

## Comparing clustering models in bank customers: Based on Fuzzy relational clustering approach

Ayad Hendalianpour<sup>a\*</sup>, Jafar Razmi<sup>a</sup> and Mohsen Gheitasi<sup>b</sup>

<sup>a</sup>*School of Industrial Engineering, College of Engineering, Tehran University, Tehran, Iran*

<sup>b</sup>*School of Industrial Engineering, College of Engineering, Shiraz Azad University, Shiraz, Iran*

### CHRONICLE

#### Article history:

Received December 5, 2015

Received in revised format

February 16 2016

Accepted August 15 2016

Available online

August 16 2016

#### Keywords:

*K-mean*

*C-mean*

*Fuzzy C-mean*

*Kernel K-mean*

*Fuzzy variables*

*Fuzzy relation clustering (FRC)*

### ABSTRACT

Clustering is absolutely useful information to explore data structures and has been employed in many places. It organizes a set of objects into similar groups called clusters, and the objects within one cluster are both highly similar and dissimilar with the objects in other clusters. The K-mean, C-mean, Fuzzy C-mean and Kernel K-mean algorithms are the most popular clustering algorithms for their easy implementation and fast work, but in some cases we cannot use these algorithms. Regarding this, in this paper, a hybrid model for customer clustering is presented that is applicable in five banks of Fars Province, Shiraz, Iran. In this way, the fuzzy relation among customers is defined by using their features described in linguistic and quantitative variables. As follows, the customers of banks are grouped according to K-mean, C-mean, Fuzzy C-mean and Kernel K-mean algorithms and the proposed Fuzzy Relation Clustering (FRC) algorithm. The aim of this paper is to show how to choose the best clustering algorithms based on density-based clustering and present a new clustering algorithm for both crisp and fuzzy variables. Finally, we apply the proposed approach to five datasets of customer's segmentation in banks. The result of the FCR shows the accuracy and high performance of FRC compared other clustering methods.

© 2017 Growing Science Ltd. All rights reserved.

## 1. Introduction

Clustering has been a widely studied problem in the machine learning literature (Filippone et al., 2008; Jain, 2010). Clustering algorithms have been addressed in many contexts and disciplines such as data mining, document retrieval, image segmentation and pattern recognition. The prevalent clustering algorithms have been categorized in different ways depending on different criteria. As with many clustering algorithms, there is a trade-off between speed and quality of the resulting results. The existing clustering algorithms can be simply classified into two categories, hierarchical clustering and partitioned clustering (Jain, 2010; Jiang et al., 2010; Feng et al., 2010). Clustering can also be performed in two different modes, hard and fuzzy. In hard clustering, the clusters are disjoint and non-overlapping in nature. Any pattern may belong to one and only one class in this case. In the case of

\* Corresponding author. Tel: +989173396702

E-mail address: [hendalianpour@ut.ac.ir](mailto:hendalianpour@ut.ac.ir) (A. Hendalianpour)

fuzzy clustering, a pattern may belong to all the classes with a certain fuzzy membership grade (Jain, 2010; Pedrycz & Rai, 2008; Peters et al., 2013). Hierarchical clustering algorithms iteratively build clusters by joining (agglomerative) or dividing (divisive) the clusters from the previous iteration (Kannappan et al., 2011; Chehreghani et al., 2009). The agglomerative approach starts from the finest clustering with one of the  $n-1$  element clusters given  $n$  objects and finishes at the most coarse clustering, with one cluster consisting of all  $n$  objects. The divisive approach works in another way, from the coarsest partition to the finest partition.

The resulting tree has nodes created at each cutoff point that can be used to generate different clustering. There is an enormous variety of agglomerative algorithms in the literature: single-link, complete-link, and average-link (Höppner, 1999; Akman, 2015). The single-link algorithm or nearest neighbor algorithm has a strong tendency to chain in a geometrical sense and not balls, an effect which is not desired in some applications; groups which are not quite well separated cannot be detected. The complete-link has the tendency to build small clusters. The average-link algorithm builds a compromise between the two extreme cases of single-linkage and complete-linkage (Eberle et al., 2012; Lee et al., 2005; Clir, & Yuan, 1995). Contrary to the agglomerative algorithms, divisive algorithms start with the largest clustering, i.e., the clustering with exactly one cluster. The cluster will be separated into two clusters in the sense that one tries to optimize a given optimization criterion (Ravi & Zimmermann, 2000; Garrido, 2011). The popular clustering algorithms has been widely used to solve problems in many areas, for instance the K-mean is very sensitive to initialization, the better centers we choose, the better results we get (Khan & Ahmad, 2004; Núñez et al., 2014), but has some of weakness and we can't use this algorithm everywhere and this algorithm can't get crisp, fuzzy and linguistic variables together. Regarding this, in this paper, we propose a new algorithm based on fuzzy variables and fuzzy relation called Fuzzy Relation Clustering (FRC) algorithm.

The organization of the remainder is as follows: section 2 reviews clustering algorithms. Section 3, present the Fuzzy variables and Fuzzy relation clustering (FRC) algorithm. Section 4 briefly introduces the three internal validity indices and the external validity indices. Section 5 describes the dataset. In section 6, we present the output of the four clustering algorithms. At the end, a concluding remark is given in section 7.

## 2. Review of clustering algorithms

### 2.1. *k*-mean

K-mean algorithm is an effective and easy algorithm for clusters in data sets (Lee et al., 2005). The process of the K-mean algorithm is as follows:

- First stage: the user is asked how many cluster  $k$ 's are formed in data sets.
- Third stage: for each record, find the nearest center cluster; to some extent, we can say the center cluster itself is a subset of records. In other words, partition representation separation of data collection, thus we have  $k$  cluster  $C_1, C_2, \dots, C_k$
- Fourth stage: for each  $k$  cluster, search center bunch and center. Update the station of each cluster to the new value of center.
- Fifth stage: continue stages 3 to 5 until reaching convergence or end.
- Usually Second stage: allocate record  $k$  to the first station of center cluster randomly.

The nearest criterion is Euclidean distance in stage 3, although the other criterion may have a better application. Suppose that we have  $n$  point data,  $(a_1, b_1, c_1), (a_2, b_2, c_2), \dots, (a_n, b_n, c_n)$ . The center of these points is compared with the center of gravity of these points and put the situation

$$\left( \sum \frac{a_i}{n}, \sum \frac{b_i}{n}, \sum \frac{c_i}{n} \right), \text{ for example, points } (1,1,1), (1,2,1), (1,3,1) \text{ with center } (2,1,1) :$$

$$\left(\frac{1+1+1+2}{4}, \frac{1+2+3+1}{4}, \frac{1+1+1+1}{4}\right) = (1.25, 1.75, 1.00)$$

End of algorithms, while that center has very few changes. In other words, the end of the algorithms, while that for all clusters  $C_1, C_2 \dots C_k$ . Obtain ownership of all the records by asking whether each center will remain a cluster in that cluster; also, although the algorithms finish, some of the convergence criterion is obtained. The algorithm ends when a certain convergence a criterion is viewed as being a major reduction in the total square error is not present:

$$SSE = \sum_{i=1}^k \sum_{p \in c_i} d(p, m_i)^2 \quad (1)$$

where  $p \in c_i$  denote each point of the data in cluster  $i$  and center cluster  $m_i$ . As was observed, k-means algorithm does not guarantee that the global minimum SSE will be found, instead, it is often placed in a local minimum. Increasing the chances for reaching the global minimum, analysis should be used for the initial cluster centers algorithm with different values. The main point is to first select place of cluster's centers in the first stage in the random form. Secondly, for the next stage the cluster's centers may be far from the first centers. One of the potential problems in employing k-mean algorithm is who decides how many clusters should be found, unless the analyst has previous knowledge about the number of fundamental clusters. In this state, there may be an increase in an external loop to algorithms. The loop from different probable quantities  $k$ . can then compare the solution of clustering for each value of  $k$ , then the value of  $k$  that has a minimum of SSE.

## 2.2. C-mean

C-mean algorithm is used for hard clustering approaches, meaning that in this way, each data is allocated just to one cluster (Filippone et al., 2008), define a family from collections on the source collection "X" in this form  $A_i, i=1,2,\dots,c$ , as  $c$  is the number of clusters or groups for clustering data. ( $2 \leq c < n$ ). The C-mean algorithm is as follows:

- Select a value for "c" as number of clusters ( $2 \leq c < n$ ) and contemplate primary matrix  $U^* \in M_c$ , then do the next stages for  $r = 1, 2, \dots$
- Calculate vector of center C:  $V_i^{(r)}$  with  $U^{(r)}$
- Update  $U^{(r)}$  and calculate updated membership functions for all  $k$  and  $i$  using the bottom connection.

$$X_{ik}^{(r+i)} = \begin{cases} d_{ik}^{(r)} = \min(d_{jk}^{(r)}) & \text{for } j \in c \\ 0 & \text{Otherwise} \end{cases} \quad (2)$$

- If the greatest value of (6) difference match elements of matrix  $U^{(r)}$  and  $U^{(r+1)}$  are smaller or equal than accepted level attention, then finish calculating  $\|U^{(r+1)} - U^{(r)}\| \leq \varepsilon$ , otherwise  $r = r + 1$  and repeat stage 2.

## 2.3. Fuzzy C- mean

This algorithm, offered by Schölkopf et al. (1998) is a skilled algorithm for fuzzy clustering of data and, in fact, is developed to the form of mean clustering  $c$ . For the development of this algorithm in clustering, define a family of fuzzy collections in form  $\{\tilde{A}_i, i = 1, 2, \dots, c\}$  under title a fuzzy separation (division) on a source collection. Now, present the algorithm for assigning fuzzy c-mean  $\tilde{U}$  for clustering of  $n$  data in  $c$  cluster. For this work an aim function  $J_m$  in an objective function, we define as follows.

$$J_m(\tilde{U}, V) = \sum_{k=1}^n \sum_{i=1}^c (\mu_{ik})^m (d_{ik})^2 \quad (3)$$

So that  $d_{ik}$  is the Euclidean distance between center of cluster  $i$  and data  $k$ .

$$d_{ik} = d(x_k - v_i) = \left[ \sum_{j=1}^m (x_{kj} - v_{ij})^2 \right] \quad (4)$$

So that  $\mu_{ik}$  is equal to membership degree of data  $k$  divided to cluster  $i$ . The least value of  $J_m$  will connect to the best clustering state. Here, a new parameter ( $m$ ) introduced by the name of parameter of weight, in which the changes Interval is in form  $m' \in [1, \infty)$ . This parameter is a distinct degree of fuzzy in clustering progress. Also, a similar previous state  $V_i$  is marked as the center coordinates of bunch  $i$ , so  $V_i = \{v_{i1}, v_{i2}, \dots, v_{im}\}$ , so that  $m$  is the number of  $V_i$  distances or is numbers of criterion similar to center coordinates of the bunch obtained from the relation shown below.

$$V_{ij} = \frac{\sum_{k=1}^n \mu_{ik}^m \cdot x_{kj}}{\sum_{k=1}^n \mu_{ik}^m} \quad (5)$$

So that  $j$  is changeable unit for showing the criterion area,  $j = 1, 2, 3, \dots, m$ . Thus, in this algorithm when optimum separation fuzzy is obtained  $J$  is minimized in the bottom relation.

$$J_m^* = (\tilde{U}^*, V^*) = \min_{M_{fc}} J(\tilde{U}, V) \quad (6)$$

The Fuzzy C-mean algorithm is as shown below:

- Selects a value for  $c$  under title of cluster's number ( $2 \leq c < n$ ) and selects a value for  $m'$ . Suppose the first separation matrix  $U^{(0)}$ , each time this algorithm is distinct with  $r, r = 0, 1, 2, \dots$
- Calculate center of cluster  $(V_i^{(r)})$  in each review.
- Update the separation matrix for  $r$  repetition  $\tilde{U}^{(r)}$  in the bottom form.

$$\mu_{(ik)}^{(r+1)} = \left[ \sum_{j=1}^c \left( \frac{d_{ik}^{(r)}}{d_{jk}^{(r)}} \right)^{\frac{2}{m-1}} \right] \quad \text{for } I_k = \phi \quad (7)$$

- So that  $I_k = \left\{ i \mid 2 \leq c < n ; d_{ik}^{(r)} = 0 \right\}, \tilde{I}_k = \{1, 2, 3, \dots, c\} - I_k, \sum_{i \in I_k} \mu_{ik}^{(r+1)} = 1$
- If  $\|U^{(r+1)} - U^{(r)}\| \leq \varepsilon_L$ , then finish the calculation and in this form return to stage 2.

#### 2.4. Kernel $k$ -mean

Given the data set  $X$ , we map our data in some feature space  $\tau$ , by means of a nonlinear map and we consider  $k$  centers in feature space ( $V_i^\phi \in J\tau, i = 1, \dots, K$ ). We call the set  $V^\phi = (V_1^\phi, \dots, V_K^\phi)$ , Feature Space Codebook, since in our representation the centers in the feature space play the same role of the code vectors in the input space. In analogy with the code vectors in the input space, we define for each center  $V_i^\phi$  its Voronoi region and Voronoi set in feature space. The Voronoi region in feature space  $R_i^\phi$  of the center  $V_i^\phi$  is the set of all Vectors in  $\tau$  for which  $V_i^\phi$  is the closest vector (Filippone et al., 2008):

$$R_i^\phi = \left\{ X^\phi \in \tau \mid i = \arg \min_i \|\phi(x) - v_i^\phi\| \right\} \quad (8)$$

The Voronoi set in feature space  $\pi_i^\phi$  of the center  $V_i^\phi$  is the set of all vectors  $\phi(x)$  in  $X$  such that  $V_i^\phi$  is the closest vector to their images  $\phi(x)$  in the feature space:

$$\pi_i^\phi = \left\{ X \in X \mid i = \arg \min_i \|\phi(x) - v_i^\phi\| \right\} \quad (9)$$

The set of the Voronoi regions in feature space define a Voronoi Tessellation of the feature space.

The Kernel K-means the algorithm has the following steps:

- Project the data set  $X$  into a feature space  $\tau$ , by means of a nonlinear mapping  $\phi$ .
- Initialize the codebook  $V^\phi = (V_1^\phi, \dots, V_K^\phi)$  with  $v_i^\phi \in \tau$ .
- Compute for each center  $v_i^\phi$  the set  $\pi_i^\phi$
- Update the code vectors  $v_i^\phi$  in  $\tau$ :

$$V_i^\phi = \frac{1}{|\pi_i^\phi|} \sum_{x \in \pi_i^\phi} \phi(x) \quad (10)$$

- Go to step 3 until any  $v_i^\phi$  changes.
- Return the feature space codebook.

This algorithm minimizes the quantization error in feature space. Since we do not know explicitly, it is not possible to compute Eq (10) directly. Nevertheless, it is always possible to compute distances between patterns and code vectors by using the kernel trick, allowing the Voronoi sets in feature space  $\pi_i^\phi$  to be obtained. Indeed, writing each centroid in feature space as a combination of data vectors in feature space, we have:

$$V_i^\phi = \sum_{h=1}^n \gamma_{ih} \Phi(x_h) \quad (11)$$

where  $\gamma_{jk}$  is one if  $x_h \in \pi_i^\phi$  and zero otherwise. Now the quantity:

$$\|\Phi(x_i) - V_j^\phi\|^2 = \left\| \Phi(x_i) - \sum_{h=1}^n \gamma_{jh} \Phi(x_h) \right\|^2 \quad (12)$$

$$\left\| \Phi(x_i) - \sum_{h=1}^n \gamma_{jh} \Phi(x_h) \right\|^2 = K_{ii} - 2 \sum_h \gamma_{jh} k_{ih} + \sum_r \sum_s \gamma_{jr} \gamma_{js} K_{rs} \quad (13)$$

This is the closest possible analog vector space model to provide a combination of  $\gamma_i^\phi$  coefficients for each update. Repeat this process until there are two possibilities and  $\gamma_i^\phi$  get the votes to change the active compound Voronoi space.

An on-line version of the kernel K-means the algorithm can be found in Clir and Yuan (1995). A further version of K-means in feature space has been proposed by Garrido (2011). In his formulation, the number of clusters is denoted by  $c$ , and a fuzzy membership matrix  $U$  is introduced. Each element  $u_{ih}$  denotes the fuzzy membership of the point  $x_h$  to the Voronoi set  $\pi_i^\phi$ . This algorithm tries to minimize the following functional with respect to  $U$ :

$$J^\Phi(U, V) = \sum_{h=1}^n \sum_{i=1}^c u_{ih} \|\Phi(x_h) - v_i^\phi\|^2 \quad (14)$$

The minimization technique used by Garrido (2011) is deterministic annealing, which is a stochastic algorithm for optimization. A parameter controls the fuzziness of the membership during the optimization and can be proportional to the temperature of a physical system. This parameter is gradually lowered during the annealing, and at the end of the procedure, the memberships have become crisp; therefore, a tessellation of the feature space is found. This linear partitioning in  $F$ , back to the input space, forms a nonlinear partitioning of the input space.

### 3. Fuzzy Relation Clustering (FRC)

This section describes the details of the computational model used for FRC algorithm. At first, it is important to note that first there is an overview of the fuzzy variables. The algorithm itself is fully unaware of the concept of customer clustering of bank, then we describe the FRC algorithm.

#### 3.1. Fuzzy variable

Many sentences in natural language express numerical sizes such as good, hot, short, young, and etc, which should be considered as a numerical scale for better understanding (Liang et al., 2005). Making a set of amounts to be constant; if  $x \in A$ , then  $x$  is high and if  $x \notin A$ , then  $x$  is not high. This process was used in traditional systems. The problem of this process is that “this is so sensitive about lack of accuracy in numerical data or its variation. In order to consider that part of no numerical information, a syntactic representation is necessary. Verbal terms are the variables which are tighter than fuzzy variables, because they accept fuzzy variables as their own amounts. The fuzzy variables, their amounts, words or sentences in one language are natural or artificial. For example, the temperature of a liquid reservoir is a fuzzy variable if it allocates amounts such as cool, cold, hot and warm. Age can be a fuzzy variable if its amounts are old, young, and etc. We can conveniently see that fuzzy variables provide a suitable tool for optimal and approximate description of complicated phenomena.

#### 3.2. Fuzzy relation

The proposed model for market segmentation is based on fuzzy relation. The key concepts in fuzzy relation are reviewed as follows:

##### 3.2.1. Fuzzy relation

Fuzzy relation is a fuzzy subsets of  $X \times Y$ , that is, mapping from  $X \rightarrow Y$ . Let  $X, Y \subseteq R$  be universal sets, then  $R$  is called a fuzzy relation on  $X \times Y$ .

$$R = \{(x, y), \mu_R(x, y) | (x, y) \in X \times Y\}$$

##### 3.2.2. Max–Min composition

Let  $R_1(x, y)$  and  $R_2(y, z)$  be two fuzzy relations,  $(x, y) \in X \times Y$  and  $(y, z) \in Y \times Z$ . Then max–min composition  $R_1 \circ R_2$  is defined by:

$$R_1 \circ R_2 = \{(x, z), \text{Max}_y \{ \text{Min} \{ \mu_{R_1}(x, y), \mu_{R_2}(y, z) \} \} | x \in X, y \in Y, z \in Z\}$$

##### 3.2.3. Fuzzy equivalence relation

A fuzzy relation  $R$  on  $X \times X$  is called a fuzzy equivalence relation if the following three conditions are met

- (1) Reflexive, i.e.,  $\mu_R(x, x) = 1; \forall x \in X$
- (2) Symmetric, i.e.,  $R(x, y) = R(y, x); \forall x \in X, y \in Y$
- (3) Transitive, i.e.  $R^2 = R \circ R \subseteq R$

##### 3.2.4. Transitive closure

The transitive closure,  $R_T$ , of a fuzzy relation  $R$  is defined as the relation that is transitive, contains  $R$  and has the smallest possible membership grades.

**Theorem 1** (Zimmermann, 1996) Let  $R$  be a fuzzy reflexive and symmetric relation on a finite universal set  $X$  with  $|X| = n$ , then the max–min transitive closure of  $R$  is the relation  $R^{(n-1)}$ . According to Theorem 1, we can get the algorithm to find the transitive closure  $R_T = R^{(n-1)}$ .

## Algorithm

Step 1: Initialize  $K = 0$ , go to step 2.

Step2:  $K = K + 1$  if  $2^K \geq (n - 1)$  then

$R_T = R^{(n-1)}$  And stop. Otherwise, go to step 3.

Step3:  $R^* = R^{2^{k-1}} \circ R^{2^{k-1}}$  if  $R^* \subseteq R$  or  $R^* = R$

Then  $R_T = R^*$  and stop

Otherwise, go to step 2.

### 3.2.5. Fuzzy relation segmentation principle

The  $\alpha$ -cut set of fuzzy relation,  $R_\alpha$  defined as:

$$R_\alpha = \{(x, y), \mu_{R_\alpha}(x, y) | \mu_R(x, y) \geq \alpha, (x, y) \in X \times Y\}$$

An equivalence relation of a finite number of elements can also be represented by a tree. In this tree, each level represents an  $\alpha$ -cut of the equivalence relation (Zimmermann, 1996).

### 3.3. Customer segmentation

In this section, we will explain the different types of market's features and formulate fuzzy equivalence relation among markets. Then place them in groups according to similarity of their features.

#### 3.3.1. Customer Features

These features are expected to cause the opinion and adjustment of market about received product or service and they are categorized in three variable sets, while these are binary, quantitative and linguistic variables. The binary variables,  $X_1, X_2, \dots, X_{n_1}$  such as marital status, are shown by vector  $P$ , i. e.,

$$P_i = (x_{i1}, x_{i2}, \dots, x_{in_1}), i = 1, 2, \dots, m$$

where,  $m$  is a number of markets and  $n_1$  is number of binary variables. The relation among markets according to the binary feature is defined as classical relation with 0 or 1 quantity. If these features are more than one, then fuzzy relation with quantity between  $[0, 1]$  will be defined.

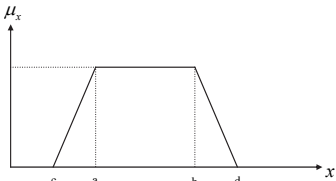
The quantitative variables,  $Y_1, Y_2, \dots, Y_{n_2}$ , such as age have real or integer values. We show them by vector  $Q$ , i.e.,  $Q_i = (y_{i1}, y_{i2}, \dots, y_{in_2}), i = 1, 2, \dots, m$  where,  $n_2$  is the number of quantitative variables. The relation among markets according to the quantitative feature depends on the distance measure of their values. Decreasing this distance makes costumer's relation strong, and vice versa. The linguistic variables,  $Z_1, Z_2, \dots, Z_{n_3}$ , have words or sentences in a natural or artificial language values, which are shown by fuzzy numbers. The vector of linguistic variables,  $V$ , is  $v_i = (A_{i1}^{L_1}, \dots, A_{ij}^{L_j}, \dots, A_{in_3}^{L_{n_3}})$  where,

$n_3$ : Number of linguistic variables

$K_j$ : Number of  $j$ -th linguistic variable values

$A_j^{L_j}$ : Value of  $j$ -th linguistic variable, ( $L_j = 1, 2, \dots, K_j$ )

The relation among markets according to a linguistic feature depends on the distance measure of their fuzzy number values. We utilize Chen and Hsieh's (Rose, 1998) modified geometrical distance algorithm based on the geometrical operation of trapezoidal fuzzy numbers. Based on this algorithm, the distance between two trapezoidal fuzzy numbers,  $A_i = (c_i, a_i, b_i, d_i)$  and  $A_k = (c_k, a_k, b_k, d_k)$  (in figure1), denoted by  $d_p(A_i, A_k)$ , and is:

$$d_p(A_i, A_k) = \begin{cases} [0.25(|c_i - c_k|^p + |a_i - a_k|^p + |b_i - b_k|^p + |d_i - d_k|^p)]^{\frac{1}{p}}, & 1 \leq p \leq \infty \\ \max\{|c_i - c_k|, |a_i - a_k|, |b_i - b_k|, |d_i - d_k|\}, & p = \infty \end{cases}$$


**Fig. 1.** Membership function of a trapezoidal fuzzy number

### 3.3.2. Customer Relations

We can get three fuzzy relation matrices,  $R_p, R_q$  and  $R_v$  from vectors  $P, Q$  and  $V$ , frequently.

$$R_p = \begin{matrix} & C_1 & C_2 & \dots & C_m \\ \begin{matrix} C_1 \\ C_2 \\ \vdots \\ C_m \end{matrix} & \begin{bmatrix} r_{11} & r_{12} & \dots & r_{1m} \\ r_{21} & r_{22} & \dots & r_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ r_{m1} & r_{m2} & \dots & r_{mm} \end{bmatrix} \end{matrix}$$

$$R_q = \begin{matrix} & C_1 & C_2 & \dots & C_m \\ \begin{matrix} C_1 \\ C_2 \\ \vdots \\ C_m \end{matrix} & \begin{bmatrix} r'_{11} & r'_{12} & \dots & r'_{1m} \\ r'_{21} & r'_{22} & \dots & r'_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ r'_{m1} & r'_{m2} & \dots & r'_{mm} \end{bmatrix} \end{matrix}$$

$$R_v = \begin{matrix} & C_1 & C_2 & \dots & C_m \\ \begin{matrix} C_1 \\ C_2 \\ \vdots \\ C_m \end{matrix} & \begin{bmatrix} r''_{11} & r''_{12} & \dots & r''_{1m} \\ r''_{21} & r''_{22} & \dots & r''_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ r''_{m1} & r''_{m2} & \dots & r''_{mm} \end{bmatrix} \end{matrix}$$

where  $C_i$  is  $i$ -th market ( $i = 1, 2, \dots, m$ ),  $0 \leq r_{ij}, r'_{ij}, r''_{ij} \leq 1$ . In fuzzy relation matrices  $r''_{ij}, r'_{ij}, r_{ij}$ , relation quantities between market  $i$  and  $j$ , are as follows:

$$r_{ij} = \frac{1}{\sum_{k=1}^{n_1} W_k^X} \sum_{k=1}^{n_1} W_k^X (1 - |x_{ik} - x_{jk}|) \quad (15)$$

$$r'_{ij} = \frac{1}{\sum_{k=1}^{n_2} W_k^Y} \sum_{k=1}^{n_2} W_k^Y \left(1 - \frac{|y_{ik} - y_{jk}|}{D_k}\right) \quad (16)$$

$$D_k = \max\{|y_{ik} - y_{jk}| \mid i, j = 1, 2, \dots, m\}, k = 1, 2, \dots, n_2 \quad (17)$$

$$r''_{ij} = \frac{1}{\sum_{k=1}^{n_3} W_k^Z} \sum_{k=1}^{n_3} W_k^Z \left(1 - \frac{d_p(A_{ik}^{L_k}, A_{jk}^{L_k})}{D'_k}\right) \quad (18)$$

$$D'_k = \max\{d_p(A_{ik}^{L_k}, A_{jk}^{L_k}) \mid i, j = 1, 2, \dots, m\}, k = 1, 2, \dots, n_3 \quad (19)$$

where,  $W_k^X$  is weight of variable  $X_k$  and ( $k = 1, 2, \dots, n_1$ ),  $W_k^Y$  is weight of variable  $Y_k$  and ( $k = 1, 2, \dots, n_2$ ) and  $W_k^Z$  is weight of variable  $Z_k$  and ( $k = 1, 2, \dots, n_3$ ). With these three matrices we can construct final fuzzy relation matrix  $R$  by the following equations:

$$R = W_p \cdot R_p + W_q \cdot R_q + W_v \cdot R_v \quad (20)$$

$$W_p + W_q + W_v = 1, (W_p, W_q, W_v \geq 0) \quad (21)$$

where,  $W_p$  is weight of  $R_p$ ,  $W_q$  is weight of  $R_q$  and  $W_v$  is weight of  $R_v$ .

### 3.3.3. Market segmentation

The fuzzy relation matrices,  $R_p, R_q$  and  $R_v$  are reflexive and symmetric because:

$$r_{ii} = r'_{ii} = r''_{ii} = 1 \quad (22)$$

$$r_{ij} = r_{ji}, r'_{ij} = r'_{ji} \text{ and } r''_{ij} = r''_{ji} \quad (23)$$



If these relations not are transitive, we can obtain transitive closure relation according to section (3.2). Then we can define relation  $R$  as an equation and make use of the fuzzy relation clustering principle to the markets segmentation according to their similarity (see section 3.2).

#### 4. Measures for evaluation of the clustering quality

Validity of clustering algorithms based on qualitative assessment of clustering is a way to resolve the issue. Generally there are three approaches for validating clustering algorithms. The first approach is based on internal criteria; external criteria on the second and the third approaches are relative criteria. The following briefly describes each of these three approaches.

- **Internal criteria:** The evaluation criteria categories are the clusters in the real structure. The aim of these criteria, the quality of clustering in real environments is derived from knowledge of clustering.
- **External criteria:** Validation of these criteria based on the comparison between the clustering with the clustering is done correctly. The evaluation of clustering algorithms to identify the performance on database is important.
- **Relative criteria:** The basis of these criteria is evaluation structure of base algorithms, with different input clustering algorithms.

In this paper, we use the internal criteria and external criteria to choose the best algorithms among K-mean, C-mean, Fuzzy C-mean and Kernel K-mean. For more details regarding internal and external criteria, the reader may refer to Aliguliyev (2009). Various cluster validity indices are available in the literature (Zhao & Karypis, 2004; Wu et al., 2009).

In internal criteria and external criteria measures, we used five indices, Below, we briefly introduce these indices.

- **Purity:** The purity gives the ratio of the dominant class size in the cluster to the cluster size itself. A large purity value implies that the cluster is a “pure” subset of the dominant class.
- **Mirkin:** This metric is obviously 0 for identical clustering’s, and positive otherwise.
- **F-measure:** The higher the F-measure, the better the clustering solution. This measure has a significant advantage over the purity and the entropy, because it measures both the homogeneity and the completeness of a clustering solution
- **V-measure:** The V-measure is an entropy-based measure that explicitly measures how successfully the criteria of homogeneity and completeness have been satisfied
- **Entropy:** Since the entropy considers the distribution of semantic classes in a cluster, it is a more comprehensive measure than the purity. Unlike the purity measure, an entropy value of 0 means that the cluster is comprised entirely of one class, while an entropy value near 1 implies that the cluster contains a uniform mixture of all classes. The global clustering entropy of the entire collection is defined to be the sum of the individual cluster entropies weighted according to the cluster size.
- **Resultant rank:** the Resultant rank is Statistical method showing the clustering algorithms ranks based on above indices.

In the next section we compare the output of popular clustering algorithms (K-mean, C-mean, Fuzzy C-mean and Kernel K-mean) and fuzzy relation clustering algorithm based on four dataset of customers segmentation in banks of Fars Province, Shiraz, Iran.

#### 5. Dataset

To compare and evaluate the output of clustering algorithms, we used the dataset of customer's segmentation in five banks of Fars Province, Shiraz, Iran. The datasets of the banks have standards for comparison among the clustering algorithms of this research. In Table 1 we describe characteristics of data set for each bank of these datasets.

**Table 1**

Characteristics of data set of five bank considered.

Attribute	Bank 1		Bank 2		Bank 3		Bank 4		Bank 5	
	Value Type	Size	Value Type	Size	Value Type	Size	Value Type	Size	Value Type	Size
Age	Linguistic	38586	Linguistic	27467	Linguistic	32654	Linguistic	25834	Linguistic	30673
Gender	Linguistic	38576	Linguistic	27456	Linguistic	32621	Linguistic	25834	Linguistic	30600
Education	quantitative	38586	quantitative	27467	quantitative	32654	quantitative	25800	quantitative	30673
Annual Income	Binary	38559	Binary	27467	Binary	32654	Binary	25834	Binary	30673
Marital status	Quantitative	38585	Quantitative	27400	Quantitative	32640	Quantitative	25834	Quantitative	30697
Average of account	Binary	38586	Binary	27467	Binary	32654	Binary	25806	Binary	30673
Occupation	Quantitative	38586	Quantitative	27467	Quantitative	32654	Quantitative	25834	Quantitative	30656
Marriage status	Quantitative	38586	Quantitative	27480	Quantitative	32633	Quantitative	25045	Quantitative	30612
Affiliation status	Quantitative	38586	Quantitative	27455	Quantitative	32600	Quantitative	25865	Quantitative	30510
Cash flow after tax	Binary	38586	Binary	26799	Binary	32630	Binary	25400	Binary	30614

Tables 2 to Table 6 present popular statistical analysis for each bank data sets. In each these tables we calculate three statistical measures such as: mean, standard deviation and Variance.

**Table 2**

Statistical analysis of data set (bank 1)

Attribute	Statistical Methods		
	Mean	Standard deviation	Variance
Age	4.55	0.671	0.143
Gender	4.09	0.921	0.196
Education	3.82	0.853	0.182
Annual Income	3.36	0.727	0.155
Marital status	4.09	0.811	0.173
Average of account	3.23	1.020	0.218
Occupation	4.09	0.684	0.146
Marriage status	3.59	0.734	0.157
Affiliation status	3.18	0.853	0.182
Cash flow after tax	3.86	0.889	0.190

**Table 3**

Statistical analysis of data set (bank 2)

Attribute	Statistical Methods		
	Mean	Standard deviation	Variance
Age	4.09	0.868	0.185
Gender	3.86	0.941	0.201
Education	3.55	0.739	0.157
Annual Income	3.41	0.590	0.126
Marital status	3.55	1.011	0.215
Average of account	4.09	0.868	0.185
Occupation	3.86	0.941	0.201
Marriage status	4.27	0.631	0.135
Affiliation status	3.55	0.739	0.157
Cash flow after tax	3.41	0.590	0.126

**Table 4**

Statistical analysis of data set (bank 3)

Attribute	Statistical Methods		
	Mean	Standard deviation	Variance
Age	4.09	0.868	0.185
Gender	3.86	0.941	0.201
Education	4.27	0.631	0.135
Annual Income	3.55	0.739	0.157
Marital status	3.41	0.590	0.126
Average of account	3.55	1.011	0.215
Occupation	4.09	0.868	0.185
Marriage status	3.86	0.941	0.201
Affiliation status	4.27	0.631	0.135
Cash flow after tax	3.55	0.739	0.157

**Table 5**  
Statistical analysis of data set (bank 4)

Attribute	Statistical Methods		
	Mean	Standard deviation	Variance
Age	2.55	0.800	0.171
Gender	4.05	0.722	0.154
Education	4.05	0.950	0.203
Annual Income	3.86	0.889	0.190
Marital status	4.23	0.685	0.146
Average of account	4.45	0.671	0.143
Occupation	3.64	0.790	0.168
Marriage status	2.55	0.800	0.171
Affiliation status	4.05	0.722	0.154
Cash flow after tax	4.05	0.950	0.203

**Table 6**  
Statistical analysis of data set (bank 5)

Attribute	Statistical Methods		
	Mean	Standard deviation	Variance
Age	4.45	0.671	0.143
Gender	3.64	0.790	0.168
Education	2.55	0.800	0.171
Annual Income	4.05	0.722	0.154
Marital status	4.05	0.950	0.203
Average of account	3.86	0.889	0.190
Occupation	4.23	0.685	0.146
Marriage status	4.45	0.671	0.143
Affiliation status	3.64	0.790	0.168
Cash flow after tax	2.55	0.800	0.171

## 6. Result

In this section, we analyze the output of five clustering algorithms (four popular clustering and also FRC algorithm). We present a set of experiments of FRC with MATLAB on a Pentium (R) CPU 2.50 GHZ with 512 MB RAM. In order to prove the clustering algorithms, five data sets are run with K-mean, C-mean, Fuzzy C-mean, Kernel K-mean and FRC algorithm, and the results are evaluated and compared respectively in terms of the objective function of density-based evaluation algorithm. The initialization of the parameters used in the FRC algorithm is summarized in Table 7.

**Table 7**  
The initialization of the parameters used in the FRC algorithms

Parameters	Value
$\alpha$ cut	0.7 and 0.8
$W_p$	0.1
$W_q$	0.3
$W_v$	0.6

Regarding the above mentioned evaluation of the clustering quality (Wu et al., 2009; Zhao & Karypis, 2014; Aliquliyev, 2009), each clustering algorithm has a high rank among other algorithms based on critical factors, which prove the algorithm better. Based on the computations of the clustering quality, the FRC had the best rank according to density-based algorithm between four of survey clustering algorithms and is better than other algorithms.

**Table 8**  
Segmentation Result

Data sets	Clusters
Bank 1	3
Bank 2	5
Bank 3	4
Bank 4	4
Bank 5	3

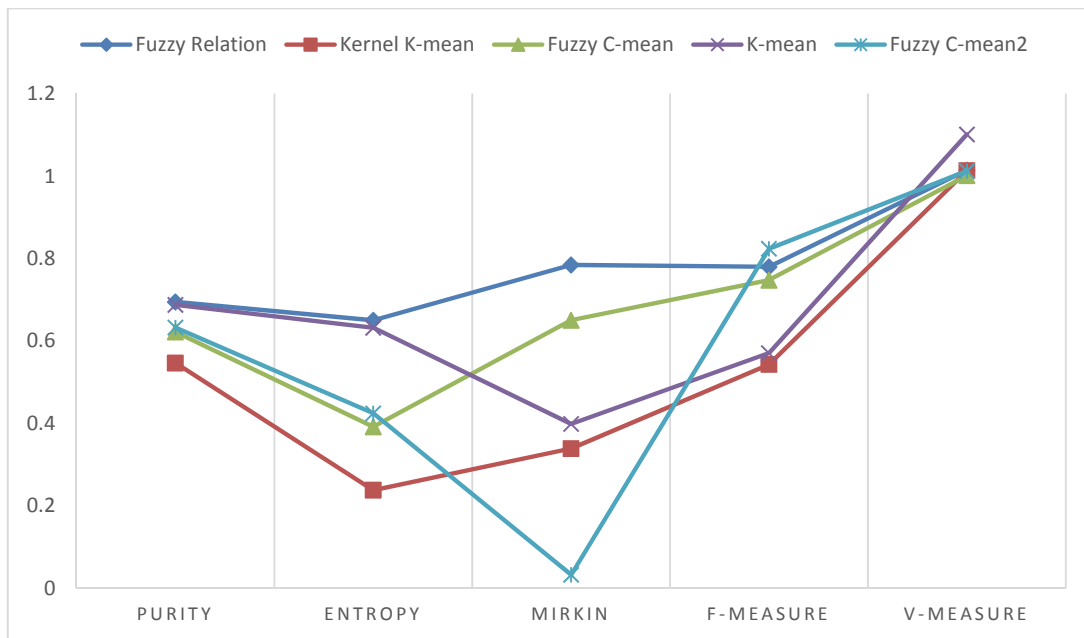
Finally, Table 8 shows the clusters of each dataset and we show our approach of clustering quality in Table 9, Table 10 and Fig. 2. Table 8 shows each data sets of bank segments in some clusters. For example, the data set of the first bank has three clusters.

**Table 9**

Average values of the validity indices for Clustering Algorithms

Algorithms	Fuzzy C-mean	K-mean	Fuzzy C-mean	Kernel K-mean	Fuzzy Relation
Purity	0.6324	0.6873	0.6218	0.5463	0.6945
Entropy	0.4244	0.6319	0.3917	0.2381	0.6501
Mirkin	0.0327	0.3986	0.6502	0.3389	0.7843
F-measure	0.8234	0.5705	0.7476	0.5428	0.7798
V-measure	1.012	1.101	1.001	1.013	1.014

Table 9 presents the five indices: Purity, Entropy, Mirkin, F-measure and V-measure validity for five clustering algorithms. From this table we can also see that the Kernel K-mean algorithm showed the worst results fewer than four indices (out of five).



**Fig. 2.** Average values of the validity indices for Clustering Algorithms

In Fig. 2, we describe the resultant rank for five clustering algorithms based on average values of the validity indices between five of survey clustering algorithms. This Figure shows a graphical comparison of the clustering methods based on five validity indices. We can find out from this figure the FCR is better than survey algorithms, because it has maximum accuracy. Regarding this, we can see FCR algorithm has high V-measure and F-measure among other algorithms. Also when V-measure and F-measure are high, the output of the model depicts accuracy.

**Table 10**

Resultant rank of clustering algorithms

Algorithms	Resultant rank
FRC	7.9745
K-mean	6.9138
Fuzzy K-mean	6.8591
Fuzzy C-mean	4.5714
Kernel K-mean	2.5001

Table 10 shows accuracy and high performance of FRC compared to other clustering methods, thus from this table it can be seen that the distance rating is very high compared to other clustering methods.

## 7. Conclusions

In this paper, we surveyed five clustering algorithms. The comparison was conducted on the banks standard dataset with widely varying numbers of clusters of Fars Province, Shiraz, Iran. The quality of a clustering result was evaluated using three validating clustering approaches: internal criteria, external criteria and relative criteria.

Regarding validating clustering approaches we found the popular clustering algorithms can't dive both crisp and fuzzy quantity variables. Based on the weak point of popular clustering algorithms we define a new clustering algorithm called FRC. In FRC, we have defined three relation matrices for binary, numeral quantities and fuzzy attributes. We proposed a FRC clustering algorithm according to object's features by fuzzy relation clustering principle. This algorithm can use different features with crisp or fuzzy quantities. These features are categorized into three variable sets, consisting of binary, quantitative and linguistic variables.

In the final analysis, the best clustering algorithm has been determined by calculating validating clustering. By calculating validating for each algorithm, considering effective feature, we realized that each of these algorithms can present the suitable clustering in these algorithms, and there are surveys which make definite and fuzzy values possible simultaneous for bank customers.

## References

- Akman, G. (2015). Evaluating suppliers to include green supplier development programs via fuzzy c-means and VIKOR methods. *Computers & Industrial Engineering*, 86, 69-82.
- Aliguliyev, R. M. (2009). Performance evaluation of density-based clustering methods. *Information Sciences*, 179(20), 3583-3602.
- Chehrehgani, M. H., Abolhassani, H., & Chehrehgani, M. H. (2009). Density link-based methods for clustering web pages. *Decision Support Systems*, 47(4), 374-382.
- Clir, G. J., & Yuan, B. (1995). Fuzzy sets and fuzzy logic. Theory and application. Prentice Hall PTR.
- Eberle, D. G., Daudi, E. X., Muiuane, E. A., Nyabeze, P., & Pontavida, A. M. (2012). Crisp clustering of airborne geophysical data from the Alto Ligonha pegmatite field, northeastern Mozambique, to predict zones of increased rare earth element potential. *Journal of African Earth Sciences*, 62(1), 26-34.
- Feng, L., Qiu, M. H., Wang, Y. X., Xiang, Q. L., Yang, Y. F., & Liu, K. (2010). A fast divisive clustering algorithm using an improved discrete particle swarm optimizer. *Pattern Recognition Letters*, 31(11), 1216-1225.
- Filippone, M., Camastra, F., Masulli, F., & Rovetta, S. (2008). A survey of kernel and spectral methods for clustering. *Pattern recognition*, 41(1), 176-190.
- Höppner, F. (1999). *Fuzzy cluster analysis: methods for classification, data analysis and image recognition*. John Wiley & Sons.
- Jain, A. K. (2010). Data clustering: 50 years beyond K-means. *Pattern recognition letters*, 31(8), 651-666.
- Jiang, H., Yi, S., Li, J., Yang, F., & Hu, X. (2010). Ant clustering algorithm with K-harmonic means clustering. *Expert Systems with Applications*, 37(12), 8679-8684.
- Jiménez-García, R., Esteban-Hernández, J., Hernández-Barrera, V., Jimenez-Trujillo, I., López-de-Andrés, A., & Garrido, P. C. (2011). Clustering of unhealthy lifestyle behaviors is associated with nonadherence to clinical preventive recommendations among adults with diabetes. *Journal of Diabetes and its Complications*, 25(2), 107-113.

- Kannappan, A., Tamilarasi, A., & Papageorgiou, E. I. (2011). Analyzing the performance of fuzzy cognitive maps with non-linear hebbian learning algorithm in predicting autistic disorder. *Expert Systems with Applications*, 38(3), 1282-1292.
- Khan, S. S., & Ahmad, A. (2004). Cluster center initialization algorithm for K-means clustering. *Pattern recognition letters*, 25(11), 1293-1302.
- Lee, J. W., Yeung, D. S., & Tsang, E. C. (2005). Hierarchical clustering based on ordinal consistency. *Pattern recognition*, 38(11), 1913-1925.
- Liang, G. S., Chou, T. Y., & Han, T. C. (2005). Cluster analysis based on fuzzy equivalence relation. *European Journal of Operational Research*, 166(1), 160-171.
- Núñez, A., De Schutter, B., Sáez, D., & Škrjanc, I. (2014). Hybrid-fuzzy modeling and identification. *Applied Soft Computing*, 17, 67-78.
- Peters, G., Crespo, F., Lingras, P., & Weber, R. (2013). Soft clustering—fuzzy and rough approaches and their extensions and derivatives. *International Journal of Approximate Reasoning*, 54(2), 307-322.
- Pedrycz, W., & Rai, P. (2008). Collaborative clustering with the use of Fuzzy C-Means and its quantification. *Fuzzy Sets and Systems*, 159(18), 2399-2427.
- Ravi, V., & Zimmermann, H. J. (2000). Fuzzy rule based classification with FeatureSelector and modified threshold accepting. *European Journal of Operational Research*, 123(1), 16-28.
- Rose, K. (1998). Deterministic annealing for clustering, compression, classification, regression, and related optimization problems. *Proceedings of the IEEE*, 86(11), 2210-2239.
- Schölkopf, B., Smola, A., & Müller, K. R. (1998). Nonlinear component analysis as a kernel eigenvalue problem. *Neural computation*, 10(5), 1299-1319.
- Wu, J., Chen, J., Xiong, H., & Xie, M. (2009). External validation measures for K-means clustering: A data distribution perspective. *Expert Systems with Applications*, 36(3), 6050-6061.
- Zhao, Y., & Karypis, G. (2004). Empirical and theoretical comparisons of selected criterion functions for document clustering. *Machine Learning*, 55(3), 311-331.
- Zimmermann, H. J. (1996). Fuzzy Control. In *Fuzzy Set Theory—and Its Applications* (pp. 203-240). Springer Netherlands.