Constructing Rough Decision Forests

Qing-Hua Hu, Da-Ren Yu, and Ming-Yang Wang

Harbin Institute of Technology, China
huqinghua@hcms.hit.edu.cn

Abstract. Decision forests are a type of classification paradigm which combines a collection of decision trees for a classification task, instead of depending on a single tree. Improvement of accuracy and stability is observed in experiments and applications. Some novel techniques to construct decision forests are proposed based on rough set reduction in this paper. As there are a lot of reducts for some data sets, a series of decision trees can be trained with different reducts. Three methods to select decision trees or reducts are presented, and decisions from selected trees are fused with the plurality voting rule. The experiments show that random selection is the worst solution in the proposed methods. It is also found that input diversity maximization doesn't guarantee output diversity maximization. Hence it cannot guarantee a good classification performance in practice. Genetic algorithm based selective rough decision forests consistently get good classification accuracies compared with a single tree trained by raw data as well as the other two forest constructing methods.

1 Introduction

Overfitting and stability are persistent problems in using tree-based learner or classifier. It's proven to be a promising technique to improve the classification performance by training multiple trees and combining their decisions with a certain decision fusion scheme. This is called a decision forest. A decision forest is a classifier consisting of a collection of tree-structured classifiers and each tree casts a unit vote for the most popular class of the input [4]. Significant improvement in classification accuracy was observed, independent of tree construction algorithms. CART, ID3 and C4.5 were employed to train the component trees. Ho [1,2] proposed that combining multiple trees, which were constructed in randomly selected subspaces, can achieve nearly monotonic improvement in generalization. Breiman presented a series of techniques to produce random forests. Due to the good performance, decision forests are made to wide-range applications. Tong combined multiple independent decision tree models for prediction of binding affinity of 232 chemicals to the estrogen receptor; consistent and significant improvement in both training and testing steps was observed [5]. Hong applied the decision forests to analyze microarray data and got a high accuracy more than 95%. Applications such as handwritten word recognition [6], digital photographs classification [7], and face recognition [8] were reported in papers.

D. Ślęzak et al. (Eds.): RSFDGrC 2005, LNAI 3642, pp. 147–156, 2005.

Roughly speaking, there are two kinds of methods to construct a decision forest for a given pattern recognition task. One is to generate multiple sample subsets from the original samples. The other is to construct trees in distinct feature subspaces. We name the first one as sample subset method (SSM) and the second feature subset method (FSM). As to SSM, the most prevailing techniques are Bagging and Boosting. Bagging randomly produces several training subsets from the original sample set [3] and train multiple trees with the subsets. While boosting generates a sequence of trees, whose training sets are determined by the performance of the former ones, where training samples misclassified will be selected in the next training set with a great probability [9]. AS to FSM, Dietterich proposed a method based on random split selection, where at each node the split is selected at random from among the K best splits [10]. As there are some data sets with very high dimensionality and small size, Ho [1,2] presented a series of random-subspace methods. These techniques randomly generate some distinct subsets of the original feature set and train the classifiers in different subspaces. SSM is applicable to the case there are a lot of samples and FSM works in many-features cases. there exists a similar problem of the two methods, namely, the methods cannot guarantee the diversity between the trained trees. It is well-accepted that the performance of multiple classifier systems (MCS) depends not only on the performance of individual classifiers in the system, but also on the diversities between them. A multiple classifier system with diverse classifiers has a good potential to improve the classification performance compared with the non-diverse ones [16].

As we know, in real-world applications a lot of data is with tens, even hundreds of reducts when a rough-set based reduction is conducted. Each reduct is considered as a minimal representation of the raw decision table without loss of the information in the rough-set view of point. Therefore each reduct can be utilized to train a decision tree. Then a series of decision trees will be built. Wu [18] presented a scheme to make full use of the total reducts without selecting. Here we will propose a selective solution for the reducts. Rough decision forests are a kind of classification systems based on a team of decision trees, where each tree is trained with a rough set based reduct of a decision table, and the decisions from the trees are combined with plurality voting or majority voting. In essence, rough decision forests are a kind of multiple classifier system based on feature subset method. Compared with random forests, one advantage of the proposed method is that training trees with reducts can guarantee good performance of all of the trees, for reducts are the minimal presentations of data without information loss.

As Zhou [13] pointed out that selectively ensembling some of the trained classifiers will lead to a good performance, instead of ensembling all of the classifiers. Especially when there are a lot of classifiers at hand, combining some of them will get a perfect performance compared with combing all of the classifiers. Experiments show that selective ensemble not only reduces the size of classification systems, but also gets higher accuracies with neural network and decision tree based classifiers. Here we will present three solutions to building rough decision

forests with a philosophy that selective ensemble of some trained trees is better than the ensemble of all trees. Some numeric experiments will show the properties of the techniques. The rest of the paper is organized as follows. Section 2 will show three techniques to building a rough decision forest. Experiments will be presented in section 3. Conclusions and future work are given in section 4.

2 Three Techniques to Construct Rough Decision Forests

There are four main procedures in constructing a selective rough decision forest: reduction, training decision trees, selecting trees and combining the selected trees. Dimensionality reduction based on rough set theory is a mature technique. Several greedy reduction algorithms were proposed to search a minimal reduct or a reduct set.An extensive review about rough set based reduction and feature selection was given in [12,17]. There are several algorithms to train a decision tree. CART, ID3, C4.5, See5.0 etc perform well in different applications. Here we use CART to train decision trees. As to decision fusion, plurality voting is employed. In this paper we will focus on the techniques to select classifiers. Three methods will be presented in this section: random selection, input diversity maximization and genetic algorithm.

2.1 Random Rough Decision Forests

Ho [2] constructed decision forests in randomly selected subspaces. This technique produces diversity between classifiers by training trees with different feature vectors. As there is an exponent combination of attributes, a great number of decision trees trained in different subspaces can be built. However, how can the accuracy of an individual be guaranteed? Obviously, the performance of a decision forest depends on the individuals as well as the diversity between the trees.

Reducts of a decision table are the attribute subsets which have the same discernibility as the whole attribute set. Theoretically, this property guarantees that all of the reducts have sufficient information to build satisfactory classifiers. Although good performance of the individuals is not a sufficient condition for constructing a good decision forest, combining good individuals has more chances to build a good ensemble system. Reduction presents a feasible solution to producing multiple feature subsets.

As there are hundreds of reducts for some data sets, it is unnecessary, sometimes impossible, to train trees with all of the reducts and combine them. Random Rough Decision Forests are a kind of multiple classifier system, in which decision trees are trained by part of rough set based reducts randomly selected from the whole reducts. Compared with existing random forests, random rough decision forests are trained with rough set based reducts.

2.2 Rough Decision Forests with Input Diversity Maximization

Improvement of classification performance brought by a multiple classifier system will be contributed to the diversity among the individuals in the system [16].

The diversity makes them complement and leads to good generalization. It's observed that great diversity between individuals will make great improvement in ensemble systems. Therefore, it is vital to build decision trees with great diversity and little correlate for a successful decision forest. Bagging and boosting belong to one kind of methods to produce diversity based on different training samples. At another angle, we can make diversity among individual trees by using different training attributes. Diversity in attribute sets will lead to diversity in tree structures and then lead to diversity in decision. So finding a group of reducts with great diversity in feature set may get improvement of classification performance.

To get a great diversity in decision trees by input space diversity maximization, a clustering algorithm is designed. The reducts with similar attribute subsets are grouped into a cluster, and then some delegates of the cluster are chosen from distinct clusters.

The similarity of reducts A and B is defined as

$$SIM(A,B) = \frac{\mid A \cap B \mid}{\mid A \mid + \mid B \mid - \mid A \cap B \mid}. \tag{1}$$

where $\mid \bullet \mid$ denotes the number of elements in the set. The demonstration is shown as figure 1. The metric of similarity has the following properties:

1) $\forall A, B, SIM(A,B) \in [0,1]$;
2) $Reflexivity : \forall A, SIM(A,A) = 1$;
3) $Symmetry : \forall A, B, SIM(A,B) = SIM(B,A)$.

Assumed there are n reducts for a given data set, computing similarity between each reduct pair, a fuzzy similar relation matrix is produced:$M = (s_{ij})_{nn}$.Here s_{ij} denotes the similarity of reducts i and j, and we have $s_{ii} = 1$ and $s_{ij} = s_{ji}$.A fuzzy equivalence relation will produce when a max-min transitivity closure operation is performed on a fuzzy similarity relation. Performing a-cuts, we will get a series of partitions of the reduct sets. Reducts from different clusters have great diversity, so input space diversity maximization can be reached by selecting reducts from different branches of the clustering tree.

2.3 Genetic Algorithm Based Rough Decision Forests

Searching the optimal ensemble of multiple classifier systems, here decision forests, is a combinational optimization problem. Genetic algorithms make a good performance in this kind of problems. The performance of decision forests not only depends on power of individuals, but also is influenced by the independence between classifiers. The idea of genetic algorithms is to survive the excellent individuals with a great probability during evolution. Some excellent population will be produced after a number of inheritance, crossover and mutation.

GA rough decision forest is a selective decision tree ensemble, where individual trees are trained by a rough set based reduction, and only parts of trained trees are included in the ensemble system with a genetic algorithm based selection.

Simple genetic algorithms can be defined as an 8-tuple:

$$SGA = (C, E, P_0, M, \Phi, \Gamma, \Psi, T)$$

where C is the coding method for individual, E is the evaluating function for a gene, P_0 is the initiation population; M is the size of population;Φ is the selecting operator,Γ is crossover operator, Ψ is mutation operator and T is the terminating condition.

In this application, the binary coding is used, namely, the classifier selected is labeled with 1, otherwise 0. And the accuracy of ensemble system on validation set is taken as the fitness function, $M = 20$. The number of maximal generation is 100. The program stops when the maximal generation reaches or improvement is less than a predefined little number. Computation of finding optimal decision forests with genetic algorithm is not time-consuming in practices.

GA based rough decision forest

Input: decision table T, reducer R, learner L
Output: ensemble C
1. reduce decision table T with R, and get N reducts
2. train N classifiers with N reducts based on learner L
3. initialize genetic algorithm (GA), generate a population of bit strings
4. evolve the population with GA, and the fitness is the accuracy of the forest
5. output the best individuals and ensemble C

It requires less than 20 generations when there are 20 decision trees to be selected.

3 Experiments

Some experiments were conducted to test and compare the proposed methods. The data sets used are from UCI databases (www. Ics.uci.edu / ∼ mlearn / ML-Summary.html). The description of data is shown in table 1. There are numeric attributes in some data sets. To discretize the numeric features, a clustering operations based on FCM are conducted on each numeric attribute, and the numeric attributes are divided into three intervals. Some forest construction methods are compared. All the trees are trained with CART algorithm and two thirds samples in each class are selected as training set, the rest are test set.

The numbers of attributes of the selected data sets vary from 13 to 35, and the numbers of reducts range between 24 and 229 in table 1, which show that there are a great number of reducts for some data sets in practice. If we just make use of one of the reducts, much useful information hidden in other reducts will do no favor for the classification task. Multiple classifier system will improve classification by combining a series of classifiers.

Here, for simplicity, 20 reducts are randomly extracted from the reduct sets of all data sets at first, respectively. Subsequent experiments are conducted on

Table 1. Data set and their numbers of reducts

Data Name	Abbreviation	Size	Attributes	reducts
Dermatology Database	Der.	366	34	44
Heart Disease Databases	Heart	270	14	24
Ionosphere database	Ionos	351	35	194
Wisconsin Diagnostic Breast Cancer	WDBC	569	32	211
Wine recognition data	Wine	168	13	135
Wisconsin Prognostic Breast Cancer	WPBC	198	34	229

Table 2. Randomly select five reducts for ensemble

	Base1	Base2	Base3	Base4	Base5	Ensemble
Der	0.88793	0.75862	0.7931	0.65517	0.66379	0.8448
Heart	0.84286	0.82857	0.85714	0.74286	0.8	0.8286
Ionos	0.84158	0.9802	0.9604	0.83168	0.90099	0.9604
WDBC	0.91716	0.90533	0.95858	0.89941	0.91716	0.9290
Wine	0.70833	0.75	0.72917	0.70833	0.72917	0.8958
WPBC	0.61765	0.55882	0.58824	0.55882	0.44118	0.5882

the extracted 20 reducts. The first method is randomly to choose some of the reducts and train trees with them. In my experiments, five reducts are included in ensemble systems. The accuracy of individuals and the forests are shown in table 2.

Compared the decision forests with their individuals; it is easy to find that there is not significantly improvement at angle of classification accuracy except data wine. We can conclude that a fine decision forest should be elaborately constructed. Roughly combining some classifiers will even produce performance decrease. Therefore, it is important to select a number of appropriate classifiers to build an effective decision forest.

Table 3 shows the classification accuracy of decision forests with input space diversity maximization. Three, five or seven reducts are selected from the clustering trees, respectively.

Size and accuracy of three kinds of rough decision forests are shown in table 4. The first one is size and accuracy of ensemble system combining all decision trees at hand. The second are with genetic algorithms and the third are with n top classifiers, n is the corresponding size of GAS decision forests.

We can get that GA rough decision forests make a good performance not only compared with the forests combining all trees, but also with the forests combining the several best classifiers. Moreover, consistent improvement is observed for each data set.

Accuracy of individual classifiers and the selected classifiers are showed as figures 2,3,4. We find that genetic algorithms don't choose all of the classifiers with best performance. While some bad decision trees are included in the forests,

Table 3. Select reducts with input diversity maximization

	Base1	Base2	Base3	Base4	Base5	Base6	Base7	Ensemble
Der	0.7155	0.8017	0.7155	0.6207	0.8966	-	-	0.8362
Heart	0.8429	0.7714	0.8571	0.8429	0.8143	0.7857	0.8286	0.8571
Ionos	0.8317	0.8614	0.9406	0.9208	0.9901	-	-	0.9307
WDBC	0.8935	0.8935	0.9053	-	-	-	-	0.9349
Wine	0.7917	0.8125	0.8542	-	-	-	-	0.9583
WPBC	0.5588	0.5294	0.6765	0.5441	0.5588	0.6765	0.6176	0.6324

Table 4. Comparison of decision forests

	All		GAS		Top	
	size	accuracy	size	accuracy	size	accuracy
Der	20	0.9052	10	0.9310	10	0.91379
Heart	20	0.8429	6	0.8857	6	0.85714
Ionos	20	0.9406	8	0.9901	8	0.94059
WDBC	20	0.9349	7	0.9704	7	0.94675
Wine	20	0.7292	7	1	7	0.97917
WPBC	20	0.5882	9	0.75	9	0.70588
Aver.	20	0.8235	7.8333	0.9212	7.8333	0.8906

Table 5. Comparison of classification accuracy based on different methods

	Original	Min-reduct	Forest 1	Forest 2	Forest 3	All-reducts	GAS
Der	0.9483	0.7155	0.8448	0.8362	0.91379	0.9052	0.9310
Heart	0.8429	0.7429	0.8286	0.8571	0.85714	0.8429	0.8857
Ionos	0.8119	0.8317	0.9604	0.9307	0.94059	0.9406	0.9901
WDBC	0.8757	0.8757	0.9290	0.9349	0.94675	0.9349	0.9704
Wine	0.7083	0.7083	0.8958	0.9583	0.97917	0.7292	1.0000
WPBC	0.5735	0.5147	0.5882	0.6324	0.70588	0.5882	0.7500
Aver.	0.7934	0.7315	0.8411	0.8583	0.8906	0.8235	0.9212

which shows the performance of individuals can not guarantee the performance of decision forests.

The comparison of all classification accuracy is shown in table 5. Original denotes classification accuracy with a single decision tree trained by the original data set. Min-reduct denotes the results with a single tree trained by the minimal reduct. Decision forest with random reduct selection is labeled as forest 1; decision forest with input space diversity maximization is labeled as forest 2, decision forests with some top classifiers is labeled as forest 3 and forests with GA selective ensemble is labeled as GAS. All-reducts denotes the performance of

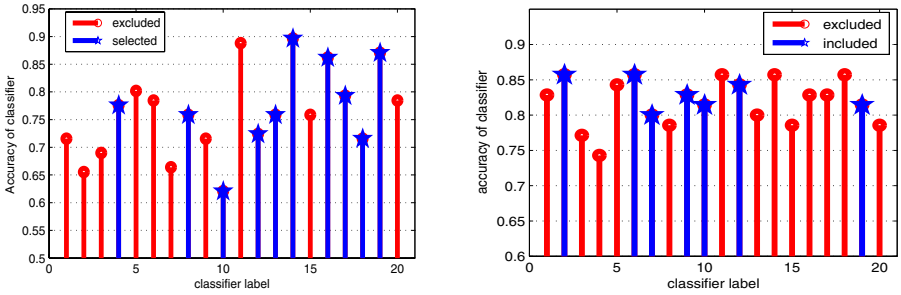

Fig. 1. Left: Der data set; ten trees are included in the decision forest. Right: Heart data set; seven trees are included in the decision forest.

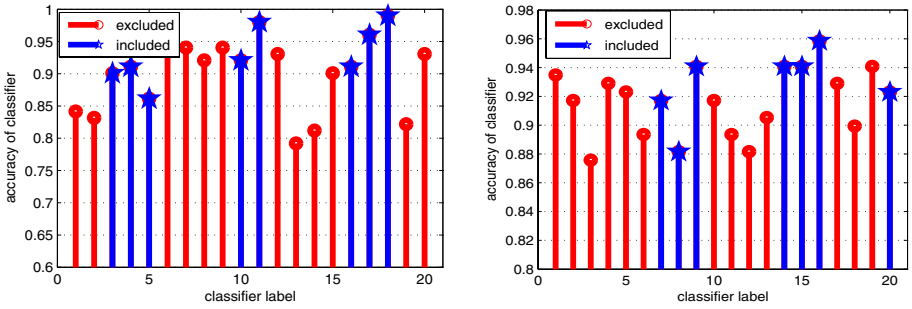

Fig. 2. Left: Ionos data set; 8 decision trees are included in the decisionforest. Right: WDBC data set; 7 decision trees are included in the decision forest.

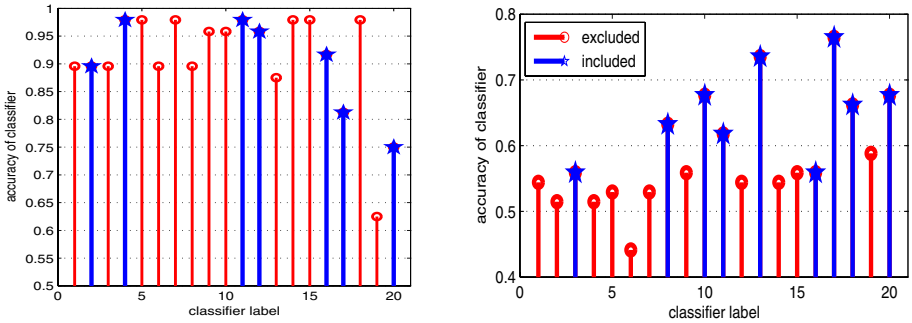

Fig. 3. Left: Wine data set; 7 decision trees are in included in the decision forest. Right: WPBC data set; 9 decision trees are included in the forest.

the decision forests combining all decision trees. We can find the decision trees trained with minimal reducts make a worst performance as for all of data sets. The best classification paradigm is GA rough decision forest, which get nearly 20% improvements relative to the minimal reducts. The second is decision forests constructed with the best decision trees, which means that the performance of individuals has great influence on the forests, but is not a sufficient condition. The average accuracy of the forests made up of all trees is worse than that of other ensemble, which shows selective ensemble is a good solution for MCS.

4 Conclusion and Future Work

Some techniques to construct rough decision forests are proposed in this paper. On one hand, as a generalization of random forests, rough decision forests can guarantee the performance of individual trees; on the other hand, rough decision forests can make full use of the output of reduction algorithms. Multiple trees are trained with different reducts and a decision forest is constructed by combining some of the trained trees. Three methods are proposed to construct a selective rough decision forest. Random selection, input space diversity maximization and genetic algorithms are tried and compared. The experimental results show that the selective decision forests are better than the decision forests combining all classifiers. GA rough decision forests get the best performance relative to the other forests.

Although good performance is produced in the experiments, the explanation of GA rough decision forests as good multiple classification systems is not clear. There are several diversity measures to analyze the properties of multiple classifier systems. We will cover a way to understand the behaviors of rough decision forests and try to propose a method for getting a consistent improvement.

References

1. Ho T. K.: Random decision forests. 3rd international conference of document analysis and recognition. 278–282
2. Ho T. K.: The random subspace method for constructing decision forests. IEEE Transactions on pattern analysis ans machine intelligence. **20** (1998) 832–844
3. Breiman L.: Bagging predictors. Machine Learning. **26** (1996) 123–140
4. Breiman L.: Random forests. Machine learning. **45** (2001) 5-32
5. Tong W., Hong H., Fang H., Xie Q., Perkins R.: Decision forest: combining the predictions of multiple independent decision tree models. Journal of chemical information ans computer sciences. **43** (2003) 525-531
6. Gunter S., Bunke H.: Handwritten word recognition using classifier ensembles generated from multiple prototypes. International journal of pattern recognition and artificial intelligence. **18** (2004) 957-974
7. Schettini R.: Automatic classification of digital photographs based on decision forests. International journal of pattern recognition and artificial intelligence. **18** (2004) 819-845

8. Cheng J., Liu Q., Lu H., et al.: Random independent subspace for face recognition. Lecture notes in computer science. **3214** (2004) 352-358
9. Freund Y., Schapire R. E.: a decision-theoretic generalization of on -line learning and application to boosting. the 2nd European conference on computational learning theory. (1995) 23-37
10. Dietterich T.: An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting and randomization. Machine Learning. (1998) 1-22
11. Gunter S., Bunke H.: Feature selection algorithms for the generation of multiple classifier systems and their application to handwritten word recognition. Pattern recognition letters. **25** (2004) 1323-1336
12. Swiniarski R., Skowron A.: Rough set methods in feature selection and recognition. Pattern recognition letters. **24** (2003) 833-849
13. Zhou Z., Wu J., Tang W.: Ensembling neural networks: Many could be better than all. Artificial intelligence. **137** (2002) 239-263
14. Hu Q., Yu D., Bao W.: Combining multiple neural networks for classification based on rough set reduction. Proceedings of the 2003 International Conference on Neural Networks and Signal processing. **1** (2003) 543 - 548
15. Hu Q., Yu D.: Entropies of fuzzy indiscernibility relation and its operations. International Journal of uncertainty, fuzziness and knowledge based systems. **12** (2004) 575-589
16. Kittler J., Hatef M., Duin R., et al.: On combining classifiers. IEEE Transactions on pattern analysis ans machine intelligence. **20** (1998) 226-239
17. Wang J., Miao D.: Analysis on attribute reduction strategies of rough set. Journal of computer science and technology. **13** (1998) 189-193
18. Wu Q., Bell D., Mcginnity M.: Multiknowledge for decision making. Knowledge and information systems. **7** (2005) 246-266