BUILDING AN ASSOCIATIVE CLASSIFIER BASED ON FUZZY ASSOCIATION RULES

Zuoliang Chen, Guoqing Chen
School of Economics and Management, Tsinghua University
Beijing 100084, China
{chenzl, chengq}@sem.tsinghua.edu.cn
www.sem.tsinghua.edu.cn

Classification based on association rules is considered to be effective and advantageous in many cases. However, there is a so-called "sharp boundary" problem in association rules mining with quantitative attribute domains. This paper aims at proposing an associative classification approach, namely Classification with Fuzzy Association Rules (CFAR), where fuzzy logic is used in partitioning the domains. In doing so, the notions of support and confidence are extended, along with the notion of compact set in dealing with rule redundancy and conflict. Furthermore, the corresponding mining algorithm is introduced and tested on benchmarking datasets. The experimental results revealed that CFAR generated better understandability in terms of fewer rules and smother boundaries than the traditional CBA approach while maintaining satisfactory accuracy.

Keywords: Associative Classification, Fuzzy Association rules, CFAR, Data Mining.

1. Introduction

Classification and association rule mining[23] are two major areas of research and applications nowadays in knowledge discovery. An association rule (AR) is of the form, $X \Rightarrow Y$ where X and Y are sets of data items. The goal of association rule mining is to generate certain associative relationships between data items with the degrees of confidence and support greater than user specified thresholds. The Apriori[23] algorithm is a well-known algorithm in this field. A typical association rule application is market baskets analysis describing, for example, the customers' buying behavior such as "*Fruit $\Rightarrow$ Meat*" meaning that customers who bought fruit also tended to buy meat, which reflects association between occurrences of data items.

Classification is used to find a logical description, namely a classifier, which results from training datasets with predetermined targets, and could group unlabeled datasets. Existing research efforts have proposed a number of approaches and systems [3,6,8,12,14,17,18,21,29,32,33]. A worth-noting type of approaches is classification based on association rules, aimed at building a classifier by discovering a small set of rules to form a so-called associative classifier. That is, an associative classifier is composed of only those association rules of the form $X \Rightarrow C$ where C is a class label treated as a special case of Y in association rules. For instance, given a customer database with each record characterized by such attributes as *income, car-owned* and *travel-plan*, a classification rule discovered may be "90 of high-income customers who own a Jeep are subscribes of Plan B; 3% of all customers have both characteristics."[30]. Apparently, such a rule could be important to a marketing manager who may concentrate on promoting Plan B among high-income top-brand car owners. There exist several ways of building associative classifiers based on association rules mining, such as CBA[3], CMAR[32], CPAR[33] and GARC[8] in light of accuracy and understandability.

Zuoliang Chen, Guoqing Chen

However, when the data of concern are associated with quantitative domains such as income, age, price, etc., which are very common in many real applications, association rule mining usually needs to partition the domains in order to apply the Apriori-type method. Thus, a discovered rule $X \Rightarrow Y$ reflects association between interval values of data items. Examples of such rules are "*Fruit*[1-5kg] $\Rightarrow$ *Meat*[5-20$]", "*Income*[20-50k$] $\Rightarrow$ *Age*[20-30]", and so on. Notably, the mining result is often dependent upon how the intervals are partitioned, especially for data values around interval boundaries. That is the so-called "sharp boundary" problem. Subsequently, the result of associative classification may also be affected in terms of accuracy and understandability.

Among those domain discretization techniques for association rule mining, fuzzy logic is regarded suitable to deal with the "sharp boundary" problem by providing a flexible and intelligent remedy[1,2,20]. This gives rise to the notion of fuzzy association rules (FAR). It is worthwhile to mention that classical association rules (AR's) are special cases of fuzzy association rules (FAR's). Moreover, the semantics of a fuzzy association rule is richer and of certain natural language nature, which are deemed desirable. For example, "*low-quantity Fruit $\Rightarrow$ normal-consumption Meat*" and "*medium Income $\Rightarrow$ young Age*" are fuzzy association rules, where X's and Y's are fuzzy sets with linguistic terms (i.e., *low, normal, medium,* and *young*). In a similar line of thinking, building an associative classifier based upon fuzzy association rules is considered meaningful in twofold: one is the need to mine large datasets with quantitative domains; the other is to generate classification rules with more general semantics and linguistic expressiveness. In this context, a rule $X \Rightarrow C$ in the built classifier reflects a classification criterion with X in general being a combination of fuzzy sets/linguistic terms, meaning that a record with data items compatible with X could be identified to class C. As a matter of fact, the previously mentioned example, "*high Income & top-brand Car $\Rightarrow$ Plan* B" is of such a nature.

This paper is aimed at dealing with the "sharp boundary" problem for quantitative domains and providing an approach to building an associative classification based on fuzzy association rules (namely CFAR). Apparently, classical associative classification approaches and algorithms are incapable, and

extensions are necessary. Notably, the proposed approach shall address a number of important issues of concern as follows. First, while fuzzy sets in forms of linguistic terms (e.g., *young, meddle, old*) are defined on a quantitative domain of attribute (e.g., domain of *Age*), a precise age value in the original dataset may belong to a fuzzy set at a degree in [0, 1], which is different from the case of partitioned intervals where the belonging is crisp: either 0 or 1. This leads to a need to reformulate the notions of support and confidence. Second, unlike the case of (crisp) partitioning where a value only belongs to one and only one interval, it is normal that a value may belong to more than one fuzzy set, each with a degree. Such "multiple belongings" may have certain impact on the support/confidence degrees. Third, since the resultant classifier usually consists of fuzzy classification rules, more rules may be relevant in identifying a data record. These issues of concern will be discussed in light of two important criteria, namely accuracy and understandability. By accuracy we mean the rate of identifying data/records into correct classes, and by understandability we mean that fewer rules are more preferable (in addition to the enhanced expressiveness of fuzzy classification rules).

The paper is organized as follows. Sec. 2 introduces the background notions including association rules and fuzzy association rules. Sec. 3 presents new measures of support and confidence and related notions in the fuzzy context. In Sec. 4, the proposed approach CFAR is discussed with corresponding algorithmic details. Sec. 5 shows data experiments in comparison with a typical associative classification approach CBA.

2. The Background

2.1. *Association rules*

Classically[23], given a transactional dataset D and a set of data items I = {$I_1, I_2 ..., I_m$}, an association rule (AR) is of the form: $X \Rightarrow Y$, expressing the semantics that "occurrence of X is associated with occurrence of Y", where X and Y are disjoint collections of data items (i.e., X and Y are called itemsets, $X,Y \subset I$, and $X \cap Y = \varnothing$). Also, X is referred to as the antecedent of an AR and Y as consequent of the AR. Let $|D|$ be the number of transactions (interchangeably referred to as records, otherwise indicated where necessary) in D, and $\|X\|$ ($\|XY\|$) be the number of transactions containing X (XY), the degrees of support and confidence (denoted as

Dsupp and *Dconf* respectively) for $X \Rightarrow Y$ are defined as: $Dsupp(X \Rightarrow Y) = \|XY\|/|D|$ and $Dconf(X \Rightarrow Y) = \|XY\|/\|X\|$. An example of an association rule in supermarket basket transactions is "*Fruit $\Rightarrow$ Meat*, with *Dsupp* = 20% and *Dconf* = 80%" meaning that "20% of the customers bought *Fruit* and *Meat* simultaneously, and 80% of the customers who bought *Fruit* also tended to buy *Meat*". Such association rules are also referred to as Boolean ARs, since the association concerned is the correspondence of the occurrence states, each being a binary value 0 or 1. There have been many efforts proposed to discover Boolean ARs in various ways[11,22,25,26,], among which the Apriori algorithm[16] is usually deemed as a typical one.

In addition to binary data values in Boolean association rules, categorical or generally quantitative data values are commonly encountered in many real applications. As shown in Table 1, for instance, *Age* and *Height* are quantitative attributes, whereas *Bald* is a categorical attribute.

Table 1: Database (*D*) with Continuous Domains.

D	Age	Height	Bald	…
ID1	31	170	1	…
ID2	25	180	0	…
ID3	16	182	0	…
ID4	52	165	1	…
…	…	…	…	…

While the Apriori algorithm is effective and efficient, it can hardly be used directly in discovery of quantitative association rules. A commonly used way is to partition the domains and therefore introduce new attributes with intervals. In doing so, the original transactional dataset *D* could be transformed into a binary database *D′* which is exemplified in Table 2. Accordingly, quantitative ARs such as "*Age*(65,100] & *Height*(180,190] $\Rightarrow$ *Bald*" can be discovered using a Apriori-type algorithm[20, 21].Table 2: Database (*D′*) Transformed from *D*.

D′	Age(0,27]	Age(27,65]	Age(60,100]	Bald	…
ID1	0	1	0	1	…
ID2	1	0	0	0	…
ID3	1	0	0	0	…
ID4	0	1	0	1	…
…	…	…	…	…	…

2.2. *Fuzzy sets*

Recall that a classical set is characterized by the membership degrees of domain elements being either 0 or 1, representing either non-belonging or full-belonging. By contrast, a fuzzy set[19] is characterized by the membership degrees of domain elements being a value in [0, 1], representing a gradual transition from "non-belonging" to "full-belonging". In many cases, linguistic terms are of fuzzy nature in concept and usually formulated by fuzzy sets.

Let *U* be the universe of discourse. Then a fuzzy set *FS* on *U* is:
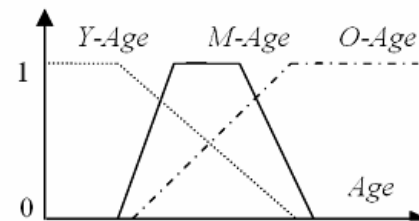
$$FS = \{(x, \mu(x)) \mid x \in U\},$$

Figure 1: Fuzzy Sets *Y-Age*, *M-Age* and *O-Age*.

$$(1)$$

where μ is the membership function reflecting a mapping from *U* to [0, 1]. Usually, there are two alternative ways to represent a fuzzy set *FS*:

$$FS = \begin{cases} \sum_{x_i \in U} \mu(x_i) & \text{if } U \text{ is discrete} \\ \int_U \mu(x) \mid x & \text{if } U \text{ is continuous} \end{cases} . \quad (2)$$

Conveniently, there are also several parameterized ways to define membership functions. Some commonly used function types among others are: trapezoidal, bell, Gaussian and triangular. A parameterized membership function can be defined in terms of a number of parameters. For example, a triangular membership function is specified by three parameters (*a*, *b*, *c*); and for a given value u, with known *a*, *b*, and *c*, the membership of u may be computed as

$$\text{triangle}(u; a, b, c) = \max\left(\min\left(\frac{u-a}{b-a}, \frac{c-u}{c-b}\right), 0\right). \quad (3)$$

For example, for attribute *Age*, some fuzzy sets may be defined on its domain U_{Age} such as *Young (Y-Age)*, *Middle (M-Age)* and *Old (O-Age)*, each with a

Zuoliang Chen, Guoqing Chen

parameterized trapezoidal membership function as shown in Fig. 1.

2.3. *Fuzzy association rules*

As indicated in subsection 2.1, domain partitioning could be a way in quantitative association rules mining. However, since the interval boundaries are sharply defined (i.e., the sharp boundary problem), the domain elements near the boundaries may be ignored or over-emphasized in the mining process, which is usually not intuitive with respect to human perception. For example, an abstract concept such as "short" to describe a person's height would be distinguished by a precise value such as (-, 170cm), often leading to a difficulty to intuitively identify nearby values such as 171 due to such a nature of artificial precisionization. Furthermore, the degrees of the elements belonging to the interval are not distinguishable, such as 150 and 170 being treated equally, which are not considered intuitively appealing. To cope with the problem, a flexible and transitional setting of the interval boundaries is regarded desirable, hence fuzzy logic is employed. Accordingly, for the data as shown in Table 1, a discovered fuzzy association rule (FAR) could be, for example, "*O-Age & Tall ⇒ Bald*". In recent years, there have been a number of efforts on discovery of FARs from various perspectives of extension[9,10,24,27,28,30]. For instance, they map the values of quantitative attributes to memberships of several new attributes in forms of linguistic terms and then mine association rules from the new database. Consider attribute *Age* in Table 1 again, three new attributes (e.g. *Y-Age*, *M-Age* and *O-Age*) in place of *Age* may be used to constitute a new database (D'') with partial belongings of original attribute values to each of the new attributes. Table 3 illustrates an example of the new database obtained from the original database, given fuzzy sets *Young* (*Y-Age*), *Middle* (*M-Age*) and *Old* (*O-Age*) as characterized by membership functions shown in Fig. 1.

Table 3: Database (D'') with Fuzzy Items.

D''	Y-Age	M-Age	O-Age	Bald	…
ID1	0.8	0.4	0.1	1	…
ID2	0.9	0.3	0	0	…
ID3	1	0	0	0	…
ID4	0.1	0.2	0.8	1	…

…	…	…	…	…	…

Generally, for original database *D* with attributes $I = \{I_1, I_2 ..., I_m\}$, each I_k ($1 \le k \le m$) can be associated with q_k fuzzy sets defined on the domain of I_k, and usually labeled as q_k new attributes. We use $F_k = \left\{ I_k^1, I_k^2, ..., I_k^{q_k} \right\}$ to represent the set of fuzzy sets associated with I_k. That is, the new database D'' is with respect to schema $R(I'')$ where $I'' = \left\{ I_1^1, ..., I_1^{q_1}, ..., I_k^1, ..., I_k^{q_k}, ..., I_m^1, ..., I_m^{q_m}, ..., \right\}$. Correspondingly, the notions of support and confidence can be extended, for example, as follows[30]:

$$support(X \Rightarrow Y) = \sum\nolimits_{d \in D''} \mu_{XY}(d) \big/ |D''|, \quad (4)$$

$$confidence(X \Rightarrow Y) = \sum\nolimits_{d \in D''} \mu_{XY}(d) \big/ \sum\nolimits_{d \in D''} \mu_X(d). \quad (5)$$

where $|D''|$ is the number of transactions (records) in database D'', X and Y are subsets of I'' (a composed set of fuzzy sets), $\mu_X(d)$ and $\mu_{XY}(d)$ are the degrees of transaction *d* belonging to *X* and *Y* respectively with T-norms as conjunction operators (e.g., min), and $\sum$ is $\sum$count operastor[22].

2.4. *Normalization*

For a transaction *d*, its value of the kth attribute $d[I_k]$ can be mapped to q_k membership degrees, and we use $\mu_k^j(d)$ to represent the degree of $d[I_k]$ in the jth category or fuzzy set of I_k. If I_k is categorical, $\mu_k^j(d) \in \{0,1\}$. If I_k is fuzzy, $0 \le \mu_k^j(d) \le 1$.

If I_k is categorical, we always have

$$\sum\nolimits_{j=1}^{q_k} \mu_k^j(d) = 1. \quad (6)$$

If I_k is fuzzy, Eq.(6) does not always hold. This may not be desirable in light of support counting semantics, as a transaction, which occurred once, is likely to be counted a total of greater than 1. Therefore, a normalization treatment is considered necessary and could be handled as follows:

$$\mu_k'^{j}(d) = \begin{cases} \mu_k^j(d) \big/ \sum\nolimits_{j=1}^{q_k} \mu_k^j(d) & \text{if } I_k \text{ is fuzzy} \\ \mu_k^j(d) & \text{if } I_k \text{ is categorical} \end{cases} \quad (7)$$

As an example, suppose $I = \{Age, Bald\}$, as mentioned before, where *Bald* is a categorical attribute with the domain of {0, 1} and *Age* is quantitative with three fuzzy sets {*Y-Age*, *M-Age* and *O-Age*}. Given a transaction $d = \{40, 1\}$, we have D'' with new values as shown in Table 4.

Without normalization (Table 4), for itemset {*Bald, Y-Age*}, *Dupp*({*Bald, Y-Age*}) = 0.7, *Dupp*({*Bald, M-Age*}) = 0.6, and *Dupp*({*Bald, O-Age*}) = 0.1. That is to say, this one transaction d = {40, 1} in *(bald, Age)* of *D*, corresponds to three itemsets in D" with their sum of degrees of support totaling 1.4 (= 0.7+0.6+0.1). Concretely, Table 4 is normalized and the resultant data are shown in Table 5. Note that such a normalization process is used in the following discussions in mining FARs.

Table 4: Without Fuzzy Normalization.

D″	Y-Age	M-Age	O-Age	Bald
ID1	0.7	0.6	0.1	1
…	…	…	…	…

Table 5: After Fuzzy Normalization.

D″	Y-Age	M-Age	O-Age	Bald
ID1	0.5	0.429	0.071	1
…	…	…	…	…

3. Fuzzy Association Rules in Classification

In classical associative classification, CBA and its modifications[4,33] are well-known, which use an Apriori-type association rule mining approach to generate classification rules. Subsequently, filters may be applied to the rules so as to eliminate non-interesting ones such as conflicts and so on. Then, a set of high confidence rules is selected to form a classifier.

In a fuzzy context, a fuzzy association rule in classification is of the following form:

$$F \Rightarrow C. \qquad (8)$$

where $F = \{f_1, f_2, ..., f_p\}$ is the subset of I'', and $|F \cap F_k| \leq 1$, where F_k represents the set of fuzzy sets associated with I_k ($1 \leq k \leq m$). The restriction $|F \cap F_k| \leq 1$ prevents associating meaningless fuzzy sets in F defined on a domain of the same attribute. For example, it tries to avoid FARs such as "*Y-Age & O-Age $\Rightarrow$ Bald*" where *Y-Age* and *O-Age* are defined on the domain of a single attribute *Age*. *C* is a class, and it is crisp. If the degree that a record *d* belongs to f_l is $\mu_l(d)$ ($1 \leq l \leq p$), we can define the degree that the record *d* belongs to *F* as

$$\mu_F(d) = \min\{\mu_1(d), \mu_2(d), ..., \mu_p(d)\}. \qquad (9)$$

3.1. *New measures*

Since classification rules are generally special cases of association rules, in both crisp and fuzzy contexts, a fuzzy classification rule $F \Rightarrow C$ could be measured directly in terms of support and confidence as follows[27]:

$$support(F \Rightarrow C) = \frac{\sum_{d[I_c]=C} \mu_F(d)}{|D|}, \qquad (10)$$

$$confidence(F \Rightarrow C) = \frac{\sum_{d[I_c]=C} \mu_F(d)}{\sum_{d \in D} \mu_F(d)}, \qquad (11)$$

where $d[I_c]$ represent the class label of record *d*.

For example, suppose $F= \{O\text{-}Age\}$ and $C=C_1$ and part of a database shown in Table 6. We have:

$$support(F \Rightarrow C) = \frac{0.1+0.7+0.8+0.5}{6} = 35.0\%,$$

$$confidence(F \Rightarrow C) = \frac{0.1+0.7+0.8+0.5}{0.1+0.7+0.8+0.9+0.5+0.1} = 67.7\%.$$

Table 6: Part of a Database Containing Membership.

M-Age	Class
0.1	C_1
0.7	C_1
0.8	C_1
0.9	C_2
0.5	C_1
0.1	C_2

It is important to note that, with fuzzy sets defined on domains, a domain element may belong to more than one fuzzy set, each having different degrees. This fact may have an impact on the degrees of confidence, further on accuracy if classification is concerned. For instance, for a particular attribute defined with a fuzzy set in *F*, if there are many values whose membership grades are very low, then the degree of confidence for a rule with this attribute may be lower than that without those low values. For illustrative purposes, let us

Zuoliang Chen, Guoqing Chen

consider a simplified case with two databases (D^I and D^{II}) as shown in Tables 7 and 8 respectively.

Table 7: Database D^I

D^I	A	$Class$	…
ID1	1	C_1	…
ID2	1	C_1	…
…	…	…	…
ID10	1	C_1	…
ID11	0	…	…
…	…	…	…
ID100	0	…	…

Table 8: Database D^{II}

D^{II}	A	$Class$	…
ID1	1	C_1	…
ID2	1	C_1	…
…	…	…	…
ID10	1	C_1	…
ID11	0.1	…	…
…	…	…	…
ID100	0.1	…	…

Apparently, *confidence*($A \Rightarrow C_1$) with database D^{II} is smaller than that with database D^I due to the existence of non-zero degree values of A relating to other classes than C_1. In other words, the dominator (i.e., $Dsupp(A)$) for D^{II} is larger than that for D^I, while the nominators (i.e., $Dsupp(AC_1)$) are the same, resulting in different confidence degrees. On one hand, these non-zeros degree values reflect the extents to which elements belong; on the other hand, too low degree values make little semantic sense in light of membership grade but are seemingly disturbing, especially in a massive database with a large number of such values. Therefore, a cut threshold ($MS \in [0, 1]$) may be introduced to discard those degree values below MS in order to reduce the risk of losing certain rules due to low confidence levels caused by such a disturbance. It is worth mentioning, however, that setting MS is in fact a matter of trade-off: semantics vs. disturbance. Moreover, the setting needs to be made in light of accuracy in the context of classification.

In line with this thought, new measures for support and confidence are proposed as a result of extension considering the cut threshold. Concretely, given MS in [0, 1], the degree of support is:

$$Dsupp(F \Rightarrow C) = \frac{\sum_{d[I_c]=C} \mu_F(d) \lceil \max(0, \mu_F(d) - MS) \rceil}{|D|}, \quad (12)$$

and the degree of confidence is:

$$Dconf(F \Rightarrow C) = \frac{\sum_{d[I_c]=C} \mu_F(d) \lceil \max(0, \mu_F(d) - MS) \rceil}{\sum_{d \in D} \mu_F(d) \lceil \max(0, \mu_F(d) - MS) \rceil}. \quad (13)$$

For example, if $MS=0.15$ for data in Table 6, then we have:

$$Dsupp(F \Rightarrow C) = \frac{0.5 + 0.7 + 0.8}{6} = 33.3\%,$$

$$Dconf(F \Rightarrow C) = \frac{0.5 + 0.7 + 0.8}{0.7 + 0.8 + 0.9 + 0.5} = 69.0\%.$$

The setting of MS can be made by users or experts according to their domain knowledge/expertise, though, supportive means and heuristics may be available upon relevant techniques. For instance, a data-based setting against accuracy was used in the experiments of this paper, which will be described in Sec. 5. Note that in the case of $MS = 0$, Eqs. (12) and (13) degenerate to Eqs. (10) and (11) respectively.

3.2. *Rule redundancy and conflicts*

Though fuzzy classification rules can be discovered using extended Apriori-type association rule mining techniques directly, the whole set of rules might be poor in quality. First, the number of rules may be too large to easily construct classifiers. More seriously, from the viewpoint of classification, there may exist conflicting rules and redundant rules. For example, $F \Rightarrow C_1$ and $F \Rightarrow C_2$ are conflicting rules (C_1 and C_2 are class labels, and $C_1 \neq C_2$). $F \Rightarrow C_1$ and $FF' \Rightarrow C_1$ with $Dconf(F \Rightarrow C_1) > Dconf(FF' \Rightarrow C_1)$ contain redundant rules. The conflicting rules will lead to identifying a transaction into two classes, while the redundancy will result in some rules useless for classification.

Definition 1: Given two fuzzy classification rules, r_i and r_j, $r_i > r_j$ (also called r_i precedes r_j) if
(1) *Dconf* of r_i is greater than that of r_j, or
(2) their *Dconfs* are the same, but *Dsupp* of r_i is greater than that of r_j, or

(3) both *Dconf* and *Dsupp* of r_i and r_j are the same, but r_i is generated earlier than r_j.

The purpose of the definition is to make sure that we can order any set of rules. Let *R* be the set of all the fuzzy classification rules. Obviously, given $r_i, r_j \in R$, we get either $r_i > r_j$ or $r_j > r_i$. Furthermore, the relation of ">" is transitive. That is to say, given $r_i, r_j \in R$, if $r_i > r_j$, $r_j > r_k$, then we get $r_i > r_k$. This can be easily proven based upon the transitivity of arithmetic comparison involved in (1), (2) and (3) of Definition 1 for numeric values in *Dconf*, *Dsupp*, and time. Thus, rules in *R* can be ordered by ">" consisting in one and only one result. For illustrative purposes, suppose we get six fuzzy classification rules as shown in Table 9. Definition 1 will lead to $r_2 > r_3 > r_5 > r_1 > r_4 > r_6$.

Table 9: Example of Rules.

NO.	Rules	Dsupp	Dconf
r1	*M-Age ,Bald* $\Rightarrow C_1$	15%	60%
r2	*M-Age* $\Rightarrow C_1$	21%	65%
r3	*Y-Age, Bald* $\Rightarrow C_1$	10%	65%
r4	*Y-Age* $\Rightarrow C_2$	14%	40%
r5	*Y-Age* $\Rightarrow C_1$	16%	60%
r6	*Y-Age, Bald* $\Rightarrow C_2$	8%	35%

Definition 2: Given two fuzzy classification rules, r_i and r_j, $r_i \succ r_j$ (also r_j is called inferior to r_i, or r_i is called superior to r_j) if the antecedent part of r_i is the subset of that of r_j, and $r_i > r_j$.

Note that the notion of "subset" is here in the classical sense of set inclusion for items in I''. For instance, {*Bald*} is a subset of {*M-Age*, *Bald*}, though *M-Age* and *Bald* are items both formulated in linguistic terms. Furthermore, the relation of "$\succ$" also is transitive. Given $r_i \succ r_j$ and $r_j \succ r_k$, we get $r_i \succ r_k$. So we can represent this as $r_i \succ r_j \succ r_k$. This property can be easily proven as well since Definition 2 involves set inclusion (i.e., $\subset$, $\subseteq$) and ">" of Definition 1, which are both transitive.

The purpose of defining "$\succ$" is to identify redundant and conflicting classification rules. Given $r_i \succ r_j$, if the consequents of them are the same, then r_j is redundant. If different, then r_j conflicts with r_i and it is the inaccurate one (because it is preceded by r_i). In general, r_j should not be contained in the classifier

once it is inferior to another rule. We will explain this by the following example.

Take Table 9 as an example, there exist $r_2 \succ r_1$, $r_5 \succ r_4 \succ r_6$ and $r_3 \succ r_6$. For simplicity, just consider $r_2 \succ r_1$ and $r_3 \succ r_6$. Compared with r_2, r_1 is considered redundant because any transaction that is compatible with *M-Age* can be classified as C_1 according to r_2 with a higher level of confidence, needless of further considering *Bald* in r_1. r_3 conflicts with r_6 because a transaction may be classified into two classes (e.g., C_1 and C_2). We can draw similar conclusion from $r_5 \succ r_4 \succ r_6$. Notice that there is no redundancy or conflicts between r_4 and r_3 according to the definition "$\succ$".

Usually, given a nonempty set *R* of discovered fuzzy classification rules, filters can be built to deal with the problem of redundancy and conflicts. It can be seen that for any *R* there exists a corresponding set *R'* with such redundancy and conflicts removed.

Definition 3: Given a set *R* of fuzzy association rules in classification. *R'* is called a compact set of *R*, denoted as CompSet(R), if (1) $\forall r \in R - R'$, $\exists r' \in R'$ such that $r' \succ r$, and (2) $\forall r' \in R'$, if $\exists r \in R$ such that $r' \succ r$, then $r \in R - R'$.

Apparently, we have $R' \subseteq R$. Moreover, *R'* is composed of those rules who are either "isolated" in *R* (i.e., to which there do no exist any other rules in *R* that are inferior or superior) or "most-superior" in *R* (i.e., to which other rules in *R* are inferior, if any). This is equivalent to Definition 3 and can be justified as follows. On one hand, suppose there exists a rule r_1 in *R'* who is not isolated and not most-superior. That is, there exists a rule r_2 in *R* who is superior to r_1. If r_2 is in *R'*, then according to (2) of Definition 3, $r_1 \in R - R'$, which is contradictory to $r_1 \in R'$. If r_2 is in $R - R'$, then according to (1) of Definition 3, there exists $r_3 \in R'$, such that $r_3 \succ r_2$. Based on transitivity, we have $r_3 \succ r_2 \succ r_1$. According to (2) of Definition 3, $r_1 \in R - R'$, which is contradictory to $r_1 \in R'$. On the other hand, let us look at (1) and (2) of Definition 3. For (1), if there exists a rule r_1 that is inferior or superior to *r* in $R - R'$, then, the most superior one in the transitivity chain containing *r* and r_1 is in *R'*, which is denoted as *r'* in *R'*. For (2), suppose that such $r \notin R - R'$, then $r \in R'$. Since $r' \succ r$, this is contradictory to *r* being most-superior.

Zuoliang Chen, Guoqing Chen

Importantly, the rules in R' are free of redundancy and conflict. First, suppose we have r_1: $F \Rightarrow C_1$ and r_2: $FF' \Rightarrow C_1$ with $Dconf(F \Rightarrow C_1) > Dconf(FF' \Rightarrow C_1)$. In classification, r_2 is considered redundant in relation to r_1. Note that $r_1 \succ r_2$ because $r_1 > r_2$ and $F \subseteq FF'$. Then according to Definition 3, $r_2 \notin$ R' and the most superior rule in the chain containing $r_1 \succ r_2$ will be included in

```
1    Q=sort(R)
2    for each rule r ∈ Q in sequence do
3        for each rule r ∈ Q and r > r' in
         sequence do
4            if r ≻ r' then
5                delete r' from Q
6        end
7    end
8    R' = {r| r ∈ Q }
9    return R' (the CompSet of R)
```

Fig. 2: Algorithm CompSet.

R'. Second, suppose we have r_1: $F \Rightarrow C_1$, and r_3: $F \Rightarrow C_2$ ($C_1 \neq C_2$). Since $F \subseteq F$, without loss of generality, suppose $r_1 \succ r_3$. r_3 cannot be in R' because r_3 is not most-superior. Consider Table 9 again, R = $\{r_1, r_2, r_3, r_4, r_5, r_6\}$. With $r_2 \succ r_1$, $r_5 \succ r_4 \succ r_6$ and $r_3 \succ r_6$, we obtain $R' = \{r_2, r_3, r_5\}$.

Given any R, we can get the unique R', independent of the order in which the removing steps are performed.

Proposition 1: The Algorithm CompSet(R) produces one and only one compact set *R'* of any set *R*.

Proof: The algorithm to obtain CompSet(R) is shown in Fig. 2. It has 3 steps:
- Step 1 (line 1): Sort the set of generated rules R according to the relation ">".
- Step 2 (line 2-7): Delete the rules that are inferior to others.
- Step 3 (line 8, 9): Return the set of the left rules.

Step 1 is based on the definition of ">", which makes sure, as mentioned above, that we can order R and get the corresponding result Q. Q could be regarded as a queue, which is ordered. Step 2 is based on the definition of "$\succ$". If there is a "$\succ$" relation, drop the inferior one. In this way, we obtain a CompSet by definition.

Suppose that R' and R'' are two CompSets of R. Without loss of generality, if there exists a

rule $r \in R'' - R'$, so $\exists r' \in R'$ and $r' \succ r$ because R'' is a Comp*Set*. Obviously, $r' \notin R''$, so $\exists r'' \in R''$ and $r'' \succ r'$. That is to say, $r'' \succ r$ and $r'', r \in R''$, which contradicts to the definition of CompSet. Thus, there exists no rule $r \in R'' - R'$, which means $R'' - R' = \phi$. Likewise, $R' - R'' = \phi$. So we have $R'' = R'$. □

For illustration, we will show how the algorithm works for Table 9. Initially, $R = \{r_1, r_2, r_3, r_4, r_5, r_6\}$, and $Q = \{r_2, r_3, r_5, r_1, r_4, r_6\}$. Set $r = r_2$, and r_1 is the first one that is inferior to r_2, so remove r_1. Since neither r_5 nor r_6 is inferior to r_2, $Q = \{r_2, r_3, r_5, r_4, r_6\}$. Secondly, set $r = r_3$, which is only superior to r_6, so $Q = \{r_2, r_3, r_5, r_4\}$. Next $r = r_5$ and r_4 is inferior to it, so $Q = \{r_2, r_3, r_5\}$. Because r_5 is the last of the queue, the process ends. Thus $R' = \{r_2, r_3, r_5\}$. Note that the transitivity of $\succ$ guarantees that removing inferior rules on a transitivity chain of rules could be carried out in any order. For instance, take $r_5 \succ r_4 \succ r_6$ in Table 9, the order of removing r_4 and r_6 does not affect the result of R'.

It is worth mentioning, however, that these notions of redundancy and conflict are particularly relevant for classification and may not be of concern for association rules in general.

4. Classification Based on Fuzzy Associations

First, we extend the algorithm[9] to generate fuzzy association rules with the new measures introduced in Eqs. (12) and (13) of Sec. 3.2. Then, the CompSet of the resultant rules is obtained. Next, the CompSet is further trained with the given training dataset based on a match measure (namely *DF*) between a fuzzy rule and a transaction.

Definition 4: Given a fuzzy rule *r:* $F \Rightarrow C$ and a transaction *d*, the confidence of classifying *d* with *r* is:

$$DF = \mu_F(d) * Dconf(r). \qquad (14)$$

where *Dconf(r)* is the degree of confidence for rule *r* as defined in Eq. (13) of Sec. 3.2, and $\mu_F(d)$ is the match that d is compatible with fuzzy set *F*.

Given a transaction *d*, the rule with the highest *DF* is used to classify d. Notably, for crisp rules, the value of $\mu_F(d)$ either is 1 where the rule *r* covers *d*, or is 0 where *r* does not cover *d*. Thus, the rule *r* which covers *d* with the highest *Dconf* will be chosen to classify *d*, which is the same with CBA and GARC classifiers.

Let R be the CompSet of the generated fuzzy rules, and D'' the training dataset with fuzzy items. Our proposed algorithm, namely CFAR, will generate a classifier in the following form:

```
1.   for each d″ ∈ D″ do
2.       for each r ∈ R do
3.              r.DF = μ_F(d″) * Dconf(r)
4.              r.use=0
5.              r.RightN=0
6.              r.WrongN=0
7.       end for
8.       sort R by r.DF desc
9.       flag=0
10.      while(flag=0) do
11.             r=fisrt rule of R
12.             if r.use=1 then
13.                    Break
14.             r.use=1
15.             if clas(r, d″ ) then
16.                    r.RightN++
17.                    flag=1
18.                    break
19.             else
20.                    r.WrongN++
21.                    move r to the bottom
         of R
22.  end for
```

Fig. 3: Procedure 1: Training.

$$<r_1, r_2, ..., r_n, default\_class>,$$

where *default_class* is the default class. In classifying an unseen transaction, sort R by DF and the first rule will be chosen to classify it. If there is no such rule that applies to the transaction, it takes on the default class. Concretely, algorithm CFAR is constituted by three procedures:

(i) Procedure 1: Training

The procedure for training the CompSet R is shown in Fig. 3, which consists of 3 steps:

  (a) Step 1 (line 2-8): calculate the DF of each rule and sort the rules by DF.
  (b) Step 2 (line 9-22): select the rule in sequence until the transaction is classified correctly. For each rule r, *RightN* and *WrongN* are used to record the number of transactions it has classified right and wrong, respectively.
  (c) Step3 (line 1): iterate steps 1 and 2 for each transaction.

(ii) Procedure 2: Delete the "worst" rule from R and compute the error rate.

We discard those rules that do not improve the accuracy of the classifier, including those whose *RightN* is zero, and the one with the smallest *RightN /(WrongN+ RightN)* whose *RightN* >0. We then compute and record the total number of errors that are made by the current classifier and the default class. This is the sum of the number of errors that have been made by all the selected rules in the classifier and the number of errors to be made by the default class in the training data.

(iii) Procedure 3: Iterate procedures 1 and 2 until the error rate on training set increases.

5. Experimental Results

This section shows an empirical performance evaluation of algorithm CFAR, along with some comparisons with the well-known CBA algorithm[4]. The experiments consist of three parts. The first part is to compare CFAR with CBA classifier on accuracy. The second part is to discuss the impact of threshold *MS* on CFAR outcomes. In the third part, we compare CFAR with the CBA classifier in terms of the number of rules produced. It should be mentioned that all the following experiments are tested based on datasets from a commonly used benchmarking database in this field, namely UCI ML Repository[5]. Since the proposed approach and corresponding CFAR is to deal with the "sharp boundary" problem where quantitative attribute domains pertain, we selected 7 relevant datasets for the experiments and did not consider other datasets whose attributes are all/almost discrete. The basic information of the datasets is listed in Table 10.

Table 10: Basic information of the datasets.

	Dataset	Attr.	NO. of attr.	NO. of tra. data	NO. of tes. data
1	Australian	Dis., con.	14	460	230
2	Breast	Con.	10	466	233
3	Cleve	Dis., con.	13	202	101
4	Crx	Dis., con.	15	490	200
5	German	Dis., con.	20	666	334
6	Heart	Con.	13	180	90
7	Wine	Con.	13	118	60

In our experiments reported below, for CFAR, we set the threshold of the *Support* (*minpsup* for short) to 10%, and for the *Confidence*, its threshold (*minpconf*) is

Zuoliang Chen, Guoqing Chen

set to be 85%. For CBA, all parameters had their default values. Discretization of quantitative domains was done using the Entropy method[29], and then triangular membership functions were specified by the discretized points for fuzzy items.

5.1. *Accuracy*

Accuracy is one of the basic performance measures for classification algorithms. In comparing CFAR with CBA, we set MS=0.15, which is the default value, and we will discuss it in later subsection on the impact of this threshold. Table 11 shows the accuracy result, which indicates that the classification accuracy of CFAR is satisfactory. On average, the accuracy of CFAR seemed at least as good as that of CBA. Moreover the CFAR appeared similarly stable as CBA in terms of standard deviations of accuracy. These findings could be further justified by statistical significance tests.

Table 11: Algorithms' accuracy on CBA and CFAR

	Dataset	CBA%	CFAR%
1	Australian	87.39	87.83
2	Breast	96.57	95.71
3	Cleve	81.19	83.17
4	Crx	80.50	80.00
5	German	72.75	72.75
6	Heart	83.33	85.55
7	Wine	86.67	93.33
	Mean	84.06	85.48
	Standard deviation	7.35	7.84

5.2. *Setting of MS*

As mentioned in previous subsections, the experiment is conducted with a setting of MS=0.15. In this section, we will discuss further the impact of this setting on the accuracy of CFAR. MS has a slightly effect on the quality of the classifier produced. If MS is set too low, the interestingness measures perform the same with the typical ones, and some useful rules may not be included. If MS is set too high, the rules generated will not be as "confident" as they appear, and some interesting rules with high *confidence* may be not generated because of their low *Support*. In this experiment, with each of the same datasets, we set MS to different values and obtain the best setting of MS in yielding the highest accuracy. Obviously, the best setting for one dataset may be

different from that for another dataset. To determine a single default setting, we chose MS=0.15, as it gives a satisfactory result. Table 12 shows the details. The result also shows that the new measures perform slightly better than the typical ones with a little higher average of accuracy.

Table 12: Setting of MS vs. accuracy on CFAR

Dataset	Highest accuracy		Accuracy at MS=0.15	Accuracy at MS=0
	%	MS		
Australian	88.70	0.15	88.70	87.83
Breast	98.71	0.15	98.71	95.71
Cleve	83.17	0.15	83.17	83.17
Crx	80.00	0.15	80.00	80.00
German	72.75	0.2	72.45	72.15
Heart	85.56	0.15	85.56	83.33
Wine	91.67	0.2	90.00	90.00
Mean	85.79		85.51	84.60

5.3. *Number of rules*

MS has effect not only on the accuracy, but also on the numbers of rules generated by CFAR. In the above subsection, we obtained the settings of MS with the highest accuracy. In this experiment, we compare the number of rules generated by CBA and CFAR.

Table 13: Number of rules generated by CBA and CFAR

	Dataset	CBA	CFAR
1	Australian	110	21
2	Breast	31	13
3	Cleve	33	28
4	Crx	58	43
5	German	9	7
6	Heart	31	22
7	Wine	13	13
	Mean	40.7	21.0

Table 13 tabulates the comparative results. Clearly, CFAR generated fewer rules than CBA, providing better understandability, even more stable performance. The main reason for fewer rules is that CFAR uses fuzzy association rules for classification. Because of smooth boundaries, a fuzzy rule can cover more transactions than a crisp one with the same original attributes and

discretized points. That is to say, CFAR need fewer rules than CBA to cover all of the transactions. In addition, the number of rules in CFAR could also be reduced by using pruning/resolution strategies such that certain conflicting and redundant rules could be dropped.

6. Conclusion

This paper has proposed a framework to integrate classification and fuzzy association rule mining. New interestingness measures have been presented to generate all fuzzy association rules in classification, and an algorithm, namely CFAR, has also been proposed to build an accurate classifier. Compared with CBA, the proposed approach has better understandability in terms of the number of rules and the smooth boundaries, while keeping the accuracy equally satisfactory. Our ongoing work now centers on exploring more properties and incorporating them into the rule generation process as pruning strategies in order to further improve computational efficiency.

Acknowledgements

References

1. A. W. C. Fu, M. H. Wong, S. C. Sze, W. C. Wong, W. L. Wong and W. K. Yu, Finding fuzzy sets for the mining of fuzzy association rules for numerical attributes, in *Proc. Int. Symposium Intelligent Data Engineering Learning* (IDEAL'98) (Hong Kong, 1998), pp. 263-268.
2. B. C. Chien, Z. L. Lin and T. P. Hong, An Efficient Clustering Algorithm for Mining Fuzzy Quantitative Association Rules, in *Proc. of the 9th Int. Fuzzy Systems Association World Congress* (Vancouver, Canada, 2001), pp. 1306-1311.
3. B. Liu, W. Hsu and Y. Ma, Integrating classification and association rule mining, in *Proc. of the Int. Conf. on Knowledge Discovery and Data Mining* (SIGKDD98) (New York, 1998), pp. 80-86.
4. B. Liu, Y. Ma and C. Wong, Classification using association rules: weaknesses and enhancements, in *Data Mining for Scientific and Engineering Applications* (2001).
5. C. Merz and P. Murphy, *UCI repository of machine learning databases* (http://www.cs.uci.edu/~mlearn/MLRepository.html, 1996).
6. D. Meretakis and B. Wüthrich, Extending naive Bayes classifiers using long itemsets, in *Proc. of 5th Int. Conf. on Knowledge Discovery and Data Mining* (San Diego, California, August, 1999).
7. G. Chen and Q. Wei, Fuzzy Association Rules and the Extended Mining Algorithms, *Information Sciences* **147** (2002) 201-228.
8. G. Chen, H. Liu, L, Yu, Q. Wei and X. Zhang, A New Approach to Classification Based on Association Rule Mining, *Decision Support System* **42** (2006) 674-689.
9. G. Chen, P. Yan and E. E.Kerre, Computationally Efficient Mining for Fuzzy Implication-based Association Rules in Quantitative Databases, *Int. Journal of General Systems* **33**(2-3) (2004) 163-182.
10. G. Chen, Q. Wei and E. Kerre, Fuzzy Data Mining: Discovery of Fuzzy Generalized Association Rules. in *Recent Research Issues on Management of Fuzziness in Databases* (Springer, Physica-Verlag, 2000).
11. G. Chen, Q. Wei, D. Liu and G. Wets, Simple association rules (SAR) and the SAR-based rule discovery, *Computers and Industrial Engineering* **43** (2002) 721–733.
12. G. Piatetsky-Shapiro, U. Fayyad and P. Smyth, From data mining to knowledge discovery, in *An Overview. Advances in Knowledge Discovery and Data Mining* (AAAI/MIT press, 1996), pp. 1-35.
13. H. Ishibuchi, T. Nakashima and T. Yamamoto, Fuzzy Association Rules for Handling Continuous Attributes, in *Proc. of the IEEE Int. Symposium on Industrial Electronics* (Korea, 2001), pp. 118-121.
14. H. Lu and H. Liu, Decision tables: Scalable classification exploring RDBMS capabilities, in *Proc. of the 26th Int. Conf. on Very Large Databases* (Morgan Kaufmann, Cairo, Egypt, 2000), pp. 373–384.
15. H. Mannila, H. Toivonen and A.I. Verkamo, Efficient Algorithms for discovering association rules, in *KDD-94: AAAI Workshop on Knowledge Discovery in Databases*, pp. 181-192.
16. H. Toivonen, Sampling large databases for association rules, in *Proc. of the 22nd Int. Conf. on Very Large Database* (Bombay, India), pp. 134-145.
17. J. Roberto and J. Bayardo, Brute-force mining of high-confidence classification rules, in *Proc. of the Third Int. Conf. on Knowledge Discovery and Data Mining* (AAAI Press, Newport Beach, California, 1997).
18. J. R. Quinlan, *C4.5: Programs for Machine Learning* (Morgan Kaufmann, 1993).
19. L.A. Zadeh, Fuzzy sets, *Information Control* **8** (1965) 338-353.
20. L. J. Mazlack, Approximate clustering in association rules, in *Proc. of 19th Int. Conf. of the North American Fuzzy Information Processing Society-NAFIPS 2000* (Atlanta, 2000), pp.256-260.
21. N. Friedman, D. Geiger and M. Goldszmidt, Bayesian network classifier, *Machine Learning* **29** (1997) 131–163.

Zuoliang Chen, Guoqing Chen

22. R. Agrawal and R. Srikant, Fast algorithm for mining association rules, in *Proc. of the 20th VLDB Conf.* (Morgan Kaufmann, Santiago, Chile, 1994), pp. 487–499.

23. R. Agrawal, T. Imielinski and A. Swami, Mining association rules between sets of items in large databases, in *Proc. of the ACM SIGMOD Int. Conf. Management of Date* (Washington, 1993), pp. 207-216.

24. R. B. V. Subramanyam and A. Goswami, Mining fuzzy quantitative association rules, *Expert Systems* **23** (2006) 212-225.

25. S. Brin, R. Motwani and J. Ullman, Dynamic itemset counting and implication rules for market basket data, in *Proc. of ACM-SIGMOD Int. Conf. on Management of Data* (ACM Press, Tucson, Arizona, 1997), pp. 255–264.

26. T. Fukuda, Y. Morimoto and S. Morishita, Data mining using two-dimensional optimized association rules: scheme, algorithms, and visualization, in *Proc. of the ACM-SIGMOD Int. Conf. on the Management of Data* (1996), pp. 12–13.

27. T. Hong, C. Kuo, S. Chi and S. Wang, Mining Fuzzy Rules from Quantitative Data Based on the AprioriTid Algorithm, in *Proc. of the ACM SAC 2000, Fuzzy Application and Soft Computing Track* (Italy, 2000), pp. 534-536.

28. T. Hong, K. Lin and S. Wang, Fuzzy data mining for interesting generalized association rules. *Fuzzy Sets and Systems* **138** (2003) 255-269.

29. U. M. Fayyad and K. B. Irani, Multi-interval discretization of continuous-valued attributes for classification learning, in *Proc. of the 13th Int. Joint Conf. on Artificial Intelligence* (1993), pp. 1022-1027.

30. W. Au and K. C. C. Chan, An effective algorithm for discovering fuzzy rules in relational databases. in *Proc. IEEE Int. Conf. Fuzzy Systems* (FUZZ IEEE 98) (1998), pp. 1314-1319.

31. W. Au and K. C. C. Chan, Classification with Degree of Membership: A Fuzzy Approach, in *Proc. of the 1st IEEE Int. Conf. on Data Mining* (San Jose, CA, 2001).

32. W. Li, J. Han and J. Pei, CMAR: Accurate and efficient classification based on multiple classification rules, in *Proc. 2001 IEEE Int. Conf. on Data Mining* (ICDM 2001) (California, 2001), pp. 369-376.

33. X. Yin and J. Han, CPAR: Classification based on predictive association rules, in *Proc. of 3rd SIAM Int. Conf. on Data Mining* (SDM'03) (San Francisco, CA, 2003).