Fuzzy Quantified Dependencies in Relational Databases

Miguel Delgado, Daniel Sánchez and Maria-Amparo Vila
Department of Computer Science and Artificial Intelligence. University of Granada
Avda. Andalucía n. 38, 18015 Granada Spain
Phone: +34-5824-4018, Fax: +34-5824-3100
email: {vila, daniel}@decsai.ugr.es, mdelgado@goliat.ugr.es

ABSTRACT: Functional dependencies have been extended in several ways. These extensions introduce some imprecision, uncertainty and/or graduality in the definition. Two main approaches, fuzzy functional dependencies and approximate dependencies, have been described in the literature. Several definitions of both fuzzy functional dependencies and approximate dependencies are available. In this paper, we want to give a general definition of association between attributes that extends the concept of functional dependency. Our approach is based on fuzzy quantifiers and resemblance relations between pairs of tuples. The degree of association between attributes is evaluated by means of type II quantified sentences. The obtained dependencies are called fuzzy quantified dependencies. The definition covers some existing approaches of fuzzy functional dependencies and approximate dependencies and merges both imprecision and uncertainty. It is also related to the amount of data that supports the association among attribute values, together with the accuracy of these associations.

KEYWORDS: Fuzzy functional dependency, approximate dependency, quantified sentences, fuzzy relations, association rules.

1 INTRODUCTION

Functional dependencies are integrity constraints based on real world restrictions that are used in the design process of a relational database. Given a relational scheme R and an instance r of R, a functional dependency $X \rightarrow Y$ with $X,Y \subseteq R$ holds in r iff for every pair of tuples t,s of r , $t[X]=s[X] \rightarrow t[Y]=s[Y]$. These dependencies have been an object of intensive study in the field of data mining, see for example Huhtala et.al. (1998) and Mannila et.al. (1994). However, previously unknown dependencies are difficult to find because one single exception tuple turns the functional dependency not to hold. Because of this, the definition of functional dependency has been extended in several ways, allowing to obtain smoothed dependencies by introducing some kind of imprecision, uncertainty or graduality in them.

There are two main approaches for extending the concept of functional dependency, called fuzzy functional dependencies and approximate dependencies (also called partial determinations). The former typically introduces some degree of imprecision in the definition by changing either the granularity level of the attribute domains to a higher level, or the equality onto a fuzzy resemblance relation, or the quantifier and implication onto fuzzy ones, or several at a time (see Bosc et.al. (1997) for a brief review and Cubero et.al. (1994a), (1994b) and (1998)). The latter allows for the existence of exceptions in the relation (i.e. tuples that do not fulfil the dependency) and measures the degree of fulfilment of a given dependency by means of probabilities, so dealing with uncertainty (see Delgado et. al. (1999a), Huhtala et.al. (1998), Kivinen et.al. (1995), Pfahringer et.al. (1995), Cubero et.al. (1998) and Piatetsky-Shapiro et.al. (1993)). In both approaches, the result can be in {0,1} or a degree of fulfilment in [0,1]. Also in each approach several definitions and algorithms for mining them are available.

Several applications have been proposed in the literature for extended functional dependencies. Among them we could mention database design, reverse engineering, data summarisation, query optimization and the general understanding of the structure of data . Gradual functional dependencies provide some kind of meta-knowledge about associations among attributes based on the association among attribute values. On the other hand, {0,1}-extended functional dependencies can be used for database design.

As we pointed out in Delgado et. al. (1999a) we think that for certain purposes not only the degree of dependence but also a measure of the degree in which the associations between attribute values are supported by data is needed. This fact was also pointed out in Pfahringer et. al. (1995), and it was the basis for the approximate dependency definition proposed in that paper. In our opinion, most of the existing definitions of approximate dependencies are not appropriate for all of the tasks we mentioned before. For example, if we want to use extended dependencies for query optimization, we can represent the extended functional dependencies as a set of (possibly imprecise and/or uncertain) associations between attribute values. This is the case of approximate dependencies that can be described by means of a set of association rules (see Agrawal et.al. (1993)). When the dependency come from a real world constraint, the importance and accuracy of the rules describing it are independent from data. However, when mining extended functional dependencies or whatever other knowledge structure from databases, the importance and accuracy are given by the data itself. In the case of association rules, the measures of support and confidence of the rule represent these concepts.

In this paper we propose a general definition of associations among attributes based on the associations among the attribute values called Fuzzy Quantified Dependencies. This definition is based on fuzzy resemblance relations defined in the domains of the attributes, and the use of type II quantified sentences. The type of the association depends on the resemblance relations and the fuzzy linguistic quantifier used. As we shall see, some existing extensions of functional dependencies are particular cases of fuzzy quantified dependencies. Another key point of fuzzy quantified dependencies is that they are gradual by definition, because a degree of fulfilment in [0,1] is obtained in general, so they are also sensitive to the method of evaluation of quantified sentences. In Delgado (1999b) we have developed new possibilistic and probabilistic methods that fulfil appropriate properties for the evaluation of type II sentences for any quantifier.

The paper is structured as follows. In section 2 a summary of the main approaches to extended functional dependencies is shown. Section 3 contains our definition of fuzzy association rule. In section 4 we define fuzzy quantified dependencies and we relate it to some of the existing extensions of functional dependencies Section 5 contains our conclusions and future works.

2 SOME APPROACHES TO THE EXTENSION OF FUNCTIONAL DEPENDENCY

2.1 FUZZY FUNCTIONAL DEPENDENCIES

Every approach to the definition of fuzzy functional dependencies introduce some imprecision in one or more of the elements of the definition of functional dependencies. Although some elements can be extended at the same time, in the following subsections we treat one extension at a time. The classification has been obtained from Bosc et.al. (1997) and Cubero et.al. (1998). Definitions in subsections 2.1.1 to 2.1.3 and 2.1.5 can be used over nonfuzzy relational databases by adding meaningful fuzzy measures over concrete values of the attributes and/or the set of tuples.

2.1.1 Functional dependencies on a fuzzy relation

In this approach we have that the relation r is fuzzy, i.e. each tuple has a degree of belonginess to r. One existing definition of extended functional dependency due to A. Kiss is the following:

$$T_r(X \to Y) = \inf_{t,s \in r} min\{ m_r(t), m_r(s), S_=(t[X], s[X]) \} \to S_=(t[Y], s[Y]) \quad (1)$$

Where $S_=$ is the classical equality function. It has been interpreted in Bosc et.al. (1997) as related to the degree of belonginess of the exception tuples to the relation. The lower the degree of the exceptions, the higher the degree of association between X and Y.

2.1.2 Relaxation of the equality

The equality of attribute values is relaxed into a resemblance relation. A resemblance relation is a fuzzy relation that is reflexive and symmetric. For every attribute $X \in R$ we must provide a resemblance relation S_X. If X is a set of attributes $X_1X_2..X_p$, then we can define S_X as

$$S_X\left((x_{11}, x_{21},..., x_{p1}), (x_{12}, x_{22},..., x_{p2}) \right) = min\{ S_{X_1}(x_{11}, x_{12}), S_{X_2}(x_{21}, x_{22}),..., S_{X_p}(x_{p1}, x_{p2}) \} \quad (2)$$

Then a fuzzy functional dependency can be defined as

$$X \rightarrow Y \Leftrightarrow \forall t, s \in R, \quad S_X\left((t[X]),(s[X])\right) \Rightarrow S_Y\left((t[Y]),(s[Y])\right) \tag{3}$$

When the implication is crisp, the value obtained is also crisp. Rescher-Gaines implication defined as { I(a,b)=1 if a≤b, 0 otherwise} is proposed. If the implication is fuzzy, the universal quantifier is substituted by a t-norm over the values of the implication, and a fuzzy degree is obtained.
The interpretation of this approach is for every pair of tuples: “the closer the values of X, the closer the values of Y”.

2.1.3 Incorporation of imprecise data

In this case, concrete attribute values are replaced by more general values. This replacement introduces a certain degree of imprecision in the data. In the generalisation approach, crisp data is replaced with fuzzy values, usually linguistic labels. For example, the attribute Age having values in the range {1,...,120} can be represented by a set of labels {young, medium, old, very old}. The degree of compatibility between the numeric value of Age and the corresponding labels measures the imprecision introduced by the replacement. A particular case of set of labels is that of crisp hierarchies, where we raise to a higher level of the hierarchy replacing the values of the previous level that appear in the relation with generalisations of the values in the actual level. In this case the imprecision takes the form of disjunctive values, because every term in a given level represents a subset of the domain of the attribute. A definition can be given of the form

$$(X,F) \rightarrow (Y,G) \Leftrightarrow \forall t \in R, \quad F(t[X]) \Rightarrow G(t[Y]) \tag{4}$$

The interpretation of an extended functional dependency of this kind is “the more X is F, the more Y is G”, where F and G are linguistic (possibly crisp) labels. However, the dependency holds in the relation once the domains have been generalized. These dependencies can be used to summarise the original relation by using labels as the domain of the attributes instead of concrete values.

2.1.4 Imprecise domains

In some models of fuzzy databases we can find attributes whose domain is (at least in part) imprecise. One example could be the attribute age with domain D_{age}=[0,120]∪{young, medium, old, very old}. One tuple can have either a concrete or imprecise value in the attribute. The main problem is that of the measure of similarity between values of the domain. When only concrete values are involved in the comparison, we compare it using classical equality. The problem arises when one or both values are possibility distributions. Existing approaches to deal with this case are: 1) the syntactic approach, that compare equality of the labels so that $\mu_=$(young,young)=1 but $\mu_=$(young,x)=0 for every other x in D_{age}, and 2) the semantic approach, where the labels are compared in terms of their representation as possibility distributions. In this case, given an attribute X with domain D_X, and a label L and a concrete value v both in D_X, we have

$$S_X(L,v) = L(v) \tag{5}$$

On the other hand, if we have L, M two labels in D_X then a weak resemblance relation is defined as (see Cubero et.al. (1994))

$$S_X(L,M) = \sup_{u,v} \ min\left(L(u), M(v), S_X(u,v)\right) \tag{6}$$

If S_X is the classical equality for concrete values, then the expression (6) becomes the measure proposed by G.Chen. as

$$S_X(L,M) = \sup_{u} \ min\left(L(u), M(u)\right) \tag{7}$$

Strong resemblance relations are also defined in Cubero et.al. (1994) as

$$S_X(L,M) = \inf_{u,v} \ max\left(1-L(u), 1-M(v), S_X(u,v)\right) \tag{8}$$

Some authors such as Chen and Cubero have followed this approach. The definition of Chen is

$$X \rightarrow_{\varphi} Y \;\; iff \;\; \forall t,s \in r \quad if \;\; t[X]=s[X] \; then \; t[Y]=s[Y] \; else \; \left(S_X(t[X],s[X]) \Rightarrow_{G\ddot{o}} S_Y(t[Y],s[Y])\right) \geq \varphi \tag{9}$$

where φ is a threshold value in [0,1]. The "if" part is not necessary if the labels in D_X and D_Y are normalized. The implication used is Gödel implication defined as {I(a,b)=1 if a≤b, b otherwise }. We can avoid the threshold and obtain a degree of fulfilment of the dependency by replacing the implication with a crisp one as described in Bosc et.al. (1997). The definition of Cubero has the semantics "If antecedent values are equal, then consequent ones are expected to be strongly resemblant". Let S_Y be a strong resemblance relation defined in D_Y. Then

$$X \rightarrow Y \;\; iff \;\; \forall t,s \in r \quad if \;\; t[X]=s[X] \; then \; S_Y(t[Y],s[Y]) \geq \varphi \tag{10}$$

where φ is a resemblance threshold associated to the attribute Y. Every attribute has associated a threshold that defines the minimum degree of resemblance to consider that two values are similar. This definition is less strict than a functional dependency (i.e. if a functional dependency X→Y holds on r, then $X \rightarrow_{\varphi} Y$ for all φ). Another definition can be more strict than a functional dependency, and involves a resemblance relation and threshold for X.

2.1.5 Relaxation of the universal quantifier

The idea is to allow for the existence of exceptions to the dependency. The measure of fulfilment is obtained by counting the exceptions and matching the percentage of tuples that agree with the dependency against a fuzzy quantifier. This corresponds in practice to approximate dependencies, which will be discussed in the next subsection.

2.2 APPROXIMATE DEPENDENCIES

Two main features characterize the existing definitions of approximate dependencies: the nature of exceptions and the measure of fulfilment of the dependency. We can define exceptions as either pairs of tuples or single tuples that break the dependency. The first approach is used by Piatetsky-Shapiro et.al. (1993), one measure by Kivinen et.al. (1995) and Delgado et.al. (1999a). The second one is used by Cubero (1998), Kivinen et.al. (1995), Huhtala et.al. (1998), Ziarko (1991) and Pfahringer et.al. (1995).

3 FUZZY ASSOCIATION RULES

3.1 EVALUATION OF QUANTIFIED SENTENCES

Quantified sentences are used in a large number of applications for representing assertions and/or restrictions about the number or percentage of objects that verify a certain property. Quantified sentences are usually classified into two classes, called type I sentences and type II sentences. A type II sentence can be described in general as:

Q of D are A

where Q is a linguistic quantifier and A and D are fuzzy sets defined over a finite set $X=\{x_1,...,x_n\}$. Type I sentences are a special case of type II sentences where D=X. The following are examples of each type of sentences:

Type I: Most of the students are young
Type II: Most of the efficient students are young

In these examples, the set X is a finite set of students, the quantifier is "Most", the set A is the property "young" and the set D is the property "efficient".
The evaluation of quantified sentences tries to obtain an accomplishment degree in the real interval [0,1] for the sentence. Different methods have been proposed to perform the evaluation following this approach. In Delgado et.al. (1999b) we have briefly reviewed the existing methods and we have discussed them in terms of a set of properties that, in our opinion, must be fulfilled by any method of evaluation. None of the existing methods fulfil all of the properties, so we propose two methods that fulfil every property, based on a possibilistic and a probabilistic approach respectively. These methods are

generalizations of two existing methods, based on the Sugeno and the Choquet integral respectively. The possibilistic method for the evaluation of the sentence Q of D are A can be expressed as

$$ZS_Q(A/D) = \max_{a \in M(A/D)} \min\left(a\ , Q\left(\frac{|(A \cap D)_a|}{|D_a|} \right) \right) \tag{11}$$

where $M(A/D) = M(A \cap D) \cup M(D)$ and $M(A) = \{a \in]0,1] \mid \exists x_i \in X \text{ such that } A(x_i) = a\}$ for all A.

The probabilistic method is defined as

$$GD_Q(A/D) = \sum_{a_i \in M(A/D)} (a_i - a_{i+1}) * Q\left(\frac{|(A \cap D)_{a_i}|}{|D_{a_i}|} \right) \tag{12}$$

where $M(A/D) = \{\alpha_1, ..., \alpha_m\}$ with $1=\alpha_1<\alpha_2 < ... <\alpha_m <\alpha_{m+1} =0$.

Some of the properties that we propose for every method (in particular (11) and (12) fulfils them) are the following:

Property 1. Crisp Case. If A and D are crisp, then the (known) result of the evaluation must be

$$Q\left(\frac{|A \cap D|}{|D|} \right)$$

Property 2. In the case D=X and for relative quantifiers, the resulting evaluation method is a valid method for the evaluation of type I sentences.

Property 3. Evaluation must be time-efficient (as much as possible), i.e. O(n) or O(nlogn).

Property 4. Evaluation must be coherent with fuzzy logic in the case of the universal quantifier. We are able to evaluate fuzzy logic quantified sentences with classical quantifiers ($\exists$ and $\forall$). In the case of $\forall$ the expected result is

$$\bigwedge_{x_i \in X} \left(D(x_i) \rightarrow A(x_i) \right)$$

using a t-norm for the intersection, and $\rightarrow$ being a fuzzy implication. In general, a conditional expression in the case of probabilistic evaluation is required.

Property 5. Evaluation must allow us to use any quantifier, i.e. any possibility distribution over [0,1].

The evaluation of quantified sentences is the basis for our definitions of fuzzy association rule and fuzzy quantified dependency.

3.2 ASSOCIATION RULES BASED ON FUZZY TRANSACTIONS

Association rules were introduced in Agrawal et.al. (1993) as empirical associations between items. Let I be a finite set of items, and let T be a set of transactions, each transaction in T being a subset of I. Let $A,B \subseteq I$. The association rule $A \Rightarrow B$ holds on T with support s and confidence c iff s% of the transactions in T contains both items A and B, and c% of the transactions that contains A also contains B. An example of a rule is ‘90% of people who buy bread also buy milk’. The confidence of the rule is 90%. The support of the rule is s iff ‘s% of people buy bread and milk’. The support measures the amount of data that agree with the rule, so measuring the importance of the rule. The confidence measures the degree of certainty/uncertainty of the rule. These are empirical measures obtained from data. In this case, our knowledge about the degree of association among items is restricted to what we can find in data.

Transactional databases are typical of some environments such as markets, and association rules were initially developed to deal with a set of market basket data transactions represented by a binary relation. In this relation, each attribute is an item, and the tuple that represents the transaction has value 1 in the attribute i_0 if the item i_0 is in the transaction, and 0 otherwise. In practice, every relation in a relational database can be transformed into a binary relation where the transactions are the tuples and the items are pairs (attribute, value).

Association rules can also be extended to obtain more accurate associations between items. Generalized and quantitative association rules perform a hierarchical partition of the domain of the attributes and obtain associations among items of the form (attribute, set of values), with better support and confidence. This technique is also used in the case of fuzzy functional dependencies as was discussed in subsection 2.1.3. We could think of rules as '90% of people who are old have high weight' defined over a relation where age and weight are numeric values. If a concrete partition is used to determine if a given value is "old", "young", etc... existing algorithms can deal with this problem using sharp boundaries between the concepts. However, it could be better to use linguistic labels for a better representation of the fuzzy concepts "old" and "high weight", so that the items (pair attribute-values) are in a given transaction with a degree $d \in [0,1]$. We call these transactions "fuzzy transactions" (fuzzy subsets of the set I of items). With this extension, not only uncertainty but also some imprecision has to be measured in the rules.

This approach is also a semantic solution to the problem of partitioning domains with many values. Previous approaches tried to obtain partitions of the domains of the attributes that provide rules with better support and confidence, but in many ocassions the obtained intervals have not a clear semantic interpretation. Moreover the partitioning process has to deal with two interactive problems called the minsup problem and the minconf problem. The problem is that if we use many small intervals in the partition we expect to obtain rules with high confidence but with very low support. On the contrary, if we use a few big intervals we expect to obtain rules with high support but with low confidence. A fuzzy partition with clear semantics allow us to forget these problems and to obtain better rules from the point of view of their understanding and semantic interpretation, by using fuzzy transactions.

This technique was used by Au and Chan (1998) to obtain fuzzy association rules using a measure of interestingness they call "adjusted difference", so they don't use the measures of confidence and support to determine whether a given rule is interesting or not. In the following definition, we extend the measures of confidence and support to the case of fuzzy transactions.

Definition 1: Let T be a set of fuzzy transactions and M be the fuzzy quantifier defined as $M(x)=x \ \forall x \in [0,1]$. Let A and B be two items, and let A_T and B_T be two fuzzy sets over T defined as follows: for every $t \in T$, $A_T(t)$ is the degree of the item A in the transaction t, and B_T is the degree of B in t. Then a fuzzy association rule $A \Rightarrow B$ holds on T with support s and confidence c, where s is the evaluation of the quantified sentence "M of T are $A_T \cap B_T$", and c is the evaluation of the quantified sentence "M of A_T are B_T".

As we said before, the items A and B are in a given transaction with a certain degree, so they can be interpreted as fuzzy sets over T. However, if $A = A_1A_2..A_n$, then the fuzzy set A_T is the fuzzy intersection of the fuzzy sets A_{1T}, A_{2T}, ..., A_{nT}. This is also valid for B_T. In the following we will use the same notation for the item and the fuzzy set, calling them A and B.

This definition is based on a method for evaluating quantified sentences. The evaluation can be performed in O(nlogn) with both methods ZS and GD, n being the number of transactions, and even in O(n) if the fuzzy sets A and B are arranged in nonincreasing order.

Lemma 1: In the crisp case, the measures of support and confidence of fuzzy association rules become the usual measures of support and confidence of association rules.
Proof: By property 1 in the crisp case (i.e. A and B are interpreted as subsets of T) the result of the evaluation of the sentence "Q of A are B" must be Q(p/q), where $p=|A \cap B|$ and $q=|A|$, so we have

$$\text{support} = M\left(\frac{|T \cap A \cap B|}{|T|}\right) = \frac{|A \cap B|}{|T|} \qquad \text{conf} = M\left(\frac{|A \cap B|}{A}\right) = \frac{|A \cap B|}{|A|} \tag{13}$$

Example 1: Table 1 shows an example of a relation (Age, Weight) who is mapped into a fuzzy item transaction using the sets of labels {young, medium, old, very old} for the age, and {light, medium, heavy, very heavy} for the weight. The possibility distribution of each label is subjective.

Age (years)	Weight (kgs.)
70	70
28	80
60	75
90	75
50	95

Table I:. Relation 'Age and Weight'

(Age,young)	(Age,med.)	(Age, old)	(Age, v.old)	(We., light)	(We.,med.)	(We., he.)	(We.,v.he.)
0	0	1	0.5	0	1	0.5	0
1	0.8	0	0	0	1	1	0
0	1	1	0	0	0.5	1	0
0	0	0.5	1	0	0.5	1	0
0	1	0.5	0	0	0.25	1	0.25

Table II:. Fuzzy transactions corresponding to the tuples of table I

Some rules that hold in the relation are the following, where the method GD of equation (12) has been used:

Rule	support	conf
(Age, Medium) ⇒ (Weight, Medium)	0.31	0.516
(Age, Medium) ⇒ (Weight, Heavy)	0.56	1
(Age, Old) ⇒ (Weight, Medium)	0.45	0.687
(Age, Old) ⇒ (Weight, Heavy)	0.5	0.75

Table III:. Fuzzy association rules obtained from table II

Another possible extension to association rules can be defined by adding a possibility value to each transaction, in the spirit of the fuzzy relational models that define a fuzzy relation as a fuzzy subset of tuples. This is the case of fuzzy functional dependencies in subsection 2.1.1. We could imagine of an association of the type "90% of people who buy bread early in the morning also buy milk". In this example the hour of each transaction could be matched against the fuzzy linguistic label "early in the morning", so that every transaction is associated a degree of fulfilment of the concept "early ...". Then we have a fuzzy set of (fuzzy or concrete) transactions. In this case, the support is obtained by evaluating the sentence "M of T_f are $A \cap B$" and the confidence is obtained by evaluating "M of $A \cap T_f$ are B", where T_f is the fuzzy subset of transactions. If both the transactions and the set of transactions are not fuzzy, the expressions for support and confidence become (13).

4 FUZZY QUANTIFIED DEPENDENCIES

Let R be a relational scheme, let r be a relation over R and let r_f be a fuzzy subset of r (i.e. a fuzzy relation, in the sense of a fuzzy subset of tuples). Let $r_f(t)$ be the degree of belonginess of the tuple t to the fuzzy relation r_f. Let S_R be a set of fuzzy similarity relations, $S_R=\{S_X \mid X \in R\}$. We define the fuzzy relation r' over the scheme R' as follows:

i) $R'=\{X' \mid X \in R\}$, $D_{X'}=[0,1]\ \forall X' \in R'$

ii) For each pair of distinct an nonordered tuples {t,s} in r we have a tuple ts in r' defined as follows:

$$ts[X'] = S_X(t[X], s[X]) \quad \forall X' \in R' \tag{14}$$

$$r_f'(ts) = min(r_f(t), r_f(s)) \quad \forall ts \in r' \tag{15}$$

iii) Let n=| r |. As a consequence of ii), n'=| r' | is

$$n' = \binom{n}{2} = \frac{n(n-1)}{2} \tag{16}$$

Definition 2: Let Q be a fuzzy quantifier and let X,Y∈R. Then, the (Q,S_R) fuzzy quantified dependency X→Y holds in r_f with degree c∈[0,1], where c is the result of the evaluation of the quantified sentence

$$Q \quad of \quad X' \cap r_f' \quad are \quad Y' \tag{17}$$

and X', r_f' and Y' are interpreted as fuzzy sets over r'. The general semantic of the definition is that c is the degree at which Q of the pairs of tuples belonging to r' that are similar in X verify that they are also similar in Y, where the similarity is given by the similarity relations S_X and S_Y. The extension of the functional dependency defined depends on: the quantifier Q, the set of similarity relations S_R, and the method of evaluation of quantified sentences.

We shall see that this definition covers and merges some existing approaches of extension of functional dependency. A very important remark is that we can design algorithms to perform the evaluation of the sentences in time O(n), although n' is O(n^2), see Delgado et.al. (1999a).

4.1 INTERPRETATION OF OTHER EXTENDED FUNCTIONAL DEPENDENCIES

4.1.1 Fuzzy relations

By property 4, when Q=∀ a t-norm and a fuzzy implication exists so that the evaluation of "Q of A are B" can be expressed as $\wedge_{t\in T}$ (A(t)$\rightarrow_f$ B(t)), or as a global conditional of A with respect to B.
Let the values of the attributes be concrete and let the similarity relation be the classical equality for every X∈R (i.e. S_X(a,b)= 1 if a=b, 0 otherwise), and let r be a fuzzy relation. In this case, we call S_R=$S_=$. Then we have that the (∀,$S_=$) fuzzy quantified dependency X→Y is the fuzzy functional dependency of A. Kiss (see subsection 2.1.1.).
Proof: Let H be a quantified sentence and let Eval(H) be the evaluation of the sentence. When evaluating (17) with Q=∀, we have that the accomplishment degree of the dependency is

$$Eval(\forall\ of\ \ X' \cap r_f'\ are\ Y') = min_{ts\in r'}\left(\left(X' \cap r_f'\right)(ts) \rightarrow Y'(ts)\right) = \ min_{ts\in r'}\left(min\left(r_f'(ts), X'(ts)\right) \rightarrow Y'(ts)\right) =$$
$$min_{ts\in r'}\left(min\left(r_f(t), r_f(s), X'(ts)\right) \rightarrow Y'(ts)\right) = \ min_{ts\in r'}\left(min\left(r_f(t), r_f(s), S_=\left(t[X], s[X]\right)\right) \rightarrow S_=\left(t[Y], s[Y]\right)\right) \ \blacklozenge$$

4.1.2 Relaxation of the equality

This extension is covered by means of the similarity relations. The expression in equation (3) is obtained by using property 4 when Q=∀.
Proof: If the tuples are not fuzzy then

$$Eval(\forall\ of\ \ X' \cap r_f'\ are\ Y') = Eval(\forall\ of\ \ X'\ are\ Y') = \bigwedge_{ts\in r'}\left(X'(ts) \rightarrow Y'(ts)\right) = \ \wedge_{t,s\in r}\ S_X\left(t[X], s[X]\right) \rightarrow S_Y\left(t[Y], s[Y]\right)$$

♦

4.1.3 Incorporation of imprecise data

This extension can be interpreted as a fuzzy association between attribute values (possibly imprecise) rather than a dependency between attributes. The difference with respect to fuzzy association rules is that the quantifier is ∀ instead of M. However as pointed out in Bosc et.al.(1997), if the rule holds it is easy to show that any pair of tuples that are similarly coherent with respect to F in the attribute X are similarly coherent with respect to G in the attribute Y. Let $S_{X(F)}$ be a similarity relation in D_X with respect to the label F defined as $S_{X(F)}$(a,b)=min(F(a),F(b)). Also let $S_{Y(G)}$ be a similarity relation in D_Y with respect to the label G defined as $S_{Y(G)}(c,d)=\min(G(c),G(d))$. We propose the extension in this case to be a fuzzy quantified dependency $(\forall, S_{X(F)Y(G)})$ where $S_{X(F)Y(G)} = \{ S_{X(F)}, S_{Y(G)}\}$. It is clear that the dependency will take into account only those tuples that are coherent with the labels F and G in the attributes X and Y, so the basic interpretation of the dependency is induced by an association among attributes rewriting F on X and G on Y. The following table shows the result of the evaluation of this dependency over the data in table II. The labels in F and G are the same used in example 1.

F.Q. Dependency (Age,F)→(Weight,G)	c
(Age, Medium) → (Weight, Medium)	0.25
(Age, Medium) → (Weight, Heavy)	1
(Age, Old) → (Weight, Medium)	0.25
(Age, Old) → (Weight, Heavy)	0.5

Table IV:. Some fuzzy quantified dependencies $(\forall, S_{Age(F)Weight(G)})$ Age→Weight obtained from table II

4.1.4 Imprecise domains

We cover this extension by defining adequate resemblance relations using (5), (6), (7) or (8) between imprecise values of attributes. If a degree of resemblance φ is used as in (10), then we have an φ-cut of a resemblance relation, that is another resemblance relation. The quantifier is, however, ∀. Let $S_{Y\varphi}$ be the φ-cut of a resemblance relation S_Y and let $S_X=S_=$. Then we have that both the quantifier and the resemblance relation are crisp, and that the relation is also crisp, so the evaluation of the quantified sentence is a value in {0,1}.
As concrete extensions, the fuzzy quantified dependency $(\forall,\{S_= , S_{Y\varphi}\})$ X→Y is the fuzzy functional dependency of Cubero defined by (10).

4.1.5 Relaxation of the universal quantifier. Approximate dependencies.

The relaxation of the quantifier is naturally performed by choosing the appropriate value of Q. As a particular case, the $(M, S_=)$ fuzzy quantified dependency is the approximate dependency defined in Delgado et.al. (1999a). When using M, we have that c is the confidence of a fuzzy association rule over the set of pairs of tuples. As we use $S_=$, we have the measures of support and confidence over the set of pairs of tuples that define the mentioned approximate dependency. In Delgado et.al. (1999a), this is precisely the definition of an approximate dependency.

A fuzzy quantified dependency (M,S_R) over r can be interpreted as a fuzzy association rule over r'. The measure of accuracy of a given fuzzy quantified dependency X→Y is then the measure of confidence of the fuzzy association rule X'⇒Y'. In the same way, we could give a measure of the support of the dependency as the evaluation of the sentence

$$M \quad of \quad r' \quad are \quad X' \cap Y' \tag{18}$$

That is, under the same assumptions than before, the measure of support of our approximate dependency (see Delgado et.al. (1999a) again).

A discussion arises here about the convenience of the use of the measure of support for every fuzzy quantified dependency. The measure of confidence takes into account by definition the support of the rule, but we think that the value of support is necessary because we are mining empirical associations between attributes based on empirical associations between attribute values.
One result described in Delgado et.al. (1999a) is that the measures of support and confidence of the approximate dependencies can be obtained from the measures of support and confidence of the association rules that describe the approximate dependency. This fact has been used to propose a meaningful interpretation of the measures of confidence and support of our approximate dependency.

5 CONCLUSIONS AND FUTURE WORKS

We have given a general definition of associations among attributes based on associations among attribute values. This definition allows us defining concrete extensions to the concept of functional dependency using the semantics of a fuzzy quantifier and a similarity/resemblance relation. Some of the existing extensions have been shown to be particular cases of this general definition. Fuzzy quantified dependencies merges imprecision and uncertainty in a semantically intuitive way to offer a final degree of fulfilment of the dependency. They are not intended for design but to provide empirical information about the structure of data.

We have also proposed a definition of fuzzy association rule based on fuzzy items and/or fuzzy transactions. Both associations are based on the use of quantified sentences evaluation methods.

Future works will be to study some of the particular definitions that could arise from distinct values of the quantifier and the resemblance relations, and to develop algorithms for mining fuzzy quantified dependencies where the values of the quantifier and the resemblance relations could be defined by the user.

REFERENCES

- Agrawal, R., Imielinski, T., Swami, A., 1993, "Mining Association Rules between Sets of Items in Large Databases." Proc. of the 1993 ACM SIGMOD Conf., Washington DC, USA.
- Au, W.H., Chan, K.C.C., 1998, "An Effective Algorithm for Discovering Fuzzy Rules in Relational Databases." Proc. IEEE Int. Conf. on Fuzzy Systems vol. 2, pp. 1314-19
- Bosc, P., Dubois, D., Prade, H., 1994, "Fuzzy Functional Dependencies – An Overview and a Critical Discussion –", Proc. IEEE Int. Conf. on Fuzzy Systems, Orlando/FL, USA, p.p. 325-330
- Bosc, P., Lietard, L., Pivert, O., 1997, "Functional Dependencies Revisited Under Graduality and Imprecision", 1997 Annual Meeting of NAFIPS, p.p. 57-62.
- Cubero, J.C., Medina, J.M., Vila, M.A., 1993, "Influence of Granularity Level in Fuzzy Functional Dependencies", Proc. ECSQARU'93, LNCS Vol. 747, Springer Verlag, p.p. 73-78
- Cubero, J.C., Vila M.A., 1994, "A New Definition of Fuzzy Functional Dependency in Fuzzy Relational Databases", Int. Journal on Intelligent Systems 9 (5), p.p. 441-448.
- Cubero, J.C., Pons, O., Vila, M.A., 1994, "Weak and Strong Resemblance in Fuzzy Functional Dependencies", Proc. IEEE Int. Conf. on Fuzzy Systems, Orlando/FL, USA, p.p. 162-166
- Cubero, J.C., Medina, J.M., Pons, O., Vila, M.A., 1996, "Discovering Fuzzy Functional Dependencies in Databases", Proc. of EUFIT'98, Aachen, Germany, p.p. 811-821.
- Cubero, J.C., Medina, J.M., Pons, O., Vila, M.A., 1998, "Fuzzy Lossless Decompositions in Databases", Fuzzy Sets and Systems 97 p.p. 145-167.
- Cubero, J.C., Cuenca, F., Blanco, I., Vila, M.A., 1998, "Incomplete Functional Dependencies versus Knowledge Discovery in Databases", Proc. of EUFIT'98, Aachen, Germany, p.p.
- Delgado, M., Sanchez, D., Vila, M.A., 1999, "Mining Approximate Dependencies Using Association Rules Measures". Submitted to IDA-99.
- Delgado, M., Sanchez, D., Vila, M.A., 1999, "Fuzzy Cardinality Based Evaluation of Quantified Sentences". Submitted to Int. Journal of Approximate Reasoning.
- Huhtala, Y., Karkkainen, J., Porkka, P., Toivonen, H., 1998, "Efficient Discovery of Functional and Approximate Dependencies using Partitions", Proc. of the 14th Int. Conference on Data Engineering, p.p. 392-401.
- Kivinen J., Mannila H., 1995, "Approximate Dependency Inference from Relations". Theoretical Computer Science 149(1) , p.p. 129-149.
- Mannila, H., Räihä, K.-J., 1994, "Algorithms for Inferring Functional Dependencies", Data & Knowledge Engineering 12(1), p.p. 83-99.
- Pfahringer, B., Kramer, S., 1995, "Compression-Based Evaluation of Partial Determinations", KDD-95 Proceedings, p.p. 234-239
- Piatetsky-Shapiro, G., Matheus, C.J., 1993, "Probabilistic Data Dependencies". Proc. of the 1993 Workshop in Machine Learning, Aberdeen, Scotland.
- Ziarko, W., 1991, "The Discovery, Analysis, and Representation of Data Dependencies in Databases", Piatetsky-Shapiro, G., Frawley, W.J. (eds.): Knowledge Discovery in Databases. AAAI Press/ The MIT Press, p.p. 195-209.