International Journal of Modern Engineering Research (IJMER)

www.ijmer.com Vol.2, Issue.3, May-June 2012 pp-1066-1072 ISSN: 2249-6645

Classification of Uncertain Data Using Selection Algorithm

K. Soundararajan[1], Dr. S. Suresh Kumar[2], C. Anusuya[3]

1, 3(Department of Information Technology, Vivekananda College of Engineering for Women, Nammakal, Tamilnadu, India)

2 (Department of Computer Science and Engineering, Vivekananda College of Technology for Women, Nammakkal, Tamilnadu, India)

ABSTRACT

Traditional machine learning algorithms assume that data are exact or precise. However, this assumption may not hold in some situations because of data uncertainty arising from measurement errors, data staleness, and repeated measurements etc., these kinds of uncertainty have to be handled cautiously, or else the mining results could be unreliable or even wrong. In this paper, we focus on classifying uncertain data by classification and prediction algorithm called setBase. This algorithm introduces new measures for generating, pruning and optimizing. Probability distribution function and uncertain data intervals are computed by this kind of algorithm .Based on these new measures, the optimal splitting attribute and splitting value can be identified and used for classification and prediction. Uncertainty in both numerical and categorical data can be processed by this measure. Our experimental results show that setBase has excellent performance even when data is highly uncertain.

Keywords: classification, data uncertainty, prediction, setBase algorithm, traditional machines.

1. INTRODUCTION

Traditional machine learning algorithms often assume that the data values are exact or precise. In many emerging applications, however, the data is inherently uncertain. Sampling errors and instrument errors are both sources of uncertainty and data are typically represented by probability distributions rather than by deterministic values. There are many learning algorithms used in the classification of deterministic data points, but few algorithms have been proposed for classification of distribution-based uncertain data objects.

Uncertain data is ubiquitous in many real world applications, such as environmental monitoring, sensor network, market analysis and medical diagnosis [1].A number of factors contributes to the uncertainty. It may be cause by imprecision measurements, network latencies, data staling and decision errors. Uncertainty can arise in categorical attributes and numerical attributes [1, 2].For example, in cancer diagnosis; it is often very difficult for the doctor to exactly classify a tumour to be benign or malignant due to the experiment precision limitation. it would be better to represent by probability to be benign or malignant[2].Since data uncertainty is ubiquitous we have to develop a data mining algorithms for uncertain datasets.

In this paper, we introduce a new selection based classification algorithm for data with uncertainty; in this process we have a number of desirable properties. Rule sets are relatively easy for people to understand[15],and rule learning systems outperform decision tree learners on many problems[16],[17].Rule sets have a natural and familiar first order version, namely prolong predicates, and techniques for learning propositional rule sets can often be extended to the first order case[18],[19]. However, when data contains uncertainty. For example, when some numerical data are instead of precise value, an interval with probability distribution function with that interval-these algorithms cannot process the uncertainty properly. We prove through extensive experiments that the proposed classification can be efficiently generated and it can classify uncertain data with potentially higher accuracies than the other classification algorithms. Furthermore, the proposed classifier is more suitable for mining uncertain data than the other mining algorithms.

In this paper, we propose a new selection based algorithm for classifying and predicting both certain and uncertain data. We integrate the uncertain data model into the selection based mining algorithm. For generating rules, we introduce a new measure called probabilistic information gain. The field of uncertain data management process a number of unique challenges on several methods which the uncertain data are discussed in Bayesian.

We also extend the set pruning measure for handling data uncertainty , we perform experiments on real datasets with both uniform and Gaussian distribution, and the experimental results demonstrate that setBase algorithm perform well even on highly uncertain data. There are many learning algorithms used in the classification of distribution based uncertain data objects.

In the rest of this paper, we first give some related works in section 2. Then we introduce preliminaries in section 3 .Our algorithmic framework is presented in section 4. Experimental studies on accuracy and performance are presented in section 5. The paper is discussed and concluded in sections 6.

2. REVIEW OF RELATED RESEARCHES

In this section, we will introduce some related works about uncertain data mining, uncertain data classification, and more. A detailed survey of uncertain data mining techniques may be found in [5].In the case of uncertain data mining, studies include clustering[12,13,14], classification [2,3,6,11],frequent itemset mining[4,7,8,9,10].

At present, existing works about classification algorithms. Qin et al, proposed a rule based algorithm to cope with uncertain data [1], later, in [2], Qin et al. presented DTU, which based on decision tree algorithm, to deal with uncertain data by extending traditional measurements, such as information entropy and information gain. In [5], Tsang et al. extended classical decision tree UDT algorithm to handle uncertain data which is

www.ijmer.com Vol.2, Issue.3, May-June 2012 pp-1066-1072 ISSN: 2249-6645

represented by probability density function (pdf).In [10],Bi et al. proposed Total Support Vector Classification (TSVC),a formulation of support vector classification to handle uncertain data.

Classification is a well studied area in data mining. There may be a numerous classification algorithms have been proposed in the literature, such as decision tree classifiers[20],rule-based classifiers[21],Bayesian classifiers[22],support vector machines(SVM)[23],artificial neural networks[24],Lazy learners, and ensemble methods[25]. Decision tree induction is the learning of a decision tree from class-labeled training tuples.

A Rule based classifier is a technique for classifying records using a collection of" if…then…"rules. Bayesian classifiers are statistical classifiers and are based on bayes theorem. SVM has its roots in statistical learning theory and has shown promising empirical results in many practical applications, from handwritten digit recognition to text categorization.

Uncertain data, also called symbolic data [36], has been studied for many years. Many works focus on clustering [37]. The key idea is that when computing the distance between two uncertain objects, the probability distributions of objects are used to calculate the expected distance. In [14], Cormode et al. show reductions to their corresponding weighted versions on data with uncertainties. In [33], Xia et al. introduce a new conceptual clustering algorithm for uncertain categorical data.

Classification is a well-studied area in data mining. Many methods have been proposed in the literature, such as decision tree [20], rule based classifications, Bayesian classifications [38] and so on. In spite of the numerous methods, building classification based on uncertain data remains a great challenge. There is early work performed on developing decision trees when data contains missing or noisy values [19]. Various strategies have been developed to predict or fill missing attribute values, for example, Dayanik presented feature interval learning algorithms which represent multi-concept descriptions in the form of disjoint feature intervals. However, the problem studied.

In this paper is different from before, instead of assuming part of the data has missing or noisy values, we allow the whole dataset to be uncertain, and the uncertainty is not shown as missing or erroneous values, but represented as uncertain intervals with probability distribution functions.

Recently, Tsang et al [6] and Qin et al [3] independently developed decision tree classifications for uncertain data. Both adopt the technique of fractional tuple for splitting tuples into subsets when the domain of its PDF spans across the cut point. Tsang et al [6] converted every numerical value into a set of sample points between the uncertain intervals [aj , bj] with the associated value f(x), effectively approximating every f(x) by a discrete distribution. Qin et al. Also proposed a rule-based classification [2]. The key problem in learning rules is to efficiently identify the optimal cut points from training data. For uncertain numerical data, an optimization mechanism is proposed to merge adjacent bins which have equal classifying class distribution. In our earlier work, we proposed a Bayesian classification method for uncertain on clustering of uncertain data. In [26], [27], the K means clustering algorithms are data. It will be compared with our new approach in this paper.

There have been studies extended so that the distance between objects are computed as the Expected Distance using a probability distribution function. For uncertain versions of *k*-means and *k*-median, Cormode et al [28] show reductions to their corresponding weighted versions on data with no uncertainties. The FDBSCAN and FOPTICS [29], [30] algorithms are based on DBSCAN and OPTICS, respectively. Instead of identifying regions with high density, these algorithms identify regions with high expected density, based on the pdfs of the objects. Aggarwal and Yu [31], [32] propose an extension of their micro-clustering technique to uncertain data. [33] Introduces a new conceptual clustering algorithm for uncertain categorical data. There is also research on identifying frequent item set and association mining [34], [35] from uncertain datasets. The support of item sets and confidence of association rules are integrated with the existential probability of transactions and items. However, none of them address the issue of developing classification and predication algorithms for uncertain datasets.

3. PRELIMINARIES

In this paper, we are mainly focus on the uncertain attributes and assume the class type is certain. In this section, we will discuss the uncertain model for both numerical and categorical attributes, which are the most common types of attributes encountered in data mining applications. Uncertain data has attracted more and more attention in the literature.

3.1. NUMERICAL DATA UNCERTAINITY

In this model, we describe numerical data uncertainty, this shows that the value of a numerical attribute is uncertain, then the attribute is said to be a uncertain numerical attribute, that must be denoted by Ai(Un). Further, we use Aj(Un) is to denote the jth instance of Ai(Un).The concept of this model for uncertain numerical data has been introduced in [1].The value of Ai(Un) is represented as a range of interval and the probability distribution function(PDF)over this range. We notice that Ai (Un) is treated as a continuous random variable. This probability distribution function is denoted by f(x) can be related to an attribute if all instances have the same distribution function and that are related to each instances has different distributions. For every uncertain numerical attribute instance Ai, let sum i = (Aij .max+Aij .min)/2 and diffi+ = (Aij .max − Aij .min) _ (Aij .max − Aij .min)/36.

Definition 1:.An uncertain interval instance of Ai(Un),is denoted by Aij(Un),U ,is an interval |Aij(Un).l,Aij(Un).r| where Aij(Un).l,Aij(Un that is belongs to R, Aij(Un).r>=Aij(Un).l. Then this realizes that this model is an application dependent.

3.2. CATEGORICAL DATA UNCERTAINITY

Under this uncertainty data model deals with categorical data, and these attributes that are allowed to take uncertain values. Then we call such attribute an uncertain categorical attributes that are denoted by Ai(Uc). Further, we use Aij(Uc) to denote the jth instance of Ai(Uc).The notion of the uncertain categorical attributes was proposed in [2].

www.ijmer.com Vol.2, Issue.3, May-June 2012 pp-1066-1072 ISSN: 2249-6645

When dealing with uncertain categorical attribute, we utilize the same model as studies in [1, 2 and 3] to represent uncertain categorical data. Under the uncertain categorical model, a dataset can have attributes that are allowed to take uncertain values [2]. And we call these attributes Uncertain Categorical Attributes, UCA. The concept of UCA was introduced in [1]. Let's write *Auci* for the *ith* uncertain categorical attribute, and *Vi = {vi*1, *vi*2, · · ·, *vi*|*Vi*|*}* for its domain. As described in [2], for instance *j* , its attribute value of *Auci* can be represented by the probability distribution over *Vi*, and formalized as *Pji = _pi*1, *pi*2, · · · , *piVi* _, such that *Pji*(*Auci = vik*) = *pik*(1 ≤ *k* ≤ |*Vi*|),and _|*Vi*|*k*=1 *pik* = 1.0, which means *Auci* takes value of *vik* with probability *pik*. Certain attribute can be viewed as a special case of uncertain attribute. In this case, the attribute value of *Auci* for instance *tj* can only take one value, *vik*, from domain *Vi*, i.e. .*Pji*(*Auci = vik*) = 1.0(1 ≤ *k* ≤ |*Vi*|), *Pji*(*Auci = vih*) = 0.0(1 ≤ *h* ≤ |*Vi*|, *h* _= *k*).

Aij (Uc) takes values from the categorical domain Dom with cardinality |Dom|=n. Within a regular relation, the value of an attribute A is a single value in Dom=1.In this case of an uncertain relation, we record the information by a probability distribution over Dom instead of a single value domains.

4. ALGORITHMS AND APPLICATION

In this section, we deal with the algorithms and its applications. To build a set base classifier, we need to extract a set of rules that show the relationships between the attributes of a dataset and the class label. Each classification is in the form of R☹Condition) +>y. Here the condition is called the set base antecedent, which is a conjunction of, y is called the set base consequent and it is the class label ,it consists of multiple sets Rs ={R1,R2,…….Rn }

The **Coverage** of a rule is the number of instances that satisfy the antecedent of a rule. The **Accuracy** of a rule is the fraction of instances that satisfy both the antecedent and consequent of a rule, normalized by those satisfying the antecedent. Ideal rules should have both high coverage and high accuracy rates.

The setBase algorithm is shown in Algorithm 1. It uses the sequential covering approach to extract rules from the datasets. This algorithm extracts the rules one class at a time for a data set. Let (*y*1, *y*2. . . *yn*) be the ordered classes according to their frequencies, where *y*1 is the least frequent class and *yn* is the most frequent class. During the *i*th iteration, instances that belong to *yi* are labeled as positive examples, while those that belong to other classes are labeled as negative examples.

A rule is desirable if it covers most of the positive examples and none of the negative examples. Our setBase algorithm is based on the RIPPER algorithm [9], which was introduced by Cohen and considered to be one of the most commonly used rule-based algorithms in practice.

Algorithm 1: setBase (Dataset *D*, ClassSet *C*)
Begin
RuleSet = ∅; //initial set of rules learned is empty
for Each Class *ci* ∈ *C* do
newRuleSet = uLearnOneRule (*D*, *ci*);
Remove tuples covered by newRuleSet from Dataset D;
RuleSet += *newRuleSet*;
End for;
Return *RuleSet*;
End

The uLearnOneRule () procedure is shown in Algorithm 2; it is the key function of the setBase algorithm. It generates the bestrule for the current class, given the current set of uncertain training tuples. The uLearnOneRule () includes two phases: growing rules and pruning rules. We will explain the first phase, growing rules, in more detail, while the other pruning rules is similar to regular rule-based classifier, thus will not be elaborated. After generating a rule, all the positive and negative examples covered by the rule are eliminated. The rule is then added into the rule set as long as it does not violate the stopping condition, which is based on the minimum description length (DL) principle. setBase also performs additional optimization steps to determine whether some of the existing rules in the rule set can be replaced by better alternative rules.

Algorithm 2 uLearnOneRule (Dataset *D*, Class *ci*)
Begin
Stop = false;
RuleSet = ∅;
Repeat
Split *D* into growData and pruneData;
Rule = uGrow (growData);
Prune Rules based on pruneData;
Add Rules to RuleSet;
Remove data covered by Rule from *D*;
Until Stop Condition is true
Return (RuleSet);
End

The process of growing rules, uGrow (), is shown in Algorithm3. The basic strategy is as follows:

1. It starts with an initial rule : *{}*-> y, where the left hand side is an empty set and the right hand side contains the target class. The rule has poor quality because it covers all the examples in the training set. New conjuncts will subsequently be added to improve the rule's quality.
2. The probabilistic information gain is used as a measure to identify the best conjunct to be added into the rule antecedent. This algorithm selects the attribute and split point which has the highest probabilistic information gain and adds them as follows.

Algorithm 3 uGrow (Instances growData)
Begin
CoverData = ∅;
While (growData.size () > 0) ∧(numUnusedAttributes >0) do
Find the attribute *Ai* and the split point *sp*, which has the highest probabilistic information gain;
Antecedent += RuleAntecedent (*Ai*, *sp*);
for (each instance *Ij*) do
if (covers(*Ij*)) then
Inst = splitUncertain (*Ij*, *Ai*, *sp*);
coverData += inst;
End if;
End for;
growData -= coverData;
end while;
End

Function splitUncertain () is shown in Algorithm 4. As the data is uncertain, a rule can partly cover an instance.

www.ijmer.com Vol.2, Issue.3, May-June 2012 pp-1066-1072 ISSN: 2249-6645

For Example, for an instance with income [100, 110], a rule "income > 105 => default Borrower = No" only partly covers it. For an instance with tumor [Benign, 0.8Malignant, 0.2], a rule "tumor = benign => Survive = yes" also partly covers it. Function splitUncertain () computes what proportion of the instances is covered by a selection based on the uncertain attribute interval and probabilistic distribution function.

Algorithm 4 splitUncertain (Instance *Ii*, attribute *Ai*, splitPoint*sp*)
Begin
if the rule fully covers instance *Ii* then
return *Ii*;
end if;
if (*Ai* is numerical or uncertain numerical) then
if the rule partially covers instance *Ii* on the right side then
Ii.w = *Ii*.w**_s f*(*x*) *dx*/*_maxmin f*(*x*) *dx*;
end if;
if the rule partially covers instance *Ii* on the left side then
Ii.w = *Ii*.w**_s f*(*x*) *dx*/*_maxmin f*(*x*) *dx*;
end if;
end if;
if (*Ai* is categorical or uncertain categorical) then
Ii.w = *Ii*.w*att.value (sp).w*P (*Ii*, *Rk*);
end if;
return *Ii*;
End

4.1 SPLITING THE DATA

It is seen that when the rules are being generated from the training dataset, the goal is to determine the best split attribute and best split point. We use a measure called probabilistic information gain to identify the optimal split attribute and split point for uncertain training dataset.

4.1.1. UNCERTAIN NUMERICAL ATTRIBUTES

The value of an uncertain numeric attribute is an interval with associated PDF. Each uncertain interval has a maximal value and a minimal value, which are called critical points. For each UNA, we can order all critical points of an uncertain numeric attributes in an ascending sort with duplicate elimination. The Class Distribution of each partition is called as Class Distribution Vector (CDV). Suppose there are N critical points after eliminating duplicates, then this UNA can be divided into N+ 1 partition. Since the leftmost and rightmost partitions do not contain data instances at all, a split definitely will not occur within them. We need only consider the rest N-1 partitions. The Probabilistic Cardinality (PC) of the dataset over the partition P_a = [a, b) is the sum of the probability of each instances whose corresponding UCA equals P_a = [a, b).

4.1.2. UNCERTAIN CATEGORICAL DATA

Uncertainty in categorical data is common place in many applications, including data cleaning, database integration, and biological annotation. In such domains, the correct value of an attribute is often unknown, but may be selected from a reasonable number of alternatives. Current database management systems do not provide a convenient means for representing or manipulating this type of uncertainty.

A rule related to an uncertain categorical attribute only covers its one value, which is called split point. An uncertain categorical attribute (UCA) is characterized by probability distribution over Domain. Datasets without uncertainty can be treated as a special case of date with uncertainty. When using a matrix to represent a categorical attribute of a dataset without uncertainty, there is at most one element per row to be 1.The probabilistic cardinality (PC) of the dataset over v_k is the sum of the probability of each instances whose corresponding UCA equals v_k. Based on the Class Distribution Vector (CDV) we can compute the probabilistic information gain if the categorical attribute is selected as the antecedent of the rule.

4.1.3. PRUNING TECHNIQUES

After growing, the rule is immediately pruned by deleting any final sequence of conditions from the rule, and chooses the deletion that maximizes the function which is known as Pruning. There are two types of pruning namely Pre-Pruning and Post-Pruning. setBase employs a general-to-specific strategy to grow a rule and the probabilistic information gain measure to choose the best conjunct to be added into the rule antecedent. The new rule is then pruned based on its performance on the validation set. The following metric is used for rule pruning.

The probabilistic prune for a rule *R* is

ProbPrune(*R*, *p*, *n*) = {*PC* (*p*) −*PC* (*n*)} / {*PC*(*p*) + *PC*(*n*)}

Here PC (p) and PC(n) is the probabilistic cardinality of positive and negative instances covered by the rule. This metric is monotonically related to the rule's accuracy on the validation set. If the metric improves after pruning, then the conjunct is removed. Like RIPPER, setBase starts with the most recently added conjunct when considering pruning. Conjuncts are pruned one at a time as long as this results in an improvement.

4.1.4. .PREDICTION TECHNIQUES

Once the rules are learned from a dataset, they can be used for predicting class types of unseen data. Like a traditional rule classifier, each rule of setBase is in form of "IF conditions THEN Class=C_i". Because each instance I_i can be covered by several rules, a vector can be generated for each instance and the vector is an Class Probability Vector (CPV). As an uncertain data instance can be partly covered by a rule, we denote the degree an instance I covered by a rule R_j. An uncertain instance may be covered or partially covered by more than one rule. We allow the test instance to trigger all relevant rules. W (I_i, R_k) to denote the weight of an instance I_i covered by different rules.

An uncertain instance may be covered or partially covered by more than one rule. We allow a test instance to trigger all relevant rules. We use *w* (*Ii*, *Rk*) to denote the weight of an instance *Ii* covered by the *k*th rule *Rk*. The weight of an instance *Ii* covered by different rules is as follows (*Ii*, *R1*) = *Ii.w* ∗ *P* (*Ii*, *R1*)*W* (*Ii*, *R2*) = (*Ii.w* − *w* (*Ii*, *R1*)) ∗ *P* (*Ii*, *R2*)*W* (*Ii*, *Rn*) = (*Ii.w* −*n*−1*k*=1*W* (*Ii*, *Rk*)) ∗ *P* (*Ii*, *Rn*).

After we compute the CPV for instance I_i, the instance will be predicted to be of the class C_j, which has the largest probability in the class probability vector. This prediction is different from a traditional rule based classifier. When predicting the class type for an instance, a traditional rule

www.ijmer.com Vol.2, Issue.3, May-June 2012 pp-1066-1072 ISSN: 2249-6645

based classifier such as RIPPER usually predicts with the first rule in the rule set that covers the instance. As an uncertain data instance can be fully or partially covered by the multiple rules, the first rule in the rule set may not be the rule that covers it best. setBase will use all the relevant rules to compute the probability for the instance to be in each class and predict the instance to be the class with highest probability.

5. EXPERIMENTAL RESULTS

In this section, we present the experimental results of the proposed setBase algorithm. We studied the setBase classifier accuracy and classifier construction time over multiple datasets. Probability vectors which are converted by uncertain categorical attributes. For example, a categorical attribute *Ai* may have *k* possible values vj, $1 \leq j \leq k$.

For an instance *Ij*, we convert its value *Aij* into a probability vector **P** =(*pj*1, *pj*2, ..., *pji*, ..., *pjk*), while *pjl* is the probability of *Aucij* to be equal to *vl*, that is, *P*(*Aucij* = *vl*) = *pjl*. For example, when we introduce 10% uncertainty, this attribute will take the original value with 90% probability, and 10% probability to take any of the other values. Suppose in the original accurate dataset *Aij* = *v*1, then we will assign *pj*1 = 90%, and assign *pjl* ($2 \leq l \leq k$) to ensure_*ki*=2 *pjl* = 10%. Similarly, we denote this dataset with 10% uncertainty in categorical data by U10.

To make numerical attributes uncertain, we convert each numerical value to an uncertain interval. For each numerical attribute, we scan all of its value and get this maximum Xmax and minimum value Xmin, respectively.

5.1. ACCURACY

In this Fig.1, we use ten-fold cross validation. Data is split into 10 approximately equal partitions; each one is used in turn for testing while the rest is used for training, that is, 9/10 of data is used for training and 1/10 for testing. The whole procedure is repeated 10 times, and the overall accuracy rate is countered as the average of accuracy rates on each partition. 2 indicate after the prediction.

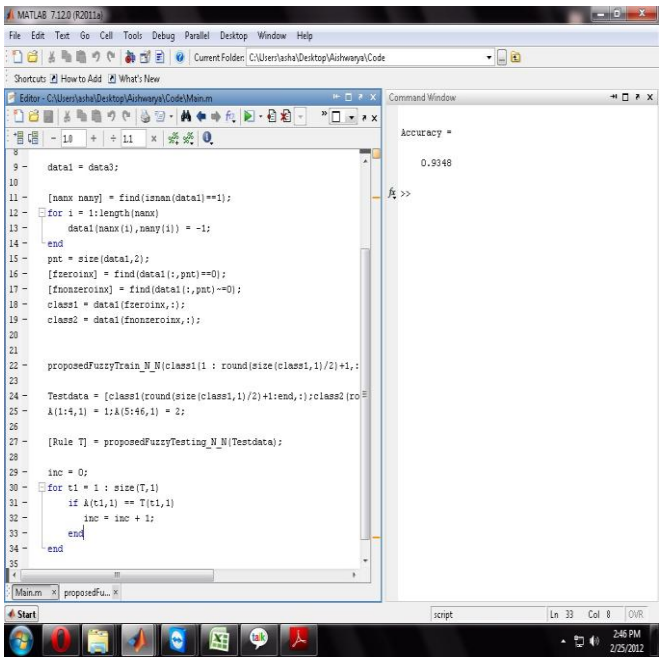

Fig.1. prediction of accuracy

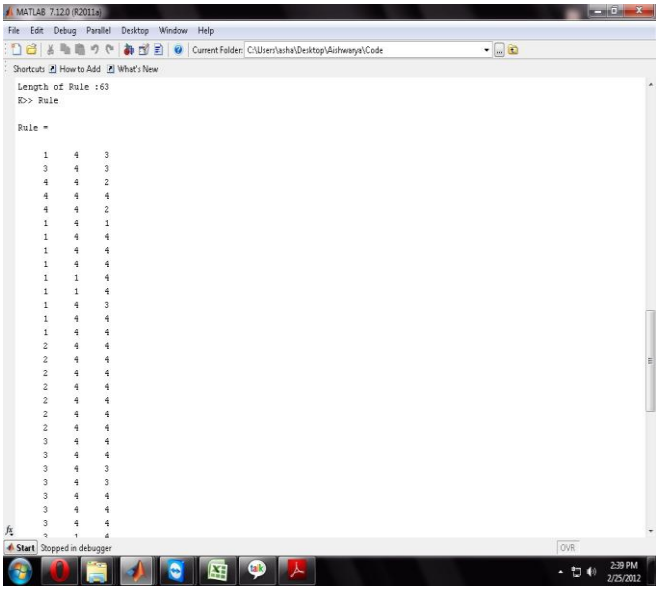

Fig.2. after prediction

5.2. COMPUTATION TIME

Table.1 depicts the absolute run time in seconds when all instances of a dataset are used to build the rule set. It is shown that it generally takes longer to construct a classifier as the uncertainty in data increases. The reason is explained in two ways of expansions. First, for uncertain data, more candidate splitting points are available and require more comparisons. Second, uncertain data can be partly covered by rules, resulting in record splitting, weight and probabilistic cardinalities computation. Furthermore, it is shown that it takes longer to generate classifier from uncertain data with Gaussian PDF than with uniform PDF.

DISTRIBUTIONS	**CLASSIFIER CONSTRUCTION TIME**					
	DATASET	***U0***	***U5***	***U10***	***U15***	***U20***
Uniform	SONAR	0.12	0.52	0.49	0.73	0.75
	GLASS	0.1	0.12	0.18	0.13	0.16
	DIABETES	0.13	0.47	0.49	0.58	0.59
Guassian	SONAR	0.12	1.1	1.73	2.0	2.62
	GLASS	0.1	0.6	0.78	0.99	1.12
	DIABETES	0.13	2.85	4.75	5.63	5.04

TABLE.1 Computation Time Calculation

6. CONCLUSION

Data Uncertainty is prevalent in many real world applications. Uncertain data often occurs in modern applications, including sensor databases, special-temporal databases, and medical or biology information systems. In this paper ,we propose a new selection based algorithm for classifying and predicting uncertain datasets. We propose new approaches for deriving optimal rules out of highly uncertain data, pruning and optimizing rules ,and class prediction for uncertain data, In this paper ,the proposed algorithm follows new paradigm of directly mining the uncertain datasets.

Our future work include developing uncertain data mining techniques for various applications, including sequential pattern mining, association mining, spatial data mining and web mining, where data can be commonly uncertain.

REFERENCES

[1] Singh,s.,Mayfield,C.,Prabhakar,s.,Shah,R.,Hambrusch, S.:Indexing *Uncertain Categorical Data.In:Proc.of ICDE 2007,pp.616-625(2007)*

[2] Qin, B., Xia, Y., Prbahakar, S., Tu, Y.: *A Rule-based Classification Algorithm for Uncertain Data. In: The Workshop on Management and Mining of Uncertain Data, MOUND (2009)*

[3] Qin, B., Xia, Y., Li, F.: DTU: *A Decision Tree for Uncertain Data. In: Theeramunkong, T., Kijsirikul, B., Cercone, N., Ho, T.-B. (eds.) PAKDD 2009. LNCS, vol. 5476, pp. 4–15. Springer, Heidelberg (2009)*

[4] Chui, C.K., Kao, B., Hung, E.: *Mining frequent itemsets from uncertain data. In: Zhou, Z.-H., Li, H., Yang, Q. (eds.) PAKDD 2007. LNCS (LNAI), vol. 4426, pp. 47–58. Springer, Heidelberg (2007)*

[5] Aggarwal, C.C., Yu, P.S.*: A survey of Uncertain Data Algorithms and Applications. IEEETransactions on Knowledge and Data Engineering 21(5), 609–623 (2009)*

[6] Tsang, S., Kao, B., Yip, K.Y., Ho, W.-S., Lee, S.D.: *Decision Trees for Uncertain Data. In: Proc. of ICDE 2009, pp. 441–444 (2009)*

[7] Chui, C., Kao, B.*: A decremental approach for mining frequent itemsets from uncertain data.In: Washio, T., Suzuki, E., Ting, K.M., Inokuchi, A. (eds.) PAKDD 2008. LNCS (LNAI), vol. 5012, pp. 64–75. Springer, Heidelberg (2008)*

[8] Leung, C.K.-S., Carmichael, C.L., Hao, B.: *Efficient mining of frequent patterns from uncertain data. In: Proc. of ICDM Workshops, pp. 489–494 (2007)*

[9] Zhang, Q., Li, F., Yi, K.: *Finding Frequent Items in Probabilistic Data. In: Proc. of SIGMOD 2008, pp. 819–832 (2008)*

[10] Bernecker, T., Kriegel, H.P., Renz, M., Verhein, F., Zuefle, A.: *Probabilistic frequent itemset mining in uncertain databases. In: Proc. of SIGKDD 2009, pp. 119–128 (2009)*

[11] Bi, J., Zhang, T.: *Support Vector Classification with Input Data Uncertainty. In: NIPS, pp.161–168 (2004)*

[12] Ngai, W.K., Kao, B., Chui, C.K., Cheng, R., Chau, M., Yip, K.Y.: *Efficient clustering of uncertain data. In: Perner, P. (ed.) ICDM 2006. LNCS (LNAI), vol. 4065, pp. 436–445. Springer, Heidelberg (2006)*

[13] Lee, S.D., Kao, B., Cheng, R.: *Reducing UK-means to K-means. In: Proc. of ICDM Work shops, pp. 483–488 (2007)*

[14] Cormode, G., McGregor, A.*: Approximation Algorithms for Clustering Uncertain Data. In: PODS 2008, pp. 191–200 (2008)*

[15] J. Catlett, "Megainduction: *A test flight," in ML, 1991, pp. 596–599.*

[16] G. Pagallo and D. Haussler, *"Boolean feature discovery in empirical learning," Machine Learning, vol. 5, pp. 71–99, 1990.*

[17] S. M. Weiss and N. Indurkhya, *"Reduced complexity rule induction," in IJCAI, 1991, pp. 678–684.*

[18] J. R. Quinlan, *"Learning logical definitions from relations," Machine Learning, vol. 5, pp. 239–266, 1990.*

[19] J. R. Quinlan and R. M. Cameron-Jones, *"Induction of logic programs: Foil and related systems," New Generation Comput., vol. 13, no. 3&4, pp. 287–312, 1995.*

[20] J.R.Quinlan, C4.5: *Programs for Machine Learning. Morgan Kaufman Publishers, 1993.*

[21] W. W. Cohen, *"Fast effective rule induction," in Proc. of the 12th Intl. Conf. on Machine Learning, 1995, pp. 115–123.*

[22] P. Langley, W. Iba, and K. Thompson, *"An analysis of bayesian classifiers," in National Conf. on Artigicial Intelligence, 1992, pp. 223– 228.*

[23] V. Vapnik, *The Nature of Statistical Learning Theory. Springer Verlag, 1995.*

[24] R. Andrews, J. Diederich, and A. Tickle, *"A survey and critique oftechniques for extracting rules from trained artificial neural networks," Knowledge Based Systems, vol. 8, no. 6, pp. 373–389, 1995.*

[25] T. G. Dietterich, *"Ensemble methods in machine learning," Lecture Notes in Computer Science, vol. 1857, pp. 1–15, 2000.*

[26] W. K. Ngai, B. Kao, C. K. Chui, R. Cheng, M. Chau, and K. Y.Yip, *"Efficient clustering of uncertain data," in IEEE International Conference on Data Mining (ICDM) 2006, pp. 436–445.*

[27] M. Chau, R. Cheng, B. Kao, and J. Ng, *"Data with uncertainty mining:An example in clustering location data," in Proc. of the Methodologiesfor Knowledge Discovery and Data Mining, Pacific-Asia Conference (PAKDD 2006), 2006.*

[28] C. G and McGregor, *"Approximation algorithms for clustering uncertain data," in PODS, 2008, pp. 191–199.*

[29] H. Kriegel and M. Pfeifle, *"Density-based clustering of uncertain data," in International Conference on Knowledge Discovery and Data Mining (KDD), 2005, pp. 672–677.*

[30] H. . Kriegel and M. . Pfeifle, *"Hierarchical density-based clusteringof uncertain data," in IEEE International Conference on Data Mining (ICDM) 2005, pp. 689–692.*

[31] A. C, *"On density based transforms for uncertain data mining," in Pro-ceedings of IEEE 23rd International Conference on Data Engineering, 2007, pp. 866–875.*

[32] A. C and Y. PS, *"A framework for clustering uncertain data streams," in Proceedings of IEEE 24rd*

International Conference on Data Engi-neering, 2008, pp. 150–159.

[33] Y. Xia and B. Xi, *"Conceptual clustering categorical data with un-certainty," in IEEE International Conference on Tools with Artificial Intelligence (ICTAI), 2007, pp. 329–336.*

[34] Z. Yu and H. Wong, *"Mining uncertain data in low-dimensional sub-space," in International Conference on Pattern Recognition (ICPR) 2006, 748–751.*

[35] C. Chui, B. Kao, and E. Hung, *"Mining frequent itemsets from uncertain data," in Proc. of the Methodologies for Knowledge Discovery and Data Mining, Pacific-Asia Conference (PAKDD) 2007, pp. 47–58.*

[36] H. H. B o ck , E . D i day, "*Anal y si s of sy mb ol i c data. E x pl or ator y me tho ds for e x tr ac ti ng stati sti c al i nfor mati on fr om c ompl e x data", Spri nge r Ve r l ag, 2000*

[37] F. Car val ho, P. B r i to, H. H. B o ck , *D y nami c c l uste r i ng for i nte r val data base d on L 2 di stanc e , Computati onal Stati sti c s, 21(2)(2006) 231-250*

[38] B . Q i n, Y. Xi a, L i F, *A baye si an c l assi fie r for unc e r tai n data, i n: P r o c e e di ngs of SAC, 2010.*