Middle-East Journal of Scientific Research 20 (10): 1207-1212, 2014
ISSN 1990-9233

DOI: 10.5829/idosi.mejsr.2014.20.10.12512

UGA: A New Genetic Algorithm-Based Classification Method for Uncertain Data

Ava Assadi and Saman Harati Zade

New Science Department,
Tehran University, Tehran, Iran

Abstract: Correct diagnosis of disease, is always one of human problems. Nowadays with advancement in the field of computer science, computer solutions can be used to solve this problem. One of these solutions is to use machine learning and data mining to detect disease. In this area, so many works have been done so far, but most of them, assumed data to be certain, whereas, in the medical field, the probability of data to be uncertain, is so much more than the other fields. Data uncertainty is common in real-world applications due to various causes, including imprecise measurement, network latency, out dated sources and privacy. To achieve this goal, there are many techniques, but physicians are very interested in rule based techniques that extract if – then rules, because these techniques are very easy to understand and also the conclusion is clear. So we decided to present a rule-based algorithm to diagnose diseases. We used genetic algorithm to extract fuzzy classification rules from data. In our suggested algorithm, we tried to improve the accuracy. Our experimental results show that our suggested algorithm has better performance than the other rule based related works.

Key words: Genetic algorithm · Rule-based algorithms · Fuzzy logic · Uncertain data

INTRODUCTION

Correct diagnosis of the disease is an important task. Recently, with advancement in the field of computer science, machine learning algorithms have come to help the medical community. But the conventional algorithms assume the data to be certain and therefore cannot deal with uncertain data. In many applications, data contains inherent uncertainty. A number of factors contribute to uncertainty, such as the random nature of the physical data generation and collection process, measurement and decision errors, unreliable data transmission and data staling [1].

The ever-increasing of uncertain data, raises the need of developing accurate and compatible machine learning algorithms to handle such data.In this paper, we focus on developing a rule-based classification algorithm for data with uncertainty. Rule-based data mining algorithms have a number of desirable properties. Rule sets are relatively easy for people to understand and rule learning systems outperform decision tree learners on many problems [1].

Previous researches have developed rule induction algorithms for coping with uncertain data [1]. In this research we use genetic algorithm for rule induction. The ability of genetic algorithms in searching large spaces is their most popular feature.

This paper is organized as follows. In the next section, we will discuss related work on data mining with uncertainty. Section 3 describes the uncertainty model for numerical data. Section 4 talks about the design of fuzzy rule–based classification systems. In Section 5, we show the details for using GA to achieve a fuzzy rule-based system. Section 6 explains how to measure the distance between two uncertain objects. Section 7 is about our strategy to classify new object. Section 8 presents our proposed algorithm. The experimental results are shown in Section 9 and Section 10 concludes the paper.

Related Works: In this section, we will introduce some related works about uncertain data mining and uncertain data classification.

There has been a growing interest in uncertain data mining. There are some previous works performed on classifying uncertain data in various applications but these methods try to solve specific classification tasks instead of developing a general algorithm for classifying uncertain data [1].

Corresponding Author: Ava Assadi, New Science Department, Tehran University, Tehran. Iran.

One of very first studies, in 2004, tried to develop SVM algorithm for uncertain data [2]. Some Other studies focus on clustering uncertain data. The main idea is to propose a model for computing the distance between two uncertain objects. The probability distributions of objects are used to compute the expected distance [1]. In 2005 Kriegel developed an uncertain clustering algorithm [3]. After that, in 2006, Ngai *et al.*, develpoed UK-means, which is a version of k- means for uncertain objects[4].

Field of Association rule mining has attracted much attention. In 2007 Chui *et al*, developed U-Apriori for association rule mining for uncertain data [5]. There are also some similar works which have been done by Zhang and Qin in 2008 and 2010 [6,7].

Ren *et al*, have developed Naïve Bayes in 2009 [8]. In 2010, He and Qin also had similar studies in this field [9, 10].

Qin in 2009 developed a decision tree algorithm for uncertain data called DTU algorithm [11].

After that, Liang, had a similar study for developing decision tree for uncertain and big data [12].

In 2010, Ge proposed an extension of neural network for uncertain data [13]. Fassetti introduced a new nearest neighbour algorithm for uncertain data [14].

A great study has been done by Qin *et al.* through extending the RIPPER algorithm.They proposed u-Rule algorithm to conduct if – then classification rules for uncertain data [1, 15, 16].

Data Uncertainty: In this section, we will discuss the uncertainty model for numerical and attributes. Here we focus on the attributes uncertainty and assume the class type is certain.

When the value of a numerical attribute is uncertain, the attribute is called an uncertain numerical attribute (UNA), denoted by A_i^{un}. Further, we use $A^{un}{}_{ij}$ to denote the jth instance of A_i^{un}. The concept of UNA has been introduced in [6]. The value of A_i^{un} is represented as a range or interval. Note that A_i^{un} is treated as a continuous random variable [1].

General Design of Fuzzy-rule-based Classification Systems: Since knowledge can often be expressed in a more natural way by using fuzzy sets, many decision support problems can be greatly simplified. We attempt to take advantage of fuzzy sets. One of the problems is discretising the domains of quantitative attributes into linguistic terms. Many algorithms have been introduced to solve this problem but most of them suffer from the following problem. The user or an expert must provide this algorithm the required fuzzy sets of the quantitative attributes and their corresponding membership functions. This problem is solved by an algorithm Gyenesei introduced in 2001 [17]. In proposed algorithm, fuzzy sets for each feature are determined using clustering techniques. In this algorithm, regardless of the clustering technique used, the best number of fuzzy sets for feature discretization is determined. We also used this idea. In our approach, we used fuzzy C means to cluster the data. Since, fuzzy c-means (FCM) is a method of clustering which allows one piece of data to belong to more than one clusters, it provides reasonable result according to the nature of our data.

In FCM, first, c cluster centres are selected randomly. Then, weight of each item, which determines the membership of that instance to cluster, calculates using formula below:

$$w_{ij}(x) = \frac{1}{\sum_{k=1}^{c}\left(\frac{d(c_j, x_i)}{d(c_k, x_i)}\right)^{\frac{2}{m-1}}}$$

where m is any real number greater than one.m specifies the fuzziness of clusters. We set m equal to 2. After that, new cluster centres should be found. For this purpose, next formula is used (18):

$$c(j) = \frac{\sum_{i=1}^{n}(m_{ij}^{m} * x_i)}{\sum_{i=1}^{n}(m_{ij}^{m})}$$

when FCM converged, we calculate a measure, which we named goodness. Goodness measure determines how accurate and separate the clusters are. This measure is used to identify the number of fuzzy sets which should be used for each attribute.

$$goodness = \sum_{i=1}^{c} \sum_{j=1, j \notin class\ of\ cluster\ i}^{n} wji + \frac{D_{\max}}{D\min}\sum_{i=1}^{c}(\sum_{j=1}^{c}|r_i - r_j|)^{-1}$$

After convergence of clustering algorithm, when final centres were determined, we should identify the class of clusters. To do this, we used the sum of sample weights in each cluster. For each cluster, the class which has the maximum value is considered as cluster class. After all, the goodness of clusters is calculated as sum of membership degree of samples which are in the wrong cluster. This is the concept of first parameter of formula above. The second one is the total separation between clusters which were introduced in [17].

Note that, we choose the number of clusters which obtain minimum value in goodness measurement.

After identifying the number of clusters and their centres, we use the algorithm which was introduced in [17] to conduct fuzzy sets and their membership function.

Genetic Algorithm: As mentioned before, genetic algorithm is one the popular evolutionary techniques which has the ability to search large spaces and also has simple implementation.

Genetic algorithms were invented by John Holland in the 1960s and were developed by Holland and his students and colleagues at the University of Michigan in the 1960s and the 1970s [19]. Holland's GA is a method for moving from one population of chromosomes to a new population by using a kind of selection together with the genetics inspired operators of crossover and mutation. Each chromosome consists of genes. The selection operator chooses those chromosomes in the population that will be allowed to reproduce, and on average the fitter chromosomes produce more offspring than the less fit ones. Crossover exchanges subparts of two chromosomes, roughly mimicking biological recombination between two single chromosome organisms. Mutation randomly changes the allele values of some locations in the chromosome.

Encoding: In our study, we use value encoding, to initialize chromosomes. Value encoding is a high level coding, in which, every gene is a string of some values.

First Population: Instead of using random population as first population in GA, we used the idea of Mansoori *et al*, which were proposed in [20]. In mentioned method, the initial population is selected from all fuzzy rules having only one active antecedent variable.

Class Detector: In initial population and in reproduction, after making a new rule, we should determine the class of the rule. To aim this purpose we used a formula which was used in Mansoori work [20].

Fitness Function: In GA, after determining the encoding, we should determine fitness function, which evaluate the goodness of each chromosome. It is important to define this function based on problem nature. Since our problem is a classification problem, we defined this function proportional to rule accuracy. At first, we formulated fitness function for rule R_i as:

$$fitness(R_i) = \frac{pANDd}{p}$$

where *pANDd* is the number of instances which satisfies both the antecedent and the consequent of R_i and p is the number of instances which satisfies only the antecedent of R_i.

Since our rules are fuzzy, we changed the definition of these parameters as below:

$$pANDd = \sum_{j=1, x_j \text{ has the same class as } R_i}^{n} \mu_j$$

$$p = \sum_{j=1}^{n} \mu_j$$

where n is the number of training instances and μ_j is the compatibility between training instance X_P and R_i which can be calculated using formula below:

$$\mu_j(x_p) = \min(\mu_{ji}(x_{pi})), \quad for\ i = 1 : n$$

where $\mu_{ji}(.)$ is the membership function of the antecedent fuzzy set. Since our instances are uncertain and are defined in an interval, to calculate the membership function of instance, we simply calculate the membership function for first and last item in the interval and use the maximum membership, as the membership function of interval.

One of the problems of using defined fitness function shows up when one rule is compatible with only one instance and classifies it correctly. With this definition, its fitness became one, which is the maximum. So we can say this fitness function consider detailed rules as better ones. So we need to add generality as a parameter of this function, therefore we redefined fitness function as:

$$fitness(R_i) = \frac{pANDd}{p} * \frac{number\ of\ pANDd}{number\ of\ same\ class}$$

In formula above, *number of pANDd*, is the number of instances which satisfies both antecedent and the consequent of R_i and *number of same class* is the number of instances which has the same class as R_i.

Selection: Selection is another important operator in GA. This operator determines which parents are selected for reproduction. The rule selection scheme in our proposed algorithm, only considers the evaluation measure of each rule to select the Q best ones through competition.

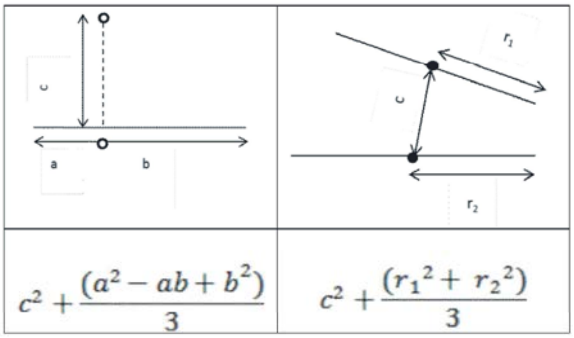
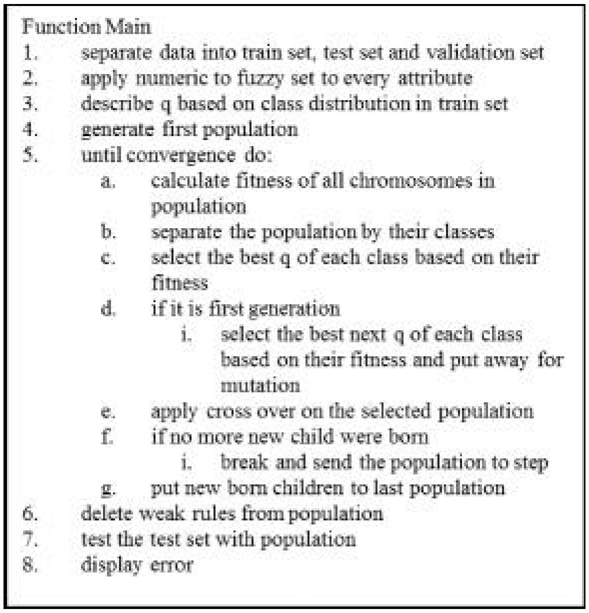

c
a
b
r1
c
r2
$c^2 + \frac{(a^2 - ab + b^2)}{3}$
$c^2 + \frac{({r_1}^2 + {r_2}^2)}{3}$

Fig. 1: Left figure shows the distance measurement between one uncertain and one certain object. Right figure shows distance measurement between two uncertain objects.

Here, unlike similar previous works, where Q was a constant number, we define Q proportional to the percentage distribution of the samples in each class. So if the number of instances of one class is greater, the greater number of chromosomes of its class will be selected for reproduction. This approach makes the data that are asymmetrical, may get a better answer.

Crossover: Now it's time to determine crossover operator. Mansoori *et al*, [20] used this strategy for reproduction. To generate offspring through reproduction, each parent rule *Rj*(with consequent class *Cj*) exploits the participation of another parent from *Cj*. This second parent *Rp*is only utilized to determine which inactive antecedent variable in *Rj*should be activated in this generation. In this way, *Rp*is selected randomly among rules of class *Cj*in the current population. After identifying *Rp,* one of its active antecedent variables *xi* is selected randomly. If *xi* is inactive in *Rj*, then the reproduction on *Rj*can take place by activating *xi* and generating all possible offspring, provided the offspring be fitter than *Rj*. These offspring will use *xi* as active beside other active antecedents in *Rj*. However, if *xi* has been active in *Rj*before reproduction, then no reproduction can occur on it. In this case, *Rj*can survive in the next generation through elitism. The consequent class of some offspring might differ from their parents' because each offspring is a more specific fuzzy rule than its parent, and so, its fuzzy subspace is smaller. Our idea is similar to this. When making a child, we active all the features in first parent which were inactive, by second parent values in that features. Before determining child class, if the coverage of offspring is lower than a user specified threshold, we discard the offspring, otherwise we call the class detector function for offspring.



```
Function Main
1.    separate data into train set, test set and validation set
2.    apply numeric to fuzzy set to every attribute
3.    describe q based on class distribution in train set
4.    generate first population
5.    until convergence do:
        a.   calculate fitness of all chromosomes in
             population
        b.   separate the population by their classes
        c.   select the best q of each class based on their
             fitness
        d.   if it is first generation
               i.   select the best next q of each class
                    based on their fitness and put away for
                    mutation
        e.   apply cross over on the selected population
        f.   if no more new child were born
               i.   break and send the population to step
        g.   put new born children to last population
6.    delete weak rules from population
7.    test the test set with population
8.    display error
```

Fig. 2: UGA Pseudocode

Calculating the Distance Between Two Uncertain Items: As mentioned before, uncertain items are described in an interval. On the other hand, in FCM, we need to be able to calculate the distance between two uncertain objects or between one certain and one uncertain object. Some efforts have been done to solve this problem, but most of them, assumed the uncertain object has described in an interval with a PDF. In real world problems we do not have the PDF of intervals. In [21] this problem has been solved. Here, we used their idea.

New Item Classification: We use weighted voting to classify a new item. In our approach we calculate the compatibility degree of item with every rule in rule set. Then we use the rules which their compatibility degree is greater than a user specified threshold. Then we sum these compatibility degrees over each class. At last, the class which has the maximum value obtained in previous step is determined as item class.

Proposed Algorithm: In this section, we present the pseudocode of our proposed algorithm.

Experiments: In this section, we present the experimental results of the proposed algorithm. We studied this classifier accuracy over multiple datasets. We used 3 real-world benchmark datasets to evaluate the performance of our proposed algorithm - diabetes, iris and sonar datasets. All of these datasets are available from the UCI Repository.

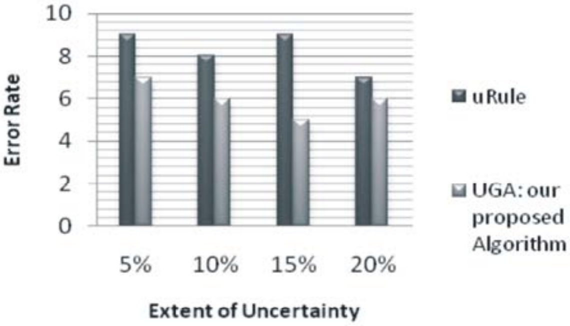


Fig. 3: Comparing error rate of our proposed algorithm with u-Rule in Iris Data set.

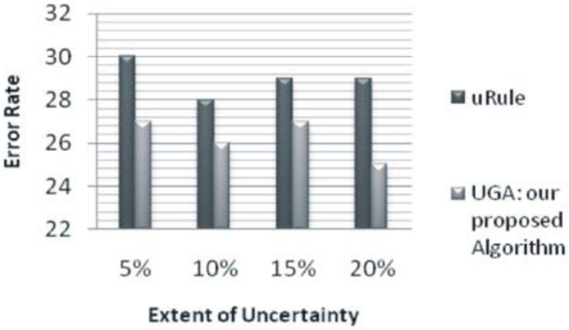


Fig. 4: Comparing error rate of our proposed algorithm with u-Rule in Diabetes Data set.

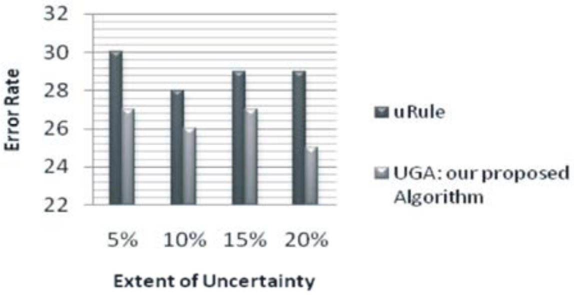


Fig. 5: Comparing error rate of our proposed algorithm with u-Rule in Sonar Data set.

Note that all these dataset are certain, and we should make them uncertain manually.

To do this, we used a method which were introduced and used in [9].

In the following experiments, we use five-fold cross validation. Data is split into five approximately equal partitions; each one is used in turn for testing while the rest is used for training. The whole procedure is repeated 50 times, and the overall accuracy rate is counted as the average of accuracy rates on each partition. Figure three to five shows the performance of proposed algorithm when uncertainty ranges from 0 to 20%.

It shows that our algorithm has better accuracies over datasets.

CONCOLUSION

Uncertain data often occur in modern applications, including sensor databases, spatial-temporal databases, and medical or biology information systems. In this paper, we propose a new rule-based algorithm for classifying and predicting uncertain datasets. The avenues of future work include developing uncertain data mining techniques for various applications.

REFERENCES

1. Qin, B., Y. Xia, R. Sathyesh, S. Prabhakar and Y. Tu, 2010. uRule: A Rule-based Classification System for Uncertain Data, in Data Mining Workshops (ICDMW), IEEE International Conference on, pp: 1415-1418.
2. Bi, J. and T. Zhang, 2004. Support Vector Machines with Input Data Uncertainty, in Advances in Neural Information Processing Systems (NIPS).
3. Kriegel, H.P. and M. Pfeifle., 2005. Density Based Clustering of Uncertain Data, in ACM SIGKDD.
4. Ngai, W., *et al*., 2006. Efficient Clustering of Uncertain Data, in Sixth IEEE Int'l Conf. Data Mining (ICDM).
5. Chui, C.K. and B. Kao, 2008. A Decremental Approach for Mining Frequent Itemsets from Uncertain Data, in 12th Pacific-Asia Conf. Knowledge Discovery and Data Mining (PAKDD).
6. Zhang, Q., Zhang, F. Li, and K. Yi, 2008. Finding Frequent Items in Probabilistic Data, in ACM SIGMOD.
7. Qin, X., Y. Zhang, X. Li and Y. Wang, 2010. Associative Classifier for Uncertain Data, in International Conference on Web-age Information ,Management (WAIM), pp: 692-703.
8. Ren, J.T., *et al*., 2009. Naive Bayes Classification of Uncertain Data, in ICDM.
9. He, J., Y. Zhang, X. Li, and Y. Wang, 2010. Naive Bayes Classifier for Positive Unlabeled Learning with Uncertainty, in SDM, pp: 361-372.
10. Qin, B., Y. Xia, and F. Li, 2010. A Bayesian Classi?er for Uncertain Data, in SAC.
11. Qin, B., Y. Xia and F. Li, 2009. DTU: A Decision Tree for Classifying Uncertain Data, in Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD).
12. Liang, C., Y. Zhang, and Q. Song, 2010. Decision Tree for Dynamic and Uncertain Data Streams, in JMLR: Workshop and Conference Proceedings, pp: 209-224.

13. Ge, J., Y. Xia, and C. H. Nadungodage, 2010. UNN: A Neural Network for Uncertain Data Classification, in 14th Pacific-Asia Conf. on Knowledge Discovery and Data Mining, pp: 449-460.
14. Angiulli, F. and F. Fassetti, 2011. Uncertain Nearest Neighbor Classification.
15. Qin, B., Y. Xia, and S. Prbahakar, 2011. Rule Induction for Uncertain Data, in Knowledge and information systems, pp: 103-130.
16. Qin, B., Y. Xia, and S. Prbahakar, 2009. A Rule-Based Classification Algorithm for Uncertain Data, in Proceedings of The IEEE Workshop on Management and Mining Of Uncertain Data (MOUND).
17. Gyenesei, A., 2001. Determining Fuzzy Sets for Quantitative Attributes in Data Mining Problems.
18. James C. Bezdek, Robert Ehrlich and William Full, 1984. FCM: The Fuzzy C - Means Clustering Algorithm, 10(2).
19. Melanie Mitchell, 1999. An Introduction to Genetic Algorithms.: MIT Press.
20. Mansoori, E.G., M.J. Zolghadri and S.D. Katebi, 2008. "SGERD: A Steady-State Genetic Algorithm for Extracting Fuzzy Classification Rules from Data,
21. Lurong Xiao and Edward Hung, 2007. An Efficient Distance Calculation Method for Uncertain Objects.