International Journal of Machine Learning and Cybernetics (2019) 10:3213–3223
https://doi.org/10.1007/s13042-019-01012-6

ORIGINAL ARTICLE

# Fuzzy rough clustering for categorical data

**Shuliang Xu**[1] · **Shenglan Liu**[2] · **Jian Zhou**[1] · **Lin Feng**[2]

Received: 27 April 2019 / Accepted: 4 September 2019 / Published online: 19 September 2019


**Abstract**
Unlabeled categorical data is common in many applications. Because there is no geometric structure for categorical data, how to discover knowledge and patterns from unlabeled categorical data is an important problem. In this paper, a fuzzy rough clustering algorithm for categorical data is proposed. The proposed algorithm uses the partition of each attribute to calculate the granularity of each attribute and introduces information granularity to measure the significance of each attribute. It is different from traditional clustering algorithms for categorical data that the proposed algorithm can transform categorical data set into numeric data set and introduces a nonlinear dimension reduction algorithm to decrease the dimensions of data set. The proposed algorithm and the comparison algorithms are executed on real data sets. The experimental results show that the proposed algorithm outperforms the comparison algorithms on the most data sets and the results prove that the proposed algorithm is an effective clustering algorithm for categorical data sets.

**Keywords** Cluster analysis · Rough set · Categorical data · Granular computing · Dimension reduction

## 1 Introduction

In information society, unlabeled categorical data is more and more common. Many fields have generated a large number of unlabeled categorical data sets such as social media, bioinformatics data, news report and web search engine [7, 11, 18, 26, 30, 32]. It will produce much overhead if data is labeled by experts and it is also impossible to obtain the labels of data at any time, therefore how to get knowledge and patterns from unlabeled categorical data is an important problem [28, 35]. Clustering is an unsupervised learning method which can mine knowledge and patterns from unlabeled data and it has been applied to text emotional analysis, bioinformatics and recommender system [12, 17, 27]. Because there is no geometric structure for categorical data, it brings a challenge for data partitioning.

✉ Lin Feng
fenglin@dlut.edu.cn

Shuliang Xu
xushulianghao@126.com

[1] Faculty of Electronic Information and Electrical Engineering, Dalian University of Technology, Dalian, China

[2] School of Innovation and Entrepreneurship, Dalian University of Technology, Dalian, China

k-modes algorithm is an important clustering algorithm for categorical data and it uses matching distance to represent the dissimilarity of two samples and data is partitioned into clusters by matching distance [5]. After k-modes is proposed, many researchers develop a series of algorithms and the algorithms can mainly divided into three categories: uncertainty-based clustering algorithm, tree clustering algorithm and subspace clustering algorithm.

Michael proposes a new dissimilarity measure for k-modes algorithm [24]; considering that the original matching method may cause the similarity of intra-cluster samples to be too low, it utilizes equivalent class of current attribute value to determine the dissimilarity; for an attribute, if the attribute values of two samples are equal, the dissimilarity is 1 on current attribute, otherwise the dissimilarity is computed by the equivalent class on the attributes; the new k-modes algorithm converts the dissimilarity degree from 0-1 match into real value which can keep more dissimilarity information. Chen et al. propose an entropy method to determine the best $k$ of clustering algorithm for categorical data [6]; in the paper, an incremental entropy is defined which is according to the merge of different clusters and the relation between $k$ and the incremental entropy is derived; when the function value is reduced severely, the corresponding $k$ value is the final result. Andritsos et al. propose a tree clustering algorithm based on entropy called as LIMBO [2];

the algorithm defines a new distance and creates a DCF tree; LIMBO scans DCF tree from top to down and divides data points into leaf nodes; the data points in the same leaf nodes are seen as a cluster; if the information loss of merging two similar clusters is less than a threshold, then the two clusters are merged; the merging steps are repeated until the number of clusters is $k$. Guha et al. propose a robust hierarchical clustering algorithm for categorical data called as ROCK [15]; ROCK introduces neighbourhood to measure the similarity of the two clusters and defines a link function which is the intersection's cardinality of two clusters and determines the evaluation criterion function; the similar clusters is merged until there is $k$ clusters; because heap is used in the clustering process, ROCK has a fast speed. Gao et al. propose a rough ensemble subspace-based clustering for categorical data [13]; the algorithm employs discernibility matrix to remove redundant attributes and forms a series of subspaces; then it uses a metric to rank subspaces and remains some subspaces with large metric values; the clustering algorithm is executed in the subspaces and the final clustering result is the fusion of multiple clustering results. Parmar et al. propose a clustering algorithm based on Min-Min-Roughness called as MMR [25]; MMR is a tree clustering algorithm and uses the models of most samples as the clustering modes; for each attribute, it computes the minimum roughness of each attribute value; then it can obtain Min-Min-Roughness of all attributes; the attribute corresponding to Min-Min-Roughness is selected as splitting attribute of the tree and the cluster in the leaf node can be split into two clusters; repeat the steps until the clustering algorithm is terminated. Li et al. propose an incremental entropy-based clustering for categorical data stream with concept drift [21]; different from traditional clustering algorithms, the incremental entropy-based clustering algorithm can deal with massive data stream and it has a self-adaption mechanism for concept drift; the algorithm defines the entropy distance between a sample and a cluster; if the entropy distance between a sample and a cluster is greater than a threshold, the sample is seen as a outlier and a new cluster is generated; in order to detect concept drift, cluster vector is defined; for a time, the cluster vector is made up of the number of samples in the clusters; if the similarity of the cluster vectors at adjacent time moments is less than a threshold, it is said that concept drift has happened.

The above works have promoted the development of the clustering for categorical data; however, for the most algorithms, how to measure the dissimilarity in categorical data set and generate compact clusters is still a problem. In this paper, a new fuzzy rough clustering algorithm (FRC) for categorical data is proposed. FRC introduces rough set to compute the information granularity of each attribute and the weights of attributes are determined by the information granularity. In addition, we define a significance of an attribute for a sample and categorical data set can be transformed into a numeric data set. Different from the existing distance for categorical data, the new distance is a weighting distance and the attribute with a good discriminative ability will be given a large weight. Therefore FRC can obtain a clustering result with minimum intra-cluster distance and maximum inter-cluster distance. The contributions of the paper are as follows:

- We propose a fuzzy rough clustering algorithm for categorical data set; FRC can obtain a clustering result with minimum intra-cluster distance and maximum inter-cluster distance.
- We define a significance measure of an attribute for a sample; categorical data set can be transformed into a numeric data set by the significance measure and we introduce a nonlinear dimension reduction algorithm to decrease the dimensions of data set.
- In FRC algorithm, we employs weighted distance to calculate the dissimilarity. For categorical data set, the weight of each attribute is determined by information granularity. After categorical data set is transformed into numeric data set, the new weight of each attribute is determined by standard deviation.

The rest of the paper is structured as follows: Sect. 2 reviews the concepts of rough set and the basic principles of k-modes algorithm; the detail theories and steps of FRC algorithm are explained in Sect. 3; FRC and the comparison algorithms are executed on the real data sets and the experimental results are discussed in Sect. 4; finally, Sect. 5 concludes the paper and gives some research directions in the future.

## 2 Preliminaries

In this section, we will give brief introductions about rough set and k-modes algorithm.

### 2.1 Rough set

Rough set is an effective method for data analysis. It can deal with categorical data without any prior knowledge. Since rough set was proposed, it has been applied to association rule extraction, attribute reduction, data classification and clustering analysis [8–10, 14, 19, 23, 29, 31, 33]. Rough set assumes that the objects with the same attribute values should be divided into the same class. The following background knowledge can be seen from the references [1, 20, 36, 37].

An information system can be described as a quadruples $< U, A, f, V >$ where $U = \{x_1, x_2, x_3, \ldots, x_n\}$ is an universe, $A = \{a_1, a_2, a_3, \ldots, a_m\}$ is an attribute set, $V$ is a set

of attribute values, $f$ is an information function which constructs the mapping between attributes and attribute values and $f : U \times A \to V$ which means $f(x, a) \in V$ if $\forall x \in U$ and $\forall a \in A$. Let $B \subseteq A$ and $B \neq \varnothing$, the indiscernibility relation of the objects on $B$ is defined as

$$IND(B) = \{(x, y) \in U \times U | f(x, a) = f(y, a), \forall a \in B\}. \quad (1)$$

If $(x, y) \in IND(B)$, it means the attribute values of $x$ and $y$ are the same on the attribute set $B$. For $\forall x \in U$, the equivalence class of $x$ on $B$ is defined as

$$[x]_B = \{y \in U | (x, y) \in IND(B)\}. \quad (2)$$

It can obtain a partition of the universe according to $IND(B)$ which is denoted as $U/B$; If $U/B = \{X_1, X_2, \ldots, X_l\}$, it is known that $\bigcup_{i=1}^{l} X_i = U$ and $X_i \cap X_j = \varnothing$ for $\forall X_i, X_j \subseteq U$ and $i \neq j$.

For an information system $< U, A, V, f >$, $B \subseteq A$, $B \neq \varnothing$ and $R_B$ is the equivalence relation. For $\forall X \subseteq U$, the lower approximation set $\underline{R_B}X$ and the upper approximation set $\overline{R_B}X$ is defined as

$$\begin{aligned} \underline{R_B}X &= \{x \in U | [x]_B \subseteq X\} \text{ and} \\ \overline{R_B}X &= \{x \in U | [x]_B \cap X \neq \varnothing\}. \end{aligned} \quad (3)$$

The universe is divided into three parts by rough set. $POS_R(X) = \underline{R_B}X$ is called as the positive region of $X$ on $B$; $NEG_R(X) = \overline{U} - \overline{R_B}X$ is the negative region of $X$ on $B$ and $BN_R(X) = \overline{R_B}X - \underline{R_B}X$ is the boundary region of $X$ on $B$. It is known that the boundary region represents the uncertainty of the set. A large boundary region means a large uncertainty. If $BN_R(X) = \varnothing$, it is said that $X$ is crisp; otherwise, $X$ is rough. For a partition, $U/B = \{X_1, X_2, \ldots, X_l\}$; in order to measure the uncertainty, the information granularity of the knowledge is defined as

$$GK_B(U) = \frac{1}{|U|} \sum_{i=1}^{l} \frac{|X_i|^2}{|U|^2}. \quad (4)$$

It is obvious that $\frac{1}{|U|^2} \leq GK_B(U) \leq \frac{1}{|U|}$. When $X_i = \{x_i\} (i = 1, 2, \ldots, |U|)$, the information granularity is minimum and the uncertainty is also minimum; when $X_i = U$ and $X_j = \varnothing$ $(j = 1, 2, \ldots, l, j \neq i)$, the information granularity is maximum and the uncertainty is also maximum.

### 2.2 k-modes algorithm

k-modes algorithm [5] is a simple and practical clustering algorithm for categorical data. k-modes algorithm uses difference degree to replace the euclidean distance. For $x_i = [x_{i1}, x_{i2}, \ldots, x_{im}]$ and $x_j = [x_{j1}, x_{j2}, \ldots, x_{jm}]$, the difference degree of two samples on each attribute is defined as

$$\varphi(x_{ip}, x_{jp}) = \begin{cases} 1 & \text{if } x_{ip} \neq x_{jp} \\ 0 & \text{otherwise} \end{cases} (p = 1, 2, \ldots, m). \quad (5)$$

Therefore the difference degree of two samples is as

$$\varphi(x_i, x_j) = \sum_{p=1}^{m} \varphi(x_{ip}, x_{jp}). \quad (6)$$

A smaller difference degree means that the two samples are more similar. k-modes algorithm is a greedy algorithm. The algorithm repeatedly adjusts the clustering result until the clustering result is convergent. Therefore k-modes algorithm is summarized as Algorithm 1.

**Algorithm 1** k-modes algorithm.

**Input:** data set $U = [x_1, x_2, \cdots, x_n]$; the number of clusters $k$.
**Output:** the clustering result $C = [C_1, C_2, \cdots, C_k]$.
1: Randomly choose $k$ modes;
2: Compute the distances between samples and modes;
3: Each sample is partitioned into the nearest cluster according Eq.(6) and update the $k$ modes;
4: Repeat the above steps until the clustering result is convergent.

From Algorithm 1, it is known that k-modes algorithm can deal with categorical data and the clustering result is local optimum although k-modes algorithm iteratively adjusts the clustering result. It is obvious that there are still some problems for k-modes algorithm. For many unlabeled data sets, how to determine the number of clusters is not easy; in addition, the weight of each attribute is equal in Eq. (6); however, the equal weight mechanism cannot make that the difference degree can represent the similarity well because the equal weight does not present the significant of each attribute. Therefore it is necessary to improve k-modes algorithm.

## 3 Fuzzy rough clustering algorithm

In this section, we will introduce the detail principles of FRC. At first, we explain how to determine the weights of attributes by rough set; then we describe the method of dimension reduction and the steps of FRC.

### 3.1 The methods determining the weights of attributes

The attribute weighting mechanism is widely used in clustering tasks. For many tasks, if there are many redundant or irrelevant attributes, the distance or difference degree cannot measure the dissimilarity well. The attribute weighting mechanism gives a weight for each attribute. The large weight means that the attribute has a greater influence on the distance or

**Table 1** The decision table of an information system

| | $a_1$ | $a_2$ | $a_3$ | $a_4$ |
|---|---|---|---|---|
| $x_1$ | 0 | 0 | 0 | 1 |
| $x_2$ | 1 | 1 | 0 | 1 |
| $x_3$ | 2 | 0 | 1 | 0 |
| $x_4$ | 2 | 0 | 1 | 0 |
| $x_5$ | 3 | 0 | 2 | 1 |
| $x_6$ | 4 | 0 | 2 | 1 |

difference degree. Therefore the attribute weighting mechanism can reduce or eliminate the adverse effects of redundant attributes and it can achieve the effect of dimension reduction without information loss. Rough set is an effective tool to analyze categorical data set and it can obtain knowledge and patterns without any prior knowledge. The performance of k-modes can be improved by rough set.

Let $U = \{x_1, x_2, \ldots, x_n\}$ and $A = \{a_1, a_2, \ldots, a_m\}$. For $\forall a \in A$, the partition of the universe on $a$ is as $U/a = \{X_1, X_2, \ldots, X_{l_a}\}$ where $l_a$ is the number of equivalence classes on $a$. Therefore the information granularity of the partition on $a$ is as

$$GK_a(U) = \frac{1}{|U|} \sum_{i=1}^{l_a} \frac{|X_i|^2}{|U|^2}. \tag{7}$$

$GK_a(U)$ is the uncertainty of the knowledge on $a$; a small information granularity means the uncertainty of the partition on $a$ is small; in other words, the objects of the universe can be discriminated well on the attribute $a$. Therefore the attribute $a$ should be given a large weight when computing the distance or difference degree. The weight of the attribute $a$ which is denoted as $w_a$ is defined as

$$Weight_a = \frac{\sum_{\forall b \in A} GK_b(U)}{GK_a(U)} \text{ and } \omega_a = \frac{Weight_a}{\sum_{\forall b \in A} Weight_b}. \tag{8}$$

In order to explain the principle of the weights of attributes, an example is introduced. Table 1 is the decision table of an information system. $U = \{x_1, x_2, x_3, x_4, x_5, x_6\}$ and $A = \{a_1, a_2, a_3, a_4\}$. From the decision table, $U/a_1 = \{\{x_1\}, \{x_2\}, \{x_3, x_4\}, \{x_5\}, \{x_6\}\}$, $U/a_2 = \{\{x_1, x_3, x_4, x_5, x_6\}, \{x_2\}\}$, $U/a_3 = \{\{x_1, x_2\}, \{x_3, x_4\}, \{x_5, x_6\}\}$ and $U/a_4 = \{\{x_1, x_2, x_5, x_6\}, \{x_3, x_4\}\}$. From the partitions of the attributes, it is obvious that the objects of the universe can be better discriminated by $a$ rather than the other attributes. Therefore the attribute $a$ has the largest weight.

$$\begin{aligned}
\therefore GK_{a_1}(U) &= \frac{1}{6} \cdot \left( \frac{1}{36} + \frac{1}{36} + \frac{4}{36} + \frac{1}{36} + \frac{1}{36} \right) \approx 0.0370 \\
GK_{a_2}(U) &= \frac{1}{6} \cdot \left( \frac{1}{36} + \frac{25}{36} \right) \approx 0.1204 \\
GK_{a_3}(U) &= \frac{1}{6} \cdot \left( \frac{4}{36} + \frac{4}{36} + \frac{4}{36} \right) \approx 0.0556 \\
GK_{a_4}(U) &= \frac{1}{6} \cdot \left( \frac{16}{36} + \frac{4}{36} \right) \approx 0.0926 \\
\therefore \omega_{a_1} &= 0.42152 \quad \omega_{a_2} = 0.12954 \\
\omega_{a_3} &= 0.28051 \quad \omega_{a_4} = 0.16843.
\end{aligned}$$

### 3.2 Dimension reduction

For $U/A = \{X_1, X_2, \ldots, X_l\}$, the partition can be seen as an initial clustering result. However, because of redundant and irrelevant attributes, the initial clustering result often does not accord with the real result. In order to improve the quality of the partition, attribute reduction is a common method for categorical data, but the complexity of rough set algorithm is high and the clustering algorithm with rough set attribute reduction will bring a large time overheard. Manifold learning is an effective algorithm to decrease the number of dimension which can reconstruct data points in a low dimensional embedding space according to their neighbors [22]. Therefore in this paper, we will transform categorical data into numerical data and utilize manifold learning to decrease the dimension of data points.

For $\forall a \in A$, let $U/a = \{X_1, X_2, \ldots, X_{l_a}\}$, it is obvious that the difference of data points in the same equivalence class should be as small as possible and the difference of data points in different equivalence classes should be as large as possible. For $\forall x_i \in U$, the dissimilarity between $x_i$ and data point in the same equivalence class is defined as follow

$$d_s(x_i, y) = \sum_{j=1}^{m} \omega_j \cdot \varphi(x_{ij}, y_j) \text{ and } y \in [x_i]_a. \tag{9}$$

For $\forall x_i \in U$, the dissimilarity between $x_i$ and data point in different equivalence classes is defined as follow

$$d_f(x_i, y) = \sum_{j=1}^{m} \omega_j \cdot \varphi(x_{ij}, y_j) \text{ and } y \notin [x_i]_a. \tag{10}$$

Therefore the significance of the attribute $a$ for $x_i$ can be defined as follow

$$sig_a(x_i) = \frac{1}{1 + \sum_{y \in U} d_s(x_i, y) / \left| [x_i]_a \right|} + \frac{1}{1 + \exp\left( - \sum_{y \in U} d_f(x_i, y) / \left| U - [x_i]_a \right| \right)}. \tag{11}$$

In Eq. (11), $sig_a(x_i)$ includes the discrimination information of equivalence partition. A large value of $sig_a(x_i)$ means that the attribute $a$ is more important for $x_i$. If the significance of each data point is calculated, a new significant matrix can be obtained. The new significant matrix $\boldsymbol{S} \in \mathbb{R}^{m\times n}$ is defined as

$$\boldsymbol{S} = \begin{bmatrix} sig_{a_1}(x_1) & \cdots & sig_{a_1}(x_n) \\ \vdots & \ddots & \vdots \\ sig_{a_m}(x_1) & \cdots & sig_{a_m}(x_n) \end{bmatrix}_{m\times n}. \tag{12}$$

In Eq. (12), $\boldsymbol{S}$ is numerical and the significant matrix can be seen as a new representation of $\boldsymbol{X}$ which means that the categorical data set $\boldsymbol{X}$ is transformed into a numerical data set. Therefore we can introduce manifold learning to decrease the dimensions of $\boldsymbol{S}$.

Let $\boldsymbol{Y} = [\boldsymbol{Y}_1, \boldsymbol{Y}_2, \ldots, \boldsymbol{Y}_n] = \boldsymbol{P}^T\boldsymbol{S}$ and $\boldsymbol{P} \in \mathbb{R}^{m\times d}(d < m)$, the data points can be reconstructed by their neighbors. Therefore the problem can be expressed as follow

$$\begin{aligned} &\min_{\boldsymbol{P}} \sum_{i=1}^{n}\sum_{j=1}^{n} W_{ij}\left\|\boldsymbol{Y}_i - \boldsymbol{Y}_j\right\|_2^2 + \alpha\|\boldsymbol{P}\|_F^2 \\ &s.t.\ \boldsymbol{P}^T\boldsymbol{S}\boldsymbol{D}\boldsymbol{S}^T\boldsymbol{P} = \boldsymbol{1} \end{aligned}. \tag{13}$$

where $W_{ij}$ is the weight between $\boldsymbol{Y}_i$ and $\boldsymbol{Y}_j$; $\alpha$ is a parameter and $\alpha > 0$. The first term of Eq. (13) is the reconstruction error; the second term is the generalization ability. Let $\boldsymbol{D}_{ii} = \sum_{j=1}^{n} W_{ij}$ and $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W}$; Eq. (13) can be also expressed as follow

$$\begin{aligned} &\min_{\boldsymbol{P}} \boldsymbol{tr}\left(\boldsymbol{P}^T\boldsymbol{S}\boldsymbol{L}\boldsymbol{S}^T\boldsymbol{P}\right) + \alpha\|\boldsymbol{P}\|_F^2 \\ &s.t.\ \boldsymbol{P}^T\boldsymbol{S}\boldsymbol{D}\boldsymbol{S}^T\boldsymbol{P} = \boldsymbol{1} \end{aligned}. \tag{14}$$

From Eq. (14), $\boldsymbol{P}$ can be solved by the following generalized eigenvalue problem:

$$\boldsymbol{S}\boldsymbol{L}\boldsymbol{S}^T\boldsymbol{P} + \alpha\boldsymbol{P} = \lambda\boldsymbol{S}\boldsymbol{D}\boldsymbol{S}^T\boldsymbol{P}. \tag{15}$$

Therefore From Eq. (15), it can known that

$$\left(\boldsymbol{S}\boldsymbol{L}\boldsymbol{S}^T + \alpha\boldsymbol{I}\right)\boldsymbol{P} = \lambda\boldsymbol{S}\boldsymbol{D}\boldsymbol{S}^T\boldsymbol{P}. \tag{16}$$

$\boldsymbol{P}$ is made up of the $d$ eigenvectors corresponding to the $d$ smallest eigenvalues. If $\boldsymbol{P}$ is solved, $\boldsymbol{Y} = \boldsymbol{P}^T\boldsymbol{S}$ is the result of dimension reduction.

### 3.3 The clustering processes of FRC

After data points is executed by the dimension reduction algorithm, a low dimension representation of data points can be obtained. Therefore it can use k-means clustering algorithm with fuzzy partition to get the final clusters. Let $k$ be the number of clusters and $\boldsymbol{Z} = [\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_k] \in \mathbb{R}^{d\times k}$ be the cluster center points; however, how to select an appropriate $k$ value is an important problem. Rough set can generate a partition of the universe for each attribute. Therefore it can employ the partition results to decrease the range of the number of clusters. Let $U$ be a categorical data set and $A$ be a attribute set; for $\forall a \in A, U/a = \{X_1, X_2, \ldots, X_{l_a}\}$; the range of $k$ should satisfy the following equation:

$$2 \le k \le \max\{|U/a|\} \quad \forall a \in A. \tag{17}$$

For the new data points $\boldsymbol{Y}$, let $\delta_i$ be the standard deviation of $\boldsymbol{Y}_i (i = 1, 2, \ldots, d)$. For a unlabeled data set, $\delta_i$ can reflect the scatter of data set on this attribute and a large $\delta_i$ means a good discernibility. $\delta_i$ can be normalized as follow

$$\delta_i \leftarrow \delta_i / \sum_{j=1}^{d} \delta_j. \tag{18}$$

It is obvious that $\delta_i$ can be seen as the weight of the $i$th attribute. Therefore the objective optimization function of clustering algorithm can be expressed as follow

$$\begin{aligned} &\min_{U,Z} \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{d}\sum_{t=1}^{k} u_{it}\cdot\delta_j\cdot\left(x_{ij} - z_{tj}\right)^2 \\ &\quad - \frac{1}{k}\sum_{j=1}^{d}\sum_{t=1}^{k}\sum_{l=t+1}^{k}\delta_j\cdot\left(z_{tj} - z_{lj}\right)^2 \\ &s.t.\ 0 \le u_{it} \le 1 \end{aligned}. \tag{19}$$

Therefore the membership $u_{it}$ is updated as

$$u_{it} = 1/\left(1 + \sum_{j=1}^{d}\delta_j\cdot\left(x_{ij} - z_{tj}\right)^2\right) \quad i = 1, 2, \ldots, n;\ t = 1, 2, \ldots, k. \tag{20}$$

The center point $z_{tj}$ is updated as

$$z_{tj} = \frac{\sum_{i=1}^{n} u_{it}x_{ij}}{\sum_{i=1}^{n} u_{it}} \quad t = 1, 2, \ldots, k;\ j = 1, 2, \ldots, d. \tag{21}$$

From the above descriptions, the detail steps of FRC algorithm are summarized as Algorithm 2.

**Algorithm 2** FRC algorithm.

**Input:** the information system $\langle U, A, V, f\rangle$;
**Output:** the clustering result $C = [C_1, C_2, \cdots, C_k]$.
1: **for** $a \in A$ **do**
2: Compute the partition of rough set $U/a$;
3: Compute the weight $\omega_a$ and $\omega \leftarrow \omega \cup \omega_a$;
4: **end for**
5: Determine the number of clusters $k$;
6: Compute $\boldsymbol{S}$ according to Eq.(12);
7: Compute $\boldsymbol{P}$ according to Eq.(16) and $\boldsymbol{Y} = \boldsymbol{P}^T\boldsymbol{S}$;
8: **for** $\boldsymbol{Y}_i \in \boldsymbol{Y}$ **do**
9: Compute the standard deviation $\delta_i$ of $\boldsymbol{Y}_i$;
10: **end for**
11: Normalize $\boldsymbol{\delta} = [\delta_1, \delta_2, \cdots, \delta_k]$ into $[0, 1]$;
12: Randomly choose $k$ center points $\boldsymbol{Z}$;
13: **while** The objective function is not convergence **do**
14: Update $u_{it}$ as Eq.(20) $(i = 1, 2, \cdots, n;\ t = 1, 2, \cdots, k)$;
15: Update $z_{tj}$ as Eq.(21) $(t = 1, 2, \cdots, k;\ j = 1, 2, \cdots, d)$;
16: **end while**
17: Obtain the final membership matrix of the universe;
18: Obtain the final clustering result $C = [C_1, C_2, \cdots, C_k]$ from the membership matrix by k-means.

**Table 2** The details of the experimental data sets

| | Samples | Attributes | Type | Classes | Distribution |
|---|---|---|---|---|---|
| student | 300 | 32 | Categorical | 3 | {50, 72, 178} |
| Germany | 300 | 20 | Categorical | 2 | {220, 80} |
| Thoracic | 470 | 17 | Categorical | 2 | {70, 400} |
| adult | 300 | 12 | Categorical | 2 | {230, 70} |
| nursery | 500 | 9 | Categorical | 4 | {112, 6, 89, 93} |
| car | 500 | 7 | Categorical | 4 | {124, 341, 20, 15} |

**Table 3** The test results of the algorithms on Germany data set

| | DKmodes | WKModes | k-modes | FRC |
|---|---|---|---|---|
| J | 0.4417 ± 0.0440 | 0.4239 ± 0.0513 | 0.4532 ± 0.0619 | **0.5435 ± 0.0190** |
| FM | 0.6124 ± 0.0416 | 0.5951 ± 0.0475 | 0.6229 ± 0.0569 | **0.7132 ± 0.0184** |
| CD | 0.6115 ± 0.0419 | 0.5937 ± 0.0477 | 0.6214 ± 0.0569 | **0.7040 ± 0.0160** |
| K | 0.6134 ± 0.0413 | 0.5964 ± 0.0474 | 0.6243 ± 0.0570 | **0.7224 ± 0.0209** |
| Time cost | 5.8688 | 1.0906 | **0.2266** | 705.5525 |

From Algorithm 2, it is known that the range of the number of clusters is decreased by the number of equivalence classes which is based on rough set. In Eq. (11), the significance of an attribute for a sample can be measured by the weight. In addition, the weighting mechanism of Eq. (19) makes that the dissimilarity can better represent the real dissimilarity. It is also known that FRC algorithm is different from k-modes algorithm and FRC algorithm can transforms categorical data set into numeric data set; therefore it can use many nonlinear dimension reduction algorithms to decrease the number of dimensions which is different from the attribute reduction algorithms based on rough set.

## 4 Experiments and results

In this section, we choose k-modes [5], DKmodes [3], WKModes [4] as comparison algorithms and all algorithms are executed on MATLAB 2017Ra. The details of data sets[1] are showed as Table 2. In order to measure the experimental results of the algorithms, the following evaluation criteria are used in this paper [34].

(1) Jaccard coefficient:

$$J = \frac{SS}{SS + SD + DS}. \tag{22}$$

(2) Fowlkes and Mallows index:

$$FM = \sqrt{\frac{SS}{SS + SD} \cdot \frac{SS}{SS + DS}}. \tag{23}$$

(3) Czekanowski–Dice index:

$$CD = \frac{2 \cdot SS}{2 \cdot SS + DS + SD}. \tag{24}$$

(4) Kulczynski index:

$$K = \frac{1}{2} \cdot \left( \frac{SS}{SS + SD} + \frac{SS}{SS + DS} \right). \tag{25}$$

where $SS$ is the number of data points which are in the same cluster and also belong to the same class; $SD$ is the number of data points which are in the same cluster but belong to different classes; $DS$ is the number of data points which are in different clusters but belong to the same classes; $DD$ is the number of data points which are in different clusters and also belong to different classes. For the evaluation criteria, $0 \leq J, FM, CD, K \leq 1$ and a larger value indicates a better performance.

In order to test the performance of FRC, we execute k-modes, DKmodes, WKModes and FRC on the experimental data sets. For FRC, $\alpha$ is set to 5, $d = 0.85 \cdot col$ where $col$ is the number of attributes. The test results are showed as Tables 3, 4, 5, 6, 7 and 8.

Tables 3, 4, 5, 6, 7 and 8 show the test results of the four algorithms on the experimental data set and the bold results are the best results. From the results, it can be seen that the J, FM, CD and K of FRC algorithm are better than the three comparison algorithms on the experimental data sets except for nursery and student data sets. The results prove that FRC algorithm is an effective algorithm for clustering task. In FRC algorithm, it introduces the attribute weighting mechanism and the data set type conversion mechanism. The attribute weighting mechanism gives a weight for each attribute and an attribute with a larger weight has a large

[1] http://archive.ics.uci.edu/ml/index.php

**Table 4** The test results of the algorithms on student data set

| | DKmodes | WKModes | k-modes | FRC |
|---|---|---|---|---|
| J | 0.2458 ± 0.0116 | 0.2531 ± 0.0236 | **0.2900 ± 0.0303** | 0.2822 ± 0.0078 |
| FM | 0.3967 ± 0.0143 | 0.4054 ± 0.0290 | **0.4503 ± 0.0369** | 0.4401 ± 0.0095 |
| CD | 0.3944 ± 0.0148 | 0.4035 ± 0.0285 | **0.4487 ± 0.0363** | 0.4401 ± 0.0095 |
| K | 0.3991 ± 0.0138 | 0.4074 ± 0.0296 | **0.4518 ± 0.0377** | 0.4402 ± 0.0095 |
| Time cost | 7.3526 | 2.4812 | **0.2824** | 1065.8253 |

**Table 5** The test results of the algorithms on nursery data set

| | DKmodes | WKModes | k-modes | FRC |
|---|---|---|---|---|
| J | **0.5078 ± 0.1159** | 0.3264 ± 0.0523 | 0.2623 ± 0.0342 | 0.2002 ± 0.0105 |
| FM | **0.6697 ± 0.1049** | 0.4927 ± 0.0594 | 0.4158 ± 0.0433 | 0.3340 ± 0.0162 |
| CD | **0.6657 ± 0.1051** | 0.4899 ± 0.0592 | 0.4145 ± 0.0430 | 0.3334 ± 0.0142 |
| K | **0.6737 ± 0.1046** | 0.4955 ± 0.0597 | 0.4172 ± 0.0435 | 0.3347 ± 0.0183 |
| Time cost | 4.0427 | 1.1652 | **0.3391** | 332.0208 |

**Table 6** The test results of the algorithms on Thoracic data set

| | DKmodes | WKModes | k-modes | FRC |
|---|---|---|---|---|
| J | 0.4955 ± 0.1075 | 0.5054 ± 0.0735 | 0.5351 ± 0.0970 | **0.7314 ± 0.0061** |
| FM | 0.6635 ± 0.0787 | 0.6735 ± 0.0583 | 0.6971 ± 0.0758 | **0.8524 ± 0.0045** |
| CD | 0.6570 ± 0.0810 | 0.6685 ± 0.0615 | 0.6923 ± 0.0787 | **0.8449 ± 0.0041** |
| K | 0.6701 ± 0.0764 | 0.6786 ± 0.0551 | 0.7019 ± 0.0729 | **0.8601 ± 0.0050** |
| Time cost | 10.5341 | 1.3735 | **0.2817** | 1226.2209 |

**Table 7** The test results of the algorithms on adult data set

| | DKmodes | WKModes | k-modes | FRC |
|---|---|---|---|---|
| J | 0.4284 ± 0.0329 | 0.4387 ± 0.0649 | 0.4905 ± 0.0912 | **0.6360 ± 0.0040** |
| FM | 0.6032 ± 0.0323 | 0.6100 ± 0.0554 | 0.6565 ± ± 0.0771 | **0.7961 ± 0.0037** |
| CD | 0.5991 ± 0.0318 | 0.6074 ± 0.0554 | 0.6536 ± 0.0768 | **0.7775 ± 0.0030** |
| K | 0.6074 ± 0.0328 | 0.6126 ± 0.0554 | 0.6594 ± 0.0776 | **0.8151 ± 0.0044** |
| Time cost | 3.0153 | 1.4524 | **0.2161** | 387.2299 |

**Table 8** The test results of the algorithms on car data set

| | DKmodes | WKModes | k-modes | FRC |
|---|---|---|---|---|
| J | 0.3615 ± 0.0570 | 0.3172 ± 0.0589 | 0.3123 ± 0.0702 | **0.4100 ± 0.0538** |
| FM | 0.5605 ± 0.0666 | 0.5073 ± 0.0721 | 0.4904 ± 0.0762 | **0.5861 ± 0.0618** |
| CD | 0.5283 ± 0.0655 | 0.4785 ± 0.0697 | 0.4718 ± 0.0787 | **0.5796 ± 0.0525** |
| K | **0.5948 ± 0.0679** | 0.5378 ± 0.0745 | 0.5099 ± 0.0740 | 0.5927 ± 0.0716 |
| Time cost | 5.7890 | 1.0897 | **0.5179** | 660.6340 |

impact on the distance. By the data set type conversion mechanism, a categorical data set can be transformed into a numeric data set and then linear or nonlinear dimension reduction algorithms can be used to decrease the dimension of data set. The above mechanisms effectively avoid the influence of redundant attributes on the performance of the algorithm. Therefore the clustering results of FRC algorithm are better than the comparison algorithms on the most data sets. Tables 3, 4, 5, 6, 7 and 8 also show the time cost of the four algorithms. From the results, it is known that k-modes algorithm cost the least time, WKModes is the second least, DKmodes is the third least and FRC algorithm spends much more time on clustering task than the comparison algorithms which means the time complexity of FRC algorithm is high. WKModes and DKModes are the developments of k-modes algorithm; WKModes uses the

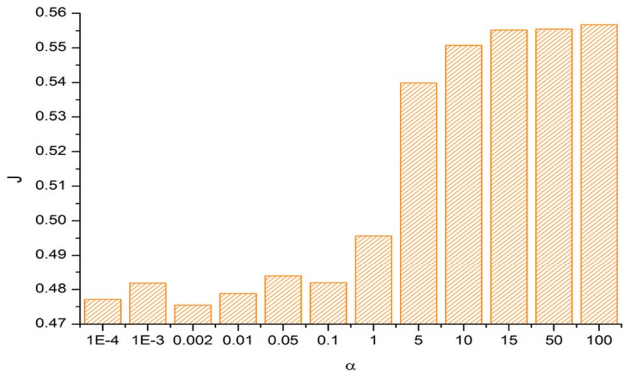


**Fig. 1** The J of FRC algorithm with different $\alpha$ values

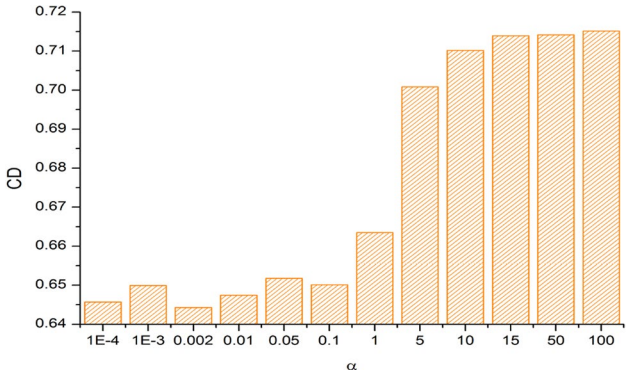


**Fig. 3** The CD of FRC algorithm with different $\alpha$ values

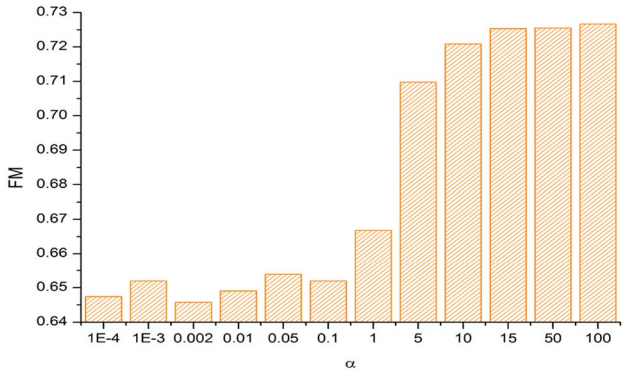


**Fig. 2** The FM of FRC algorithm with different $\alpha$ values

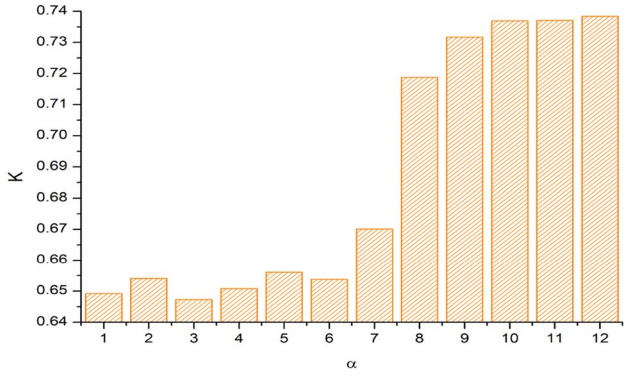


**Fig. 4** The K of FRC algorithm with different $\alpha$ values

equivalence partition of current attribute to determine the importance of current attribute in each cluster; DKmodes considers the frequency of mode components in the current cluster when it determines the dissimilarity between data point and center point; it is obvious that WKModes and DKModes introduce the more complex dissimilarity measurement methods to measure the dissimilarity, therefore WKModes and DKModes cost more time than k-modes algorithm; for FRC algorithm, it uses information granularity of equivalence partition on each attribute to determine the weight of the attribute and also uses the significance of each attribute for each data point to transform categorical data set into numeric data set; the time complexity of the above steps is $\mathcal{O}(mn^2)$; the time complexity of the dimension reduction of FRC algorithm is $\mathcal{O}(n^2)$; the time complexity of Eq(19) is $\mathcal{O}(n \cdot k \cdot d \cdot I) \approx \mathcal{O}(n)$ where $I$ is the number of iterations; therefore the time complexity of FRC algorithm is $\mathcal{O}(mn^2)$. For the comparison algorithms, there is no the data set type conversion mechanism; therefore the time complexity of each comparison algorithm is not larger than $\mathcal{O}(mn^2)$. From the analysis, it is known that the complexity of FRC algorithm is highest.

In order to test the impacts of the parameter $\alpha$ on the performance of FRC algorithm, we choose Germany as the experimental data set and FRC algorithm is executed on the data set; $\alpha$ is changed in [0, 100] and the other parameter is set as the above experiment; the test results are showed as Figs. 1, 2, 3 and 4.

Figures 1, 2, 3 and 4 show the results of FRC algorithm tested on Germany data set with different $\alpha$ values. From the results, it can be seen that the improvement of the performance of FRC algorithm is the largest when $\alpha = 5$. When $\alpha < 15$, the performance of FRC algorithm increases with the increase of $\alpha$ value; after $\alpha$ is larger than 15, the performance of FRC algorithm almost keeps stable. For FRC algorithm, $\alpha$ is a parameter which can affect the generalization ability of FRC algorithm. If $\alpha \to 0$, Eq. (13) is equal to LPP algorithm [16]. If $\alpha > 0$, the optimization problem of Eq. (13) considers the the generalization ability, therefore FRC algorithm can obtain a better generalization ability and the test results of Figs. 1, 2, 3 and 4 also prove the conclusion. If $\alpha$ is a large value, the impact of the first term of Eq. (13) will be decreased. If $\alpha$ is too large, the impact of the first term of Eq. (13) is almost

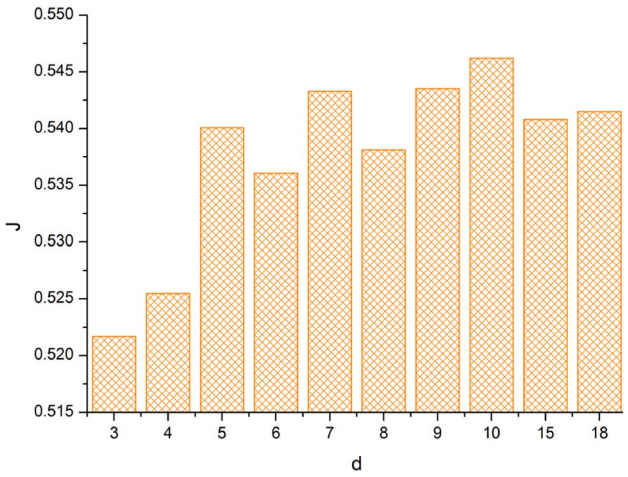


**Fig. 5** The J of FRC algorithm with different dimensions

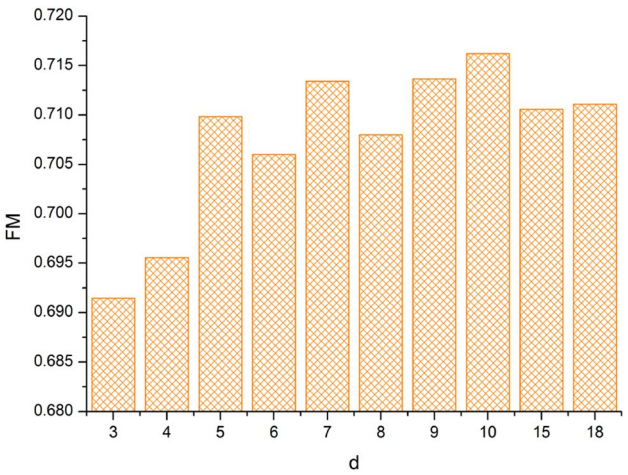


**Fig. 6** The FM of FRC algorithm with different dimensions

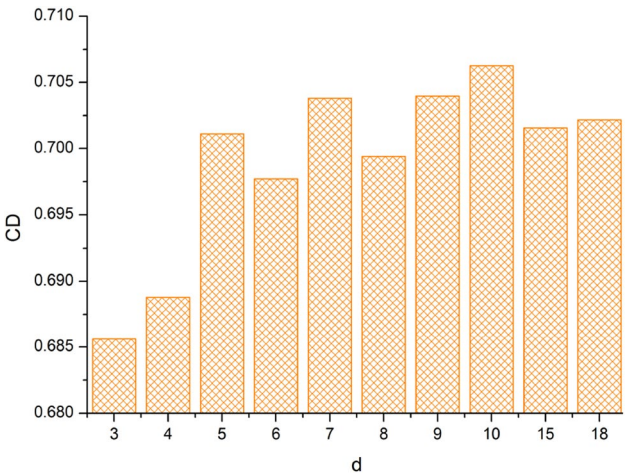


**Fig. 7** The CD of FRC algorithm with different dimensions

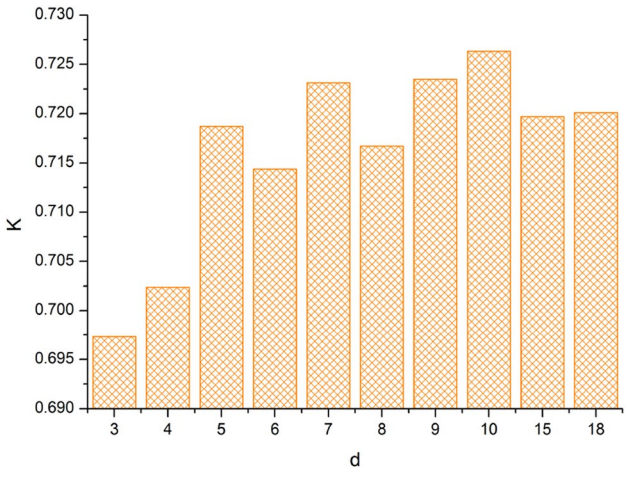


**Fig. 8** The K of FRC algorithm with different dimensions

ignored. In a word, an appropriate $\alpha$ value is important for the performance of FRC algorithm.

In order to test the scalability of FRC algorithm, we choose Germany as the experimental data set; the parameter is set as $\alpha = 5$; FRC is executed on adult data set with different dimensions. The test results are showed as Figs. 5, 6, 7 and 8 and Table 9.

Figures 5, 6, 7 and 8 show the test results of FRC algorithm on adult data set with different dimensions. From the results, it can be seen that the values of evaluation criteria are different when $d$ is different. In other words, $d$ can effect the performance of FRC algorithm. For FRC algorithm, $d$ determines the dimensions of data set after dimension reduction. If $d$ is too small, much effective discriminant information is removed from data. If $d$ is too large, it is obvious that much reductant or ineffective information is also added into data which decreases the performance of FRC algorithm. Table 9 shows the time cost of FRC algorithm with different dimensions. From the results, it is known that the overall change of time cost is increased. It is known that the time cost of FRC algorithm is mainly determined by the data processing of Sects. 3.1 and 3.2 and the time complexity is $\mathcal{O}(mn^2)$; if $d$ is changed, it can effect the time cost of the clustering steps and the time complexity of Eq. (19) is $\mathcal{O}(n \cdot k \cdot d \cdot I)$. Therefore the change of the time cost of FRC algorithm is also changed when $d$ is changed.

In order to test the effectiveness of FRC algorithm, we choose Germany as the experimental data set and execute FRC algorithm and FRC algorithm without dimension reduction which is denoted as UFRC. For the two algorithm, $\alpha$ is set to 5, $d = 3$. The results are showed as Fig. 9.

Figure 9 shows the results of FRC algorithm and UFRC algorithm on Germany data set. From the results, it is known that FRC algorithm outperforms UFRC algorithm on J, FM, K and CD evaluation criteria. It means the dimension

**Table 9** The time cost of FRC algorithm on adult data set with different dimensions

| d | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 15 | 18 |
|---|---|---|---|---|---|---|---|---|---|---|
| Time cost | 633.5584 | 708.4287 | 728.4625 | 736.2965 | 775.9457 | 799.7833 | 751.7610 | 835.5449 | 852.2152 | 727.2187 |

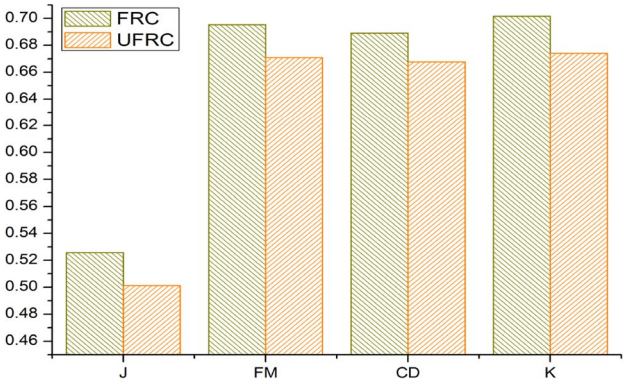


**Fig. 9** The test results of FRC algorithm and FRC algorithm on Germany data set

reduction can improve the performance of the algorithm. For FRC algorithm, dimension reduction is to remove and eliminate irrelevant and redundant information. In the dimension reduction algorithm, nonlinear transformation is introduced; redundant or redundant information can be removed, therefore the dimension reduction algorithm can improve the performance of FRC algorithm.

## 5 Conclusions

In this paper, a fuzzy rough clustering algorithm (FRC) for categorical data is proposed. FRC algorithm uses the partition of rough set to compute the information granularity of each attribute and introduces information granularity to determine the weights of attributes. Different from original k-modes algorithm, FRC algorithm transforms categorical data set into numeric data set and employs nonlinear dimension reduction algorithm to decrease the dimensions of data set; the objective optimization function of RFC algorithm considers intra-cluster distance and inter-cluster distance of clustering result and FRC algorithm can obtain a clustering result with minimum intra-cluster dissimilarity and maximum inter-cluster dissimilarity. FRC algorithm and the comparison algorithms are executed on real data sets. The experimental results show that RFC algorithm outperforms the comparison algorithms on the most data sets and it proves that FRC algorithm is an effective clustering algorithm for categorical data. However, FRC algorithm randomly chooses $k$ modes which reduces the convergence speed; in addition, the weights of attributes are the same for each cluster which is not fit for the theory of subspace learning. Therefore we can introduce a method to select high quality initial center points and use subspace clustering to further improve the performance of FRC algorithm in the future.

**Acknowledgements** This work was supported by National Key Research and Development Program of China (Nos.2017YFB1300200, 2017YFB1300203), National Natural Science Fund of China (Nos.61972064, 61672130, 61602082, 61627808, 91648205), the Open Program of State Key Laboratory of Software Architecture (No.SKLSAOP1701), LiaoNing Revitalization Talents Program (No. XLYC1806006), the Fundamental Research Funds for the Central Universities (Nos. DUT19RC(3)012, DUT17RC(3)071) and the development of science and technology of Guangdong province special fund project (No.2016B090910001). The authors are grateful to the editor and the anonymous reviewers for constructive comments that helped to improve the quality and presentation of this paper.

## References

1. An S, Hu QH, Yu DR (2015) Robust rough sets and applications. Tsinghua University Press, Tsinghua
2. Andritsos P, Tsaparas P, Miller RJ, Sevcik KC (2004) Limbo: scalable clustering of categorical data. In: International conference on extending database technology. Springer, pp. 123–146
3. Cao F, Liang J, Li D, Bai L, Dang C (2012) A dissimilarity measure for the k-modes clustering algorithm. Knowl Based Syst 26:120–127
4. Cao F, Liang J, Li D, Zhao X (2013) A weighting k-modes algorithm for subspace clustering of categorical data. Neurocomputing 108:23–30
5. Chaturvedi A, Green PE, Caroll JD (2001) K-modes clustering. J Class 18(1):35–55
6. Chen K, Liu L (2005) The“ best k” for entropy-based categorical data clustering. In: international conference on scientific and statistical database management, pp 253–262
7. Correa ES, Freitas AA, Johnson CG (2006) A new discrete particle swarm algorithm applied to attribute selection in a bioinformatics data set. In: Proceedings of the 8th annual conference on Genetic and evolutionary computation. ACM, pp 35–42
8. Fan J, Niu Z, Liang Y, Zhao Z (2016) Probability model selection and parameter evolutionary estimation for clustering imbalanced data without sampling. Neurocomputing 211:172–181
9. Fan JC, Li Y, Tang LY, Wu GK (2018) Roughpso: rough set-based particle swarm optimisation. Int J Bio-Inspir Comput 12(4):245–253
10. Feng L, Xu S, Wang F, Liu S, Qiao H (2019) Rough extreme learning machine: a new classification method based on uncertainty measure. Neurocomputing 325:269–282
11. Fern XZ, Brodley CE (2003) Random projection for high dimensional data clustering: a cluster ensemble approach. In: Proceedings of the 20th international conference on machine learning (ICML-03), pp 186–193

12. Fu L, Niu B, Zhu Z, Wu S, Li W (2012) Cd-hit: accelerated for clustering the next-generation sequencing data. Bioinformatics 28(23):3150–3152
13. Gao C, Pedrycz W, Miao D (2013) Rough subspace-based clustering ensemble for categorical data. Soft Comput 17(9):1643–1658
14. Gong Z, Zhang X (2017) The further investigation of variable precision intuitionistic fuzzy rough set model. Int J Mach Learn Cybern 8(5):1565–1584
15. Guha S, Rastogi R, Shim K (2000) Rock: A robust clustering algorithm for categorical attributes. Information systems 25(5):345–366
16. He X, Niyogi P (2004) Locality preserving projections. In: Advances in neural information processing systems, pp 153–160
17. Hu X, Tang J, Gao H, Liu H (2013) Unsupervised sentiment analysis with emotional signals. In: Proceedings of the 22nd international conference on World Wide Web. ACM, pp 607–618
18. Kim M, Kim I, Lee M, Jang B (2018) Worldwide emerging disease-related information extraction system from news data. In: Proceedings of the 16th ACM conference on embedded networked sensor systems. ACM, pp 331–332
19. Li C, Zhu L, Luo Z (2018) Underdetermined blind separation via rough equivalence clustering for satellite communications. In: 2018 international symposium on networks, computers and communications (ISNCC). IEEE, pp 1–5
20. Li W, Jia X, Wang L, Zhou B (2019) Multi-objective attribute reduction in three-way decision-theoretic rough set model. Int J Approx Reason 105:327–341
21. Li Y, Li D, Wang S, Zhai Y (2014) Incremental entropy-based clustering on categorical data streams with concept drift. Knowl Based Syst 59:33–47
22. Lin T, Zha H (2008) Riemannian manifold learning. IEEE Trans Pattern Anal Mach Intell 30(5):796–809
23. Nath B, Bhattacharyya D, Ghosh A (2013) Incremental association rule mining: a survey. Wiley Interdiscip Rev Data Min Knowl Discov 3(3):157–169
24. Ng MK, Li MJ, Huang JZ, He Z (2007) On the impact of dissimilarity measure in k-modes clustering algorithm. IEEE Trans Pattern Anal Mach Intell 3:503–507
25. Parmar D, Wu T, Blackhurst J (2007) Mmr: an algorithm for clustering categorical data using rough set theory. Data Knowl Eng 63(3):879–893
26. Rekik R, Kallel I, Casillas J, Alimi AM (2018) Assessing web sites quality: a systematic literature review by text and association rules mining. Int J Inf Manag 38(1):201–216
27. Song L, Tekin C, van der Schaar M (2016) Online learning in large-scale contextual recommender systems. IEEE Trans Serv Comput 9(3):433–445
28. Steinbach M, Karypis G, Kumar V et al (2000) A comparison of document clustering techniques. In: KDD workshop on text mining, vol 400. Boston, pp. 525–526
29. Tiwari AK, Shreevastava S, Som T, Shukla KK (2018) Tolerance-based intuitionistic fuzzy-rough set approach for attribute reduction. Expert Syst Appl 101:205–212
30. Wang R, Wang XZ, Kwong S, Xu C (2017) Incorporating diversity and informativeness in multiple-instance active learning. IEEE Trans Fuzzy Syst 25(6):1460–1475
31. Wang XZ, Wang R, Xu C (2017) Discovering the relationship between generalization and uncertainty by incorporating complexity of classification. IEEE Trans Cybern 48(2):703–715
32. Wang XZ, Xing HJ, Li Y, Hua Q, Dong CR, Pedrycz W (2014) A study on relationship between generalization abilities and fuzziness of base classifiers in ensemble learning. IEEE Trans Fuzzy Syst 23(5):1638–1654
33. Wang XZ, Zhang T, Wang R (2019) Noniterative deep learning: incorporating restricted boltzmann machine into multilayer random weight neural networks. IEEE Trans Syst Man Cybern Syst 49(7):1299–1380
34. Xie J (2016) Unsupervised learning methods and applications. Publishing Hourse of Electronics Industry, Beijing
35. Xu R, Wunsch D (2005) Survey of clustering algorithms. IEEE Trans Neural Netw 16(3):645–678
36. Yang Q, Du Pa, Wang Y, Liang B (2018) Developing a rough set based approach for group decision making based on determining weights of decision makers with interval numbers. Oper Res 18(3):757–779
37. Yao Y (2007) Decision-theoretic rough set models. In: International conference on rough sets and knowledge technology. Springer, pp 1–12