Joint Conference of 1996 International Computer Symposium
December 19~21, Kaohsiung, Taiwan, R.O.C.

An Inductive Learning Strategy with Fuzzy Sets

Chang-Jiun Tsai[1], *Ching-Hung Wang*[1,2], * *Tzung-Pei Hong*[3] and *Shian-Shyong Tseng*[1]

Institute of Computer and Information Science, National Chiao-Tung University.[1]
Chunghwa Telecommunication Laboratories[2]
Department of Information Management, Kaohsiung Polytechnic Institute[3]
E-mail: tphong @ nas05.kpi.edu.tw

Abstract

In real applications, data provided to a learning system usually contain noisy and fuzzy information which greatly influences concept descriptions derived by conventional inductive learning methods. Modifying learning methods to learn concept descriptions in noisy and vague environments is thus very important. In this paper, we apply fuzzy set concept to machine learning to solve this problem. A fuzzy learning algorithm based on the AQR strategy is proposed to manage noisy and fuzzy information. The proposed algorithm generates fuzzy linguistic rules from fuzzy instances. In the experiment, the Iris Flower classification problem is used to compare the accuracy of the proposed algorithm with that of some other learning algorithms. Experimental results show that our method yields high accuracy.

Keyword: fuzzy set, fuzzy AQR, hypothesis space, instance space, inductive learning.

1. Introduction

Among machine learning approaches [6][8][17], *inductive learning* from instances may be the most commonly used in real world application domains. Inductive learning is basically a process of inferring concept descriptions that include all positive instances and exclude all negative instances. These traditional inductive learning procedures are however inapplicable to some application domains, since data in the real world usually contain fuzzy information. The fuzzy information will in general greatly influence the use of the concepts derived [15][19][20]. Some kinds of inductive learning problems arising in vague environments were discussed in [2][3][5][10][11]. Modifying traditional inductive learning methods to work well in vague environments is then very important. Several successful learning strategies based on ID3 have been proposed [7][19][20][21][23]; most of these use tree-pruning and fuzzy logic techniques. In [22], Wang *et al* proposed a fuzzy version space learning algorithm to manage both noisy and fuzzy data. In this paper, we will propose a fuzzy learning algorithm based on the AQR learning strategy [8] that induces a fuzzy rule set from fuzzy data. The learning approach can overcome problems of inductive learning in vague learning environments.

The remainder of this paper is organized as follows. Some related concepts and terms are reviewed in Section 2. The AQR learning strategy is reviewed in Section 3. The fuzzy AQR learning algorithm is proposed in Section 4. Experimental results from the IRIS flower classification problem are presented in Section 5. Finally, conclusions are given in Section 6.

2. Review of Related Concepts and Terms

In this section, we review some concepts and terms which are related to this paper. They are described as follows.

2.1 Fuzzy Set Concepts

A fuzzy set is an extension of a crisp set. Crisp sets allow only full membership or no membership at all, whereas fuzzy sets allow partial membership. In other words, an element may belong to more than one set. In a crisp set, the membership or non-membership of an element x in set A is described by a characteristic function $u_A(x)$, where

$$u_A(x) = \begin{cases} 1 & if\ x \in A \\ 0 & if\ x \notin A \end{cases}.$$

Fuzzy set theory extends this concept by defining partial membership, which can take values ranging from 0 to 1 :

$$u_A: X \rightarrow [0,1],$$

where X refers to the universal set defined for a specific problem.

Assuming that A and B are two fuzzy sets with membership functions of $u_A(x)$ and $u_B(x)$, then the following fuzzy operators can be defined.

* To whom all correspondence should be sent.

(1) The *intersection* operator:

$$u_{A \cap B}(x) = u_A(x)\ \tau\ u_B(x),$$

where τ: [0, 1] $*$ [0, 1] $\rightarrow$ [0, 1] is a *t-norm* operator satisfying the following conditions [14]:
for each *a, b, c* $\in$ [0, 1]:

(i) $a\ \tau\ 1 = a$;
(ii) $a\ \tau\ b = b\ \tau\ a$;
(iii) $a\ \tau\ b \geq c\ \tau\ d$ if $a \geq c, b \geq d$;
(iv) $a\ \tau\ b\ \tau\ c = a\ \tau\ (b\ \tau\ c) = (a\ \tau\ b)\ \tau\ c$.

Some instances of a *t-norm* operator $a\,\tau\,b$ are $\min(a, b)$ and $a * b$.

(2) The *union* operator:

$$u_{A \cup B}(x) = u_A(x)\ \rho\ u_B(x),$$

where ρ: [0, 1] $*$ [0, 1] $\rightarrow$ [0, 1] is an *s-norm* operator satisfying the following conditions [14]:
for each *a, b, c* $\in$ [0, 1]:

(i) $a\ \rho\ 0 = a$;
(ii) $a\ \rho\ b = b\ \rho\ a$;
(iii) $a\ \rho\ b \geq c\ \rho\ d$ if $a \geq c, b \geq d$;
(iv) $a\ \rho\ b\ \rho\ c = a\ \rho\ (b\ \rho\ c) = (a\ \rho\ b)\ \rho\ c$.

Some instances of an *s-norm* operator $a\,\rho\,b$ are $\max(a, b)$ and $a + b - a * b$.

2.2 Machine Learning Concepts

An *instance space* is a set of instances that can be legally described by a given instance language. Instance spaces can be divided into two classes: *attribute-based* instance spaces and *structured* instance spaces [18]. In an attribute-based instance space, each instance can be represented by *one* or *several* attributes. Attribute-based instance spaces are of the main concern here.

A *hypothesis space* is a set of hypotheses that can be legally described by a concept description language (generalization language). The form of a hypothesis space is restriction to concepts that can be expressed in conjunctive or disjunctive forms.

In the following sections, these concepts will be used in our learning algorithm to derive fuzzy knowledge.

3. Review of the AQR Learning Strategy

AQR is an inductive learning system [8] that uses the basic AQ algorithm [16] to generate a set of classification rules, one for each class. When building classification rules, AQR performs a heuristic search through the hypothesis space to determine the descriptions that account for all positive instances and no negative instances. AQR processes the training instances in stages; each stage generates a single rule, and then removes the instances it covers from the training set. This step is repeated until enough rules have been found to cover all the instances of the chosen class. The algorithm of AQR is described as follows:

AQR algorithm:
Let POS be a set of positive instances.
Let NEG be a set of negative instances.
STEP 1. Let COVER be the empty cover.
STEP 2. While COVER does not cover all instances in POS, process the following steps. Otherwise, stop the procedure and return COVER.
STEP 3. Select a SEED, i.e., a positive instance not covered by COVER.
STEP 4. Call procedure GENSTAR to generate a set STAR which is a set of complexes that covers SEED but that covers no instances in ENG.
STEP 5. Let BEST be the best complex in STAR according to the user-defined criteria.
STEP 6. Add BEST as an extra disjunct to COVER.

GENSTAR procedure:
STEP 1. Let STAR be the set containing the empty complex.
STEP 2. While any complex in STAR covers some negative instances in NEG, process the following steps. Otherwise, stop the procedure and return STAR.
STEP 3. Select a negative instance E_{neg} covered by a complex in STAR
STEP 4. Specialize complexes in STAR to exclude E_{neg} by:
a. Let EXTENSION be all selectors that cover SEED, but not E_{neg}.
b. Let STAR be the set { x $\cap$ y | x $\in$ STAR, y $\in$ EXTENSION}
c. Remove all complexes in STAR subsumed by other complexes.
STEP 5. Repeat this step until size of STAR $\leq$ maxstar (a user-defined maximum).
Remove the worst complex from STAR.

The concepts derived from AQR are represented as the multiple-valued logic propositional calculus with typed variables, which can be represented as follows:
If *<cover>* then predict *<class>*, where
<cover> = *<complex 1>* or .... or *<complex m>*,
<complex> = *<selector 1>* and ... and *<selector n>*,
<selector> = *<attributes r values>*,

<r> = *relation operator.*

A *selector* relates a variable to a value or a disjunction of values. For example, "color = red', "height = tall", and "weight > 60 kg" are all *selectors*. A conjunction of *selectors* forms a *complex*. A *cover* is a disjunction of *complexes* describing all positive all positive instances and none of the negative ones of the concept.

Unfortunately, AQR only works well in ideal domains where no noisy or fuzzy data is present. When such data are presented, the derived rules usually provides wrong classification information. However, the effective use of learning systems in real-world applications substantially depends upon their capability in handling noisy and fuzzy information. In this paper, we thus apply the concept of fuzzy sets to the AQR learning strategy to solve the above problem.

4. The Fuzzy AQR Learning Strategy

Conventionally, inductive learning is to find a concept description R that describes all the positive instances and none of the negative ones. If E is the set of positive and negative instances, P is the set of positive instances, and N is the set of negative instances, then we find a concept description R such that

$$"\forall e^+ \in P, e^+ \subset R, \forall e^- \in N, e^- \not\subset R",$$

where e^+ is a positive instance and e^- is a negative one, $\subset$ and $\not\subset$ are relation operators that mean "*covered by*" and "*not covered by*" respectively.

The concept description R, however, may not be easily found by conventional inductive learning approaches under imprecision and noise. In order to overcome these problems, the conventional inductive learning problem is then generalized as:

$$"\tilde{\forall} e \in \tilde{P}, e \tilde{\subset} \tilde{R}, \tilde{\forall} e \in \tilde{N}, e \tilde{\not\subset} \tilde{R}",$$

where $\tilde{R}$ is a fuzzy concept description, $\tilde{\forall}$ is a linguistic quantifier of type "almost all", "most", [14] etc., $\tilde{P}$ denotes a set of instances with the "soft" positiveness and $\tilde{N}$ denotes a set of instances with the "soft" negativeness, $\tilde{\subset}$ and $\tilde{\not\subset}$ are fuzzy relation operators that mean "*probably covered by*" and "*probably not covered by*" respectively. Each instance e can be considered as a *soft instance*. *Soft instances* differ from conventional instances in that they have class membership values. The membership value $u_{\tilde{P}}(e)$ specifies the degree to which instance e belongs to the positive class $\tilde{P}$, and the membership value $u_{\tilde{N}}(e)$ specifies the degree to which instance e belongs to the negative class $\tilde{N}$. The inductive learning is generalized to find a concept description, $\tilde{R}$, which can describes almost all of the "soft" positive instances and almost none of the "soft" negative instances. The term "describes" may have a fuzzy sense here, i.e., to a degree from 0 to 1.

The representation of a "soft" training instance is usually in terms of selectors with a class membership value. Each selector is represented as [*A r 'a'*], where A is an attribute, r is a crisp or fuzzy relation, a is a crisp or fuzzy value. An example for a "soft" training instance is shown as follows.

e=[height>190cm]&[weight>80kg] $\Rightarrow$ [basketballer],
with membership value $u_{basketballer}(e)$=0.8,

where [height > 190 cm] and [weight > 80 kg] are both selectors, and $u_{basketballer}(e)$ is a class membership value to specify the degree to which instance e belongs to the class *basketballer*.

The selectors used to describe instances may, however, be different from that of the derived concepts since some selectors in the derived concepts may be expressed in fuzzy terms. For example, the fuzzy concept derived from instances may be represented as :

Rule: *If* [height = 'tall']&[weight = 'heavy'] Then the player is a baseketballer, with membership value $u_{basketballer}(Rule)$=0.8,

where $u_{basketballer}(Rule)$ is represented the certainty of the rule.

It is clear that selectors used in the *instance space* to describe instances and ones in the *hypothesis space* to describe the derived concepts need not be the same. Since a dichotomy of attribute values is certainly too rigid and unrealistic in practice, we allow for a degree of identify between the value of attribute A_i in the instance e and the selector s_i in the hypothesis space. This degree $u_{s_i}(e)$ can be obtained by fuzzy matching. The values of $u_{s_i}(e)$ between 0 and 1 are used to represent the degree of covering instance e by S_i, 0 indicates the definite exclusion and 1 is the definite inclusion.

If we have an instance $e = S_{i_1}S_{i_2} \ldots S_{i_l}S_{j_1}S_{j_2} \ldots S_{j_m}S_{k_1}S_{k_2} \ldots S_{k_n}$ and a complex $C_j = S_{j_1}S_{j_2} \ldots S_{j_m}$, then the degree of covering instance e by a complex C_j is evaluated as

$$u_{c_j}(e) = u_{s_{j_1}}(e) \wedge u_{s_{j_2}}(e) \wedge \cdots \wedge u_{s_{j_m}}(e)$$

or more generally

$$u_{c_j}(e) = u_{s_{j_1}}(e)\, \tau\, u_{s_{j_2}}(e)\, \tau \cdots \tau\, u_{s_{j_m}}(e),$$

where τ is a *t-norm* operator.

The concept description $\tilde{R}$ is the disjunction of the complexes, say $C_1, C_2, \cdots, C_x$, denoted as

$\tilde{R} = C_1 \cup C_2 \cdots \cup C_x$. The degree of covering instance e by concept description $\tilde{R}$ is evaluated as

$$u_{\tilde{R}}(e) = u_{C_1}(e) \vee u_{C_2}(e) \vee \cdots \vee u_{C_x}(e)$$

or more generally

$$u_{\tilde{R}}(e) = u_{C_1}(e)\, \rho\, u_{C_2}(e)\, \rho \cdots \rho\, u_{C_x}(e),$$

where ρ is a *t-norm* operator.

Here, we propose a fuzzy AQR learning algorithm to induce fuzzy concepts from a set of "soft" training instances. In the proposed method, the concept descriptions no longer necessarily *include/exclude* all *positive/negative* instances presented, since fuzzy information exists in the training set. Fuzzy measure functions $u_{\tilde{V}^+}(C)$ and $u_{\tilde{V}^-}(C)$ are used to evaluate the "goodness" of C. The fuzzy measure function $u_{\tilde{V}^+}(C)$ used to evaluate the degree of including "soft" positive instances by C is defined as follows.

$$u_{\tilde{V}^+}(C) = \sum_{e \in E} (u_{\tilde{P}}(e)\, \tau\, u_C(e)) / \sum_{e \in E} u_{\tilde{P}}(e).$$

Similarly, $u_{\tilde{V}^+}(\tilde{R})$ used to evaluate the degree of including "soft" positive instances by complex $\tilde{R}$ is defined as follows.

$$u_{\tilde{V}^+}(\tilde{R}) = \sum_{e \in E} (u_{\tilde{P}}(e)\, \tau\, u_{\tilde{R}}(e)) / \sum_{e \in E} u_{\tilde{P}}(e).$$

The fuzzy measure function $u_{\tilde{V}^-}(C)$ used to evaluate the degree of excluding "soft" negative instances by C is defined as follows

$$u_{\tilde{V}^-}(C) = \sum_{e \in E} (u_{\tilde{N}}(e)\, \tau\, (1-u_C(e))) / \sum_{e \in E} u_{\tilde{N}}(e).$$

Correspondingly, $u_{\tilde{V}^-}(\tilde{R})$ used to evaluate the degree of excluding "soft" negative instances by complex $\tilde{R}$ is defined as follows

$$u_{\tilde{V}^-}(\tilde{R}) = \sum_{e \in E} (u_{\tilde{N}}(e)\, \tau\, (1-u_{\tilde{R}}(e))) / \sum_{e \in E} u_{\tilde{N}}(e).$$

If the concept description $\tilde{R}$ is a disjunction of complexes $C_1, C_2, \cdots, C_k$, then

$$u_{\tilde{V}^+}(\tilde{R}) = \frac{\sum_{e \in E} (u_{\tilde{P}}(e)\, \tau\, (u_{C_1}(e)\, \rho \cdots \rho\, u_{C_k}(e)))}{\sum_{e \in E} u_{\tilde{P}}(e)}$$

$$= \frac{\sum_{e \in E} ((u_{\tilde{P}}(e)\, \tau\, u_{C_1}(e))\, \rho \cdots \rho\, (u_{\tilde{P}}(e)\, \tau\, u_{C_k}(e)))}{\sum_{e \in E} u_{\tilde{P}}(e)}$$

$$= \frac{\sum_{e \in E} (u_{\tilde{P}}(e)\, \tau\, u_{C_1}(e))\, \rho \cdots \rho \sum_{e \in E} (u_{\tilde{P}}(e)\, \tau\, u_{C_k}(e))}{\sum_{e \in E} u_{\tilde{P}}(e)}$$

$$= u_{\tilde{V}^+}(C_1)\, \rho\, u_{\tilde{V}^+}(C_2)\, \rho \cdots \rho\, u_{\tilde{V}^+}(C_k).$$

$$u_{\tilde{V}^-}(\tilde{R}) = \frac{\sum_{e \in E} (u_{\tilde{N}}(e)\, \tau\, (1-u_{C_1}(e)\, \rho \cdots \rho\, u_{C_k}(e)))}{\sum_{e \in E} u_{\tilde{N}}(e)}$$

$$= \frac{\sum_{e \in E} (u_{\tilde{N}}(e) \tau (1-u_{C_1}(e))) \rho \cdots \rho (u_{\tilde{N}}(e) \tau (1-u_{C_k}(e)))}{\sum_{e \in E} u_{\tilde{N}}(e)}$$

$$= \frac{\sum_{e \in E} (u_{\tilde{N}}(e) \tau (1-u_{C_1}(e))) \rho \cdots \rho \sum_{e \in E} (u_{\tilde{N}}(e) \tau (1-u_{C_k}(e))}{\sum_{e \in E} u_{\tilde{N}}(e)}$$

$$= u_{\tilde{V}^-}(C_1)\, \rho\, u_{\tilde{V}^-}(C_2)\, \rho \cdots \rho\, u_{\tilde{V}^-}(C_k).$$

However, a complex C with a higher membership value $u_{\tilde{V}^+}(C)$ possesses more truth to include "*soft*" positive training instances; a complex C with a higher $u_{\tilde{V}^-}(C)$ possesses more truth to exclude "*soft*" negative training instances. A complex that includes much positive information may possibly include much negative information. Correspondingly, a complex that excludes much negative information may also possibly exclude much positive information. Clearly these kinds of complexes are not sure to be better than the complexes that include both less fuzzy positive and also less fuzzy negative information, or than the ones which exclude both less fuzzy negative information and less fuzzy positive information. Which complex C are suitable then depends on both $u_{\tilde{V}^+}(C)$ and $u_{\tilde{V}^-}(C)$. $u_{\tilde{V}^+\tilde{V}^-}(C)$ is then used to achieve this purpose, which is defined as follows.

$$u_{\tilde{V}^+\tilde{V}^-}(C) = u_{\tilde{V}^+}(C)\, \rho\, u_{\tilde{V}^-}(C),$$

where ρ is a *union* operators.

Similarly, $u_{\tilde{V}^+\tilde{V}^-}(\tilde{R})$ is used to evaluated the performance of the derived concept description $\tilde{R}$ which is defined as follows.

$$u_{\tilde{V}^+\tilde{V}^-}(\tilde{R}) = u_{\tilde{V}^+}(\tilde{R})\, \rho\, u_{\tilde{V}^-}(\tilde{R}).$$

If $\tilde{R} = C_1 \cup C_2 \cdots \cup C_k$, then $u_{\tilde{V}^+\tilde{V}^-}(\tilde{R}) = (u_{\tilde{V}^+}(C_1)\, \rho_1 \cdots \rho_1\, u_{\tilde{V}^+}(C_k))\, \rho\, (u_{\tilde{V}^-}(C_1)\, \rho_2 \cdots \rho_2\, u_{\tilde{V}^-}(C_k))$,

where ρ, ρ_1, and ρ_2 are *union* operators.

The fuzzy AQR learning strategy consists of two main phases: generating and testing. The generating phase generates and collects possible complexes into a large set; the testing phase then evaluate each element of this set according to the value of $u_{\tilde{V}^+\tilde{V}^-}$. The best complex as an extra disjunct is added to the set of concept description. The same procedure is repeated until all "soft" positive instances have been probably covered by the set of concept description. The fuzzy AQR learning algorithm is stated as follows:

INPUT:
(1) A set of "soft" positive training instances, denoted as $\tilde{P}$, each "soft" positive instance e^+ with class membership value $u_{\tilde{P}}(e^+)$.
(2) A set of "soft" negative training instances, denoted as $\tilde{N}$, each "soft" negative instance e^- with class membership value $u_{\tilde{N}}(e^-)$.

OUTPUT:
Find a fuzzy concept description $\tilde{R}$ that includes almost all of "soft" positive instances and excludes almost all of "soft" negative ones.

Fuzzy Inductive Learning Algorithm:
Initially, $\tilde{R}$ is an empty set that can not cover any "soft" positive instances.
While "soft" positive instances in $\tilde{P}$ have not been probably covered by $\tilde{R}$ (*i.e.* $\exists\ e^+ \in \tilde{P}$, $e^+ \tilde{\not\subset} \tilde{R}$), do
{
Select a "soft" positive one as a *SEED* that is not probably covered by $\tilde{R}$ but having the highest positiveness $u_{\tilde{P}}(e^+)$.
Generate a set of *complexes*, $\hat{C}_{set}$, that probably covers *SEED* but that covers no instances in $\tilde{N}$ (*i.e.* $\forall\ C_i \in \hat{C}_{set}, \forall\ e^- \in \tilde{N}, SEED \tilde{\subset} C_i \ \&\ e^- \tilde{\not\subset} \tilde{N}$) as follows:
{
Let $\hat{C}_{set}$ be a set of selectors that probably cover *SEED*. (i.e., $\hat{C}_{set} = \{S_i \mid S_i$ is a *selector*, $SEED = \bigcap_i S_i\}$)
While any *complex* in $\hat{C}_{set}$ probably covers negative instances in $\tilde{N}$ (*i.e.* $\exists\ e^- \in \tilde{N}$, $\exists\ C_j \in \hat{C}_{set}$, $e^- \tilde{\subset} C_j$), do
{
Select a *complex* C_j with the smallest value $u_{\tilde{N}}(C_j)$ in $\hat{C}_{set}$.
Select a negative instance e^- covered by C_j with the highest negativeness $u_{\tilde{N}}(e^-)$.
Specialize *complex* C_j as C_j', to exclude e^- as follows:
{
a. Let $\hat{S}$ be a set of selectors that probably cover *SEED*, but not any negative instances.
b. Let $\hat{C}_{set}$ be the set $\{C_j' \mid C_j' = C_j \wedge S_k, C_j \in \hat{C}_{set}, S_k \in \hat{S}\}$.
c. Remove all *complexes* in $\hat{C}_{set}$ subsumed by other *complexes*.
}
}
}
Select the best *complex* C_{best} that has the highest value of $u_{\tilde{P}^+\tilde{N}^-}(C_{best})$ in $\hat{C}_{set}$.
Add C_{best} as an extra disjunct to $\tilde{R}$.
(*i.e.* $\tilde{R} = \tilde{R} \cup C_{best}$)
}

Fuzzy AQR performs a heuristic search through the hypothesis space to determine the fuzzy concept descriptions that include almost all of "soft" positive instances and exclude almost all of "soft" negative ones. It induces the rules in stages; each stage generates a rule (a *complex*). When the learning process terminates, the *complexes* are output to form a set of rules. If the concept description $\tilde{R}$ is a disjunction of complexes $C_1, C_2, \cdots, C_k$, then the description is represented as the form of rules shown as follows.

Rule 1: IF C_1 Then positive class, with the membership value $u_{\tilde{P}^+\tilde{N}^-}(C_1)$,
Rule 2: IF C_2 Then positive class, with the membership value $u_{\tilde{P}^+\tilde{N}^-}(C_2)$,
⋮
Rule *k*: IF C_k Then positive class, with the membership value $u_{\tilde{P}^+\tilde{N}^-}(C_k)$.

5. Experiments

To demonstrate the effectiveness of the proposed fuzzy AQR learning algorithm, we applied it to classify Fisher's Iris Data which contain 150 training instances. The data are inconsistent, so the original AQR learning algorithm yields wrong concept descriptions.

The Iris problem is as follows. There are three species of iris flowers to be distinguished: *Setosa*, *Versicolor*, and *Verginica*. There are 50 training instances for each class. Each training instance is described by four attributes: *Sepal Length* (*S.L.*), *Sepal Width* (*S.W.*), *Petal Length* (*P.L.*), and *Petal Width* (*P.W.*). All four of the attributes are numerical domains. Assume the membership functions of each attribute as shown in Fig. 1.

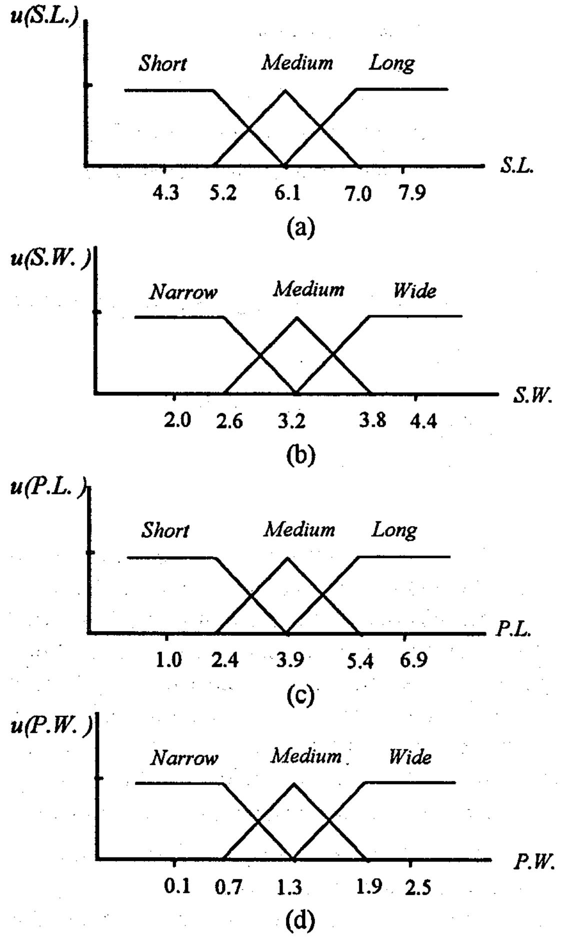

Figure 1. The given membership functions of each attribute

Since the training set includes only 150 instances, a method called *N-fold cross validation* [4] was adopted for this small set of samples. All instances were randomly divided into *N* subsets of as nearly equal size as possible. For each *n*, *n* = 1, .., *N*, the *n*-th subset was used as a test set, and the other subsets were combined into a training set. In the experiments, the data were partitioned into ten subsets, each with fifteen instances composed of five positive training instances and ten negative training instances. The fuzzy learning algorithm then ran on training instances to derive promising rules. Finally, the most promising rules derived were then tested on the remaining data subset. Classification rates were then averaged across all possible groups.

The fuzzy AQR learning algorithm was implemented in C language on a SUN SPARC/2 workstation. The algorithm was run 100 times, using different random partitions on the sample set. The classification accuracy converged to 1 for *Setosa*, 0.92 for *Versicolor*, and 0.96 for *Virginia*. The accuracy of some other learning algorithms on the Iris Flower Classification Problem was examined in [12] by Hirsh. The methods studied were Hirsh's Incremental Version Space Merging [12], Aha and Kibler's noise-tolerant NT-growth [1], Dasarathy's pattern-recognition approach [9], and Quinlan's C4 [20]. The generalized version space learning algorithm (GVS) was examined in [13] by Hong and Tseng. Table 1 compares the accuracy of our learning algorithm with that of the others. It can easily be seen that the accuracy of our method is as high as or higher than the other learning methods.

Table 1 Accuracy of six learning algorithms on the iris flower problem

class / algorithm	Setosa	Viginica	Versicolor	Average
FAQR	100	92	96	96.00
GVS	100	94	94	96.00
IVSM	100	93.33	94.00	95.78
NTgrowth	100	93.50	91.13	94.87
Dasarathy	100	98	86	94.67
C4	100	91.07	90.61	93.89

7. Conclusion

In this paper, we propose a fuzzy AQR learning algorithm to generate fuzzy rules from numerical data. This approach can overcome problems conventional learning methods have with noisy and fuzzy information, and find promising inference rules. These inference rules can be applied to infer the classes of input data by the inference process in Fuzzy Set Theory. Experimental results show that our method yields accuracy as high as or larger than that of some other learning algorithms. The proposed method is then a flexible and efficient fuzzy inductive learning method.

REFERENCES

[1] D. W. Aha and D. Kibler, Detecting and removing noisy instances from concept descriptions, *Technical Report University of California* (1989) 12-88.

[2] A. F. Blishun, Fuzzy Learning models in expert systems, *Fuzzy Sets and Systems* 22 (1987) 57-70.

[3] M. Botta, A.Giordana and L. Saitta, Learning in uncertain environments, *An Introduction to Fuzzy Logic Applications in Intelligent Systems* (1992) 281-296.

[4] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. stone, Classification and Regression Tree (1984).

[5] L. M de Campos and S. Moral, Learning rules for a fuzzy inference model, *Fuzzy Sets and Systems* 59 (1993) 247-257.

[6] J. Cendrowska, PRISM: An algorithm for inducing modular rules, *Int. J. of Man-Machine Studies* 27 (1987) 349-370.

[7] R. L. P. Chang and T. Pavliddis, Fuzzy decision tree algorithms, *IEEE Transactions on Systems, Man, and Cybernetics* 7 (1977) 28-35.

[8] P. Clark and T. Niblett, The CN2 induction algorithm, *Machine Learning* 3 (1989) 261-283.

[9] B. V. Dasarathy, Noising around the neighborhood: a new system structure and classification rule for recognition in partially exposed environments, *PAMI* 2 (1980) 67-71.

[10] M. Delgado and A. Gonzalez, An inductive learning procedure to identify fuzzy systems, *Fuzzy Sets and Systems* 55 (1993) 121-132.

[11] A. Gonzalez, A learning methodology in uncertain and imprecise environments, *Int. J. of Intelligent Systems* 10 (1995) 357-371.

[12] H. Hirsh, "Generalizing version spaces, *Machine Learning* 17 (1994) 5-46.

[13] T. P. Hong and S. S. Tseng, A generalized version space learning algorithm for noisy and uncertain data, *IEEE Transactions on Knowledge and Data Engineering* (accepted).

[14] J. Kacprzyk and C. Iwanski, Fuzzy logic with linguistic quantifiers in inductive learning, *Fuzzy Logic for the Management of Uncertainty* (1992) 465-478.

[15] Y. Kodratoff and M. Manago, Generalization and noise, *Int. J. of Man-Machine Studies* 27 (1987) 181-204.

[16] R. S. Michalski, On the quasi-minimal solution of the general covering problem, Proceedings of the Fifth International Symposium on Information Processing, (1969), pp. 125-128.

[17] T. M. Mitchell, Generalization as search, *Artificial Intelligence* 18 (1982) 203-226.

[18] T. M. Mitchell, Version Space: an approach to concept learning, Ph.D. Thesis, Stanford University, 1978.

[19] J. R. Quinlan, C4.5: programs for machine learning (1993).

[20] J. R. Quinlan, Decision tree as probabilistic classier, Proceeding of the Fourth International Machine Learning Workshop (1987) 31-37.

[21] J. Rives, FID3: Fuzzy induction decision tree, *The First International Symposium on Uncertainty Modeling and Analysis* (1990) 457-462.

[22] C. H. Wang, T. P. Hong, and S.S. Tseng, Inductive learning from fuzzy examples, accepted and to appear in *the Fitfth IEEE International Conference on Fuzzy Systems, FUUZ-IEEE'96.*

[23] Y. Yuan and M. J. Shaw, Induction of fuzzy decision trees, *Fuzzy Sets and Systems* 69 (1995) 125-139.