Journal of Computers Vol. 31 No. 1, 2020, pp. 119-133
doi:10.3966/199115992020023101010

Effective Hierarchical Cluster Analysis Based on New Clustering Validity Index

Er-Zhou Zhu[1], Yin-Yin Ju[1], Da-Wei Liu[1], Yang Li[1], Dong Liu[1], Zhu-Juan Ma[2*]

[1] School of Computer Science and Technology, Anhui University, Hefei 230601, China
ezzhu@ahu.edu.cn,{yinyinju, liudawei96h, liycs, dongliucn723}@gmail.com

[2] School of Economic and Technical, Anhui Agricultural University, Hefei 230036, China
zjmsjtu16@gmail.com

Received 28 May 2018; Revised 13 September 2018; Accepted 18 November 2018

Abstract. Clustering analysis plays an important role in finding natural structures of datasets. It is widely used in many areas, such as data mining, pattern recognition and image processing. By generating a set of nested partitions of datasets, hierarchical clustering algorithms provide more information than partitional clustering algorithms. However, due to the generated clustering hierarchies are too complex to analyze, many existing hierarchical clustering algorithms cannot properly process many non-spherical and overlapping datasets. Clustering validity index is the key technique for forming the optimal clustering partitions and evaluating the clustering results generated by clustering algorithms. However, many existing clustering validity indexes suffer from instability and narrow range of applications. Aiming at these problems, the traditional Average-Linkage hierarchical clustering algorithm is firstly improved for better processing the above irregular datasets. Then, a new clustering validity index (MSTI) is defined to stably and effectively evaluate the clustering results of the improved algorithm. Finally, the new algorithm for determining the optimal clustering number is designed by leveraging the improved Average-Linkage hierarchical clustering algorithm and the new MSTI. Experimental results have shown that our new clustering method is stable, accurate and efficient in processing many kinds of datasets.

Keywords: clustering validity index, hierarchical clustering, optimal clustering number

1 Introduction

Clustering analysis belongs to the unsupervised machine learning method. Similar to the principle of "birds of a feather flock together", clustering divides dataset into clusters in the absence of prior information [1]. It is widely used in many areas, such as data mining, pattern recognition and graph processing [2]. Researches on cluster analysis are mainly focusing on two directions: clustering algorithms and cluster validity indexes (CVIs) [3].

The aim of any clustering algorithm is to find the natural structure of target datasets. Up to now, many clustering algorithms are proposed, but they can be broadly classified into two categories: partitional clustering algorithms and hierarchical clustering algorithms [4]. Partitional clustering algorithms partition the input dataset into groups or clusters. Famous implementations of partitional clustering algorithms are K-means, K-medians, K-medoids and fuzzy C-mean (FCM) [5]. On the other hand, hierarchical clustering algorithms build nested partitions of datasets. Implementations of hierarchical clustering algorithms can be divided into two categories [6]: the divisive hierarchical clustering (DHC) [7] and the agglomerative hierarchical clustering (AHC) [8]. By generating a set of nested partitions of datasets, hierarchical clustering algorithms provide more information than partitional clustering algorithms. However, due to the generated clustering hierarchies are too complex to analyze, many existing hierarchical clustering algorithms cannot properly process many non-spherical and overlapping datasets.

[*] Corresponding Author

In this paper, based on the Euclidean geometry theory, we improve the traditional Average-Linkage hierarchical clustering algorithm. The improved algorithm is able to stably and effectively process many kinds of datasets, including non-spherical and overlapping datasets.

Clustering algorithms divide datasets into several clusters. However, the number of clusters in a dataset is usually in a fuzzy interval, it is difficult to determine the optimal clustering number (K_{opt}) in practice [9]. CVI has always been the focus of cluster analysis [10]. Researches of CVIs mainly use mathematical knowledge to model the validity index. For example, when the clustering number takes different values, the clustering results are evaluated respectively. As the optimal value of a CVI is taken, the corresponding clustering result is the optimal clustering partition of the dataset, and the corresponding clustering number is the K_{opt}.

By now, a variety of CVIs have been proposed and been applied to various fields. However, each CVI has its own advantages and limitations. There is no clustering algorithm integrated with certain CVI can handle all kinds of structured datasets [11]. For example, the DBI-index [12] is only suitable for measuring the clusters of datasets with “within-cluster compactness, between-cluster separation”. The I-index [13] is only suitable for dealing with some datasets with less clustering numbers. The DI-index [14] is too sensitive to noise data points. The smaller the COP-index [4], the better clustering results of datasets are acquired. Generally speaking, since the cluster size and density are different, most of the existing CVIs cannot process datasets with non-spherical distribution and datasets with a large number of outliers and overlapping very well. In this paper, based on the knowledge of cost spanning tree in graph theory, we designed a new clustering validity index (MSTI). The new MSTI is able to stably and effectively form the K_{opt} for many kinds of datasets.

Generally speaking, this paper makes the following contributions:

Improves the traditional Average-Linkage hierarchical clustering algorithm. Due to the generated clustering hierarchies are too complex to analyze, many existing hierarchical clustering algorithms cannot properly process many non-spherical and overlapping datasets. Based on the Euclidean geometry theory, the traditional Average-Linkage hierarchical clustering algorithm is improved for stably and effectively processing these irregular datasets.

Proposes a new clustering validity index—MSTI. By adopting the knowledge of cost spanning tree in graph theory, the minimum spanning trees among the clusters and maximal spanning tree within each cluster are constructed. By doing this, the MSTI is defined as the ratio of the cost of the minimum spanning tree among clusters to the cost of maximum spanning tree in each cluster. Under this circumstance, the K_{opt} is acquired when the above ratio reaches the biggest value.

Designs a new algorithm for optimizing and determining the $\mathbf{K_{opt}}$. This algorithm is designed by combining the revised Average-Linkage clustering algorithm and the new proposed MSTI. By this algorithm, optimal clustering numbers and optimal clustering partitions of many kinds of datasets can be effectively acquired. The experimental results have also shown that the new proposed algorithm is reliable and accurate while without incurring much runtime overhead on processing many kinds of datasets.

The remainder of this paper is organized as follows: Section 2 discusses the related work. Section 3 presents the new MSTI and the corresponding new hierarchical clustering algorithm for determining the K_{opt}. Section 4 evaluates the experimental results. Finally, Section 5 briefly concludes this paper and outlines our future work.

2 Related Work

Since the number of clusters in a dataset is usually in a fuzzy interval, it is difficult to determine the K_{opt} and the optimal clustering partition in practice. As an important method to get the optimal clustering numbers and the optimal clustering partitions for datasets, CVI plays an important role in the cluster analysis. Generally speaking, commonly used CVIs can be divided into 3 categories [1]: indexes based on the fuzzy division of datasets, indexes based on the statistical information of datasets and indexes based on the geometric structure of a datasets.

2.1 Fuzzy Division Based CVIs

Fuzzy division based CVIs are commonly used to evaluate the results of fuzzy clustering algorithms. WGLI [15] is a representative fuzzy division based CVI that utilizes the optimum membership as its global property and the modularity of bipartite network as its local independent property. Xie-Beni [16] is a famous CVI that combines the objective function of fuzzy clustering, the structure of the dataset itself and the nature of the fuzzy membership degree. By utilizing two factors, a normalized partition coefficient and an exponential separation measure for each cluster, Wu and Yang [17] creates the PCAES validity index. The fuzzy division based CVIs can objectively evaluate the clustering results, but they are not suitable to evaluate the results of hard clustering algorithms. The improved Average-Linkage hierarchical clustering algorithm in our paper is one of the hard clustering algorithms, so the fuzzy division based CVIs are not suitable to evaluate the results of our improved algorithm.

2.2 Statistical Information Based CVIs

Statistical information from datasets can be used to design CVIs. IGP [18] is a representative CVI based on the statistical information of datasets. It uses the in-group radio of the intra data points to evaluate the clustering results. Based on the risk calculated by loss functions and possibilities, Yu et al. [19] design a clustering validity evaluation function. This is an automatic method by extending the decision-theoretic rough set model to clustering. Since it only focuses on the adjacent consistency, this kind of CVIs is not stable for many datasets. This means the number of clusters generated by these indexes is usually less than the actual number.

2.3 Geometric Structure Based CVIs

Based on the geometric structure of a dataset, many CVIs have been proposed. The DBI-index [12] is presented which indicates the similarity of clusters. This index is suitable for measuring the clusters of datasets with “within-cluster compactness, between-cluster separation”. So the greater the overlap of dataset is, the worse the performance of DBI-index clustering evaluation. The DI-index [14] is used in detecting the compact well-separated clusters. This CVI is too sensitive to the noise data. It is difficult to find the K_{opt} for datasets with outliers. By calculating the average distance between the sample points of each cluster to its center, the COP-index [4] measures the compactness of the sample distribution within each cluster. The separability among clusters is measured by farthest distance. Accordingly, the smaller the COP-index, the better clustering results of dataset. The CH-index [20] is proposed for identifying clusters of points in a multi-dimensional Euclidean space. Through extensive contrasts in different datasets, we find that, in most cases, the CH-index is superior to most CVIs in evaluating the clustering performance and determining the K_{opt}. Yue et al. [10] firstly developed a new measure, called as dual center, to represent the separation among clusters. Then, according to the new defined measure, a validity index (SMV) is proposed for evaluating the clustering performance of partitional algorithms. The SMV-index exhibits high accuracy but narrow range of applications. The I-index [13] is suitable for dealing with some datasets with less clustering numbers, but it relies too much on preset parameters.

The optimal numbers of clusters derived by CVIs, DI-index, DBI-index, CH-index, I-index, COP-index, SMV-index, are based on the assumption that the optimal partition is already made for the target datasets [3]. As a matter of fact, the correctness of the clustering results is not verified. In addition, due to the random selection of initial clustering centers and different settings of parameters, even the same clustering algorithm will divide a single dataset into different clustering partitions. Many of the existing CVIs have good clustering performance for the dataset of “within-cluster compactness, between-cluster separation” and “each cluster presenting a spherical distribution”. However, most of them cannot properly deal with datasets with non-spherical distributions and datasets with a large degree of overlapping. A lot of studies [21] have shown that there is no CVI that can optimally process all datasets. Of course, there are already many CVIs for datasets with non-spherical distribution [22], different clusters with different sample sizes and density [23] and large degree overlap among clusters [24]. Compared with these CVIs, our new MSTI is able to process many kinds of datasets. Meanwhile, it incurs relatively lesser time cost and exhibits higher stability than above CVIs.

3 The New Hierarchical Clustering Method

In this section, the traditional Average-Linkage hierarchical clustering algorithm is firstly revised by utilizing the Euclidean geometry theory. Then, the new clustering validity index, MSTI, is given by utilizing the cost spanning tree in the graph theory. Finally, a new algorithm for optimizing and determining the K_{opt} is designed by leveraging the revised Average-Linkage clustering algorithm and the new proposed MSTI.

3.1 The Improved Average-Linkage Hierarchical Clustering Algorithm

In the Euclid space R^m, a dataset X={x_1, x_2, …, x_n} containing n samples is given. Where, each sample point x_i={x_{i1}, x_{i2}, …, x_{im}} has m feature attributes. By the clustering algorithm, X is divided into K clusters C={C_1, C_2, …, C_K}, and a K×n partition matrix, represented as $U(D)$=[u_{ki}], is derived. u_{ki} (k=1, 2, …, K; i=1, 2, …, n) is the degree of membership of sample point x_i to cluster C_k. Clustering can be divided into hard clustering and soft clustering (fuzzy clustering). Since the Average-Linkage hierarchical clustering algorithm used in this paper is in the category of hard clustering, the membership degree u_{ki} should be satisfied with:

$$u_{ki} = \begin{cases} 1, & if\ x_i \in C_k \\ 0, & otherwise \end{cases}. \tag{1}$$

Where, the clustering results must be satisfied with $C_i \neq \emptyset$, $X=C_1 \cup C_2 \cup ... \cup C_K$, $C_i \cap C_j = \emptyset$, $i \neq j$, i, j=1, 2, …, K.

Definition 1. The Euclid space distance between sample points x_i={x_{i1}, x_{i2}, …, x_{im}} and x_j={x_{j1}, x_{j2}, …, x_{jm}} and (marked as $d(x_i, x_j)$) can be calculated as (each sample point contains m feature attributes):

$$d(x_i, x_j) = \| x_i \text{-} x_j \| = \sqrt{(x_{i1} \text{-} x_{j1})^2 + ... + (x_{im} \text{-} x_{jm})^2}\ . \tag{2}$$

By clustering and partitioning the datasets at different levels, the hierarchical clustering finally forms a tree clustering structure. Generally, the hierarchical clustering utilizes two strategies to partition the target datasets, i.e. the “top down” splitting strategy and the “bottom up” merging strategy. In this paper, we use the later one. By this algorithm, each sample point in the target dataset is initially set as a cluster. Then, the distance between each of the two clusters pair is calculated, and the two clusters with the shortest distance are merged into a single cluster. The above steps are executed repeatedly until the target dataset is divided into K clusters.

Based on the above definitions, the key step of the Average-Linkage hierarchical clustering algorithm is to calculate the distance between each of the two clusters pair. By drawing the Euclid space distance between two sample points defined in Definition 1, the distance between clusters C_i and C_j is defined as Definition 2.

Definition 2. During the clustering and partitioning process of the Average-Linkage hierarchical clustering algorithm, the distance between clusters C_i and C_j (marked as $d_{avg}(C_i, C_j)$ can be calculated as the following formula:

$$d_{avg}(C_i, C_j) = \frac{1}{|C_i \| C_j|} \in_{x_k \in C_i} \in_{y_k \in C_j} d(x_k, y_t)\,. \tag{3}$$

In the formula, $|C_i|$ and $|C_j|$ represent the number of sample points in the cluster C_i and C_j respectively. $d(x_k, y_t)$ is the Euclid space distance between sample point x_k and y_t. Base on this definition, the traditional Average-Linkage hierarchical clustering algorithm is improved in Fig. 1. In this figure, each sample point in the target dateset X is initially treated as a single cluster (lines 1-2). Then, the distance between each of the two clusters is calculated according to Definition 2 (lines 3-5). The initial clustering number is set as the number of sample points in X (line 6). Repeatedly merges the nearest two clusters in to a single one until the number of the clusters reaches to K (lines 7-18). More specifically, line 8 is utilized to merge the two nearest clusters; lines 9-10 are utilized to renumber the clusters after two nearest clusters are merged; lines 11-16 are utilized to adjust the distances after the cluster merging. The improved Average-Linkage hierarchical clustering algorithm can properly resolve the datasets with non-spherical distribution and datasets with a large number of overlapping points.

```
Input: (1) Dataset $X=\{x_1, x_2, \ldots, x_n\}$; (2) Clustering number $K$;
       (3) The function $d_{avg}=(C_i, C_j)$ for calculating the distance between cluster $C_i$ and $C_j$.
Output: Dataset $X$ is divided into $K$ clusters: $C=\{C_1, C_2, \ldots, C_K\}$.
for $i$=1,2,…, $n$ do
    $C_i \leftarrow \{x_i\}$;
 for $i$=1,2,…, $n$-1 do
    For $j$=$i$+1, $i$+2,…, $n$ do
        $\Gamma(i, j) \leftarrow d_{avg}(C_i, C_j)$;
 $m \leftarrow n$; //set the initial clustering number
 while $m \neq K$ do
    Find the nearest two clusters: $C_i$* and $C_j$*, then $C_i$* ←$C_i$*∪$C_j$*;  // suppose $i^*<j^*$
    for $j= j^*+1, j^*+2, \ldots m$ do
      Renumbering the cluster $C_j$ as $C_{j-1}$;
    for $i$=1,2,…, $m$ do // Since cluster $C_j$* is deleted, the distance associate with this cluster is also delete.
      if $i<j^*$ delete $\Gamma(i, j^*)$
      else if $i>j^*$  delete  $\Gamma(j^*, i)$ ;
    for $j$=1,2,…, $m$ do //Since the original cluster $C_i$* is changed, recalculates the distance associate with new $C_i$*
      if $j< i^*$   $\Gamma(j, i^*) \leftarrow d_{avg}(C_j, C_i^*)$
      else if $j> i^*$   $\Gamma(i^*, j) \leftarrow d_{avg}(C_i^*, C_j)$;
    m←m-1;
 end while.
```

Fig. 1. The improved Average-Linkage hierarchical clustering algorithm

3.2 The Proposed MSTI

The traditional PRIM algorithm used for constructing the minimal and maximum cost spanning trees is firstly introduced. Then, the MSTI is designed to leverage the theory of Euclidean geometry and the PRIM algorithm.

PRIM algorithm. Supposing in a weighted connected graph G=(V, E), V and E represent the vertexes set and the edges set of G respectively. Each edges in E is assigned with a weight (in this subsection, the weight is omitted for simplicity). V_T and E_T are the vertexes set and the edges set of minimal (or maximum) cost spanning tree produced by the PRIM algorithm.

At the beginning of the PRIM algorithm, a vertex u_0 is selected randomly from V as the only element of V_T ($V_T=\{u_0\}$, $u_0 \in V$). Meanwhile, the edges set E_T is set null (E_T = Ø). Then, for all edges (u, v)∈E, where $u \in V$ and $v \in V\text{-}V_T$, the edge with the minimum (or maximum) cost ((u_i, v_i)) is selected and put into the edges set E_T.

Meanwhile, v_i is put into the vertexes set V_T. The above steps is repeatedly executed until V_T is equal to V (V_T=V). After the algorithm is completed, the minimum (or maximum) cost spanning tree T=(V_T, E_T) is constructed. Where, set E_T contains n-1 edges. Fig. 2 describes the pseudo code of the PRIM algorithm.

```
Input: a weighted connected graph $G$=($V$, $E$).
Output: the minimal (or maximum) cost spanning tree of $G_T$=($V_T$, $E_T$).
Initialize a empty tree: $E_T \leftarrow \Phi$, $V_T \leftarrow \Phi$;
Randomly select a vertex $u_0$ from $V$, and let $V_T = \{ u_0\}$;
while ($V$ - $V_T$)≠$\Phi$ do
    for all edges in $E$, the minimum / maximum cost edge ($u$, $v$) ($u$   $V_T$, $v$   $V$-$V_T$) is selected;
    $E_T \leftarrow E_T \cup \{(u, v)\}$;
    $V_T \leftarrow V_T \cup \{v\}$;
end while.
```

Fig. 2. The pseudo code of the PRIM algorithm

In the PRIM algorithm, the key step is to construct a weighted connected graph. To this end, the weighted connected graphs are constructed by connecting all the sample points in each cluster (for the maximum cost spanning trees) or by connecting all the clustering centers of all the clusters (for the minimum cost spanning tree). By doing this, the weighted connected graph contains n×(n-1)/2 edges (n is the number of sample points of a cluster or the number of the clusters in a dataset). Where, the weight of an edge is defined as the Euclidean distance between its associated two vertexes. The time complexity of the PRIM is O(n^2), which is irrelevant to the number of edges in the graph. By this feature, the PRIM algorithm is more suitable to construct the minimum (or maximum) cost spanning tree from the graph with dense edges.

The design of MSTI. Combing the Euclidean geometry theory and the PRIM algorithm, the new clustering validity index MSTI is designed for the Average-Linkage hierarchical clustering algorithm.

Definition 3. In the Euclid space R^m, a dataset $X=\{x_1, x_2, \ldots, x_n\}$ containing n samples is given. By the Average-Linkage hierarchical clustering algorithm, X is divided into K clusters $C=\{C_1, C_2, \ldots, C_K\}$, and a partition matrix, U_K, is derived. Then, the CVI V_K is used to evaluate the effect of the clustering. When the clustering parameter K takes different values, the corresponding partition matrix of the clustering results is U_i (i=2, 3,…,$\sqrt{n}$); its corresponding clustering index value is V_i (i=2, 3, …, $\sqrt{n}$). According to the property of the CVI, the optimal clustering partition of the dataset is obtained as follows:

$$U_m = U_2 \uparrow U_3 \uparrow \ldots \uparrow U_{\sqrt{n}} . \tag{4}$$

Where, $U_p=U_i\uparrow U_j$ indicates that the partition matrix of the optimal clustering result is assigned to U_p. Therefore, U_m in formula (4) is the optimal partition matrix of the dataset X, and m is the value of the K_{opt}.

Definition 4. In the Euclid space R^m, a dataset $X=\{x_1, x_2, \ldots, x_n\}$ containing n samples is given. By the Average-Linkage hierarchical clustering algorithm, X is divided into K clusters $C=\{C_1, C_2, \ldots, C_K\}$. Then, K weighted connected graphs $G=\{G_1, G_2, \ldots, G_K\}$ are constructed by connecting all the sample points in each cluster. In each graph, the weight of an edge is defined as the Euclidean distance between its associated two vertexes. For each graph G_i, i=1, 2, …, K, the PRIM algorithm described in Fig. 2 is used to construct the maximum cost spanning tree T_i, i=1, 2, …, K. By doing this, a trees set $T=\{T_1, T_2, \ldots, T_K\}$ that contains K maximum cost spanning trees is constructed. Finally, the compactness inner the cluster k (k=1, 2, …, K) can be evaluated as follows:

$$\theta_k = max\{V_{T_1}, V_{T_2}, \ldots, V_{T_K}\} . \tag{5}$$

Where, $V_{T_K} = \lambda_i /(|C_i|\text{-}1)$; λ_i is the sum of weights on all edges of T_i (constructed from cluster C_i); $|C_i|$ is the number of sample points in cluster C_i, i=1, 2, …, K.

Definition 5. In the Euclid space R^m, a dataset $X=\{x_1, x_2, \ldots, x_n\}$ containing n samples is given. By the Average-Linkage hierarchical clustering algorithm, X is divided into K clusters $C=\{C_1, C_2, \ldots, C_K\}$. Where, v_i is the clustering center point of C_i, i=1, 2, …, K. By connecting all the clustering center points v_i (i=1, 2, …, K), the weighted connected graph N is constructed. In this graph, the weight on each edge is defined by the Euclidean distance between its associated two vertexes. Then, the PRIM algorithm described in Fig. 2 is used to construct the minimal cost spanning tree T_N. Finally, the separation among the clusters can be evaluated as follows ($V_{T_N} = \delta /(K\text{-}1)$; δ is the sum of weight on all edges of T_N):

$$\varphi_K = V_{T_N} . \tag{6}$$

Definition 6. In the Euclid space R^m, a dataset $X=\{x_1, x_2, \ldots, x_n\}$ containing n samples is given. By the Average-Linkage hierarchical clustering algorithm, X is divided into K clusters $C=\{C_1, C_2, \ldots, C_K\}$. Where, v_i is the clustering center point of C_i, i=1, 2, …, K. By connecting all the clustering center points v_i (i=1, 2, …, K), the weighted connected graph N is constructed. In this graph, the weight on each edge is defined by the Euclidean distance between its associated two vertexes. Then, the PRIM algorithm described in Fig. 2 is used to construct the minimal cost spanning tree T_N. Finally, the new clustering validity index, MSTI, is defined as follows:

$$MSTI(K) = \frac{\varphi_K}{\theta_k} = \frac{V_{T_N}}{max\{V_{T_1}, V_{T_2}, \ldots V_{T_K}\}} . \tag{7}$$

Where, θ_k and φ_K are defined by formulas (5) and (6) respectively. The K_{opt} can be derived by the following means (in this formula, the number of clustering number follows the empirical rule: $2 \le K_{max} \le \sqrt{n}$):

$$K_{opt} = \{K \mid max_{2 \le K \le \sqrt{n}}\{MSTI(K)\}\} . \tag{8}$$

3.3 The K Value Determination and Optimization Algorithm Based on New MSTI

By combining the improved Average-Linkage algorithm and the new proposed MSTI, the new hierarchical clustering algorithm for determining and optimizing the *K* value is designed (as shown in Fig. 3). In the algorithm, the range of the maximum clustering number K_{max} is firstly specified by the number of sample points (*n*) in data set $X=\{x_1, x_2, \ldots, x_n\}$ and the empirical rule $2 \le K_{\max} \le \sqrt{n}$ (line 1). Then, for different *K* in $[2, \sqrt{n}]$, the value of *MSTI*(*K*) is calculated by the improved Average-Linkage hierarchical clustering algorithm shown in Fig. 1 and formula (7) (line 3-5). Lastly, the minimal *MSTI*(*K*), $K \in [2, \sqrt{n}]$, is selected (line 6-13). By doing so, the value of *K* is the corresponding K_{opt} of dataset *X*. Consequently, the dataset *X* is optimal divided into $C=\{C_1, C_2, \ldots, C_{Kopt}\}$ (line 14).

```
Input: (1) Dataset $X=\{x_1, x_2, \ldots, x_n\}$; (2) Clustering number $K$;
       (3) The function $d_{avg}=(C_i, C_j)$ for calculating the distance between cluster $C_i$ and $C_j$.
Output: (1) The optimal clustering number $K_{opt}$;
        (2) The optimal clustering partition $C=\{C_1, C_2, \ldots, C_K\}$ of dataset $X$.
1.  According to the number of sample points $n$ in dataset $X$ and the empirical rule, we can get: $2 \le K_{max} \le \sqrt{n}$ ;
2.  for $K=2,3,\ldots,\sqrt{n}$ do
3.      Using the improved Average-Linkage hierarchical clustering algorithm on dataset $X$;
4.      Evaluate the clustering result according to the new $MSTI(K)$ described as formula (7);
5.  end for;
6.  let $max \leftarrow MSTI(2)$;
7.  for $K = 3, 4, \ldots, \sqrt{n}$ do
8.      if $max < MSTI(K)$
9.          then $max \leftarrow MSTI(K)$;
10.             $K_{opt} \leftarrow K$;
11.         else keeping $max$ unchanged;
12.     end if;
13. end for;
14. The optimal clustering partition $C=\{C_1, C_2, \ldots, C_{Kopt}\}$ of the dataset $X$ is get when the clustering validity index
    reaches the maximum value. Consequently, the value of $K$ is the corresponding optimal clustering number $K_{opt}$.
```

Fig. 3. The new hierarchical clustering algorithm for determining K_{opt}

4 Experimental Results

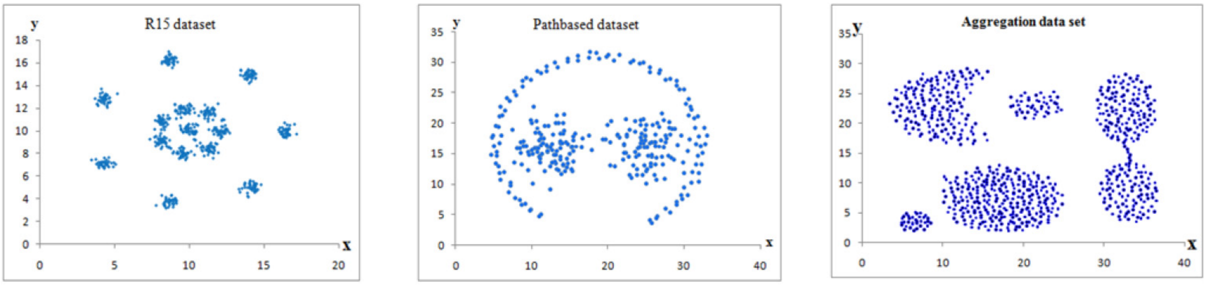
This section presents the detailed results of simulation experiments on verifying the performance of our new clustering method. The test environment of this section consists of an Intel Pentium CPU (E6700 at 3.2GHz), 2.0GB RAM and Windows 7 OS. Meanwhile, the MyEclipse 8.6 with jdk 1.8 is selected for running our Java programs. The tested datasets in this section consist of three simulated datasets (R15, Pathbased and Aggregation (http://cs.joensuu.fi/sipu/datasets/)) and three UCI machine learning databases (Iris, Pima and Seeds (http://archive.ics.uci.edu/ml/data sets.html)). A detailed description of the 6 experimental datasets is shown in Table 1.

Table 1. Description of tested datasets

Dataset name	Sample number	Clustering number (*K*)	Dimensions	Range of *K*
R15	600	15	2	2<=K<=24
Pathbased	300	3	2	2<=K<=17
Aggregation	754	6	2	2<=K<=27
Iris	150	3	4	2<=K<=12
Pima	768	2	8	2<=K<=27
Seeds	210	3	7	2<=K<=14

4.1 Spatial Distribution of Tested Datasets

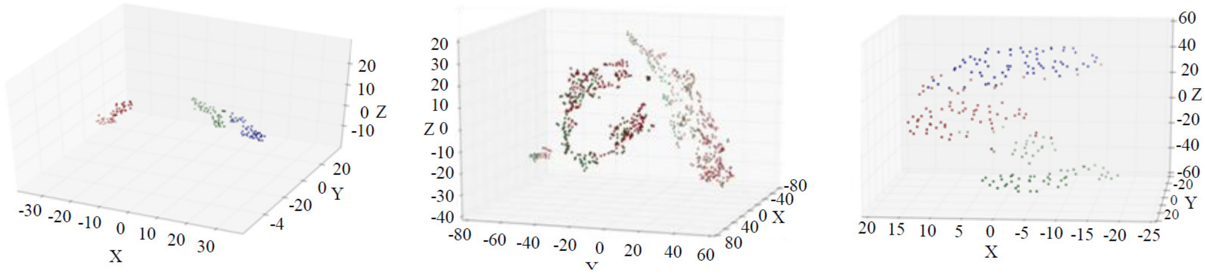
Fig. 4 shows the spatial distributions of the R15, Pathbased and Aggregation datasets. As Fig. 4(a) shows, there are 15 clusters in the R15 analog dataset. Among them, the peripheral seven clusters are characterized by “within-cluster compactness, between-cluster separable”. The internal eight clusters are characterized by “within-cluster compactness, between-cluster linear separable”. As shown in Fig. 4(b), the spatial distribution of Pathbased dataset is similar to “human face”. From Fig. 4(b), we can see that there are three clusters in this dataset. The spherical distributions of the internal two clusters are characterized by “within-cluster compactness, between-cluster linear separable”. The spatial distribution of the outside cluster exhibits a big arc. So, there is a great difference among the spatial distributions of different clusters. Fig. 4(c) describes the spatial distribution of the Aggregation dataset. From the figure, we can see that the dataset is comprised of 6 clusters, and each cluster has a uniform and compact distribution of sample points. However, the number of sample points contained by each cluster varies greatly.



(a) Pathbased (b) and Aggregation (c) datasets

Fig. 4. The spatial distribution of the R15

Fig. 5. shows the spatial distributions of the Iris, Pima and Seeds datasets. Iris is a commonly used experimental dataset for clustering and classification. It is collected by Fisher based on the characteristics of iris plants. The Iris dataset is divided into 3 clusters, 50 samples per cluster, and each sample has four attribute values. Since the Iris dataset is four dimensions, it is necessary to reduce the dimensions to be displayed in the low dimensional space. Currently, there are mainly two kinds of high dimensional data visualization tools, the linear dimensionality reduction tools and the nonlinear dimensionality reduction tools. In this paper, we select the widely used nonlinear dimensionality reduction tool T-SNE [25] to process datasets in Fig. 5. From Fig. 5(a), we can observe that the sample points of two clusters in the Iris dataset have a small degree of overlap. But the two clusters are linearly separable. The other one is far apart from these two clusters. From Fig. 5(b), we can see that the Pima dataset can be divided into two clusters. However, the large degree of overlap between the two clusters brings huge challenges to clustering algorithms and CVIs. Fig. 5(c) is a three-dimensional spatial distribution graph of the Seeds dataset after dimensionality reduction by the T-SNE method. Actually, the sample points in this dataset can be divided into three clusters, and each cluster represents a different wheat variety. However, the spatial distribution shown in Fig. 5(c) demonstrated that, among the three clusters, two clusters with near distance have large overlap and they are linearly separable to the third cluster. For this reason, 2 is the more reasonable optimal clustering number for this dataset.



(a) Pima (b) and Seeds (c) datasets

Fig. 5. The spatial distribution of the Iris

4.2 Effectiveness Evaluation

For the 6 datasets listed in Table 1, the empirical rule $K \leq \sqrt{n}$ is firstly used to get the range of *K*. Then, for different *K* in $[2, \sqrt{n}]$, our improved clustering algorithm is used to perform the clustering partition for different datasets. At last, the five CVIs are compared with each other to evaluate the effectiveness of the clustering results. Because of too much difference among the values of the 5 tested CVIs, it is needed to standardize the CVI values to facilitate displaying and analyzing the experimental results. In this paper, the following methods are adopted:

$$MaxI = max\{FI(1), FI(2), ..., FI(n)\} . \quad \textbf{(9)}$$

$$FI_S(K) = \frac{FI(K)}{MaxI} \times 40, \qquad 2 \leq K \leq \sqrt{n} . \quad \textbf{(10)}$$

Where, *FI*(*K*) is the clustering validity index function (there are *n* CVIs participated in the comparison). The inequality $2 \leq K \leq \sqrt{n}$ is the empirical rule. $FI_S(K)$ is the standardized CVI values which will be displayed and analyzed in the following subsections. Through the standardization of the above methods, the values of the 5 CVIs will be limited to the interval of [0, 40].

Table 2 lists the standardized CVI values of R15 dataset evaluated by 5 CVIs. Through the standardization of the methods of the formula (9) and (10) and the empirical rule $2 \leq K \leq \sqrt{n}$, values of different CVIs are limited to the interval [2, 40]. In this table, the values of *K* locate at the same row with the underlined bold index values are the K_{opt} calculated by different CVIs. As the optimal clustering number of the R15 dataset is 15 (K_{opt}=15), so CVIs, DBI-index and MSTI can get the optimal clustering partition. However, the I-index, COP-index and SVM-index cannot obtain the optimal clustering partition (that is, the optimal clustering numbers 7 and 8 they got are not the optimal clustering results of the R15 dataset).

Table 2. Standardized CVI values of R15 dataset

K	BDI	I	COP	SMV	MSTI
2	40.000	11.680	36.952	40.000	10.825
3	36.883	12.264	37.850	36.022	12.785
4	32.710	14.521	40.000	30.880	13.296
5	30.574	14.065	39.444	28.527	15.496
6	25.066	0.887	32.348	23.052	15.234
7	20.753	**40.000**	25.259	17.680	14.981
8	14.759	1.102	**18.805**	**11.810**	20.162
9	20.337	5.869	28.463	14.015	18.928
10	19.153	1.243	28.295	14.027	21.425
11	19.053	1.479	26.790	14.183	23.740
12	18.591	1.356	26.272	13.153	22.771
13	17.757	7.418	23.641	12.863	22.802
14	15.818	1.760	21.006	12.601	22.362
15	**13.329**	2.023	18.908	11.947	**40.000**
16	13.962	1.907	19.950	12.901	37.820
17	13.972	1.528	19.901	12.676	36.588
18	14.132	1.412	19.779	13.279	35.977
19	14.569	1.449	20.955	14.228	34.721
20	14.898	0.011	21.237	14.662	33.389
21	15.871	1.342	21.406	15.006	31.621
22	18.382	1.315	23.166	17.286	30.998
23	18.253	1.234	23.178	16.776	29.884
24	19.028	1.255	23.785	16.534	28.852

Table 3 lists the standardized CVI values of Pathbased dataset processed by different CVIs. In this table, index values with underlined bold fonts specify the K_{opt} that calculated by different CVIs. As a matter of fact, the optimal cluster number of the Pathbased dataset is 3. From Table 3, we can see that only the cluster validity index MSTI proposed in this paper can get the optimal clustering partition of this dataset. The I-index can obtain the near optimal cluster partition. However, DBI-index, COP-index and SMV-index cannot get the optimal clustering number.

Table 3. Standardized CVI values of Pathbased dataset

K	DBI	I	COP	SMV	MSTI
2	0.8007	**4.4093**	0.3735	0.7265	0.6635
3	0.6309	2.6375	0.3176	0.6272	**0.9780**
4	0.6237	2.7725	0.3069	0.6423	0.9757
5	**0.5724**	2.5361	**0.2955**	0.5978	0.8664
6	0.6029	1.8025	0.3047	**0.5346**	0.7620
7	0.7052	0.2634	0.3169	0.6143	0.8648
8	0.6831	0.9326	0.3160	0.5527	0.8079
9	0.7114	0.1064	0.3291	0.6141	0.9330
10	0.6860	0.2739	0.3252	0.6367	0.8626
11	0.6595	0.0735	0.3144	0.6328	0.8393
12	0.6956	0.0725	0.3372	0.6485	0.8448
13	0.6918	0.3966	0.3347	0.6384	0.8214
14	0.6751	0.0643	0.3263	0.6500	0.7868
15	0.6563	0.0685	0.3200	0.6396	0.7388
16	0.7142	0.0840	0.3501	0.6623	0.8572
17	0.6952	0.2096	0.3428	0.6436	0.8062

Table 4 gives the standardized CVI values of Aggregation dataset processed by different CVIs. In this table, index values with underlined bold fonts specify the optimal cluster numbers K_{opt} calculated by different CVIs. As a matter of fact, the optimal cluster number of the Aggregation dataset is 6. From Table 4, we can see that CVIs, DBI-index, COP-index and MSTI, can get the optimal clustering number. The SMV-index can obtain the near optimal cluster partition. However, the I-index cannot get the optimal clustering number.

Table 4. Standardized CVI values of Aggregation dataset

K	BDI	I	COP	SMV	MSTI
2	40.000	38.474	40.000	40.000	26.85
3	27.386	34.562	33.786	31.581	31.102
4	24.102	**40.000**	30.732	31.645	38.939
5	22.736	1.052	29.193	**27.508**	35.795
6	**20.857**	8.516	**27.800**	30.036	**40.000**
7	24.342	1.965	29.943	30.174	33.762
8	29.379	12.752	34.767	29.639	32.470
9	31.122	7.071	35.406	29.001	35.564
10	30.349	0.859	34.690	30.311	33.450
11	32.425	7.120	36.564	30.704	31.372
12	34.848	1.948	38.697	33.067	32.199
13	35.947	2.003	39.205	33.348	31.972
14	36.783	0.775	39.495	33.956	30.908
15	37.342	1.964	39.584	33.862	29.527
16	36.070	0.743	39.366	34.777	28.679
17	36.349	1.868	39.881	34.493	32.596
18	36.078	0.914	39.486	34.430	31.580
19	35.726	1.747	38.966	34.131	33.908
20	34.876	1.731	38.320	34.456	33.250
21	34.841	1.900	38.153	34.296	31.996
22	35.125	0.662	38.385	34.543	32.188
23	35.128	0.839	38.570	34.483	31.614
24	35.040	0.300	38.508	34.536	31.680
25	35.259	0.699	38.425	35.117	30.402
26	35.411	0.381	38.060	35.376	30.824
27	35.795	0.677	38.387	35.656	30.183

Table 5 lists the experimental results of the standardized CVI values of Iris dataset processed by different CVIs. As shown in Fig. 5(a), there are three clusters in this data. But the sample points of two clusters in the Iris dataset have a small degree of overlap. For this reason, CVIs, DBI-index, COP-index and SMV-index, treat them as a single one. Thus, they only obtain the near optimal cluster partition for the dataset. As a matter of fact, the overlapped two clusters in the Iris dataset are linearly separable. It is more reasonable for the dataset being divided into 3 clusters. In this experiment, the I-index and our proposed MSTI can get the optimal clustering number for this dataset.

Table 5. Standardized CVI values of Iris dataset

K	DBI	I	COP	SMV	MSTI
2	**0.3836**	0.3751	**0.1767**	**0.4497**	1.1869
3	0.6588	**0.4422**	0.2525	0.4568	**1.2462**
4	0.6267	0.3535	0.2726	0.5117	0.9936
5	0.6860	0.3298	0.2848	0.5857	0.9623
6	0.6393	0.0881	0.2860	0.5435	0.9061
7	0.7363	0.1328	0.3169	0.6254	0.8882
8	0.7965	0.1246	0.3148	0.6226	0.8429
9	0.8246	0.0097	0.3500	0.5793	0.8994
10	0.7773	0.0847	0.3593	0.5290	0.8829
11	0.7896	0.0818	0.3670	0.5638	0.8582
12	0.8352	0.1664	0.3644	0.6023	0.9010

Table 6 lists the experimental results of Pima dataset evaluated 5 tested CVIs. In this table, index values with underlined bold fonts specify the optimal cluster numbers K_{opt} calculated by different CVIs. Fig. 5(b) has shown that it is more reasonable to divide this dataset into 2 clusters. From Table 6, we can see that CVIs, DBI-index, COP-index and SMV-index and MSTI can get the optimal clustering number. The I-index could only get the near optimal clustering partition for this dataset.

Table 6. Standardized CVI values of Pima dataset

K	BDI	I	COP	SMV	MSTI
2	**11.366**	0.179	**13.300**	**20.059**	**40.000**
3	24.692	**40.000**	27.660	30.266	36.155
4	25.134	0.755	28.230	32.678	34.354
5	22.425	1.353	28.190	34.338	29.270
6	28.479	12.52	30.741	37.758	32.371
7	27.679	10.12	30.658	34.972	30.804
8	29.756	8.937	30.806	37.039	28.665
9	30.239	7.866	33.358	36.878	27.623
10	32.760	4.850	36.879	38.381	29.988
11	34.231	0.046	36.979	37.929	28.315
12	34.530	1.555	35.205	39.170	27.498
13	38.969	0.042	36.204	40.000	27.371
14	38.317	0.330	36.906	39.900	26.409
15	36.823	4.810	36.899	37.614	25.318
16	33.864	0.000	36.839	33.296	23.877
17	34.874	0.381	38.784	33.874	24.159
18	36.538	0.000	40.000	34.192	24.972
19	36.609	1.204	39.919	34.423	24.305
20	36.055	0.381	39.994	34.626	24.127
21	36.321	0.056	39.499	35.345	26.274
22	37.235	0.000	39.530	35.658	25.634
23	39.283	0.000	39.406	37.889	24.914
24	40.000	0.263	39.555	38.308	27.572
25	39.205	0.264	39.450	37.806	27.338
26	38.637	0.018	39.378	36.597	27.143
27	37.111	0.245	39.264	34.117	26.901

The Seeds dataset has 3 clusters. But the spatial distribution shown in Fig. 5(c) has demonstrated that, among the three clusters, 2 clusters with near distance have large overlap and they are linearly separable to the third one. So, it is reasonable to divide this dataset into 2 clusters. Table 7 lists the experimental results of Seeds dataset evaluated 5 tested CVIs. In this table, index values with underlined bold fonts specify the optimal cluster numbers K_{opt} calculated by different CVIs. From the experimental results, we can see that CVIs, DBI-index, I-index and MSTI can get the K_{opt}. The COP-index and the SMV-index cannot get the optimal clustering partition for this dataset.

Table 7. Standardized CVI values of Seeds dataset

K	BDI	I	COP	SMV	MSTI
2	**0.6381**	**2.2854**	0.3179	0.6796	**0.9462**
3	0.7604	1.0533	0.3087	0.6333	0.9109
4	0.7149	0.0250	**0.3004**	0.5989	0.8366
5	0.7203	0.5640	0.3091	**0.5814**	0.7821
6	0.8672	0.0423	0.3730	0.6936	0.7910
7	0.7730	0.0835	0.3315	0.6670	0.7297
8	0.8428	0.1066	0.3468	0.6756	0.7571
9	0.8340	0.1350	0.3558	0.7154	0.7007
10	0.8407	0.0745	0.3611	0.7088	0.8253
11	0.8147	0.0195	0.3571	0.6874	0.8099
12	0.8067	0.1114	0.3541	0.6973	0.7710
13	0.7680	0.0125	0.3521	0.6656	0.7625
14	0.7642	0.0894	0.3587	0.6692	0.7466

From Table 2-Table 7, we can see that, for all tested datasets with different spatial distributions, the proposed MSTI can get the optimal clustering number K_{opt}.

4.3 Performance Evaluation

In order to better display the 5 CVIs' execution time on processing the six different datasets, the following standard methods are adopted:

$$ExeMax = max\{Exe(1), Exe(2),..., Exe(n)\} . \qquad \textbf{(11)}$$

$$STDExe(k) = \frac{Exe(k)}{ExeMax} \times 40, \qquad 2 \le k \le n . \qquad \textbf{(12)}$$

Where, *Exe(k)* is the execution time of certain dataset processed by CVI *k* (there are *n* CVIs participated in the comparison); *STDExe(k)* is the standardized execution time processed by CVI *k*. Through the standardization of the above methods, the execution time of each CVI on certain dataset will be limited to the interval of [0, 40].

Fig. 6 shows the efficiency of the 5 CVIs in solving the optimal clustering numbers of the 6 experimental datasets. From the figure, we can see that our proposed MSTI do not incurs more execution time on resolving the optimal clustering numbers compared with the other 4 CVIs.

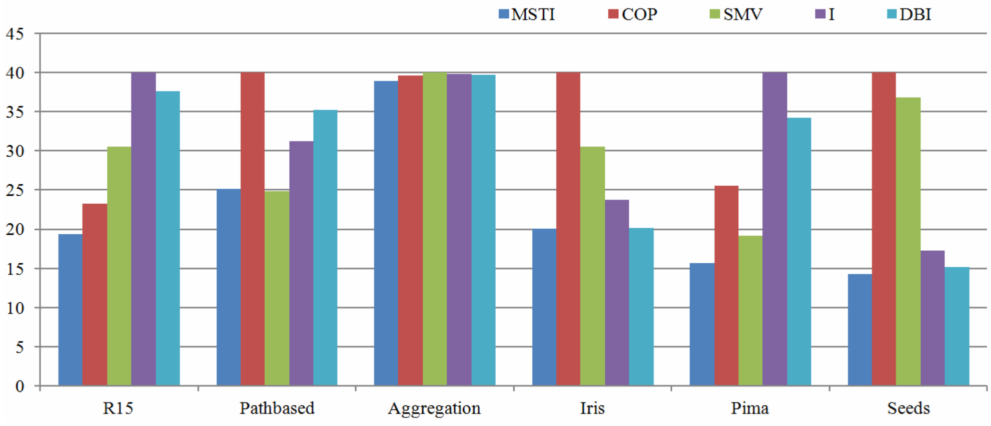


Fig. 6. The standardized execution time comparison of different CVIs on processing different datasets

4.4 Accuracy Evaluation

The third column of Table 8 lists the clustering accuracy rates when the proposed algorithm (shown in Fig. 3) is used to process different testing datasets. From the results, we can conclude that, no matter the tested datasets with non-spherical distribution, overlap among different clusters, density and number of sample points vary greatly among different clusters, our proposed algorithm can get high clustering accuracy rate.

Table 8. Accuracy evaluation on different datasets

Dataset	Clustering number	Accuracy rate
R15	15	99.50%
Pathbased	3	87.67%
Aggregation	6	99.60%
Iris	3	94.67%
Pima	2	76.10%
Seeds	3	90.95%

In order to verifying the correctness of the proposed MSTI, 12 other datasets from UCI machine learning da tabases (http://archive.ics.uci.edu/ml/datasets.html) are tested. Table 9 lists the experimental results. In the table, "*SamNum*", "*AttrNum*", and "*CluNum*" represent for "*Sample Points Number*", "*Attribute Number*" and "*Cluster Number*" respectively. From this table, we can see that the proposed MSTI exhibits relatively high accuracy. As there is no label for each sample, we cannot evaluate the accuracy of "S1" dataset by MSTI. Meanwhile, except "Glass", it can get all the optimal clustering numbers.

Table 9. Accuracy evaluation on 12 other UCI datasets

Datasets	SamNum	AttrNum	CluNum	K_{opt}	MSTI	Accuracy
Cancer	699	9	2	2	0.7467	94.37%
Haberman	306	3	2	2	1.0146	89.95%
Glass	214	9	7	6	1.0323	68.85%
Breast Tissus	106	9	6	6	6.305	72.91%
Parkinsons	195	22	2	2	1.5116	85.38%
Ecoli	336	7	8	8	0.8647	79.76%
Spectf	267	44	2	2	0.9982	81.04%
Energy_efficiency	768	8	12	12	19.879	93.75%
statlog(German)	1000	24	2	2	1.103	91.00%
S1	5000	2	15	15	1.2673	No Label
Ionosphere	351	34	2	2	0.8543	74.39%
Libras movement	360	90	15	15	0.6987	81.39%

5 Conclusion and Future Works

When the traditional cluster partition algorithms solve the clustering problems, it is necessary to set the value of the clustering number *K* in advance. But in practice, the numbers of clusters (*K* values) are usually in a fuzzy interval. This seriously limits the further application of the traditional cluster partition algorithms. For the purpose of elevating the stability and enlarging the application scope of clustering, this paper proposed an effective hierarchical clustering algorithm based on the new designed clustering validity index (MSTI). In the algorithm, the traditional Average-Linkage hierarchical clustering algorithm was firstly improved for better processing the irregular datasets. Then, the new MSTI was defined by utilizing the cost spanning trees in the graph theory. Lastly, the new clustering algorithm was proposed by integrating the revised Average-Linkage hierarchical clustering algorithm and the new designed MSTI. The experimental results demonstrated that the new algorithm was accurate while without scarifying much time cost. However, the proposed algorithm does not solve the intrinsic defects of the traditional hierarchical clustering algorithm, so it still cannot process the dataset with a large number of noise points. The efficiency of the algorithm still needs to be improved when it is used to process large scale datasets. Therefore, these defects and deficiencies should be improved as our future work.

Acknowledgements

This work was supported by the National Natural Science Foundation of China [Grant No. 61300169] and the Natural Science Foundation of Education Department of Anhui province (China) [Grant No. KJ2018A0022].

Reference

[1] E.-Z. Zhu, R.-H. Ma, An effective partitional clustering algorithm based on new clustering validity index, Applied Soft Computing 71(2018) 608-621.

[2] J.-H. Huang, Z.-L. Yu, Z.-G. Gu, A clustering method based on extreme learning machine, Neurocomputing 277(2018) 108-119.

[3] O. Arbelaitz, I. Gurrutxaga, J. Muguerza, J.M. Pèrez, I. Perona, An extensive comparative study of cluster validity indices, Pattern Recognition 46(2013) 243-256.

[4] I. Gurrutxaga, I. Albisua, O. Arbelaitz, J.I. Martín, J. Muguerza, J.M. Pérez, I. Perona, SEP/COP: An efficient method to find the best partition in hierarchical clustering based on a new cluster validity index, Pattern Recognition 43(10)(2010) 3364-3373.

[5] S.-B. Zhou, Z.-Y. Xu, A novel internal validity index based on the cluster centre and the nearest neighbour cluster, Applied Soft Computing 71(2018) 78-88.

[6] M. Roux, A comparative study of divisive and agglomerative hierarchical clustering algorithms, Journal of Classification 35(2018) 345-366.

[7] S.-N. Li, W.-J. Li, J. Qiu, A novel divisive hierarchical clustering algorithm for geospatial analysis, International Journal of Geo-Information 6(1)(2017) Article No. 30.

[8] S. A. Mondal, An improved approximation algorithm for hierarchical clustering, Pattern Recognition Letters 104(2018) 23-28.

[9] A. B. Said, R. Hadjidj, S. Foufou, Cluster validity index based on Jeffrey divergence, Pattern Analysis and Applications 20(1)(2017) 21-31.

[10] S. Yue, J. Wang, J.-J. Wang, X. Bao, A new validity index for evaluating the clustering results by partitional clustering algorithms, Soft Computing 20(3)(2016) 1127-1138.

[11] R. Kashef, M. S. Kamel, Cooperative clustering, Pattern Recognition 43(6)(2010) 2315-1329.

[12] D.L. Davies, D.W. Bouldin, A cluster separation measure, IEEE Transaction on Pattern Analysis and Machine Intelligence 1(2)(1979) 224-227.

[13] U. Maulik, S. Bandyopadhyay, Performance evaluation of some clustering algorithms and validity indices, IEEE Transactions on Pattern Analysis and Machine Intelligence 12(24)(2001) 1650-1654.

[14] J.C. Dunn, A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters, Journal of Cybernetics 3(3)(1973) 32-57.

[15] D.-W. Zhang, M. Ji, J. Yang, A novel cluster validity index for fuzzy clustering based on bipartite modularity, Fuzzy Sets and Systems 253(2014) 122-137.

[16] X.L. Xie, G. Beni, A validity measure for fuzzy clustering, IEEE Transactions on Pattern Analysis and Machine Intelligence 13(8)(1991) 841-847.

[17] K.-L. Wu, M.-S. Yang, A cluster validity index for fuzzy clustering, Pattern Recognition Letters 26(2005) 1275-1291.

[18] A.V. Kapp, Are clusters found in one dataset present another dataset?, Biostatistics 8(1)(2007) 9-31.

[19]H. Yu, Z.-G. Liu, G.-Y. Wang, An automatic method to determine the number of clusters using decision-theoretic rough set, International Journal of Approximate Reasoning 55(2014) 101-115.

[20] T. Calinski, J. Harabasz, A dendrite method for cluster analysis, Communications in Statistics 3(1)(1974) 1-27.

[21] P.-L. Lin, P.-W. Huang, C.-H. Wu, S.M. Huang, An efficient validity index method for datasets with complex-shaped clusters, in: Proc. the 2016 Int. Conf. on Machine Learning & Cybernetics, 2016.

[22] N.R. Pal, J. Biswas, Cluster validation using graph theoretic concepts, Pattern Recognition 30(6)(1997) 847-857.

[23] T. Pei, A. Jasra, D.J. Hand, A.-X. Zhu, C. Zhou, DECODE: a new method for discovering clusters of different densities in spatial data, Data Mining and Knowledge Discovery 18(3)(2009) 337-369.

[24] J. Wu, H. Yuan, H. Xiong, G. Chen, Validation of overlapping clustering: a random clustering perspective, Information Sciences 180(22)(2010) 4353-4369.

[25] L.V.D. Maaten, G. Hinton, Visualizing data using t-SNE, Journal of Machine Learning Research 9(2605)(2017) 2579-2605.