Deep Learning and Analyses of Clustering Algorithms

Yang Li

School of Electronic Engineering, Xidian University, Xi’an, 710126, China

Abstract: The research actuality and new progress in clustering algorithm in recent years are summarized in this paper. First, the analyses and induction of some representative clustering algorithms have been made from several aspects, such as the ideas of algorithm, key technology, advantage and disadvantage. On the other hand, several typical clustering algorithms and known data sets are selected simulation experiments are implemented from both sides of accuracy and running efficiency, and clustering condition of one algorithm with different data sets is an analyzed by comparing with the same clustering of the data set under different algorithms. Finally, the research hotspot, difficulty shortage of the data clustering and some pending problems are addressed by the integration of the aforementioned two aspects information. The above work can give a valuable reference for data clustering and data mining.

Keywords: Clustering algorithm, k-means, Data sets, Refining initial points, Cluster

1. Introduction

Clustering algorithm has been researched for decades, at the same time, clustering is indispensable to research of data mining and pattern recognition. For pattern recognition, clustering mainly used as speech recognition and character recognition. In machine learning, clustering algorithm is applied to image segmentation and machine vision. For image processing, clustering algorithm is applied to data compression and message retrieval, and another important usage of clustering is to apply into data mining, sequence, heterogeneous data analyses, etc.

In this paper, we analyzed clustering algorithm representatively that proposed in recent years about algorithm ideas, key technologies, advantages and disadvantages. And we choose some well-known data sets in experiments, after that, we made conclusion in terms of the analyses[1-3].

Section 1 is introduction of clustering; cluster process and algorithms; section 2 focuses on seventeen representative algorithms; section 3 described eight results of simulation about clustering algorithm, and combined reference [4] for analyses; section 4 is conclusion.

2. Clustering and categories of Algorithm

2.1 Concept of Clustering and Cluster Process

There is no accepted definition of clustering in academia so far. In this paper, we show one definition mentioned by Everitt[5] in 1974: Entities in the same class clusters are analogous, entities in different class clusters are dissimilar;

\[ J(W, P) = \sum_{i=1}^{n} \left( \sum_{j=1}^{n} \| x_i - p_j \|^2 \right) \]

Typical clustering processes include data preparation, feature selection and extraction, proximity calculating, clustering (Grouping), making effectively evaluation of clustering results[1,2,3].

Clustering process;
1) Data Preparation: Standardize features and reduce dimensionality.
2) Feature Selection: Choose the most effective feature
from the initial feature, and store it into vector.
3) Feature Extraction: Transform the feature which was selected, in order to shape a new prominent feature.
4) Clustering(Grouping): Choosing a suitable characteristic distance function, measure its closeness, then clustering or grouping.

2.2 Category of Clustering Algorithm

There is no clustering algorithms could be generally used to reveal various structures appeared from multidimensional data aggregation. Clustering algorithm includes several classifications, in this paper, we classify this algorithm into hierarchical clustering algorithm, disconnected clustering algorithm, clustering algorithm based on density and grid, and other clustering algorithm.

3. Clustering Algorithms

3.1 Hierarchical clustering algorithm

Hierarchical clustering algorithm is called Trees clustering algorithm\(^8,9\)\). It repeatedly retrievals or clusters data by means of hierarchical structure, in order to form a clustering answer to hierarchical sequence. The complexity of computation is \(O(n^3)\), this algorithm is used in classification of small data sets.

Assume a sample set \(S = \{O_1, O_2, ..., O_n\}\) has \(n\) samples in all

**HA1[Initialization].** Regard every sample \(o_i\) as a class; /*form \(n\) classes in all: \(o_1, o_2, ..., o_n\)*/

**HA2[Figure out two nearest classes].**

\[
dis \tan ce(o_i, o_j) = \min_{o_k, o_l \in S, o_k = o_l} \; dis \tan ce(o_k, o_l) ;
\]

/* Locate the nearest two classes from all classes \(o_i\) and \(o_k\)*/

**HA3[Cluster \(o_i\) and \(o_k\)].** Then we get a new class \(o_{i+k}\);
/*Existing class will be reduced by 1*/

**HA4.** If all the samples belong to a same class, end this algorithm; otherwise, back to HA2.

3.2 Traditional clustering rules

Methods of measure the distance between two classes is one important parts of traditional hierarchical aggregation algorithm. In this paper, we use Euclidean distance to measure similarity. Connection rules contain single connection rules, fully connection rules, average connection between class rules, average connection within one class rules, and Ward rules\(^10\) (where \(||x-y||\) is Euclidean norm, \(n_i\) and \(n_k\) are the numbers of samples in class \(o_i\) and \(o_k\), total methods of two different elements chosen from \(n_i+n_k\) denoted as \(C(n_i+n_k, 2)\));

Single-connection clustering rules:

\[
d(0_i, 0_k) = \min_{x\in o_i, y\in o_k} ||x-y|| ;
\]

Full-connection clustering rules:

\[
d(0_i, 0_k) = \max_{x\in o_i, y\in o_k} ||x-y|| ;
\]

Average-connection between multiple classes rules:

\[
d(0_i, 0_k) = 1/n_i \sum_{x\in o_i} (\sum_{y\in o_k} ||x-y||) ;
\]

Average-connection within one class rules:

\[
d(0_i, 0_k) = 1/C(n_i+n_k, 2) \sum_{x\in o_i, y\in o_k} ||x-y|| ;
\]

Ward rules: \(d(0_i, 0_k) = (1/(n_i+n_k)) \sum_{x\in o_i, y\in o_k} ||x-n||^2\),

where \(n\) is the center of fusion clustering.

3.3 New Hierarchical clustering algorithm

(1) Binary-Positive

In 2007, Glebard\(^4\) et al. suggested a new hierarchical clustering algorithm, which is called Binary-Positive. There are many methods represented in Dice to measure various Binary-Positive similarity\(^10,11\).

Glebard et al. adopted these four data sets: Wine, Iris, Ecologic and Psychology balance, to experiment between eleven kinds of clustering algorithms. The results show that all the algorithms are well-used in experiments.

(2) Rough clustering of sequential data(RCOSD)

In 2007, Kumar\(^12\) et al. argued a new hierarchical clustering algorithm based on indistinguishable crude aggregation: RCOSD. This algorithm introduce \(S'M\) as a way to measure similarity. This algorithm could merger more than two classes every time, so it could accelerate the

**Figure 1:** The classification chart of clustering algorithms

**Volume 4 Issue 12, December 2015**
speed of hierarchical aggregation.

The results show that RCOSD is available, this algorithm could help Webers discriminate potentially significant user groups.

3.4 Partition clustering algorithm

Partition clustering algorithm should specify the number of clusters or center of clusters in advance. By using duplicate iteration calculations, gradually reduce the error of the objective function. When the value of the objective function converges, we get final result of clustering.

3.4.1 K-means clustering algorithm

In 1967, Mac Queen mentioned K-means clustering algorithm for the first time, so far, several clustering missions have chosen this classical algorithm. The main idea of this algorithm is to find out K clustering centers: \( c_1, c_2, ..., c_k \) in order to minimize every data points \( x_i \) and the sum of square of the distance of the closest clustering center \( c_i \) (This sum of square of the distance is called deviation \( D \)).

K-means clustering algorithm\(^{[10]} \) (clustering of \( n \) samples)

**K1[Initialization].** Assign K clustering centers randomly\((c_1, c_2, ..., c_k)\);

**K2[Assign x].** For every sample \( x_i \), find the closest clustering center \( c_v \), and assign \( x_i \) into class \( c_v \) that indicated;

**K3[Correct c_v].** Put every \( c_v \) into indicated class center;

**K4[Calculate deviation].**

\[
D = \sum_{i=1}^{n} \left[ \min_{v=1...K} d(x_i, c_v) \right]^2;
\]

**K5[D is convergent or not?].** If \( D \) is convergent, then return\((c_1, c_2, ..., c_k)\) and end this algorithm; Otherwise, back to K2.

Advantage: K-means clustering algorithm could classify large-scale dataset efficiently, and this algorithm calculates faster than Hierarchical clustering algorithm.

3.4.2 K-modes algorithm

(1) K-modes-Huang algorithm\(^{[14]} \)

Introduction of Means and Modes:

In K-means algorithm, mean is center of clusters, and can be defined randomly at first. In K-modes algorithm, mode is defined as: data set \( X = \{x_1, x_2, ..., x_n\}, \forall x \in X \) described by \( m \) categorical attributes \( \{A_1, A_2, ..., A_m\} \), \( X \) is denoted as vector \( <x_1, x_2, ..., x_m>\); \( Q \) is one of the modes \( X \); \( Q \) is denoted as vector \( <q_1, q_2, ..., q_m>\), and \( Q \) satisfies \( \sum_{i=1}^{m} d_i(X_i, Q) \) minimum. \( d_i(X_i, Q) \) is distance between \( X_i \) and \( Q \).

Same as K-means algorithm, K-modes algorithm is also able to produce locally optimal solution, relying on the choice of modes initialization and sequence of centralized data.

In 1999, Huang\(^{[15]} \) et al. made the conclusion K-means algorithm can only converge local minimum.

(2) K-modes-CGC algorithm\(^{[16]} \)

In 2001, Chaturvedi et al. suggested a new nonparametric clustering method, which was called K-modes-CGC algorithm. This algorithm is similar to traditional K-means algorithm (the prior algorithm). K-modes-CGC algorithm optimizes one loss function based on norm \( L_p \).

In Monte Carlo simulation, Chaturvedi et al. used K-modes-CGC\(^{[17]} \) and latent class algorithm to restore a known latent class structure. The results show that, both algorithm have equivalent efficiency.

In 2003, Huang\(^{[18]} \) et al. proved that K-modes-CGC algorithm is equivalent to K-modes-Huang algorithm.

3.4.3 Refining initial points for K-modes

In 2002, Sun\(^{[19]} \) et al. applied refining initial points for K-means suggested by Bradley\(^{[20]} \) into K-modes (Huang, 1998). Sun et al. proposed an experiment based on refining initial points for K-modes. Sun’s experiment based on well-known soybean disease dataset. Data of soybean disease contain 47 records, every record was described by 35 characteristics, and every record was signed one of the following four diseases: Diaporthe StemCanker, Charcoal Rot, Rhizoctonia Root Rot, and Phytophthora Rot. Apart from Phytophthora Rot has seventeen records, other three diseases have ten records. There are two programs about K-modes:

Program 1: Randomly select initial point sets;
Program 2: Use refining initial points for K-modes to select initial point sets.

In addition, program 2 produces precision reliable clustering results.

3.4.4 K-means Consistency Preservation algorithm (K-means-CP)

In 2004, Ding\(^{[22]} \) et al. put forward K-means Consistency Preservation algorithm. Nearest neighbor consistency is an important concept in statistical pattern recognition, they expended this concept into data clustering: As any data points in one class, its nearest neighbor and mutual nearest neighbor of \( k \) should belong to this class. They proposed improved algorithm for \( k\)NN and \( 4\)NN, and regarded the nearest neighbor and mutual nearest neighbor of \( k \) as an important way of metrics.

K-means-CP:

1[Initialization]. Choose K points as centers of the initial class randomly\((c_1, c_2, ..., c_k)\)

2[Assign neighbor sets]. Assign a neighbor set \( S \); /*Assign \( S \) into nearest class \( C_p \)

\[
p = \arg \min_{v=1...K} \sum_{x_i \in C_p} (x_i - m_v)^2
\]

3[Update clustering center]. Define \( m_v = \sum_{x_i \in C_v} x_i / n_v \);

/* Update cluster center (centroid), \( m_v \) is center of class \( C_v \), \( n_v \) = |\( C_v \)|
4[Convergent or not?]. If centroid no longer move, then end the algorithm; Otherwise, back to Step2.

\[ J_{km} = \sum_{i=1}^{n} \sum_{j=1}^{K} w_{ij}^2 \left[ \sum_{m=1}^{n} \lambda_{jm} \left( x_i - m_j \right)^2 \right] \]

\[ \text{and judge it is convergent or not.} \]

### 3.4.5 Fuzzy clustering algorithm

In 1969, Rusmini applied fuzzy sets theory into clustering analyses for the first time, and proposed fuzzy clustering means(FCM). FCM is one of the most popular algorithms dealing with image segmentation. FCM can retain more information of initial images. In this paper, we simply introduce the newest study.[23,24]

In 2006, Li Jie[25] et al. proposed new algorithm NFWFCA based on feature weighting. Traditional fuzzy K-means, K-modes, and K- prototype all assume that contributions of dimensional feature for each sample vector are the same. In fact, contributions of dimensional feature for each sample vector are discriminate. Based on K- prototype, NFWFCA used Relief[26] algorithm in order to determine weight of each dimensional feature:

\[ \lambda^r = \frac{\text{diff} \_ \text{hit}^r}{R} + \frac{\text{diff} \_ \text{miss}^r}{R} \]

Property feature weights compute as:

\[ \lambda^e = \frac{\text{diff} \_ \text{hit}^e}{R} + \frac{\text{diff} \_ \text{miss}^e}{R} \]

Revise the object function:

\[ J(W,P) = \sum_{i=1}^{n} \sum_{j=1}^{K} w_{ij}^2 \left[ \sum_{m=1}^{n} \lambda_{jm} \left( x_i - p_{jm} \right)^2 \right] + \sum_{j=1}^{n} w_{ij}^2 \sum_{q=1}^{n} \lambda_{jq} \delta(x_{jq}^r, x_{jq}^e) \]

### 4. Clustering Algorithm based on gird and density

Clustering algorithm based on gird and density is one important clustering algorithm, and it is widely used in many sphere especially in spatial information processing.

Different from traditional clustering algorithm, this algorithm could discover arbitrary shape clustering by using data density; this clustering algorithm familiarly combine with other algorithms, especially clustering algorithm based on density.

In 2001, Zhao and Song[27] mentioned a clustering algorithm GDILC which based on grid density contours. The main idea of GDILC is: use density contours image to depict data sample distribution, and use network to calculate each density of data sample. The results show that GDILC has high accuracy and efficient characteristic.

In 2004, Ma[28] proposed a new algorithm SGC which based on shifting grid concept. SGC is a non-parametric algorithm, it does not need users to input parameters, for it divide every latitude of grid structures into one data space. SGC produces displacement concept of whole grid structures, hence, could improve results accurate and efficient.

In 2005, Pileva[29] et al. advanced gird clustering algorithm(GCHL) base on large, high-dimensional database. GCHL combines density-gird clustering algorithm with parallel shaft partitioning strategy, in order to make sure the high density region. This algorithm can be used excellently in random spatial database.

In 2006, Micro[30] et al. faced to moving object trajectory data processing sphere, in view of simple concept of distance between the track, put forward an adaptive clustering algorithm base on density(TFCTMO).
In 2007, Derya et al. expanded DBSCAN (density—based spatial clustering of applications with noise), and then proposed a new algorithm based on density which was called ST-DBSCAN (spatial-temporal DBSCAN), comparing with existing clustering algorithm based on density, ST-DBSCAN has ability in discovering class clusters relying on non-space value, space value, and tense value.

### 4.1 Other clustering algorithms

#### 4.1.1 ACODF clustering algorithm

In 2004, Tsai proposed a novel with different preferences ant colony system (novel AS)—ACODF (a novel data clustering approach for data mining in large databases), in order to solve clustering problems. ACODF is able to obtain optimal solution quickly, it contain three important strategies as following:

1) The application of different preferences (favorable) ACO strategy. Each ant only visit one tenth of all the cities, then successively reduce number of cities each visit; After several cycles, the concentration of pheromone increased between two closer points, and the concentration of pheromone reduced between two distant points. Therefore, ants would like to visit closer nodes, and use pheromones to strengthen this path, finally, form a high concentration path, then clustering completed.

2) We design two formulas:

\[ ns(t+1) = ns(t) \times T \]

where \( ns \) is the number of nodes that ants visit in \( T_0 \); \( ns(t+1) \) is the number of nodes that ants have been visited yet; \( ns(t) \) is the number of nodes that ants visited in the last cycle; \( T \) is a constant (\( T = 0.95 \)).

\[ \eta f(t+1) = 2 \times ns(t)/3 - i \times ns(t)/(run \times 3) \]

where \( \eta f \) is the number of nodes that ants visit in \( T_0 \); \( \eta f(t+1) \) is the number of nodes that ants have been visited yet; \( \eta f(t) \) is the number of nodes that ants visited in the last cycle; \( run = 2, i \in \{1, 2\} \).

(1) Using tournament selection strategy. Different from traditional ACO, ACODF uses tournament selection strategy in order to select path. That is select \( K \) paths randomly in \( N \) paths, then choose the shortest path in these \( K \) paths (\( N > K \)).

### 5. Experiments

In this paper, we choose five data sets: Iris, Wine, Soybean, Zoo and Image. Image data set is used to compared with Iris and Wine datasets.

For numerical model data, respectively use Iris, Wine, Image for experiments.

Iris includes 3 classes, each class has 50 elements, and every class represent one kind of flowers, 150 samples—equidistribution in 3 clusters; among them, one class of linear separable with other two class, and other two class are partially overlapped. Data set Wine has good clustering structure, including 178 samples, 13 numerical model property, divided into 3 classes, where contain different amount of samples. Image derived from UCI machine learning data sets, randomly chosen from 7 outdoor image sets.

For categorical attribute data, respectively used Soybean and Zoo dataset for experiments.

Dataset Soybean has 47 samples, including 35 attributes, divided into 4 classes for linear separable. Its attributes are all categorical attributes. Dataset Zoo has 101 records, divided into 7 classes for linear inseparable. In data set Zoo, 16 attributes used to describe samples, 15 of them are Boolean property value \{0,1\} and 1 categorical attribute (leg count) \{0,2,4,5,6,8\}.

#### 5.1 Soybean disease data set experiment

We use the formula of accuracy:

\[ r = \sum_{i=1}^{k} (a_i / n) \]

Where \( a_i \) the sample number which emerges in the \( i \)-th cluster (obtain from this algorithm) and initial-points, \( k \) is cluster number (\( k=4 \)), \( n \) is total samples number (\( n=47 \)).

Tab.1 and Tab.2 show the experiment result of this algorithm.

<table>
<thead>
<tr>
<th>Case</th>
<th>Algorithm</th>
<th>K-modes</th>
<th>Iterative</th>
<th>Initial-points</th>
<th>K-modes</th>
</tr>
</thead>
<tbody>
<tr>
<td>98</td>
<td>5</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>94</td>
<td>6</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>89</td>
<td>0</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>77</td>
<td>0</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>70</td>
<td>7</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>68</td>
<td>2</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### 5.2 Wine dataset experiment

Basing on the running time, we can figure out that iterative initial-points refinement K-modes runs longer than K-modes algorithm.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Average running time(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>K-modes</td>
<td>0.00811331</td>
</tr>
<tr>
<td>Iterative initial-points</td>
<td>0.01178265</td>
</tr>
<tr>
<td>refinement K-modes</td>
<td></td>
</tr>
</tbody>
</table>

Table 2: Average run time of 20 random tests for soybean disease data set on 2 algorithms
5.2 Hierarchical clustering and $K$-means algorithm

For numerical model data, we randomly did 20 cluster experiments using data set Iris, Wine, Image, and the results show in Tab.4.

Table 4: Clustering results of 20 random tests for Iris, Wine, Image data sets on several algorithms

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Average accuracy of running 20 cycles(%)</th>
<th>Average running time(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nearest neighbor</td>
<td>Iris 68.00, Wine 42.70, Image 30.00</td>
<td>Iris 1.5831, Wine 3.1346, Image 5.2414</td>
</tr>
<tr>
<td>Furthest neighbor</td>
<td>Iris 8.00, Wine 67.40, Image 39.00</td>
<td>Iris 1.5042, Wine 3.1434, Image 5.6708</td>
</tr>
<tr>
<td>Between groups average</td>
<td>Iris 74.70, Wine 61.20, Image 37.00</td>
<td>Iris 1.5027, Wine 3.1526, Image 5.7853</td>
</tr>
<tr>
<td>Ward method</td>
<td>Iris 89.30, Wine 35.60, Image 60.00</td>
<td>Iris 2.3793, Wine 4.7757, Image 8.5999</td>
</tr>
<tr>
<td>$K$-means</td>
<td>Iris 81.60, Wine 87.96, Image 56.00</td>
<td>Iris 0.0026, Wine 0.0038, Image 0.0457</td>
</tr>
</tbody>
</table>

The results show us, operating efficiency of these five clustering algorithms were significantly different between data sets. Thus, in a real world application, we should use different algorithms for different data sets question

5.3 Comparison between $K$-means and $K$-means-CP algorithm

In order to figure out whether $K$-means-CP is obviously better than $K$-means or not, and the relationship between $k$NN consistency and clustering quality, we did 20 random experiments on $K$-means, 1 $K$-means-CP($k=1$, denoted as $cp1$), and 2 $K$-means-CP($k=2$, denoted as $cp2$), and evaluated the results based on accuracy and quality. The difference within one class, the difference between multiple classes in a whole cluster, and quality can be compute as following formula(1)~(3):

1. \[ \sum_{v=1,...,k} \sum_{x \in C_v} d(x, \overline{x_v})^2 \]  
2. \[ \sum_{i \neq j \in \mathbb{K}} d(x_j, \overline{x_i})^2 \]  
3. \[ \frac{\sum_{i \neq j \in \mathbb{K}} d(x_j, \overline{x_i})^2}{\sum_{v=1,...,V} \sum_{x \in C_v} d(x, \overline{x_v})^2} \]

where \( k \) is the number of cluster in clustering result, \( C_v \) denotes cluster \( v \), \( \overline{x_v} \) denotes the centroid of \( C_v \), \( x_j, x_i \) respectively denote the centroid of cluster \( j \) and \( i \), \( d \) is distance function. Tab.6 shows the result that $K$-means-CP not better than $K$-means, $k$NN consistency has nothing to do with clustering quality.

Table 6: Clustering results of 20 random tests for 5 data sets on $K$-means, $cp1$&$cp2$ algorithm

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Average accuracy (20 times)</th>
<th>Average quality (20 times)</th>
</tr>
</thead>
<tbody>
<tr>
<td>cp1(1NN)</td>
<td>Iris 0.623 571 428 571 428 778 036 756 839 360 0</td>
<td></td>
</tr>
<tr>
<td>cp2(2NN)</td>
<td>Iris 0.609 523 809 523 809 764 759 617 717 611 0</td>
<td></td>
</tr>
<tr>
<td>$K$-means</td>
<td>Iris 0.632 380 952 380 952 734 076 358 291 719 0</td>
<td></td>
</tr>
<tr>
<td>Imagene</td>
<td>Average accuracy (20 times)</td>
<td>Average quality (20 times)</td>
</tr>
<tr>
<td>cp1(1NN)</td>
<td>Iris 0.840 000 000 000 000 238 626 172 448 124 0</td>
<td></td>
</tr>
<tr>
<td>cp2(2NN)</td>
<td>Iris 0.892 333 333 333 333 322 489 157 412 046 0</td>
<td></td>
</tr>
<tr>
<td>$K$-means</td>
<td>Iris 0.862 333 333 333 334 290 693 364 311 0</td>
<td></td>
</tr>
<tr>
<td>Wine</td>
<td>Average accuracy (20 times)</td>
<td>Average quality (20 times)</td>
</tr>
<tr>
<td>cp1(1NN)</td>
<td>Iris 0.898 314 606 741 573 0.045 433 239 324 063 5</td>
<td></td>
</tr>
<tr>
<td>cp2(2NN)</td>
<td>Iris 0.903 337 078 651 385 0.045 153 360 976 765 9</td>
<td></td>
</tr>
<tr>
<td>$K$-means</td>
<td>Iris 0.946 910 112 359 550 0.049 098 735 880 057 5</td>
<td></td>
</tr>
<tr>
<td>Class</td>
<td>Average accuracy (20 times)</td>
<td>Average quality (20 times)</td>
</tr>
<tr>
<td>cp1(1NN)</td>
<td>Iris 0.511 915 887 850 467 0.400 881 509 658 679</td>
<td></td>
</tr>
<tr>
<td>cp2(2NN)</td>
<td>Iris 0.531 542 016 074 766 0.404 061 886 906 006</td>
<td></td>
</tr>
<tr>
<td>$K$-means</td>
<td>Iris 0.542 523 364 485 981 0.453 522 047 430 965</td>
<td></td>
</tr>
<tr>
<td>Ionosphere</td>
<td>Average accuracy (20 times)</td>
<td>Average quality (20 times)</td>
</tr>
<tr>
<td>cp1(1NN)</td>
<td>Iris 0.691 880 341 880 342 0.003 812 476 851 314 12</td>
<td></td>
</tr>
<tr>
<td>cp2(2NN)</td>
<td>Iris 0.682 051 282 051 282 0.003 555 311 462 034 7</td>
<td></td>
</tr>
<tr>
<td>$K$-means</td>
<td>Iris 0.710 256 410 256 410 0.003 784 599 450 916 1</td>
<td></td>
</tr>
</tbody>
</table>

6. Conclusion

By means of experiments for several clustering algorithm, we can figure out most clustering algorithms need prescribed parameters. Thus, promoting non-prescribed parameters clustering algorithm, combining clustering algorithm with parameters autogeneration algorithm may have good prospect. And RCOSD can efficiently work in
data mining, helping us to understand the results of clustering.

The algorithms discussed in this paper apply to data sets with different properties, accordingly, researchers should use different algorithms and methods for divergent data problems. As a supplement for above conclusion, we compared 11 different algorithms\(^4\) and 8 algorithms proposed in this paper. Tab. 7 shows the comparative results of some typical clustering algorithms.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Years</th>
<th>Sort</th>
<th>Similarity measure</th>
<th>Parameter</th>
<th>Noise</th>
<th>Cluster shape</th>
<th>Scaled dimension</th>
<th>Others</th>
</tr>
</thead>
<tbody>
<tr>
<td>K-means</td>
<td>1967</td>
<td>Partition</td>
<td>Distance function</td>
<td>1</td>
<td>Sensitive</td>
<td>Hypersphere</td>
<td>Large numeric</td>
<td>—</td>
</tr>
<tr>
<td>K-means-Huang</td>
<td>1998</td>
<td>Partition</td>
<td>Category similarity function</td>
<td>1</td>
<td>Sensitive</td>
<td>Sphere</td>
<td>Large category</td>
<td>Describe cluster well</td>
</tr>
<tr>
<td>K-means-CP</td>
<td>2004</td>
<td>Partition</td>
<td>Distance function</td>
<td>1</td>
<td>Sensitive</td>
<td>Sphere</td>
<td>Large-Scale</td>
<td>1NN consistency is irrelevant with clustering accuracy</td>
</tr>
<tr>
<td>MDS_CLUSTER</td>
<td>1998</td>
<td>Partition</td>
<td>Euclidean distance</td>
<td>1</td>
<td>In-Sensitive</td>
<td>Arbitrary non-overlap</td>
<td>—</td>
<td>One simple parameter</td>
</tr>
<tr>
<td>Feature weighted fuzzy clustering</td>
<td>2004</td>
<td>Partition</td>
<td>Euclidean distance category similarity measure</td>
<td>1</td>
<td>In-Sensitive</td>
<td>Sphere</td>
<td>Small, mix</td>
<td>Feature weighted</td>
</tr>
<tr>
<td>Nearest neighbor</td>
<td>1967</td>
<td>Hierarchy</td>
<td>Distance function</td>
<td>1</td>
<td>In-Sensitive</td>
<td>Filamentary</td>
<td>Small and middle-dimension</td>
<td>—</td>
</tr>
<tr>
<td>Furthest neighbor</td>
<td>1967</td>
<td>Hierarchy</td>
<td>Distance function</td>
<td>1</td>
<td>—</td>
<td>Sphere</td>
<td>Small and middle-dimension</td>
<td>—</td>
</tr>
<tr>
<td>Between groups average</td>
<td>1967</td>
<td>Hierarchy</td>
<td>Distance function</td>
<td>1</td>
<td>—</td>
<td>Manifold</td>
<td>Small and middle-dimension</td>
<td>—</td>
</tr>
<tr>
<td>Sequence data rough clustering</td>
<td>2007</td>
<td>Hierarchy</td>
<td>SPM</td>
<td>2</td>
<td>—</td>
<td>Sequence data</td>
<td>Large-Scale</td>
<td>Depict cluster feature</td>
</tr>
<tr>
<td>SGC</td>
<td>2004</td>
<td>Density</td>
<td>Distance function</td>
<td>None</td>
<td>In-Sensitive</td>
<td>Arbitrary shape</td>
<td>Large and middle-dimension</td>
<td>Mostly used for spatial data processing</td>
</tr>
<tr>
<td>GCHL</td>
<td>2005</td>
<td>Grid</td>
<td>Euclidean distance</td>
<td>2</td>
<td>In-Sensitive</td>
<td>Arbitrary shape</td>
<td>Oversize high-dimension</td>
<td>Information processing</td>
</tr>
<tr>
<td>ACODF</td>
<td>2004</td>
<td>Others</td>
<td>Euclidean distance</td>
<td>1</td>
<td>—</td>
<td>Sphere, non-sphere</td>
<td>Small, high-dimension</td>
<td>Get optimal value fast</td>
</tr>
</tbody>
</table>

Combining both algorithms in references and the methods proposed in this paper, we can make the conclusion that clustering algorithms and the results are unpredictable, in practical researches, we should choose appropriate clustering algorithms to solve different kind of data problems in order to obtain the best clustering results.
References


