

Fuzzy Genetic Data Mining for Customer Buying Patterns using K-Means Clustering

Shaikh Nikhat Fatma
Department Of Computer ,
Mumbai University ,

Pillai's Institute Of Information Technology, New
Panvel

Jagdish W Bakal
Phd,Principal ,
Mumbai University ,

Shivajirao S Jondhale College of Engineering,
Dombivli(E)

ABSTRACT

Data mining is the process of extracting desirable knowledge or interesting patterns from existing databases for specific purposes. Most conventional data-mining algorithms identify the relationships among transactions using binary values. Transactions with quantitative values are however commonly seen in real-world applications. The fuzzy concepts are used to represent item importance, item quantities, minimum supports and minimum confidences. Each attribute uses only the linguistic term with the maximum cardinality in the mining process. The number of items is thus the same as that of the original attributes, making the processing time reduced. A fuzzy-genetic data-mining algorithm for extracting both association rules and membership functions from quantitative transactions is shown in this paper. It used a combination of large 1-itemsets and membership-function suitability to evaluate the fitness values of chromosomes. The calculation for large 1-itemsets could take a lot of time, especially when the database to be scanned could not totally fit into main memory. In this system, an enhanced approach, called the cluster-based fuzzy-genetic mining algorithm. It divides the chromosomes in a population into clusters by the k-means clustering approach and evaluates each individual according to both cluster and their own information.

Keywords

k-means Clustering, data mining, fuzzy set, genetic algorithm, Fuzzy Association Rules, Quantitative transactions

1. INTRODUCTION

Data Mining is commonly used in attempts to induce association rules from transaction data. An association rule is an expression $X \rightarrow Y$, where X is a set of items and Y is a single item. It means in the set of transactions, if all the items in X exist in a transaction, then Y is also in the transaction with a high probability. Transaction data in real-world applications, however, usually consist of quantitative values.[10].

The goal of data mining is to discover important associations among items such that the presence of some items in a transaction will imply the presence of some other items. To achieve this purpose, proposed several mining algorithms based on the concept of large item sets to find association rules in transaction data mining process into two phases. In the first phase, candidate item sets were generated and counted by scanning the transaction data. If the number of an item set appearing in the transactions was larger than a pre-defined threshold value (called minimum support), the item set was considered a large item set. Item sets containing only one item were processed first. Large item sets containing only single items were then combined to form candidate item sets containing two items. This process was repeated until all large

item sets had been found. In the second phase, association rules were induced from the large item sets found in the first phase.

All possible association combinations for each large item set were formed, and those with calculated confidence values larger than a predefined threshold (called minimum confidence) were output as association rules. In addition to proposing methods for mining association rules from transactions of binary values, from those with quantitative attributes. Their proposed method first determined the number of partitions for each quantitative attribute, and then mapped all possible values of each attribute into a set of consecutive integers. It then found large item sets whose support values were greater than the user-specified minimum-support levels. These large item sets were then processed to generate association rules, and rules of interest to users were output.

Basically, the fuzzy mining algorithms first used membership functions to transform each quantitative value into a fuzzy set in linguistic terms. The algorithm then calculated the scalar cardinality of each linguistic term on all the transaction data. The mining process based on fuzzy counts was then performed to find fuzzy association rules.

Data mining is the process of extracting desirable knowledge or interesting patterns from existing databases for specific purposes. Most conventional data-mining algorithms identify the relationships among transactions using binary values. Transactions with quantitative values are however commonly seen in real-world applications. Therefore a fuzzy mining algorithm by which each attribute used only the linguistic term with the maximum cardinality in the mining process. The number of items was thus the same as that of the original attributes, making the processing time reduced. The fuzzy association rules derived in this way are not complete.

In the remaining paper, in section II we take over view of cluster-based fuzzy-GA mining Frameworks. In section III we take the review of how cluster-based fuzzy-GA mining is actually done and how it actually works. In section IV we have the algorithm for cluster-based fuzzy-GA mining. In section V we have an example and in section VI we make a conclusion followed by the references.

2. PROPOSED CLUSTER-BASED FUZZY- GA MINING FRAMEWORK

- The proposed framework will maintain a population of sets of membership functions, and use the genetic algorithm to automatically derive the resulting one.
- It will first transform each set of membership functions into a fixed-length string.
- Each chromosome will represent a set of membership functions used in fuzzy mining. Then, it will use the K-means clustering approach to gather similar chromosomes into groups.

- All the chromosomes in a cluster will use the number of large 1-itemsets derived from the representative chromosome in the cluster and their own suitability of membership functions to calculate their fitness values. Since the number for scanning a database will decrease, the evaluation cost can thus be reduced.
- The evaluation results can be utilized to choose appropriate chromosomes for mating in the next generation.
- The offspring membership function sets will then undergo recursive —evolution— until a good set of membership functions has been obtained.
- Finally, the derived membership functions will be used to mine fuzzy association rules.

For coding non-isosceles triangles and trapezes, three and four points are needed instead of two for isosceles triangles. Besides, the number of membership function for each item can be different. In this application, we assume each item has three membership functions to illustrating the proposed method. The number of membership functions may have impact of the results.[1]

3. CLUSTER-BASED FUZZY-GENETIC MINING APPROACH

A) Chromosome Representation:

Each set of membership functions is encoded. All pairs of (c.w)'s for a certain item are concatenated to represent its membership functions. Since and are both numeric values, a chromosome is thus encoded as a fixed-length real-number string rather than a bit string. Other types of membership functions (e.g. non-isosceles trapezes) can also be adopted in this method.[1][2]

B) Initial Population:

A genetic algorithm requires a population of feasible solutions to be initialized and updated during the evolution process. As mentioned above, each individual within the population is a set of isosceles-triangular membership functions. Each membership function corresponds to a linguistic term in a certain item. The initial set of chromosomes is randomly generated with some constraints or forming feasible membership function.[1][2]

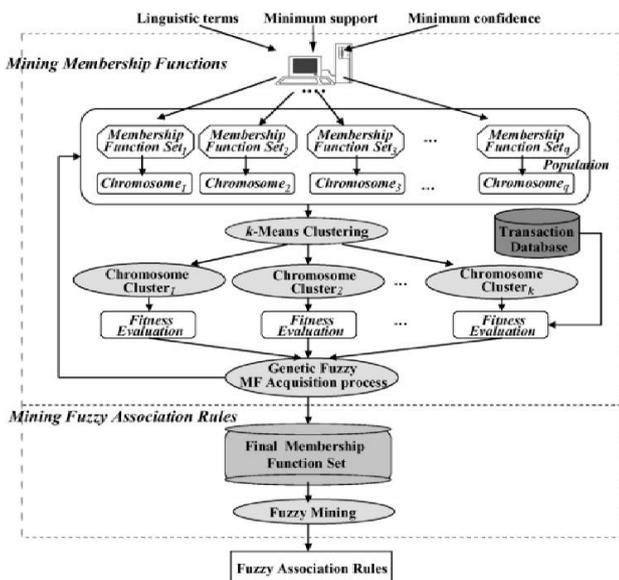


Fig 1: Cluster Based Fuzzy Genetic Mining Frame work

C) Fitness and Selection:

In order to develop a good set of membership functions from an initial population, the genetic algorithm selects *parent* membership function sets with its probability values for mating. An evaluation function is then used to qualify the derived membership function sets. It is shown as follows:

$$f(C_q) = \frac{|L_{1q}|}{\text{Suitability}(C_q)}$$

Where $|L_{1q}|$ is the number of large 1-itemsets obtained by using the set of membership functions in chromosome C_q and suitability of (C_q) represents the shape suitability of (C_q) . Suitability of (C_q) is defined as:

$$\sum_{j=1}^m [\text{overlap_factor}(C_{qj}) + \text{coverage_factor}(C_{qj})]$$

Where m is the number of item. $\text{Overlap_factor}(C_{qj})$ represents the overlap factor of the membership functions for an item I_j in the chromosome C_q and is defined as

$$\text{overlap_factor}(C_{qj}) = \sum_{k \neq j} \left[\max \left(\frac{\text{overlap}(R_{jk}, R_{ji})}{\min(w_{jk}, w_{ji})}, 1 \right) - 1 \right]$$

Where $(R_{j1}, R_{j2}, \dots, R_{ji})$ is the coverage range of the membership functions I_j , i is the number of membership functions for I_j and $\max(I_j)$ is the maximum quantity of in the transactions.[1][2]

D) Clustering Chromosomes:

The coverage factors and overlap factors of all the chromosomes are used to form appropriate clusters. The k -means clustering approach is adopted here to cluster chromosomes. Since the chromosomes with similar coverage factors and overlap factors will form a cluster, they will have nearly the same shape of membership functions and induce about the same number of large 1-itemsets. For each cluster, the chromosome which is the nearest to the cluster center is thus chosen to derive its number of large 1-itemsets. All chromosomes in the same cluster then use the number of large 1-itemsets derived from the representative chromosome as their own. Finally, each chromosome is evaluated by this number of large 1-itemsets divided by its own suitability value.

E) Genetic Operators

Two genetic operators, the *max-min arithmetical (MMA) crossover* proposed in and the *one-point mutation*, are used in the genetic-fuzzy mining framework. Assume there are two parent chromosomes.

$$C_u^{rt} = (c_1, \dots, c_h, \dots, c_z) \text{ and } C_w^{rt} = (c_1, \dots, c_h, \dots, c_z)$$

The max-min-arithmetical (MMA) crossover operator will generate the following four candidate chromosomes from them:

- 1) $C_1^{t+1} = (c_{11}^{t+1}, \dots, c_{1h}^{t+1}, \dots, c_{1Z}^{t+1})$, where $c_{1h}^{t+1} = dc_h + (1-d)c'_h$;
- 2) $C_2^{t+1} = (c_{21}^{t+1}, \dots, c_{2h}^{t+1}, \dots, c_{2Z}^{t+1})$, where $c_{2h}^{t+1} = dc'_h + (1-d)c_h$;
- 3) $C_3^{t+1} = (c_{31}^{t+1}, \dots, c_{3h}^{t+1}, \dots, c_{3Z}^{t+1})$, where $c_{3h}^{t+1} = \min\{c_h, c'_h\}$;
- 4) $C_4^{t+1} = (c_{41}^{t+1}, \dots, c_{4h}^{t+1}, \dots, c_{4Z}^{t+1})$, where $c_{4h}^{t+1} = \max\{c_h, c'_h\}$.

Where the parameter d_i is either a constant or a variable whose value depends on the age of the population.

4. THE CLUSTER-BASED FUZZY-GENETIC MINING ALGORITHM

Notation used in this paper are stated as follows.

n : the total number of transaction data;

m : the total number of attributes;

A_j : the j^{th} attribute, $1 \leq j \leq m$;

$|A_j|$: the number of fuzzy regions for A_j ;

R_{jk} : the k^{th} fuzzy region of A_j , $1 \leq k \leq |A_j|$;

$D^{(i)}$: the i^{th} transaction datum, $1 \leq i \leq n$;

$V_j^{(i)}$: the quantitative value Of A_j for $D^{(i)}$

$F_{(ij)}^{(i)}$: the fuzzy set converted from $v_j^{(i)}$

$f_{jk}^{(i)}$: the membership value of v in Region R_{jk} ;

count_{jk} : the summation of f_{jk} for $i=1$ to n ;

α : the predefined minimum support level;

λ : the predefined minimum confidence value;

C_r : the set of candidate itemsets with r attributes (items);

L_{tr} : the set of large itemsets with r attributes (items).[2]

INPUT:

A body of n quantitative transactions, a set of m items, each with a number of linguistic terms, a parameter k for k -means clustering, a population size P , a crossover rate P_c , a mutation rate P_m , a support threshold α , and a confidence threshold λ .

OUTPUT:

A set of fuzzy association rules with its associated set of membership functions. Set of membership functions.

Step 1:

Randomly generate a population P of individuals; each individual is a set of membership functions for all the m items.

Step 2:

Encode each set of membership functions into a string representation according to the schema stated in section III.

Step 3:

Calculate the coverage_factor and the overlap_factor of each chromosome using the formulas defined in Section.III C

Step 4:

Divide the chromosomes into k clusters by the k -means clustering approach based on the two attributes (coverage_factors and overlap_factors); Find out the representative chromosome in each cluster, which is the nearest to the center.

Step 5:

Calculate the number of large 1-itemsets for each

representative chromosome by the following Sub steps.

Step 5.1:

For each transaction datum D_i $i=1$ to n and for each item I_j and $j=1$ to m transfer the quantitative value $v_j^{(i)}$ into a fuzzy set $F_j^{(i)}$ represented as

$$\frac{f_{j1}^{(i)}}{R_{j1}} + \frac{f_{j2}^{(i)}}{R_{j2}} + \dots + \frac{f_{jl}^{(i)}}{R_{jl}} \quad (1)$$

Using the corresponding membership functions represented by the representative chromosome, where $v_j^{(i)}$ is the quantitative value of I_j in D_i , R_{jk} is the k^{th} fuzzy region (term) of item I_j , $f_{jk}^{(i)}$ is $v_j^{(i)}$'s fuzzy membership value in region R_{jk} , and $l(=|I_j|)$ is the number of linguistic terms for I_j .

Step 5.2:

For each item region R_{jk} , calculate its count from the transactions as follows:

$$\text{count}_{jk} = \sum_{i=1}^n f_{jk}^{(i)} \quad (2)$$

STEP 5.3:

For each R_{jk} , $1 \leq j \leq m$ and $1 \leq k \leq |I_j|$, check whether its count count_{jk} is larger than or equal to the minimum support threshold α . If R_{jk} satisfies the above condition, put it in the set of large 1-itemsets L_1 that is

$$L_1 = \{R_{jk} \mid \text{count}_{jk} \geq \alpha, 1 \leq j \leq m \text{ and } 1 \leq k \leq |I_j|\} \quad (3)$$

STEP 5.4:

Set $|L_1|$ as the number of large 1-itemsets for the representative chromosome.

STEP 6:

Calculate the fitness value of each chromosome using the number of large 1-itemsets of its representative chromosome and the suitability value of its own according to the formula defined in Section III -C.

Step 7:

Execute the crossover operation on the population.

Step 8:

Execute the mutation operation on the population.

Step 9:

Use the Roulette-wheel selection operation to choose appropriate individuals for the next generation.

Step 10:

If the termination criterion is not satisfied, go to Step 3; otherwise, do the next step.

Step 11:

Get the set of membership functions with the highest fitness value.

Step 12: Mine fuzzy rules using the set of membership functions.[1][2]

5. EXAMPLE

Assume there are four items in a transaction database: milk, bread, cookies and beverage. Assume the membership functions for each item are given as shown in Fig. According to the proposed encoding scheme mentioned above, the chromosome for representing the membership functions in Fig. 2 is encoded as shown in Fig.3 Since the item milk has three possible linguistic terms Low, Middle and High, the membership functions for milk are thus encoded as (5, 5, 10, 5, 15, 5) according to Fig. 2.

Bread also has three possible linguistic terms, Low, Middle and High and its associated membership functions are thus encoded as (6, 6, 12, 6, 18, 6). Similarly, the membership functions for cookies and beverage are, respectively, encoded as (3, 3, 6, 3, 9, 3) and (4, 4, 8, 4, 12, 4). Other types of membership functions (e.g. non isosceles trapezes) can also be adopted in our method. For coding non-isosceles triangles and trapezes, three and four points are needed instead of two for isosceles triangles. Each chromosome consists of a set of membership functions for all the items. This representation allows genetic operators to search for appropriate solutions.[2] Assume each item has three fuzzy regions: Low, Middle, and High. Thus, three fuzzy membership functions must be derived for each item.[1]

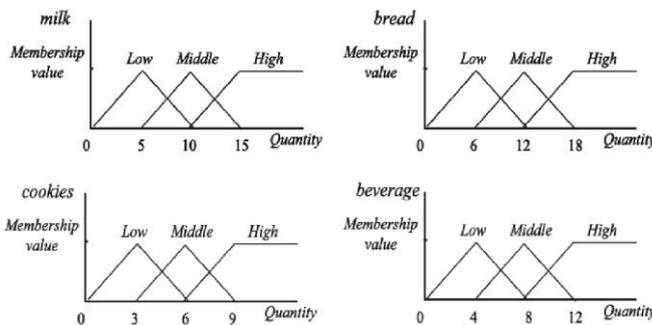


Fig 2: Example of membership function for four items

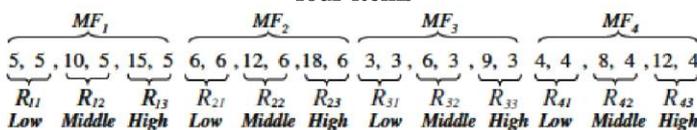


Fig.3: The chromosome representation for the membership functions in Fig.2

STEP 1: Individuals are randomly generated as the initial population. In this example, is set at 10. Each individual is a set of membership functions for all the four items including milk, bread, cookies, and beverage.

Table 1 Six transactions in this example

TID	Items
T1	(milk, 5); (bread, 10); (cookies, 7); (beverage, 7)
T2	(milk, 7); (bread, 14); (cookies, 12)
T3	(bread, 15); (cookies, 12)
T4	(milk, 2); (bread, 5); (cookies, 5)
T5	(bread, 9)
T6	(milk, 13); (beverage, 12)

STEP 2: Each set of membership functions is encoded into a chromosome. Assume the ten individuals are generated as follows:

- C₁: 5, 5, 10, 5, 15, 5, 6, 12, 6, 18, 6, 3, 3, 6, 3, 9, 3, 4, 4, 8, 4, 12, 4;
- C₂: 5, 5, 10, 5, 15, 5, 4, 6, 10, 6, 16, 6, 4, 3, 7, 3, 10, 3, 4, 4, 8, 4, 12, 4;
- C₃: 4, 3, 7, 3, 10, 3, 6, 6, 12, 6, 18, 6, 6, 5, 11, 5, 16, 5, 6, 4, 10, 4, 14, 4;
- C₄: 5, 2, 7, 2, 9, 2, 5, 4, 9, 4, 13, 4, 6, 5, 11, 5, 16, 5, 6, 4, 10, 4, 14, 4;
- C₅: 4, 3, 7, 3, 10, 3, 6, 6, 12, 6, 18, 6, 5, 3, 8, 3, 11, 3, 3, 4, 7, 4, 11, 4;
- C₆: 6, 3, 9, 3, 12, 3, 5, 5, 10, 5, 15, 5, 4, 4, 8, 4, 12, 4, 6, 4, 10, 4, 14, 4;
- C₇: 3, 3, 6, 3, 9, 3, 6, 2, 8, 2, 10, 2, 6, 5, 11, 5, 16, 5, 4, 4, 8, 4, 12, 4;
- C₈: 4, 3, 7, 3, 10, 3, 6, 6, 12, 6, 18, 6, 5, 6, 11, 6, 17, 6, 6, 4, 10, 4, 14, 4;
- C₉: 3, 3, 6, 3, 9, 3, 6, 3, 9, 3, 12, 3, 6, 5, 11, 5, 16, 5, 6, 2, 8, 2, 10, 2;
- C₁₀: 4, 3, 7, 3, 10, 3, 6, 2, 8, 2, 10, 2, 6, 5, 11, 5, 16, 5, 6, 3, 9, 3, 12, 3.

Step 3: The Coverage factor and the Overlap factor of each chromosome are calculated by using the formulas defined as:

Overlap Factor

$$\text{overlap_factor}(C_{qj}) = \sum_{k \neq i} [\max(\frac{\text{overlap}(R_{jk}, R_{ji})}{\min(w_{jk}, w_{ji})}, 1) - 1]$$

Coverage factor:

$$\text{Coverage_factor}(C_{qj}) = \frac{1}{\frac{\text{Range}(R_{j1}, \dots, R_{jl})}{\max(I_j)}}$$

where the “suitability” is represented as the pair(Coverage factor and the Overlap factor) is defined as:

$$\sum_{j=1}^m [\text{overlap_factor}(C_{qj}) + \text{coverage_factor}(C_{qj})]$$

STEP 4: The k-means clustering approach is executed to divide the ten chromosomes into k clusters. In this example, assume the parameter k is set at 3. Then three clusters found

STEP 5: The number of large 1-itemsets for each representative chromosome is calculated by the following sub steps.

STEP 5.1: The quantitative value of each transaction datum is transformed into a fuzzy set according the membership functions in the representative chromosome. Take the item “bread” in transaction by the membership functions denoted by the representative chromosome as an example. The membership

functions for bread from are represented as (6, 6,12, 6, 18, 6), which are shown in Fig.4

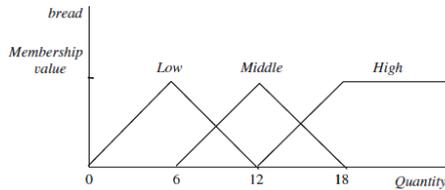


Fig 4: Membership function for item bread

The amount “9” of the item bread is then converted into the fuzzy set $(0.5/bread.low) + (0.5/bread.middle)$ according to the membership functions in Fig.4. The conversion results for all the transactions are shown in Table 2

Table 2: The fuzzy sets transformed from the data in Table 1

TID	Fuzzy set
T1	$(\frac{1.0}{milk.Low} + \frac{0.33}{bread.Low} + \frac{0.67}{bread.Middle} + \frac{0.67}{cookies.Middle} + \frac{0.33}{cookies.High}) (\frac{0.25}{beverage.Low} + \frac{0.75}{beverage.Middle})$
T2	$(\frac{0.6}{milk.Low} + \frac{0.4}{milk.Middle} + \frac{0.67}{bread.Middle} + \frac{0.33}{bread.High} + \frac{1}{cookies.High})$
T3	$(\frac{0.5}{bread.Middle} + \frac{0.5}{bread.High} + \frac{1}{cookies.High})$
T4	$(\frac{0.4}{milk.Low} + \frac{0.83}{bread.Low} + \frac{0.33}{cookies.Low} + \frac{0.67}{cookies.Middle})$
T5	$(\frac{0.5}{bread.Low} + \frac{0.5}{bread.Middle})$
T6	$(\frac{0.4}{milk.Middle} + \frac{0.6}{milk.High} + \frac{1}{beverage.High})$

where the item . term notation is called a fuzzy region.

STEP 5.2: The scalar cardinality of each fuzzy region in the transactions is calculated as the count value. Take the fuzzy region milk.Low as an example. Its scalar cardinality = $(1.0 + 0.6 + 0.0 + 0.4 + 0.0 + 0.0) = 2.0$. The counts for all the fuzzy regions are shown in Table 3

Table 3: The counts of the fuzzy regions

Item	Count	Item	Count
<i>milk.Low</i>	2.00	<i>cookies.Low</i>	0.33
<i>milk.Middle</i>	0.80	<i>cookies.Middle</i>	1.33
<i>milk.High</i>	0.60	<i>cookies.High</i>	2.33
<i>bread.Low</i>	1.67	<i>beverage.Low</i>	0.25
<i>bread.Middle</i>	2.33	<i>beverage.Middle</i>	0.75
<i>bread.High</i>	0.83	<i>beverage.High</i>	1.00

STEP 5.3: The count of any fuzzy region is checked against the predefined minimum support value α . Assume in this example, α is set at 2.0. Since all the count values of *milk.Low*, *bread.Middle* and *cookies.High* are larger than 2.0, these items are put in L_1 .

Table 4: The set of large 1-itemsets (L_1) in this example

Itemset	Count
<i>milk.Low</i>	2.0
<i>bread.Middle</i>	2.33
<i>cookies.High</i>	2.33

STEP 6: The fitness value of each chromosome is calculated. Since there are three large 1-itemsets for the membership functions of C_1 and its suitability is calculated as 4, the fitness value of C_1 is thus $3/4 (= 0.75)$. The fitness values of all the chromosomes are shown in Table 5. Its fitness value is thus according to the formula.

$$f(C_q) = \frac{|L_{1q}|}{\text{Suitability}(C_q)}$$

Results for the other two clusters can be similarly derived.

Table 4: The fitness values of the chromosomes in the initial population

Chromosome	f	Chromosome	f
C_1	0.75	C_6	0.54
C_2	0.53	C_7	0.32
C_3	0.27	C_8	0.54
C_4	0.3	C_9	0.31
C_5	0.58	C_{10}	0.32

STEP 7: The crossover operation is executed on the population. Assume d is set at 0.35 and the crossover rate is set at 0.8. Take the crossover of the two chromosomes, C_1 and C_2 , as an example. According to the MMA crossover operator. The following four candidate offspring chromosomes are generated as follows:

C_1 : 5, 5, 10, 5, 15, 5, 6, 6, 12, 6, 18, 6, 3, 3, 6, 3, 9, 3, 4, 4, 8, 4, 12, 4

C_2 : 5, 5, 10, 5, 15, 5, 4, 6, 10, 6, 16, 6, 4, 3, 7, 3, 10, 3, 4, 4, 8, 4, 12, 4

1) C_1^{t+1} : 5, 5, 10, 5, 15, 5, 4.7, 6, 10.7, 6, 16.7, 6, 3.65, 3, 6.65, 3, 9.65, 3, 4, 4, 8, 4, 12, 4

2) C_2^{t+1} : 5, 5, 10, 5, 15, 5, 5.3, 6, 11.3, 6, 17.3, 6, 3.35, 3, 6.35, 3, 9.35, 3, 4, 4, 8, 4, 12, 4

3) C_3^{t+1} : 5, 5, 10, 5, 15, 5, 4, 6, 10, 6, 16, 6, 3, 3, 6, 3, 9, 3, 4, 4, 8, 4, 12, 4

4) C_4^{t+1} : 5, 5, 10, 5, 15, 5, 6, 6, 12, 6, 18, 6, 4, 3, 7, 3, 10, 3, 4, 4, 8, 4, 12, 4

The fitness value of the above four candidates are then evaluated, with results shown in Table 5.

Table 6: The fitness value of the four candidate offspring

Chromosome	f	Chromosome	f
C_1^{t+1}	0.8	C_3^{t+1}	0.8
C_2^{t+1}	0.8	C_4^{t+1}	0.8

Since all the four chromosomes have the same fitness value, any two of them can be chosen. Here assume C_1^{t+1} and C_3^{t+1} are chosen.

STEP 8: The mutation operation is executed on the population to generate possible offspring. The operation is the same as the traditional one except that rearrangement may

need to be done. Here, after mutation, if the fitness value of new chromosome is better old chromosome, then the new one is instead of the old one and updated into population.

STEP 9 TO STEP 12:The Roulette-wheel selection operation is executed to generate ten chromosomes as the next population. The same procedure is then executed until the termination criterion is satisfied. The best chromosome (with the highest fitness value) is output as the membership functions for deriving fuzzy rules. After the membership functions are derived, the fuzzy mining is then used to mine fuzzy association rules.

6. CONCLUSION

Therefore the proposed cluster-based fuzzy-genetic mining algorithm is used for extracting both fuzzy association rules and membership functions from quantitative transactions. The proposed algorithm can adjust membership functions by genetic algorithms and uses them to fuzzify quantitative transactions. It can also speed up the evaluation process and keep nearly the same quality of solutions by clustering chromosomes. Each chromosome represents a set of membership functions used in fuzzy mining. The proposed algorithm first divides the chromosomes in a population into k clusters by using the K-means clustering approach. All the chromosomes in a cluster then use the number of large 1-itemsets derived from the representative chromosome in the cluster and their own suitability of membership functions to calculate the fitness values. The evaluation cost can thus be greatly reduced due to the time-saving in finding 1-itemsets.

7. REFERENCES

- [1] Chun-Hao Chen, Tzung-Pei Hong, —Cluster-Based Evaluation in Fuzzy-Genetic Data Mining, IEEE transactions on fuzzy systems, Vol. 16, No. 1, February 2008 249, pp. 249-262.
- [2] T. P. Hong, C. H. Chen, Y. L. Wu, and Y. C. Lee, “A GA-based fuzzy mining approach to achieve a trade-off between number of rules and suitability of membership functions,” *Soft Computing*, vol. 10, no. 11, pp. 1091–1101, 2006.
- [3] Hung-Pin Chiu, Yi-Tsung Tang, “A Cluster-Based Mining Approach for Mining Fuzzy Association Rules in Two Databases”, *Electronic Commerce Studies*, Vol. 4, No.1, Spring 2006, Page 57-74.
- [4] Tzung-Pei Hong, Chan-Sheng Kuo, Sheng-Chai Chi, “Trade-Off Between computation time and number of rules for fuzzy mining from quantitative data”, *International Journal of Uncertainty, Fuzziness and Knowledge-Based systems*, Vol. 9, No. 5, 2001, page 587- 604.
- [5] M. Sulaiman Khan, Maybin Mueyba, Frans Coenen, “Fuzzy Weighted Association Rule Mining with Weighted Support and Confidence Framework”, *The University of Liverpool, Department of Computer Science, Liverpool, UK*.
- [6] H. Ishibuchi and T. Yamamoto, “Rule weight specification in fuzzy rule-based classification systems,” *IEEE Trans. on Fuzzy Systems*, Vol. 13, No. 4, pp. 428-435, August 2005.
- [7] Miguel Delgado, Nicolás Marín, Daniel Sánchez, and María-Amparo Vila, “Fuzzy Association Rules: General Model and Applications”, *IEEE transactions on fuzzy systems*, vol. 11, no. 2, April 2003
- [8] Tzung-Pei Hong, Li-Huei Tseng and Been-Chian Chien, “ Learning Fuzzy Rules from Incomplete Quantitative Data by Rough Sets”.
- [9] H. J. Zimmermann, “Fuzzy set theory and its applications”, *Kluwer Academic Publisher*, Boston, 1991.
- [10] Tzung-Pei Hong, Ming-Jer Chiang and Shyue-Liang Wang “Mining from Quantitative Data with Linguistic Minimum Supports and Confidences”, 2002 IEEE Proceedings.
- [11] S. Yue, E. Tsang, D. Yeung, and D. Shi, “Mining fuzzy association rules with weighted items,” in *Proc. IEEE Int. Conf. on Systems, Man and Cybernetics*, 2000, pp. 1906–1911.
- [12] M. Kaya, R. Alhadj , “Genetic algorithm based framework for mining fuzzy association rules”, 2004 Elsevier B.V.