Mining Fuzzy Association Rules in Databases

Chan Man Kuok, Ada Fu, Man Hon Wong

Department of Computer Science and Engineering
The Chinese University of Hong Kong
Shatin, New Territories, Hong Kong

Abstract

Data mining is the discovery of previously unknown, potentially useful and hidden knowledge in databases. In this paper, we concentrate on the discovery of association rules. Many algorithms have been proposed to find association rules in databases with binary attributes. We introduce the fuzzy association rules of the form, 'If X is A then Y is B', to deal with quantitative attributes. X, Y are set of attributes and A, B are fuzzy sets which describe X and Y respectively. Using the fuzzy set concept, the discovered rules are more understandable to human. Moreover, fuzzy sets handle numerical values better than existing methods because fuzzy sets soften the effect of sharp boundaries.

1 Introduction

During the past years, boolean association rule mining has received considerable attention. Boolean association rule mining tries to find consumer behavior in retail data. The discovered rule can tell, for example, people buy butter and milk will also buy bread. Such rules can be used in customizing marketing program, advertisement and sales promotion. However, binary association rule mining restricts the application area to binary one.

Recently, people are interested in quantitative attributes. In [12], mining quantitative association rules has been proposed. The algorithm finds the association rules by partitioning the attribute domain and combining adjacent partitions, then transforms the problem into binary one. Although this method can solve problems introduced by infinite domain, it causes the sharp boundary problem. We either ignore or overemphasize the elements near the boundaries in the mining process.

In this paper, we propose an algorithm for mining fuzzy association rule of the form, If X is A then Y is B. X, Y are attributes and A, B are fuzzy sets which characterize X and Y respectively. The Fuzzy set concept is better than the partition method because fuzzy sets provide a smooth transition between member and non-member of a set. Because of the smooth transition, there are fewer boundary elements being excluded. Moreover, the fuzzy association rule is more understandable because of linguistic terms associated with fuzzy sets.

This paper is organized as follows. In the following section, we will describe different ways to handle quantitative attributes. We will give definition of fuzzy association rules and interest measures of itemsets and rules in section 3. In section 4, the experimental results will be given. We will give a brief conclusion in section 5.

2 Quantitative Attributes

In [1, 2, 10, 5, 11], algorithms to find binary association rules in large databases have been proposed. However, a database may also contain quantitative attributes, e.g. integer, categorical, numerical attributes. Since we cannot directly apply the binary algorithms, we either have to transform the quantitative problem into binary one or to find new algorithms.

In figure 1, the discrete interval method [12] divides the attribute domain into discrete intervals. Each element will contribute weight to its own interval. We can use the weights to estimate the importance of an interval. However, we may miss some interesting intervals because of excluding some potential elements near the sharp boundaries.

The effect of sharp boundary is shown in figure 1. The first graph is the data distribution of *age*. The attribute domain has been partitioned into 5 intervals. Suppose the intervals, 10 to 20, 20 to 30 and 30 to 40, only have 20% support and the minimum support is 25%. In this case, all these intervals will not have enough support. However, the interval, 20

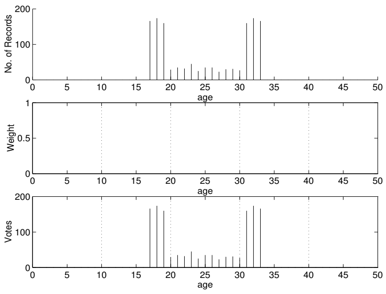


Figure 1: Discrete Intervals.

to 30, should be interesting if we consider the values near both sides.

Another attribute partitioning method [12] is to divide attribute domain into overlapped regions and is shown in figure 2. In the second graph, we can see that the boundaries of intervals are overlapped with each other. As a result, the elements located near the boundary will contribute to more than one interval such that some intervals may become interesting in this case. It is, however, not reasonable for an element near the boundaries to contribute the same as those located within an interval. This will surely overemphasize the importance of an interval.

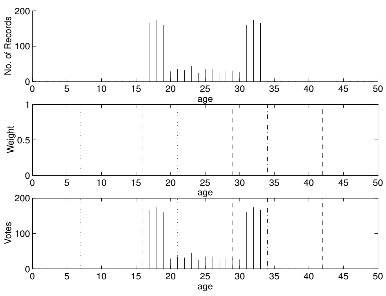


Figure 2: Overlapped Intervals.

The above attribute partitioning methods are subject to the effect of sharp boundaries because of the classical set theory. In the fuzzy set theory, however, an element can belong to a set with set membership value in [0,1]. This value is assigned by the membership function associated with each fuzzy set. For attribute x and its domain D_x, the mapping of the membership function is $m_{f_x}(x) : D_x \rightarrow [0, 1]$.

Fuzzy set provides a smooth change between the boundary and the effect is shown in figure 3. The second graph shows the curve of a traditional fuzzy set. In the third graph, we can see that the values located outside the interval have been considered. Therefore, the sharp boundary problem has been tackled. Moreover, the contribution of a value has been restricted by the membership function as illustrated in figure 3.

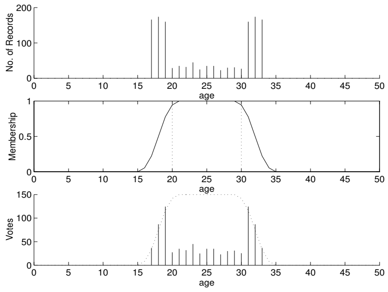


Figure 3: Fuzzy Set.

3 Problem Definition

Mining fuzzy association rule is the discovery of association rules using fuzzy set concepts such that the quantitative attributes can be handled. In this section, we will give the definition of fuzzy association rule first. Then we will discuss the interest measures of itemsets and the rules.

3.1 Fuzzy association rule

Let $T = \{t_1, t_2, ..., t_n\}$ be the database and t_i represents the i^{th} tuple in T. Moreover, we use $I = \{i_1, i_2, ..., i_m\}$ to represent all attributes appeared in T and i_j represents the j^{th} attribute. Since I contains set of items, we call I an *itemset* which appeared in existing papers. Table 1 is a sample database with quantitative attributes.

Retired	*Children*	*Salary*
Yes	2	0
No	3	15000
No	0	10000
No	1	20000
Yes	2	0

Table 1: A Sample Database.

We have $T = \{t_1, t_2, t_3, t_4, t_5\}$ and $I = \{Retired, Children, Salary\}$. We can retrieve the value of attribute i_k in the j^{th} record simply by $t_j[i_k]$. For example, if we want to know the value of *Salary* of the forth record, we can use $t_4[Salary]$ and get the value 20000.

Besides, each attribute i_k will associate with several fuzzy sets. We use $F_{i_k} = \{f^1_{i_k}, f^2_{i_k}, ..., f^l_{i_k}\}$ to represent set of fuzzy sets associated with i_k and

$f^j_{i_k}$ represents the j^{th} fuzzy set in F_{i_k}. For example, if the attribute *Salary* has three fuzzy sets: *high*, *medium* and *low*, we will have $F_{Salary} = \{high, medium, low\}$. The fuzzy sets and the corresponding membership functions are provided by domain experts.

Given a database T with attributes I and those fuzzy sets associated with attributes in I, we want to find out some interesting, potentially useful regularities in a guided way. Our proposed fuzzy association rule is in the following form:

$$\text{If } X \text{ is } A \text{ then } Y \text{ is } B.$$

In the above rule, $X = \{x_1, x_2, ..., x_p\}$ and $Y = \{y_1, y_2, ..., y_q\}$ are itemsets. X and Y are subsets of I and they are disjoint which means that they share no common attributes. $A = \{f_{x_1}, f_{x_2}, ..., f_{x_p}\}$ and $B = \{f_{y_1}, f_{y_2}, ..., f_{y_q}\}$ contain the fuzzy sets associated with the corresponding attributes in X and Y. For example, an attribute x_k in X will have a fuzzy set f_{x_k} in A such that $f_{x_k} \in F_{x_k}$ is satisfied.

The first part of the rule 'X is A' is called the antecedent and 'Y is B' is called the consequent of the rule. The semantics of the rule is when 'X is A' is satisfied, we can imply that 'Y is B' is also satisfied. Here, *satisfied* means there are sufficient amount of records which contribute their votes to the *attribute-fuzzy set* pairs and the sum of these votes is greater than a user specified threshold.

If a rule is interesting, it should have enough *significance* and a high *certainty* factor. We use significance and a certainty factor to determine the satisfiability of itemsets and rules.

3.2 Significance factor

To generate fuzzy association rule, we have first to find out all *large k-itemsets* which are itemsets with *significance* higher than a user specified threshold. The significance factor is calculated by first summing all votes of each record with respect to the specified itemset, then dividing it by the total number of records. Each record contributes a vote which falls in [0,1]. Therefore, a significance factor reflects not only number of records supporting the itemset, but also their degree of support. We use the following formula to calculate the significance factor of $\langle X, A\rangle$, i.e. $S_{\langle X,A\rangle}$.

$$
\begin{aligned}
Significance &= \frac{\text{Sum of votes satisfying } \langle X, A\rangle}{\text{Number of records in } T} \\
S_{\langle X,A\rangle} &= \frac{\sum_{t_i \in T} \prod_{x_j \in X} \{\alpha_{a_j}(t_i[x_j])\}}{total(T)}
\end{aligned}
$$

where

$$
\alpha_{a_j}(t_i[x_j]) = \begin{cases} m_{a_j \in A}(t_i[x_j]) & \text{if } m_{a_j} \geq \omega, \\ 0 & \text{otherwise.} \end{cases}
$$

In the above equation, $\langle X, A\rangle$ represents the *itemset-fuzzy set* pair, where X is set of attributes x_j and A is the set of fuzzy sets a_j. A record satisfies $\langle X, A\rangle$ means that the vote of the record is greater than zero. The vote of a record is calculated by the membership grade of each x_j in that record. The membership grade should not be less than the user specified threshold ω such that low membership values will not be considered. We use $t_i[x_j]$ to obtain the value of x_j in the i^{th} records, then transform the value into membership grade by $m_{a_j \in A}(t_i[x_j])$ which is the membership function of x_j. After obtaining all membership grades of each x_j in a record, we use $\prod_{x_j \in X}\{m_{a_j \in A}(t_i[x_j])\}$ to calculate the vote of t_i. After summing up the votes of all records, we divide the value by the total number of records.

In fact, we can use operators other than $\prod$ (*mul*), e.g. *min*, *max*, but *mul* gives the simplest and reasonable results. It takes the membership of all attributes of an itemset into account. Table 2 illustrates why we use *mul*.

			Max	Min	Mul
0.9	0.2	0	0.9	0	0
0.9	0.9	0.2	0.9	0.2	0.162
0.3	0.3	0.2	0.3	0.2	0.018

Table 2: The Effect Of Functions.

$\langle Salary, high\rangle$	$\langle Balance, low\rangle$
0.9	0.2
0.2	0.7
0.5	0.4
0.3	0.7
0.6	0.3

Table 3: Database Containing Membership.

We use an example to illustrate the computation of the significance factor. Let $X = \{Salary, Balance\}$ and $A = \{high, low\}$ and a part of database shown in table 3. The significance of $\langle X, A\rangle$ is as follows.

$$
\begin{aligned}
S_{\langle X,A\rangle} &= (0.18 + 0.14 + 0.2 + 0.21 + 0.18)/5 \\
&= 0.182
\end{aligned}
$$

3.3 Certainty factor

We use the discovered *large itemsets* to generate all possible rules. The criteria for a rule to be interesting is called *certainty* factor. If the union of antecedent and consequent has enough *significance* and the rule has sufficient *certainty*, this rule will be considered as interesting. There are two ways to calculate the *certainty* factor.

Using significance

When we obtain a *large itemset* $\langle Z, C\rangle$, we want to generate fuzzy association rules of the form, 'If X is A then Y is B.', where $X \subset Z$, $Y = Z - X$, $A \subset C$ and $B = C - A$. Having the *large itemset*, we know its significance as well as the fact that all of its subsets will be also *large*. We can calculate the certainty factor as follows.

$$
\begin{aligned}
Certainty &= \frac{\text{Significance of } \langle Z, C\rangle}{\text{Significance of } \langle X, A\rangle} \\
C_{\langle\langle X,A\rangle,\langle Y,B\rangle\rangle} &= \frac{\sum_{t_i \in T} \prod_{z_k \in Z} \{\alpha_{c_k}(t_i[z_k])\}}{\sum_{t_i \in T} \prod_{x_j \in X} \{\alpha_{a_j}(t_i[x_j])\}} \\
&\text{where} \\
\alpha_{c_k}(t_i[z_k]) &= \begin{cases} m_{c_k \in C}(t_i[z_k]) & \text{if } m_{c_k} \geq \omega, \\ 0 & \text{otherwise.} \end{cases} \\
& Z = X \cup Y, C = A \cup B
\end{aligned}
$$

Since the significance factor of an itemset is the measure of the degree of support given by records, we use significance to help us estimate the interestingness of the generated fuzzy association rules. In the above equation, we divide the *significance* of $\langle Z, C\rangle$ by *significance* of $\langle X, A\rangle$. The certainty reflects fraction of votes support $\langle X, A\rangle$ will also support $\langle Z, C\rangle$. We will use the information in table 3 to illustrate the calculation of certainty factor. Given the rule, 'If *Salary* is *high* then *Balance* is *low*.', i.e. $X = \{Salary\}$, $A = \{high\}$, $Y = \{Balance\}$ and $B = \{low\}$, the certainty is as follows.

$$
\begin{aligned}
C_{\langle\langle X,A\rangle,\langle Y,B\rangle\rangle} &= \frac{0.18 + 0.14 + 0.2 + 0.21 + 0.18}{0.9 + 0.2 + 0.5 + 0.3 + 0.6} \\
&= 0.364
\end{aligned}
$$

Using correlation

Another way to calculate the *certainty* factor of a rule is to compute the *correlation* of $\langle X, A\rangle$ and $\langle Y, B\rangle$. In this paper, the correlation, which is different from statistics, is called *XYCorrelation*. The calculation of expectation of the antecedent is similar to statistics except that we have to take the user specified membership ω into account. The vote of record will be zero if the membership grade of $\langle X, A\rangle$ in that record is less than ω. However, the vote of consequent will also be zero if the vote of the antecedent is less than ω. The following equation is used for computing the certainty.

$$
\begin{aligned}
Certainty &= \text{XYCorrelation of } \langle X, A\rangle \text{ and } \langle Y, B\rangle \\
C_{\langle\langle X,A\rangle,\langle Y,B\rangle\rangle} &= \frac{Cov(X,Y)}{\sqrt{Var(X) \times Var(Y)}} \\
&\text{where} \\
Cov(X,Y) &= E[\langle Z, C\rangle] - E[\langle X, A\rangle] \times E'[\langle Y, B\rangle] \\
& Z = X \cup Y, C = A \cup B \\
Var(X) &= E[\langle X, A\rangle^2] - E[\langle X, A\rangle]^2 \\
Var(Y) &= E'[\langle Y, B\rangle^2] - E'[\langle Y, B\rangle]^2 \\
E[\langle X, A\rangle] &= \frac{\sum_{t_i \in T} \prod_{x_j \in X} \{\alpha_{a_j}(t_i[x_j])\}}{total(T)} \\
\alpha_{a_j}(t_i[x_j]) &= \begin{cases} m_{a_j \in A}(t_i[x_j]) & \text{if } m_{a_j} \geq \omega, \\ 0 & \text{otherwise.} \end{cases} \\
E'[\langle Y, B\rangle] &= \frac{\sum_{t_i \in T} \beta[t_i]}{total(T)} \\
\beta[t_i] &= \begin{cases} \prod_{y_k \in Y} \{\alpha_{b_k}(t_i[y_k])\} & \text{if } \gamma \geq \omega, \\ 0 & \text{otherwise.} \end{cases} \\
& \gamma = \prod_{x_j \in X} \{\alpha_{a_j}(t_i[x_j])\}
\end{aligned}
$$

In data mining, an association rule $X \rightarrow Y$ usually means X implies Y and we cannot assume Y also implies X because of the data distribution of X and Y. Therefore, we change the calculation of expectation such that we can accommodate the meaning of fuzzy association rules. In the above equations, we can see that the calculation of $E[\langle X, A\rangle]$ is similar to an ordinary expectation except it has taken the membership threshold ω into account. $E'[\langle Y, B\rangle]$ calculates the expectation of the consequent. If the product of membership grades of the antecedent of a record is less than ω, the vote of the consequent of that record will be zero.

The value of the certainty is ranging from -1 to 1. Only positive value tells that the antecedent and consequent are related. The higher the value is, the more related they are. Therefore, if the rule 'If X is A then Y is B.' holds, the certainty of this rule should be at least greater than zero.

Given the database in table 3, we can calculate the certainty factor of the rule, 'If *Salary* is *high* then *Balance* is *low*.' as follows.

$$
\begin{aligned}
C_{\langle\langle X,A\rangle,\langle Y,B\rangle\rangle} &= \frac{0.182 - 0.23}{\sqrt{0.06 \times 0.0424}} \\
&= -0.96
\end{aligned}
$$

4 Experimental Results

In this section, we will examine the accuracy and performance of discrete interval method and the methods proposed in this paper. We will describe the parameter settings and the results of different methods.

4.1 Experiment One

In this experiment, we use two attributes to illustrate how the fuzzy set concept can solve the problem of sharp boundary. We assume that there are three intervals/fuzzy sets for each attribute.

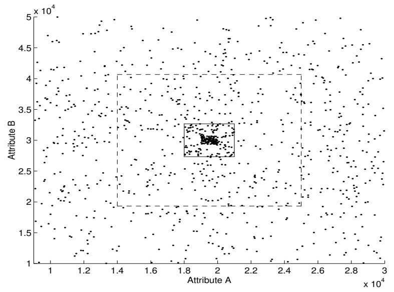


Figure 4: Data of Experiment One.

In figure 4, the horizontal axis represents attribute A and the vertical axis represents attribute B. We generate the database such that records are clustered in the inner box.

	$S = 0.25$		
Methods	C	L	R
Discrete1	15	6	0
Discrete2	3	3	2
Significance	7	3	2
Correlation	7	3	2

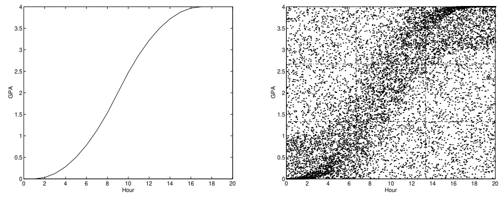
Table 4: Result Of Experiment One.

In table 4, S is the significance factor and C, L, R are numbers of candidate itemsets, large itemsets and rules. The *confidence* and *certainty* have been set to 50%. Moreover, we have set the user specified membership threshold to 0.6. *Discrete1* uses the inner box as the interesting region and *Discrete2* uses the outer box.

We can see that all methods discover similar results except that Discrete1 cannot find rules. Therefore, Discrete2 uses large region in order to find the missing rules. However, the region is so large that the semantics of the rules become meaningless. On the contrary, the fuzzy sets have not overemphasized the sparse elements but still give similar results.

4.2 Experiment Two

We assume there is a relation between the working hour and the GPA of a student. The relation of the two attributes is shown in figure 5(a). The meaning of the relation is that the GPA of a student will be high if he works hard. Otherwise, he will get low GPA. The data are generated according to the relation curve in figure 5(a). In figure 5(b), we can see the data distribution of the two attributes.



(a) The Relation. (b) The Database.

Figure 5: Data of Experiment Two.

The two attributes, *Hour* and *GPA*, have three intervals/fuzzy sets such that the plane of *Hour* and *GPA* is divided into nine regions. In figure 5(b), we can see that at least four areas are heavily shaded which means that several rules should exist in the database. The results in table 5 are quite similar to those of experiment one. The significance factor has been set to 0.2 and 0.25 and the certainty factor is 50%.

	$S = 0.2$			$S = 0.25$		
Methods	C	L	R	C	L	R
Discrete	15	7	2	15	6	0
Significance	15	11	5	15	10	5
Correlation	15	11	10	15	10	8

Table 5: Result Of Experiment Two.

In this experiment, we can see that the method using correlation to calculate certainty factor gives the highest number of expected interesting rules. The discrete interval method again find fewest rules than our methods.

4.3 Experiment Three

In this experiment, we will give the experimental results on the performance of the three methods. There are three attributes in the database. Each attribute has three intervals/fuzzy sets. We have set the user specified parameters such that all three methods will give same number of rules. We have run the programs with database size ranging from 5000 to 100000 records. Figure 6, shows the execution time of the three methods.

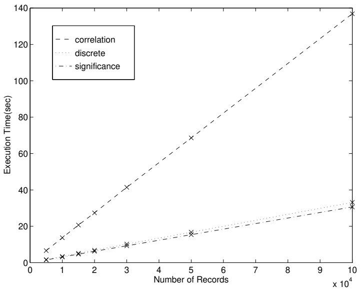


Figure 6: Result of Experiment Three.

In figure 6, the execution of all methods grow linearly as the number of records increased. In previous experiments, the method using correlation will give more rules than others. However, the performance of this method turns out to be the worst because we have to scan the database again when we calculate the certainty factor. The method using significance give comparable performance with respect to the discrete interval method and it finds more relevant rules than the discrete interval method. Therefore, the trade-off between significance and correlation methods is performance and number of rules to be discovered.

5 Conclusion

In this paper, we have proposed a method to handle quantitative attributes. We assign each attribute with several fuzzy sets which characterize the quantitative attribute. Using the fuzzy set concept, we want to find fuzzy association rule. We have defined the significance factor of itemset, the definition and certainty factor of fuzzy association rule. Moreover, we have performed several experiments. In those experiments, we have shown that our algorithm has solved the problem of sharp boundary. We have used two methods to measure the certainty of fuzzy association rules, i.e. significance and correlation. In the experiments, we have found that the method using significance as certainty will give a better performance. On the other hand, the method using correlation as certainty will give more accurate results.

References

[1] R. Agrawal, T. Imielinski, and A. Swami. Mining association rules between sets of items in large databases. In *SIGMOD*, pages 207–216, Washington D.C., May 1993.

[2] R. Agrawal and R. Srikant. Fast algorithms for mining association rules in large databases. In *20th International Conference on Very Large Databases*, Santiago, Chile, Sept. 1994.

[3] H. Bandemer and W. Näther. *Fuzzy data analysis.* Kluwer Academic Publishers, Dordrecht, Netherlands; Boston, 1992.

[4] D. Dubois and H. Prade. *Possibility theory : an approach to computerized processing of uncertainty.* Plenum Press, New York, 1988.

[5] U. M. Fayyad and R. Uthurusamy. Efficient algorithms for discovering association rules. In *AAAI Workshop on KDD, Eds*, pages 181–192, Seattle, Washington, July 1994.

[6] A. Geyer-Schulz. *Fuzzy rule-based expert systems and genetic machine learning.* Physica-Verlag, Heidelberg, 1995.

[7] J. Han and Y. Fu. Discovery of multiple level association rules from large databases. In *21st Int'l Conf. on VLDB*, Zürich, Switzerland, Sept. 1995.

[8] A. Kandel. *Fuzzy expert systems.* CRC Press, Boca Raton, Fla., 1992.

[9] G. J. Klir and T. A. Folger. *Fuzzy sets, uncertainty, and information.* Prentice Hall, Englewood Cliffs, N.J., 1988.

[10] J. S. Park, M.-S. Chen, and P. S. Yu. An effective hash-based algorithm for mining associatin rules. In *SIGMOD*, pages 175–186, San Jose, 1995. ACM.

[11] A. Sarasere, E. Omiecinsky, and S. Navathe. An efficient algorithm for mining association rules in large databases. In *21st Int'l Conf. on VLDB*, Zürich, Switzerland, Sept. 1995.

[12] R. Srikant and R. Agrawal. Mining quantitative association rules in large relational tables. 1995.

[13] L. A. Zadeh and J. Kacprzyk. *Fuzzy Logic for the Management of Uncertainty.* John Wiley & Sons, Inc., 1992.