

# Basis of a Fuzzy Knowledge Discovery System

Maurice Bernadet

IRIN / Ecole Polytechnique de l'Université de Nantes  
Rue Christian Pauc, La Chantrerie  
BP 60601, 44306 Nantes Cedex 3, France  
Tel: (+33 2)/(02) 40 68 30 00  
Fax: (+33 2)/(02) 40 68 30 66  
E-mail: Maurice.Bernadet@irin.univ-nantes.fr

**Abstract.** Considering a fuzzy knowledge discovery system we have realized we describe here the main features of such systems. First, we consider possible methods to define fuzzy partitions on numerical attributes in order to replace continuous or symbolic attributes by fuzzy ones. We explain then how to generalize statistical indexes to evaluate fuzzy rules, detailing a special index, the intensity of implication and its generalization to fuzzy rules. We describe then one algorithm use to extract fuzzy rules. Since many fuzzy operators are available, we propose a method to choose one fuzzy conjunction, one fuzzy implication and one fuzzy aggregation, and we explain how this choice may be validated by comparing the results of the Generalized Modus Ponens applied on the premises of the examples to the effective conclusions in the database. To reduce the important number of fuzzy rules extracted, we consider also some methods to aggregate fuzzy rules, showing that usage of classical reduction schemes requires specific choices of fuzzy operators.

## 1 Introduction

Knowledge discovery has been defined as "The non-trivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data" [9]. To realize a set of knowledge discovery tools, one has some major choices to achieve: the discovery process may be supervised or not, knowledge representation may use decision trees, association rules, neural networks, ... Having to design such tools, we have chosen, for simplicity reasons, a supervised process and a knowledge representation by rules. However, considering that rules using intervals on continuous attributes are often difficult to interpret and that strict thresholds are often too abrupt, we have decided to use fuzzy logics.

Fuzzy logics may be considered as extensions of multivalued logics, allowing usage of intermediate truth-values between false and true [18]. They allow expression of knowledge in a more natural way than classical binary logics, using graduated attributes as in "X is rather high" (for "X is high" is rather true) ... Fuzzy logics offer many logical operators [12], which permits a good expressiveness of various knowledge forms. In this paper, we detail the primary operations needed to extract fuzzy knowledge from a database.

First, usage of fuzzy logics in knowledge discovery needs to convert numerical attributes to their fuzzy representations; for this, it is necessary to define for each

classical attribute, a mapping from its possible values to a set of truth-values for each fuzzy attribute. This mapping is often realized by a fuzzy partition, or rather by a fuzzy pseudo partition, and it is then possible to translate classical attributes by valuations on their fuzzy correspondents. These operations are called fuzzification.

To extract rules from a database, one needs to evaluate each possible rule in order to establish which rules must be kept; for this purpose, many indexes are available, of which we have only retained three indexes: the confidence of a rule, its support and a less usual index, called the intensity of implication. After recalling the principles of these indexes, we expose how they can be evaluated in fuzzy logics.

It is then possible to use a knowledge extraction algorithm using the same principles as in classical logics. Our algorithm is an exploratory search in a tree of possible rules, with evaluation of each rule. Fuzzy logics also allow specific methods, based on genetic algorithms, to search the most representative set of weights for a general set of fuzzy rules, but we will not consider this possibility here.

Several evaluations of the fuzzy logical operators are possible. If one only wants to extract rules for a human expert, the nature of the operators does not matter, but if these rules are to be processed by an expert system, a choice of fuzzy operators is necessary. To find the most adequate set of fuzzy operators, we expose a justified restriction to only four possible sets of fuzzy operators and we give a method to find amongst them, the more consistent with a database.

To reduce the huge set of rules that can then be extracted, one may want to use classical reduction schemes. We show that to be valid in a fuzzy logic, classical reduction schemes need specific choices of fuzzy operators. We conclude by recalling the interest to use fuzzy attributes instead of numerical intervals for continuous attributes in the database, and by considering some possible improvements of the systems we have described.

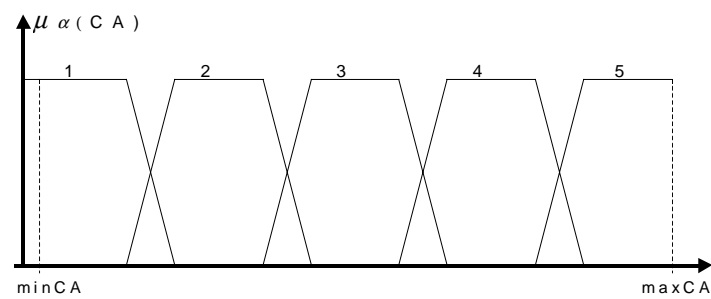
## 2 The Process of “Fuzzification”

### 2.1 Definition of Fuzzy Partitions

Let us first recall that fuzzy logics evaluate the truth-value of a fuzzy proposition “ $X$  is  $A$ ”, as the degree to which  $X$  belongs to the fuzzy set  $A$ :  $Truth("X \text{ is } A") = \mu_A(X)$ ,  $\mu_A(X)$  being the membership (or characteristic) function of the fuzzy set  $A$ .

Fuzzy sets allow definition of fuzzy C-partitions or "pseudo partitions" in which each value of a continuous attribute may be classified into several fuzzy classes, with a total membership of 1. These fuzzy pseudo partitions allow conversion of continuous attributes into fuzzy ones, giving then the truth-value of fuzzy propositions. For a continuous attribute  $CA$ , varying from  $\min CA$  to  $\max CA$ , one can define a fuzzy pseudo partition several ways ([5], [13]).

The simplest method divides the interval  $[\min CA, \max CA]$  in  $n$  sub-intervals, with a small percentage of coverage between two adjacent ones, and to give each sub-interval a symbolic name related to their position; for instance one may divide the interval  $[\min CA, \max CA]$  in 5 sub-intervals with an overlap of about 20%, giving then 5 fuzzy attributes, attributes such as: *strong negative*, *rather negative*, *medium*, *rather positive* and *strong positive* (Fig. 1).



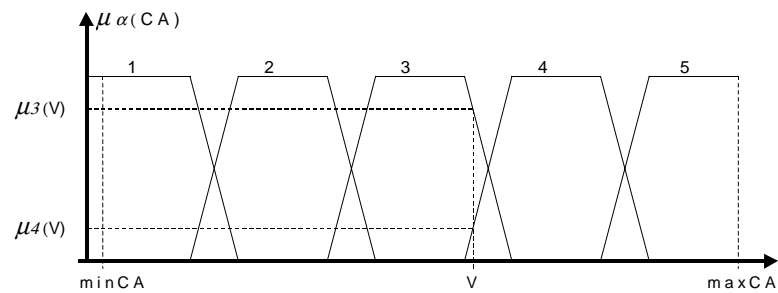
**Fig. 1.** A fuzzy C-partition ( $\alpha = 1$ , strong negative;  $\alpha = 2$ , rather negative;  $\alpha = 3$ , medium;  $\alpha = 4$ , rather positive;  $\alpha = 5$ , strong positive).

The fuzzy classes may also be defined by the experts; otherwise one may propose 3 or 5 classes as standard options. Different numbers of classes may also be used, but a too high number of classes risks to heavily slow down the knowledge discovery process.

Another kind of method extracts the number of classes and defines the fuzzy C-classes from the database. It considers the values of the attributes giving the same conclusion and, when possible, cluster these values in the same fuzzy sets, with a membership value equal to the rate of samples that give this conclusion. These methods often use histograms of the values of the attributes for each possible conclusion. Moreover, it is possible to conceive a more satisfactory method by generalizing to fuzzy logics optimal discretization methods such as those studied in [20].

**2.2 Fuzzification of a Database**

Once the fuzzy classes have been defined for each continuous attributes, one may convert the related value of each item, by mapping this value to the membership values of each fuzzy class defined for the corresponding classical attribute (Fig. 2).



**Fig. 2.** Mapping from the value V of the continuous attribute CA into membership values of fuzzy attributes (here, only  $\mu_3$  and  $\mu_4$  are non zero).

### 3 Evaluation of Fuzzy Rules

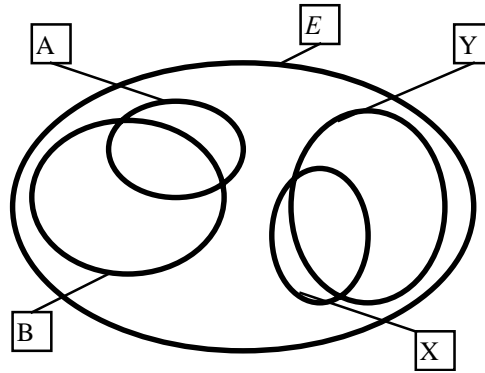
#### 3.1 Indexes to Evaluate Classical Rules

Several indexes may be used to evaluate classical rules, and we have chosen three of them: the confidence, the support and a less usual index, the intensity of implication.

*The confidence* of a rule "if  $a$  then  $b$ " expresses the conditional probability of  $b$  when  $a$  is true; it may be evaluated by  $n_{a \wedge b} / n_a$ , calling  $n_{a \wedge b}$  the number of items verifying  $a$  and  $b$  and  $n_a$  the number of items verifying  $a$ .

*The support* of a rule "if  $a$  then  $b$ " may be defined as the rate of occurrences of items verifying  $a$  and  $b$  related to all items of the database; so, the support of the rule is  $n_{a \wedge b} / nE$ , with  $n_{a \wedge b}$  the number of items verifying  $a$  and  $b$  and  $nE$  the total number of items in the database.

*The intensity of implication* is an index expressing the quality of a rule. This index, defined by R. Gras and A. Larher [8], is based on simple probability concepts: since the cardinalities of two subsets  $A$  and  $B$  are determined by the objects of the database belonging to  $A$  and  $B$ , we consider two random subsets  $X$  and  $Y$  having respectively the same cardinalities as  $A$  and  $B$ . The implication  $a \Rightarrow b$  is characterized by the relation  $A \subset B$  and its counter examples are associated to the subset  $A \cap \bar{B}$ . We compare the cardinality of  $A \cap \bar{B}$  (given by the database) with the random variable given by the cardinality of  $X \cap \bar{Y}$ , supposing that there is no link between  $X$  and  $Y$  (Fig. 3). If the cardinality of  $A \cap \bar{B}$  is unusually small compared to the expectation of the distribution on the cardinalities of  $X \cap \bar{Y}$ , we accept "if  $a$  then  $b$ " as a rule.



**Fig. 3.**  $X$  and  $Y$  vary randomly in  $E$ .

The intensity of  $a \Rightarrow b$  is therefore the complement to one of the probability for the random variable "cardinality of  $X \cap \bar{Y}$ " to be smaller than cardinality of  $A \cap \bar{B}$ . It may be defined by

$$\varphi(a, \bar{b}) = 1 - P[\text{Card}(X \cap \bar{Y}) \leq \text{Card}(A \cap \bar{B})] \quad (1)$$

Calling  $n = \text{Card}(E)$ ,  $n_a = \text{Card}(A)$ ,  $n_{\bar{a}} = \text{Card}(\bar{A})$ ,  $n_b = \text{Card}(B)$ ,  $n_{\bar{b}} = \text{Card}(\bar{B})$ ,  $n_{a \wedge b} = \text{Card}(A \cap B)$ ,  $n_{a \wedge \bar{b}} = \text{Card}(A \cap \bar{B})$ , the random variable  $\text{Card}(X \cap \bar{Y})$  obeys an hyper geometric distribution [7]:

$$P[\text{Card}(A \cap \bar{Y}) = k] = \frac{C_{n_a}^k \cdot C_{n-n_a}^{n-n_b-k}}{C_n^{n-n_b}} = \frac{C_{n_a}^k \cdot C_{n_{\bar{a}}}^{n_{\bar{b}}-k}}{C_{n_{\bar{b}}}^{n_{\bar{b}}}}. \quad (2)$$

and

$$P[\text{Card}(X \cap \bar{Y}) \leq \text{Card}(A \cap \bar{B})] = \sum_{\substack{i=0 \\ i \geq n_a - n_b}}^{\text{Card}(A \cap \bar{B})} \frac{C_{n_a}^i \cdot C_{n_{\bar{a}}}^{n_{\bar{b}}-i}}{C_{n_{\bar{b}}}^{n_{\bar{b}}}}. \quad (3)$$

The intensity of implication has interesting properties [6]. First, its value increases with the size of the learning set, while other indexes stay constant; some new counter-examples for a strong implication do not change much its value, but progressively doubts come, and finally a few more counter-examples cause its fall. This index is also well adapted to noisy data since a small number of counter-examples does not necessary invalidate the implication; it also proscribes rules such as  $a \Rightarrow b$  when proposition  $b$  is true for nearly all examples of the learning set (since it is not surprising then that nearly all examples with  $a$  true are examples with  $b$  true).

### 3.2 Adaptation of Classical Indexes to Fuzzy Rules

Two categories of indexes evaluate fuzzy rules; indexes based exclusively on fuzzy set theory ([1], [13]) or indexes developed in classical logic and generalized to fuzzy knowledge ([14], [16], [19]). Choosing this second class, we have generalized the three indexes for classical rules by applying a concept from fuzzy probability of fuzzy events [17], for which the number of elements that satisfy a proposition  $p$  associated to a fuzzy set  $P$  with membership function  $\mu_p$ , is the crisp cardinality of the fuzzy set  $P$ :  $\text{Card}(P) = \sum_{x \in E} \mu_p(x)$ .

For a crisp implication  $a \Rightarrow b$  on two fuzzy propositions  $a$  and  $b$ , linked to fuzzy sets  $A$  and  $B$  with membership functions  $\mu_a$  and  $\mu_b$ , we generalize the three indexes by using cardinalities of fuzzy sets instead of cardinalities of crisps sets [2]. With a fuzzy conjunction  $T$  (T-norm) and a fuzzy complement  $\mu_{\bar{A}}(x) = 1 - \mu_A(x)$ , one can write:

$$\begin{aligned} n_A &= \text{Card}(A) = \sum_{x \in E} \mu_A(x), & n_{\bar{A}} &= \text{Card}(\bar{A}) = \sum_{x \in E} \mu_{\bar{A}}(x) = \sum_{x \in E} (1 - \mu_A(x)), \\ n_{\bar{B}} &= \text{Card}(\bar{B}) = \sum_{x \in E} \mu_{\bar{B}}(x) = \sum_{x \in E} (1 - \mu_B(x)), \\ n_{A \cap B} &= \text{Card}(A \cap B) = \sum_{x \in E} \mu_{A \cap B}(x) = \sum_{x \in E} T(\mu_A(x), \mu_B(x)), \\ n_{A \cap \bar{B}} &= \text{Card}(A \cap \bar{B}) = \sum_{x \in E} \mu_{A \cap \bar{B}}(x) = \sum_{x \in E} T(\mu_A(x), (1 - \mu_B(x))) \end{aligned}$$

The confidence of a rule, its support and its intensity of implication are then expressed by the same formulas as above, but they use these cardinalities of the fuzzy sets instead of the cardinalities of crisp sets.

#### 4. A Knowledge Extraction Algorithm

Our algorithm uses a depth first strategy to evaluate all the rules that can be constructed from a set of propositions. To limit the number of rules to evaluate, we restrict the number of propositions in the premise of a rule; thus, we use 4 thresholds  $\alpha, \beta, \gamma, \delta$  and one rule is kept if its confidence  $C$  is greater than  $\alpha$ , its support  $S$  is greater than  $\beta$ , its intensity of implication  $I$  over  $\gamma$  and if the length  $L$  of its premise has at most  $\delta$  propositions.

Let us call  $E = \{e_1, e_2, \dots, e_n\}$  the learning set,  $R$  the set of rules retained,  $P = \{p_1, p_2, \dots, p_n\}$  the set of propositions describing the examples in  $E$ ,  $P'$  the set of propositions associated to the possible conclusions,  $D = \{a_1, a_2, \dots, a_m\}$  the set of attributes in the possible propositions of the premise,  $F_{decision}$  the fuzzy partition associated to the attribute of the classifying decision; the algorithm is then:

Algorithm: Knowledge Extraction

```

(1)  $R = \emptyset$ 
(2) For all values  $v_i \in F_{decision}$  do
(3)   Let  $T$ , the tree of rules in  $P$  with conclusion
       $\{a_{decision} = v_i\}$ 
(4)   Let  $B$ , the set of observations in  $E$  with proposition
       $\{a_{decision} = v_i\}$  true
(5)   CurrentNode = Forward( $T, R$ )
(6)   While the tree  $T$  has not been totally searched do
(7)     Let  $r: Premise \rightarrow p_i \in P'$ , the rule associated to
      CurrentNode
(8)     Let  $A$ , the set of examples in  $E$  for which Premise
      is true
(9)     Compute  $C, S, I$  from the cardinalities of the sets
       $E, A$  and  $B$ 
(10)    Let  $L$ , the length of Premise
(11)    If  $(C \geq \alpha)$  and  $(S \geq \beta)$  and  $(I \geq \gamma)$  and  $(L \leq \delta)$ 
(12)      Then  $R = R \cup \{Premise \rightarrow p_i\}$ 
(13)    End If
(14)    If  $(S < \beta)$  or  $(L > \delta)$  or (CurrentNode terminal)
(15)      Then CurrentNode = Backward( $T$ , CurrentNode)
(16)      Else CurrentNode = Forward( $T$ , CurrentNode)
(17)    End If
(18)  End While
(19)End For
End.
```

## 5. The Choice of Fuzzy Operators

The knowledge extraction algorithm highlights a set of interesting rules, but needs another mechanism to evaluate the fuzzy implications. Fuzzy operators accept many possible definitions [12]. A fuzzy conjunction (fuzzy "and") may be chosen inside several classes of T-norms, such as:

$$\begin{aligned} \text{Zadeh's minimum:} \quad & \mu_{a \cap b}(x) = \min(\mu_a(x), \mu_b(x)) \\ \text{probabilistic intersection:} \quad & \mu_{a \cap b}(x) = \mu_a(x) * \mu_b(x) \\ \text{Lukasiewicz's intersection:} \quad & \mu_{a \cap b}(x) = \max(\mu_a(x) + \mu_b(x) - 1, 0) . \end{aligned}$$

As a fuzzy implication, one may use:

$$\begin{aligned} \text{Reichenbach's} \quad & \mu_{a \Rightarrow b}(x, y) = 1 - \mu_a(x) + \mu_a(x) * \mu_b(y) \\ \text{Zadeh's} \quad & \mu_{a \Rightarrow b}(x, y) = \max(1 - \mu_a(x), \min(\mu_a(x), \mu_b(y))) \\ \text{Kleene-Dienes' } \quad & \mu_{a \Rightarrow b}(x, y) = \max(1 - \mu_a(x), \mu_b(y)) \\ \text{Lukasiewicz's} \quad & \mu_{a \Rightarrow b}(x, y) = \min(1, 1 - \mu_a(x) + \mu_b(y)) \\ \text{Gödel-Brouwer's} \quad & \mu_{a \Rightarrow b}(x, y) = 1 \quad \text{if } \mu_a(x) \leq \mu_b(y) \\ & \mu_{a \Rightarrow b}(x, y) = \mu_b(y) \quad \text{otherwise} \end{aligned}$$

However, since the extracted rules can be used in an expert system by application of the generalized modus ponens (G.M.P.), our choices must be coherent with the operators chosen then. Let us recall that the G.M.P. is the following inference scheme:

$$\begin{array}{c} \text{If } X \text{ is } A \quad \text{then } Y \text{ is } B \\ \quad \quad \quad X \text{ is } A' \\ \hline \quad \quad \quad Y \text{ is } B' \end{array} \quad \begin{array}{l} \text{And for one T-norm } T \text{ and the implication} \\ \mu_{a \Rightarrow b}(\mu_a(x), \mu_b(y)) = I(\mu_a(x), \mu_b(y)) , \\ \text{one has } \mu_{b'}(y) = \sup_{x \in A'} T(\mu_a(x), I(\mu_a(x), \mu_b(y))) . \end{array}$$

A comparative study of fuzzy implication operators [11] has shown that the generalized modus ponens gives very good results with four combinations: Lukasiewicz's implication and bold intersection, Kleene-Dienes' implication and minimum, Kleene-Dienes' implication and bold intersection, Gödel-Brouwer's implication and bold intersection. So we have limited our trials to these four combinations. Our algorithms also need an aggregation operator; and since we want an averaging evaluation of the implication and a mechanism to exclude abnormal records, we have chosen the arithmetic mean, which allows usage of standard deviations.

The algorithm to compute the fuzzy implication for each rule highlighted by the knowledge extraction algorithm is given below (more details are given in [3]). Each rule may be composed of one conjunction of fuzzy propositions in premise and one fuzzy proposition in conclusion, and the algorithm prospects a random sample of the database to compute the number of examples and counter examples for each couple (x, y) of the implication, then it evaluates the rate of good examples. If the rate of good examples satisfies the expert, the set of fuzzy operators is kept; otherwise, another set of operators is tried.

Algorithm: Evaluation of a set of fuzzy operators

- 1) For each example:
  - Evaluate the premise's fuzzy value using the T-norm,
  - Compute the values of the selected implication.
- 2) Compute the arithmetic mean  $\eta(x, y)$  and the standard deviation  $\sigma(x, y)$  for each pair  $(x, y)$ , calling
  - $x$ , one truth-value of the premise,
  - $y$ , one truth-value of the conclusion.
- 3) For each example:
  - If the value of its implication for the pair  $(x, y)$  fits  $[\eta(x, y) - 2 * \sigma(x, y), \eta(x, y) + 2 * \sigma(x, y)]$  (interval of confidence of 95% for a normal law),
    - Note it as one positive example for  $(x, y)$ ,
  - Else
    - Note it as one negative example.
- 4) The values of the implication are then given by the arithmetic means of the good examples for each couple  $(x, y)$  ( $x$  for the premise,  $y$  for the conclusion).
- 5) Evaluate the adequacy of the implication:
  - For each positive example of the sample,
  - Apply the Generalized Modus Ponens,
  - Compute the distance between the truth-values of the inferred conclusions and the observation,
  - If this distance is greater than a threshold chosen by the operator,
  - add the example to the set of records for which the rule is inadequate
  - Else add the example to the set of records for which the rule is correct
- 6) The rates of good examples are then given by
 
$$\rho(x, y) = n_+(x, y) / (n_+(x, y) + n_-(x, y))$$

End.

These algorithms were processed on databases of the UCI KDD Archive; they often gave good results, but sometimes none of the possible fuzzy implication proved really adequate. For these cases, we have proposed a statistical evaluation of the conclusions given the truth-values of the consequent ([4]).

## 6. Reduction of Fuzzy Rules

In some applications, rules are not extracted in order to build expert systems, but to give human experts a synthetic view of the database records; then, since the number of rules extracted is often high, we have studied methods to aggregate rules.

In classical logics, one may write  $(a \Rightarrow c) \wedge (b \Rightarrow c) \vdash (a \vee b) \Rightarrow c$ , and we wanted to process similarly with fuzzy rules without having to reevaluate the fuzzy rule  $(a \vee b) \Rightarrow c$ . So, we wished to write:  $\mu_{(a \Rightarrow c) \wedge (b \Rightarrow c)}(x, y) = \mu_{(a \vee b \Rightarrow c)}(x, y)$  or  $\mu_{(a \Rightarrow c)}(x, y) \wedge \mu_{(b \Rightarrow c)}(x, y) = \mu_{(a \vee b \Rightarrow c)}(x, y)$ , using fuzzy operators chosen among the sets defined above. Therefore, we had to find which, if any, of the above operators

sets allow to write the condensed form:  $\mu_{I(\alpha, \gamma)} T \mu_{I(\beta, \gamma)} = \mu_{I(\alpha C \beta, \gamma)}$ ,  $T$  being a T-norm,  $C$  its complementary T-co-norm, and noting  $\mu_{I(\alpha, \gamma)} = \mu_{(a \Rightarrow c)}(x, y)$ ,  $\alpha = \mu_a(x, y)$ ,  $\beta = \mu_b(x, y)$  and  $\gamma = \mu_c(x, y)$ . We have proved [4] that the Kleene-Dienes' implication, associated with *min* for T-norm and *max* for T-co norm is the only solution in the four sets of fuzzy operators. Similarly, to keep the classical reduction scheme  $(a \Rightarrow c) \wedge (a \Rightarrow c) \vdash a \Rightarrow b \wedge c$  or, with the above condensed notations  $\mu_{I(\alpha, \gamma)} T \mu_{I(\alpha, \gamma)} = \mu_{I(\alpha, \beta T \gamma)}$ , we have shown that the same set of fuzzy operators must be used.

Adaptations of classical reduction schemes to fuzzy logics must be studied further, but we give these results to show that fuzzy rules often may not be handled as classical rules.

## 7. Conclusion

We have described here a generalization of knowledge discovery mechanisms to fuzzy logics; first, one needs to fuzzify the continuous attributes of the database. One needs then indexes to evaluate possible rules, and we have generalized to fuzzy logics three indexes: the confidence, the support and a less usual index, the intensity of implication. We have given a KDD algorithm that can be used to extract fuzzy rules.

Since fuzzy logics allow usage of numerous operators, we have justified their restriction to four sets and we have developed a method to choose on set of operators, computing the rates of good examples by applying the Generalized Modus Ponens on a sample extracted from the database.

Finally, to reduce the number of rules proposed to human experts, we have studied methods to cluster fuzzy rules, and we have shown that among the previously retained sets of operators, the association of Kleene-Dienes' implication with *min* as T-norm and *max* as T-co norm is the most interesting choice.

We must remark that the increase in computing complexity induced by usage of fuzzy logics is relatively small for rule extraction, since instead of increasing by one the counters on the number of examples and counter examples, fuzzy logics add membership degrees. The operations of fuzzification, the choice of fuzzy operators and the rules reductions are more complex, but the advantages of using fuzzy logics may compensate for this: intervals on continuous attributes are expressed by more expressive (fuzzy) labels, and abrupt threshold are avoided.

Among future perspectives, we plan to improve automatic configuration of the fuzzy partitions in connection to the samples extracted of the databases; we want to study other possible rules reduction mechanisms, and we also intend to study parallelisation methods of extraction algorithm, using for this a multi-agent platform. A more complete study of fuzzy operators is also desirable, to consider the Goguen implication, and examine cases when the generalized modus ponens is not only an approximate inference but also a valid logical inference [10].

## References

1. Aguilar Martin J. and Lopez De Mantaras R.: The Process of Classification and Learning the Meaning of Linguistic Descriptors of Concepts. In: M.M. Gupta et E. Sanchez (eds.) *Approximate reasoning in decision analysis*, North Holland, pp. 165-175, 1982
2. Bernadet M., Rose G., Briand H.: FIABLE and Fuzzy FIABLE: two learning mechanisms based on a probabilistic evaluation of implications. In: *Conference IPMU'96*, Granada (Spain), July 1996, pp. 911-916.
3. Bernadet M.: A knowledge discovery mechanism with evaluation of fuzzy implications. In: *Conference IPMU'98*, Paris (France), July 1998.
4. Bernadet M.: reduction of rules in a fuzzy knowledge discovery system. Research Report RR 00.8, IRIN, Univ. of Nantes, France
5. Bezdek, J.C. and Harris, J.D. (1978): Fuzzy Partitions and Relations: An Axiomatic Basis for Clustering. In: *Fuzzy Sets and Systems*, 1. pp. 111-127.
6. Briand H., Djeraba C., Fleury L., Masson Y. and Philippe J.: Contribution of the implication intensity in rules evaluations for knowledge discovery in databases. In: *ECML'95*, Heraklion, Crete, April 1995.
7. Fleury L. and Masson Y.: Intensity of implication: a measurement in learning machine. In: *IEAAIE'95*, 8th International Conference on Industrial and Engineering applications of AI and Expert Systems, 6-8 June 1995, Melbourne, Australia.
8. Gras R., Larher A.: L'implication statistique, une nouvelle méthode d'analyse de données. In: *Mathématiques, Informatique et Sciences Humaines* n°120.
9. Frawley W.J., Piatetsky-Shapiro G. and Matheus C.J.: Knowledge Discovery in Databases: An Overview. In: *AI Magazine*, Fall 1992, pp. 213-228.
10. Hajek P.: *Metamathematics of Fuzzy Logic*. Kluwer Academic Publisher, 1998.
11. Kerre Etienne E.: A comparative study of the behavior of some popular fuzzy implications on the generalized modus ponens. In: *Fuzzy Logic for the management of uncertainty*, L. Zadeh, J Kacprzyk., Wiley 1992, pp. 281-295.
12. Klir George J, Bo Yuan: *Fuzzy Sets, and Fuzzy Logics- Theory and Applications*. PrenticeHall. Englewood Cliffs, USA, 1995.
13. Lesmo L., Saitta L. and Torasso P.: Fuzzy production rules : a learning methodology. In : P.P. Wang (ed.) *Advances in Fuzzy Sets, Possibility Theory and Applications*. New York : Plenum Press, pp. 181-198
14. Rives J.: FID3 : Fuzzy Induction Decision Tree. In: *ISUMA'90*, Maryland, USA, pp. 457-462, December 1990
15. Turksen I.B.: Measurement of membership functions and their acquisition. In: *Fuzzy Sets and Systems*, 40:5--38, 1991.
16. Weber R.: Fuzzy-ID3: a Class of Methods for Automatic Knowledge Acquisition. In: *Proceedings of the 2nd International Conference on Fuzzy Logic and Neural Networks*, Iizuka, Japan, pp. 265-268, July 1992
17. Zadeh L.A.: Probability Measures of Fuzzy Events. In: *Journal of Mathematical Analysis and Applications*, Vol. 23, pp. 421-427, 1968
18. Zadeh L.A.: Fuzzy Logic and its application to approximate reasoning. In: *Information Processing*, 74:591--594, 1974.
19. Zeidler J., Schlosser M.: Fuzzy Handling of Continuous-Valued Attributes in Decision Trees. In: *Proc. ECML-95 Mlnet Familiarization Workshop Statistics, Machine Learning and Knowledge Discovery in Databases*, Heraklion, Crete, Greece, April 1995, pp. 41-46.
20. Zighed D.A., Rabaseda S., Rakotomalala R., Feschet F.: Discretization methods in supervised learning. In: *Encyclopedia of Computer Science and Technology*, vol. 40, pp. 35-50, Marcel Dekker inc., 1999.