For the first three groups of two-dimensional clustering data sets, we show the clustering results in a two-dimensional plot to evaluate the clustering performance intuitively. The distance among different objects of these data sets is evaluated by Euclidean distance. For the face detection data set, the distance among different pictures is computed with the method introduced in [1]. It is noted that, RLClu needs users to find out the clustering centers with the help of the decision graph. To make a fair comparison, the number of clustering centers used in RLClu is set to the same as that identified by the STClu. And the parameter cutoff distance d_c used in RLClu is set to the default value suggested in [1].

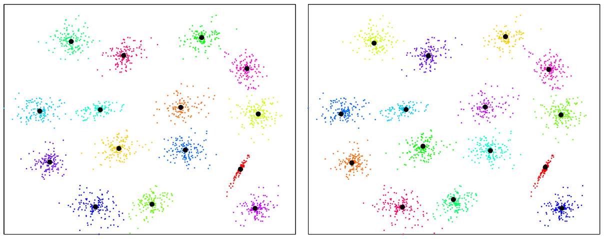

(a) STClu (b) RLClu

Fig. 5. Clustering results on data set S1

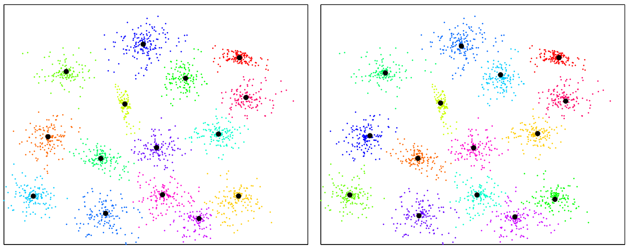

(a) STClu (b) RLClu

Fig. 6. Clustering results on data set S2

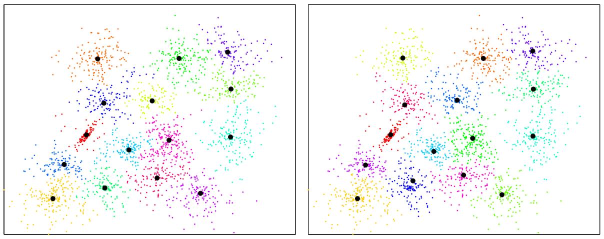

(a) STClu (b) RLClu

Fig. 7. Clustering results on data set S3

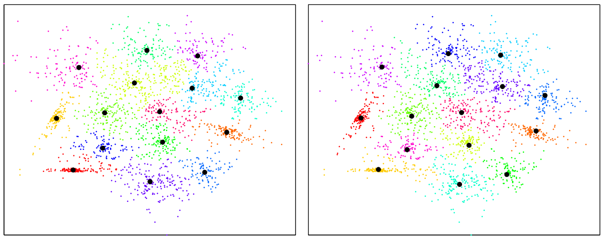
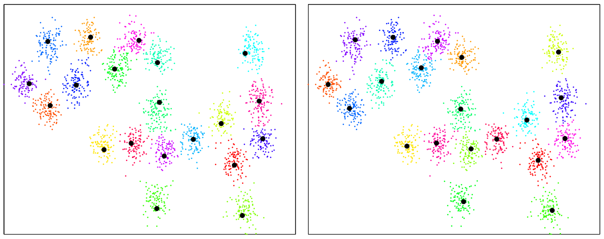

(a) STClu (b) RLClu

Fig. 8. Clustering results on data set S4

4.2.1 Results on S-sets

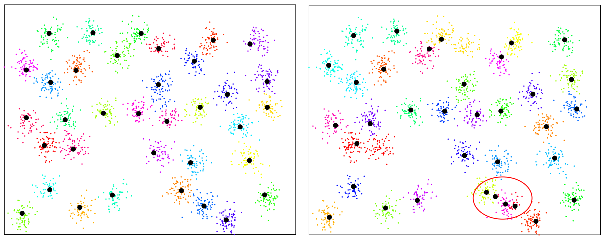
Figs. 5, 6, 7, 8 present the clustering results of algorithms STClu and RLClu. For each figure, the black circles demonstrate the clustering centers identified by the corresponding algorithms; and the points drawn in different colors correspond to different clusters. The same representation holds in the clustering results over A-sets and Shape Sets.

From these four figures, we can observe that, for data sets S1 and S2 (See Figs. 5 and 6) where the degree of the



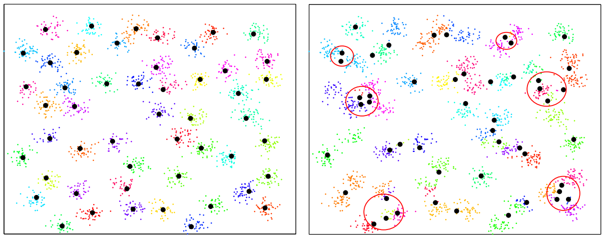
(a) STClu (b) RLClu

Fig. 9. Clustering results on data set A1



(a) STClu (b) RLClu

Fig. 10. Clustering results on data set A2



(a) STClu (b) RLClu

Fig. 11. Clustering results on data set A3

clustering overlapping is quite small, both of STClu and RLClu can effectively identify the correct clustering centers and achieve a good clustering results.

For data sets S3 and S4 (See Figs. 7 and 8), as the degree of clustering overlapping increases, it will be more difficult to distinguish different clusters. In this case, both of STClu and RLClu can still get the correct clusters. RLClu detects the clustering centers via observing the decision graphs of data sets S3 and S4 (See Fig.1 in the Supplementary materials). From their decision graphs, we can observe that the first 15 objects with greater $\rho \times \delta$ are far away other objects for all the four data sets. So these objects are identified as clustering centers by RLClu. However, with the degree of the clustering overlapping increases, some of these clustering centers get closer to other objects. This makes it difficult for RLClu to identify the clustering centers by observing the decision graph. However, STClu identifies the correct number of clustering centers via statistical test automatically, and shows advantage compared to RLClu that detects the clustering centers by observing the decision graph manually.

4.2.2 Results on A-sets

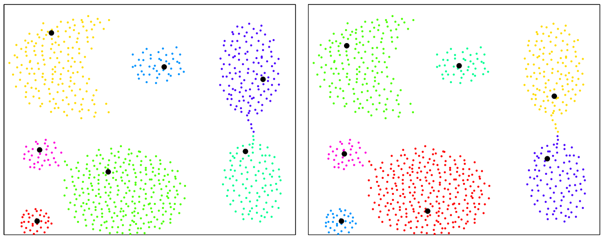
Figs. 9, 10 and 11 show the clustering results on data sets A-sets. From these figures, we can observe that:
1) For data set A1 with 20 clusters, both STClu and RLClu algorithms can detect the correct clustering centers. From the decision graph of A1 (See Fig.2(a) in the Supplementary materials), we can observe that the first 20 objects with greater $\rho \times \delta$ are relatively away from other objects. So these objects are identified as the clustering centers by RLClu. However, STClu detects the correct 20 clustering centers automatically by the statistical test technique.
2) For data set A2 with 35 clusters, STClu algorithm can detect all the correct clustering centers automatically via statistical test. But for clustering algorithm RLClu, there are still a small fraction of clustering centers identified incorrectly (such as the area marked by red circle in Fig. 10(b)). This can be demonstrated by the decision graph of A2 (See Fig.2(b) in the Supplementary materials). From the decision graph of A2, we can observe that some of the first 35 objects with greater $\rho \times \delta$ are quite close to the other objects. In this case, it is difficult to identify the correct clustering centers from the objects being close with each other only via the decision graph.
3) With the increase of the number of clusters, for data set A3 with 50 clusters, STClu still can correctly identify the clustering centers automatically. But the 50 clustering centers identified by RLClu are not all correct (such as the area marked by red circles in Fig. 11(b)). The reason can be demonstrated as follows: with the number of clusters increases, there are significant overlaps among some clusters. This kind of overlap increases the difficulty to distinguish different clusters. The local density ρ and the minimum density-based distance δ used in RLClu fail to distinguish the clustering centers from the other objects. This can also be observed by the decision graph of A3 (See Fig.2 (c) in the Supplementary materials). According to the decision graph of A3, almost half of the first 50 objects with greater $\rho \times \delta$ are difficult to distinguish from the other objects. This leads to that RLClu can not identify all correct clustering centers. However, STClu employs K-density $\hat{\rho}$ which shows better performance in distinguishing different objects than ρ used in RLClu, and further detects the clustering centers via the statistical test automatically instead of the decision graph. This is the advantage of STClu especially when it is difficult to distinguish different objects based on the decision graph.

4.2.3 Results on Shape Sets

Figs. 12, 13, 14 and 15 show the clustering results of the two clustering algorithms on the Shape sets. From these figures, we can observe that for all these data sets, i) the clustering centers identified by STClu and RLClu are different (see the black circles in these figures) since that the metrics used to measure the local density of each object are different in these two algorithms; ii) both STClu and RLClu can get the correct clusters.

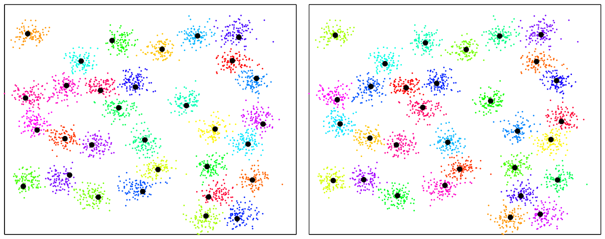
Yet, for algorithm STClu, the number of clusters are identified automatically by the statistical test in Proposition 1. For algorithm RLClu, we set the number of clusters to a specific value in advance manually (i.e., 6 for flame, 31 for D31, 2 for flame and 2 for Spiral). In fact, RLClu detects the clustering centers based on the decision graph with the guideline to select the objects being far away from other objects. According to these decision graphs (See Fig.3 in the Supplementary materials), except for data set Spiral, the number of clustering centers are easily detected as 3. For the other three data sets, i) according to the decision graph of data set Aggregation, the number of clustering centers could be also one of {7, 8, 9} except for 6; ii) according to the decision graph of data set D31, the number of clustering

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2016.2535209, IEEE Transactions on Knowledge and Data Engineering



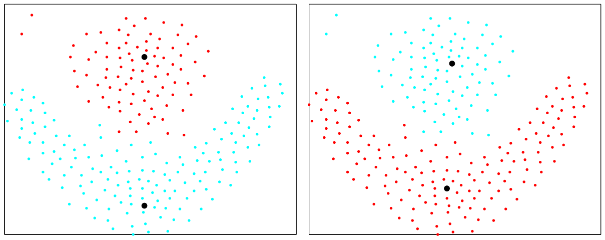
(a) STClu (b) RLClu

Fig. 12. Clustering results on data set Aggregation



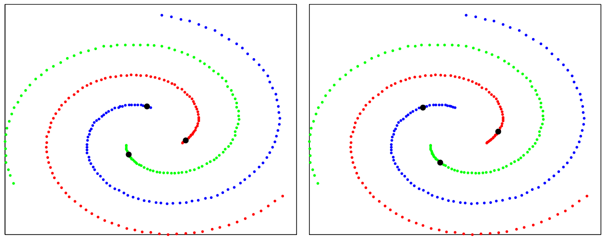
(a) STClu (b) RLClu

Fig. 13. Clustering results on data set D31



(a) STClu (b) RLClu

Fig. 14. Clustering results on data set flame



(a) STClu (b) RLClu

Fig. 15. Clustering results on data set Spiral

centers would be smaller than 31; iii) according to the decision graph of data set flame, except for the first two objects (overlap with each other), the number of clustering centers could be either 3 or 4. All the other numbers will result in RLClu getting inappropriate clustering results on these data sets.

4.2.4 Results on High-dimensional Data Sets

Different from the two-dimensional data sets, for high-dimensional data sets, it is impossible to give a graphical representation of the clustering results of the algorithms on them. It is noted that the ground truth partitions of these high-dimensional data sets are known in advance, and these information can be used as the baseline to evaluate the performance of the clustering algorithms. So in our experiments, we employ the well-known metric, *Normalized Mutual Information* (NMI) [54] to evaluate the performance of the clustering algorithm on high-dimensional data sets. Table 1 shows the NMI of STClu and RLClu algorithms on the six high-dimensional data sets. From this table, we can observe that the NMI value reaches the highest value 1.0 for each algorithm on all the six data sets. This means that both of STClu and RLClu can perfectly find the correct cluster structures for all data sets. Yet, for the clustering algorithm STClu proposed in this paper, the number of clusters and the clustering centers for each data set are identified automatically by the statistical test in Proposition 1; while RLClu detects the clustering centers by manually observing the decision graph of each data set (See Fig.4 in the Supplementary materials) with the guideline to select the objects far away from others as the clustering centers. According to these decision graphs (See Fig.4 in the Supplementary materials), it is easy to detect the objects falling in the right up corner as the clustering centers by RLClu since they are far away from other objects. This is due to the fact that different clusters are well separated with each other and the clustering structures implied in these high-dimensional data sets are relatively obvious.

TABLE 1
NMI of STClu and RLClu on high-dimensional data sets

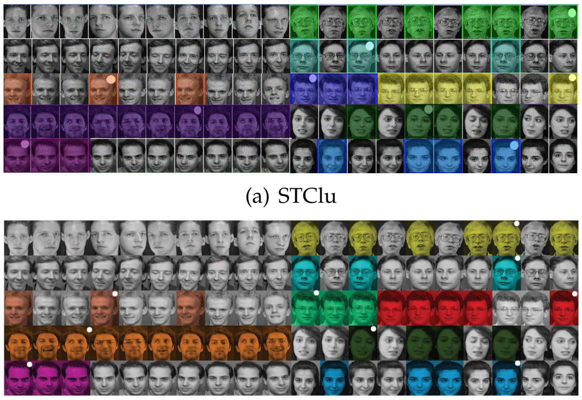
Algorithm	Dim32	Dim64	Dim128	Dim256	Dim512	Dim1024
STClu	1.0	1.0	1.0	1.0	1.0	1.0
RLClu	1.0	1.0	1.0	1.0	1.0	1.0

4.2.5 Results on Real World Data Sets

For clustering the pictures in Olivetti Face Database by RLClu, i) Rodriguez and Laio [1] have stated that, unlike the other clustering examples above, the exact number of clusters is not clear according to the decision graph; ii) they also claimed that the density estimator is unavoidably affected by large statistical error since there are only 10 pictures for each people. In this case, the parameter K used to calculate K-density is set to 3 instead of the default value. Meanwhile, in order to get more accurate clustering results, after clustering center identification, they employed a more restrictive criterion to assign a picture into a cluster which considers not only the local density but also the cutoff distance d_c. Specifically, a picture is grouped into the same cluster of its nearest picture with higher local density and the distance between them is smaller than d_c. With the constrain of d_c, the pictures further than d_c from any other picture of higher density are not assigned into any cluster [1].

In the proposed algorithm STClu, we first employ the Outward statistical method to detect the number of clusters automatically. Afterwards, in order to make a fair comparison, similar to RLClu, we follow the clustering process which performs on the K-density and the cutoff distance d_c is set to 0.07 as suggested in [1]. STClu algorithm identifies 46 clusters automatically with the outward statistical test method in Proposition 1. Moreover, we also give the decision graph of Olivetti Face Database and mark out the first 46 objects with the largest $\rho \times \delta$ (See Fig. 5 in the Supplementary materials).

For algorithm RLClu, we set the number of clustering centers as 46 identified by STClu manually in advance. In fact, if we identify the clustering centers based on the decision graph of these pictures, it is difficult to get that 46 is a proper number of clusters (See the Fig.5 in the


(a) STClu

(b) RLClu

Fig. 16. Clustering results on the first 100 Olivetti Face Data

Supplementary materials). However, if we only focus on the first 100 pictures for clustering, the nine pictures with greater ρ and δ in these 100 pictures (also of course in the 46 pictures marked in the decision graph) are easily picked up as the clustering centers according to the decision graph. Therefore, Rodriguez and Laio [1] gave the performance of their algorithm by using these nine clustering centers.

Figs. 16(a) and 16(b) show the clustering results of the first 100 pictures for the algorithms STClu and RLClu, respectively. In these figures, the clustering centers are labeled with white circles; the pictures with the same color belong to the same cluster, whereas other pictures are not assigned to any cluster.

From these two figures, we can observe that, i) both of the algorithms STClu and RLClu identify the same nine clusters, and the pictures of each cluster are derived from the identical people; ii) the clustering centers for some clusters identified by STClu are different from those identified by RLClu due to the fact that the metrics used to evaluate the local density of each picture in these two algorithms are different. For clustering algorithm RLClu, there is not any effective quantization method to answer why the nine clusters are proper on the decision graph in [1]. In contrast, for STClu proposed in this paper, the nine clustering centers among the first 100 pictures are detected by the statistical testing method automatically.

When we make the analysis on all 400 pictures, the decision graph used in RLClu still does not allow identifying clearly the number of clusters (See the Fig.5 in the Supplementary materials). As already mentioned, we pick up the 46 pictures with the largest $\rho \times \delta$ as the clustering centers for RLClu to make a fair comparison with STClu. The figures in Fig.6 of Supplementary materials show the clustering results of all the 400 pictures for the algorithms STClu and RLClu. From these figures, we can get that, i) due to the fact that the number of clusters identified by STClu (also set by RLClu) is greater than the real number of clusters 40, some pictures originally belonging to the same cluster are divided into multiple some smaller clusters. However, these small clusters remain pure, that is, each cluster only includes pictures of the same people; ii) similar to the results on the first 100 pictures, the clustering centers of some clusters identified by STClu are again different from those identified by RLClu. The reason for these differences is that the metric to evaluate the local density of each picture in STClu is different from that in RLClu.

Moreover, it is noted that the cutoff distance d_c is employed in the process of assigning a picture into a cluster. This is different from standard clustering process of both STClu and RLClu which only consider the local density. If one does not apply d_c to the process of clustering in STClu and RLClu, there would not exist any picture being unassigned. And some pictures of different people might be clustered into the same cluster. In this case, following the evaluation metrics r_{true} and r_{false} used in [1], [55] to evaluate the performance of different clustering algorithms. Where r_{true} is the ratio of pairs of pictures of the same people correctly associated with the same cluster and r_{false} is the ratio of pairs of pictures of different people erroneously grouped to the same cluster. The r_{true} is 64.50% for STClu and 65.89% for RLClu, and the r_{false} is 1.48% for STClu and 1.10% for RLClu. It seems that RLClu is slightly better than STClu. However, for clustering algorithm RLClu, the number of the clusters is set manually, and it is difficult to set this number by only observing the decision graph. This will be the biggest barrier to apply RLClu algorithm in practice. In contrast, STClu algorithm can identify the number of clustering automatically, and its performance is still comparable to the state-of-art image clustering method in [55]. That is, the algorithm STClu proposed in this paper shows better usability in practice.

4.3 Sensitivity Analysis of Number of Nearest Neighbors K on STClu

As we know, the K-density $\hat{\rho}$ plays a fundamental role in the proposed algorithm STClu since that both of the new minimum density-based distance $\hat{\delta}$ and the centrality metric $\hat{\gamma}$ are defined based on $\hat{\rho}$. There is a parameter, the number of the nearest neighbors K, affecting the calculation of $\hat{\rho}$. So in this section, we will analyze the impact of the parameter K on the performance of STClu.

For a given data set with n objects, the possible value of the number of K can vary from 1 to $n-1$. It is difficult and impractical to represent the clustering results by two-dimensional plots with respect to all the possible K on the data sets in Section 4.1. It is noted that all these data sets are known the ground truth partition in advance except for A-set. The ground truth partition is baseline to evaluate the performance of the clustering algorithm. Therefore, in this section, we employ a well-known metric, *Normalized Mutual Information* [54], which has been widely used to evaluate the performance of the clustering algorithm for sensitive analysis. We implement the proposed clustering algorithm STClu on the data sets in S-set, Shape-set, high-dimensional data set and the Olivetti Face Database under all the possible settings of K, and calculate the Normalized Mutual Information with respect to STClu under each possible K.

The impact of the number of nearest neighbors K on the performance of STClu in terms of NMI (Normalized Mutual Information) is shown in Fig. 7 in the Supplementary materials due to the length of the paper. From the figure of the sensitive analysis of K, it is observed that some of these figures do not list all possible numbers of the nearest neighbors. The reason is that when the number of the nearest neighbors K is large enough, with the increase of K, the performance of STClu either becomes stable or decreases. From these figures (Fig. 7 in the Supplementary materials), we can observe that:

1) For all the four data sets S1, S2, S3 and S4 in S-set, with the increase of K, NMI of STClu first gets the maximum value rapidly and holds this maximum value for dozens of K, and then drops down quickly and keeps steady. Holding the maximum value on a set of continuous K means that the algorithm is not so sensitive to K. And the default value of $K = \lceil\sqrt{n}\rceil$ falls into the range of K where the NMI gets maximum value. Meanwhile, it is noted that the maximum value of NMI approaches 1 for S1 and deceases from S1 to S4. This is due to the fact that the clusters in S1 are well separated, but for the other three data sets, there exists overlapping among different clusters and the overlapping ratio increases from S2 to S4. It is difficult to distinguish the objects lying on the overlapping region, and this reduces the NMI. However, STClu can still detect the correct clustering centers automatically.

2) For data set D31 with clusters in spherical shape in Shape-set, the variation of NMI with respect to K is similar to the data set in S-set. For data set Aggregation with complex clusters in different shapes, with the increasing of K, NMI of STClu first shows fluctuation when $K < 50$ but the NMI keeps relatively stable and gets relatively large value for a set of continuous K; then, after a small reduction, it becomes stable at a relative large value. The default value of $K = \lceil\sqrt{n}\rceil$ still falls in the region where NMI is large. It is noted that the NMI corresponding to the default K does not reach the best value 1. This is due to the fact that there exists a "bridge" among different clusters and the objects on the "bridge" are incorrectly clustered. For data set flame, with the increase of K, NMI of STClu first shakes sharply, then holds the maximum value 1 for a while and finally drops down to a quite small value. For data set Spiral, NMI can also be up to the theoretical maximum value 1 with respect to a set of continuous K. All these results show that the proper settings of K for STClu are a set of consecutive integers. This means that there will be multiple candidates K instead of some special points to explore the true structure of a clustering data.

3) For the six high-dimensional data sets, NMI of STClu first reaches the maximum value rapidly and keeps this maximum value when $K > 5$ for each data set. And the default value of $K = \lceil\sqrt{n}\rceil$ still falls in the region where NMI gets the maximum value. This means that the performance of proposed algorithm STClu is not sensitive to the parameter K, and the default setting of K is rational for these high-dimensional data sets.

4) For the Olivertti Face Database, the variation of NMI is quite special and different from the above ones. That is, NMI of STClu gets relatively large value when the number of nearest neighbors K is small (i.e., $K \leq 4$). When $K > 4$, with the increasing of K, NMI drops rapidly and becomes quite small. This is due to the fact that the ground truth partition constitutes 40 clusters and only 10 pictures in each cluster. The statistical error of the estimated density on such a small set of pictures will be large [1]. Therefore, for the clusters consisting of only a small set of objects, the setting of K should be carefully designed in our experiment, a small K will be better.

In summary, the performance of the proposed algorithm STClu is related to the number of the nearest neighbors K. For most data sets, the default setting of $K = \lceil\sqrt{n}\rceil$ usually falls into a set of consecutive integral values of K resulting in a better clustering performance. That is, the clustering results are stable when K varies around $\lceil\sqrt{n}\rceil$. This indicates that STClu is not so sensitive to the parameter K. The metric K-density used in STClu is robust in density evaluation.

5 Conclusion

In this paper, we have proposed a statistical test based clustering algorithm (STClu) that can automatically identify the clustering centers and further cluster the objects in an effective way. We first defined a new metric, K-density $\hat{\rho}$, to measure the local density of each object. Based on K-density, we established a new metric $\hat{\delta}$ to evaluate the distance of an object to its neighbors with higher density. Then, a product of these two metrics $\hat{\gamma} = \hat{\rho} \times \hat{\delta}$ was used to evaluate the centrality of each object. Afterwards, by analyzing the distribution of these metrics, we found that $\hat{\gamma}$ could be represented by a long-tailed distribution, and further transformed the clustering center identification into a problem of extreme-value detection from a long-tailed distribution. Finally, we employed an outward statistical testing method to detect the clustering centers with $\hat{\gamma}$ automatically and then completed the clustering process by assigning each of the rest objects to the cluster that contains its nearest neighbor with higher K-density. Extensive experiments have been conducted on both synthetic and real world data sets; the experimental results show the effectiveness and robustness of the proposed algorithm STClu.

Acknowledgments

The authors would like to the editors and the anonymous reviewers for their insightful and helpful comments and suggestions, which resulted in substantial improvements to this work.

References

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

[2] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu, "An efficient k-means clustering algorithm: analysis and implementation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 881 – 892, 2002.

[3] A. Gersho and R. M. Gray, *Vector quantization and signal compression*. Springer, 1992.

[4] A. Shama and S. Phadikar, "Automatic color image segmentation using spatial constraint based clustering," in *Emerging Trends in Computing and Communication*. Springer, 2014, pp. 113–121.

[5] G. Dong and M. Xie, "Color clustering and learning for image segmentation based on neural networks," *IEEE Transactions on Neural Networks*, vol. 16, no. 4, pp. 925–936, July 2005.

[6] T. Warren Liao, "Clustering of time series data-a survey," *Pattern Recognition*, vol. 38, no. 11, pp. 1857–1874, Nov. 2005.

[7] C.-P. Lai, P.-C. Chung, and V. S. Tseng, "A novel two-level clustering method for time series data analysis," *Expert Systems with Applications*, vol. 37, no. 9, pp. 6319–6326, 2010.

[8] N. Jardine and C. J. V. Rijsbergen, "The use of hierarchic clustering in information retrieval," *Information Storage and Retrieval*, vol. 7, pp. 217–240, 1971.

[9] H. X. Weili Wu, Shashi Shekhar, *Clustering and Information Retrieval*. Kluwer Academic Publishers, 2003.

[10] V. Estivill-Castro and I. Lee, "Argument free clustering for large spatial point-data sets via boundary extraction from delaunay diagram," *Computers, Environment and Urban Systems*, vol. 26, no. 4, pp. 315–334, 2002.

[11] W. C. Xiankun Yang, "A novel spatial clustering algorithm based on delaunay triangulation," *J. Software Engineering & Applications*, vol. 3, pp. 141–149, 2010.

[12] o. Dongquan Liu, Sourina, "Free-parameters clustering of spatial data with non-uniform density," in *IEEE conference on cybernetics and intelligent systems*, 2004, pp. 387 – 392.

[13] I. Rui Xu, Donald C. Wunsch, "Clustering algorithms in biomedical research: a review," *IEEE Reviews in Biomedical Engineering*, vol. 3, pp. 120–154, 2010.

[14] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review," *ACM Computing Surveys*, vol. 31, pp. 264–323, 1999.

[15] P. Berkhin, "A survey of clustering data mining techniques," in *Grouping multidimensional data*. Springer, 2006, pp. 25–71.

[16] A. K. Jain, "Data clustering: 50 years beyond k-means," *Pattern recognition letters*, vol. 31, no. 8, pp. 651–666, 2010.

[17] P. Rai and S. Singh, "A survey of clustering techniques," *International Journal of Computer Applications*, vol. 7, no. 12, pp. 156–162, 2010.

[18] U. Von Luxburg, "A tutorial on spectral clustering," *Statistics and computing*, vol. 17, no. 4, pp. 395–416, 2007.

[19] P. H. A. Sneath and R. R. Sokal, "Numerical Taxonomy," *Nature*, vol. 193, pp. 855–860, 1962.

[20] B. King, "Step-Wise Clustering Procedures," *Journal of The American Statistical Association*, vol. 62, pp. 86–101, 1967.

[21] P. S. Bradley, O. L. Mangasarian, and W. N. Street, "Clustering via concave minimization," *Advances in neural information processing systems*, pp. 368–374, 1997.

[22] D. Arthur and S. Vassilvitskii, "k-means++: The advantages of careful seeding," in *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*. Society for Industrial and Applied Mathematics, 2007, pp. 1027–1035.

[23] G. Mclanchan and T. Krishan, "The em algorithm and extensions," *Series in Probability and Statistics*, vol. 15, no. 1, pp. 154–156, 1997.

[24] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise." in *Proceedings of International Conference on Knowledge Discovery and Data Mining*, vol. 96, no. 34, 1996, pp. 226–231.

[25] W. E. Donath and A. J. Hoffman, "Lower bounds for the partitioning of graphs," *IBM Journal of Research and Development*, vol. 17, no. 5, pp. 420–425, 1973.

[26] L. Hagen and A. B. Kahng, "New spectral methods for ratio cut partitioning and clustering," *IEEE Transactions on Computer-aided design of integrated circuits and systems*, vol. 11, no. 9, pp. 1074–1085, 1992.

[27] A. Y. Ng, M. I. Jordan, Y. Weiss *et al.*, "On spectral clustering: Analysis and an algorithm," *Advances in neural information processing systems*, vol. 2, pp. 849–856, 2002.

[28] B. Nadler and M. Galun, "Fundamental limitations of spectral clustering," in *Advances in Neural Information Processing Systems*, 2006, pp. 1017–1024.

[29] G. W. Milligan and M. C. Cooper, "An examination of procedures for determining the number of clusters in a data set," *Psychometrika*, vol. 50, no. 2, pp. 159–179, 1985.

[30] C. Biernacki and G. Govaert, "Using the classification likelihood to choose the number of clusters," *Computing Science and Statistics*, pp. 451–457, 1997.

[31] M. M.-T. Chiang and B. Mirkin, "Intelligent choice of the number of clusters in k-means clustering: an experimental study with different cluster spreads," *Journal of classification*, vol. 27, no. 1, pp. 3–40, 2010.

[32] B. Mirkin, "Choosing the number of clusters," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 1, no. 3, pp. 252–260, 2011.

[33] A. Kolesnikov, E. Trichina, and T. Kauranne, "Estimating the number of clusters in a numerical data set via quantization error modeling," *Pattern Recognition*, vol. 48, no. 3, pp. 941–952, 2015.

[34] G. James and C. Sugar, "Finding the number of clusters in a dataset: An information-theoretic approach," *Journal of the American Statistical Association*, vol. 98, no. 463, pp. 750–764, 2003.

[35] L. Zelnik-Manor and P. Perona, "Self-tuning spectral clustering," in *Advances in neural information processing systems*, 2004, pp. 1601–1608.

[36] D. Doval, S. Mancoridis, and B. Mitchell, "Automatic clustering of software systems using a genetic algorithm," *International Workshop on Software Technology and Engineering Practice*, vol. 0, p. 73, 1999.

[37] H. He and Y. Tan, "A two-stage genetic algorithm for automatic clustering," *Neurocomputing*, vol. 81, no. 1, pp. 49–59, 2012.

[38] S. Saha and S. Bandyopadhyay, "A generalized automatic clustering algorithm in a multiobjective framework," *Applied Soft Computing*, vol. 13, no. 1, pp. 89–108, 2013.

[39] K. K. Pavan, A. A. Rao, and A. Rao, "An automatic clustering technique for optimal clusters," *International journal of Computer Sciene Engineering and Applications*, pp. 133–144, 2011.

[40] S. Das, A. Abraham, and A. Konar, "Automatic Clustering Using an Improved Differential Evolution Algorithm," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 38, pp. 218–237, 2008.

[41] A. Vattani, "k-means requires exponentially many iterations even in the plane," *Discrete and Computational Geometry*, vol. 45, no. 4, pp. 596–616, 2011.

[42] W. A. Trybulec, "Pigeon hole principle," *Journal of Formalized Mathematics*, vol. 2, no. 199, pp. 1–5, 1990.

[43] Bauldry and C. William, *Introduction to Real Analysis*. John Wiley and Sons, Inc., 2011.

[44] W. H. Rogers and J. W. Tukey, "Understanding some long-tailed symmetrical distributions," *Statistica Neerlandica*, vol. 26, no. 3, pp. 211–226, 1972.

[45] S. Foss, D. Korshunov, and S. Zachary, *An introduction to heavy-tailed and subexponential distributions*. Springer, 2011.

[46] C. Schluter and M. Trede, "Identifying multiple outliers in heavy-tailed distributions with an application to market crashes," *Journal of Empirical Finance*, vol. 15, pp. 700–713, 2008.

[47] P. Fränti and O. Virmajoki, "Iterative shrinking method for clustering problems," *Pattern Recognition*, vol. 39, no. 5, pp. 761–775, 2006.

[48] I. Kärkkäinen and P. Fränti, *Dynamic local search algorithm for the clustering problem*. University of Joensuu, 2002.

[49] A. Gionis, H. Mannila, and P. Tsaparas, "Clustering aggregation," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 1, no. 1, pp. 1–30, 2007.

[50] C. T. Zahn, "Graph-theoretical methods for detecting and describing gestalt clusters," *IEEE Transactions on Computers*, vol. 100, no. 1, pp. 68–86, 1971.

[51] L. Fu and E. Medico, "Flame, a novel fuzzy clustering method for the analysis of dna microarray data," *BMC bioinformatics*, vol. 8, no. 1, pp. 1–15, 2007.

[52] C. J. Veenman, M. J. T. Reinders, and E. Backer, "A maximum variance cluster algorithm," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 9, pp. 1273–1280, 2002.

[53] P. Franti, O. Virmajoki, and V. Hautamaki, "Fast agglomerative clustering using a k-nearest neighbor graph," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 11, pp. 1875–1881, 2006.

[54] N. X. Vinh, J. Epps, and J. Bailey, "Information theoretic measures for clusterings comparison: is a correction for chance necessary?" in *Proceedings of the 26th Annual International Conference on Machine Learning*. ACM, 2009, pp. 1073–1080.

[55] D. Dueck and B. J. Frey, "Non-metric affinity propagation for unsupervised image categorization," in *IEEE International Conference on Computer Vision*. IEEE, 2007, pp. 1–8.

PLACE
PHOTO
HERE

Guangtao Wang

PLACE
PHOTO
HERE

Qinbao Song