LETTER

Specific Random Trees for Random Forest

Zhi LIU[†a)], ***Student Member***, **Zhaocai SUN**[††], ***and*** **Hongjun WANG**[†], ***Nonmembers***

SUMMARY In this study, a novel forest method based on specific random trees (SRT) was proposed for a multiclass classification problem. The proposed SRT was built on one specific class, which decides whether a sample belongs to a certain class. The forest can make a final decision on classification by ensembling all the specific trees. Compared with the original random forest, our method has higher strength, but lower correlation and upper error bound. The experimental results based on 10 different public datasets demonstrated the efficiency of the proposed method.
key words: *random forest, multiclass classification, specific random trees*

1. Introduction

Learning method that combines decision trees using several bootstrap samples randomly, Random forest (RF), is a popular ensemble learning methodRF has been applied successfully in several fields, such as data classification and data mining to its efficiency [2]. The generalization error of a forest depends on the strength of the individual tree and the correlation between the trees. If the strength is greater and/or the correlation is smaller, the generalization error is lower and the performance of the RF is better.

With its tree structure, RF can handle both binary and multiclass problems. However, in a multiclass problem, different classes interact with each other, the grown trees are complex, and the performance of the classifier may be affected [2].

To address this problem, we propose a novel method using specific trees to generate a forest. The structure of our specific tree is simple. It only estimates whether or not a new object belongs to a specific class. The advantage of this method is its ability to decrease the correlation between individual trees. According to the theory of Breiman [1], the upper error bound of our forest will also be decreased. Experiments have also verified that our forest outperforms the original forest method on multiclass classification problem.

2. Proposed Algorithm

For multiclass classification, the original RF theories use each tree of the random forest as a multi-classifier for all instances. In the current study, the concept of specific random tree (SRT), which is a decision tree built for a certain class, i.e., SRT, was proposed and used for justifying whether a sample belongs to a certain class. The voting based on all trees makes the final classification results. In other words, the basic classifier in the proposed algorithm focuses on solving binary classification, but not multiclass classification. Meanwhile, the independency among the specific trees will be strengthened relative to the original RF. Therefore, enough specific random trees can obtain satisfactory classification performance. The proposed algorithm is called specific random forest (SRF).

Our algorithm can be described as follows:

1) Let n be the size of the original training set T. n instances are randomly drawn with replacement to form the bootstrap [3] samples.

2) A training data subset T_i ($i = 1 \dots N$, where N is the number of the trees) is randomly sampled for each specific random tree corresponding to a certain class C_i ($i = 1 \dots N$).

3) Based on T_i, SRT is built according the following rules: Let m be the dimensionality of the original feature space and the number of the features for classification $k \in [1 \dots m]$ be a preliminary fixed parameter. For the node of the tree, a subset of k features is randomly drawn without replacement. The best split is then selected among these features. Only one node exists for each tree. Therefore, the two groups split by the binary classifier belong to C_i and $\overline{C_i}$ respectively, where $\overline{C_i}$ denotes the class, including the data that does not belong to C_i. The tree is then built without pruning.

4) For a certain class C_i, if a random testing sample belong to C_i, then it is labeled as 1, otherwise, it is labeled as 0.

The main difference between our method and Breiman's original RF [2] is that SRT is introduced to generate the forest. This difference can be illustrated by a triple-class classification in Fig. 1. Each tree is a triple-class classifier according to the original RF, as shown in Fig. 1 (a). They can classify an object into one of the three classes (circle, plus, or triangle). The forest ensembling the trees will outperform any single tree. The forest is generated by the SRT, as shown in Fig. 1 (b). Each specific tree is a binary classifier for a certain class. For example, Tree 1 (the red circles) only estimates whether or not an object belongs to the class of circles. The correlations between the trees are decreased; thus, our method outperforms the original RF.

Manuscript received July 4, 2012.
Manuscript revised October 22, 2012.
[†]The authors are with School of Information Science and Engineering, Shandong University, Jinan, China.
[††]The author is with the Department of Computing, Harbin Institute of Technology, Shenzhen Graduate School, Shenzhen, China.
a) E-mail: liuzhi@sdu.edu.cn
DOI: 10.1587/transinf.E96.D.739

Copyright © 2013 The Institute of Electronics, Information and Communication Engineers

3. Experimental Results

The datasets used in these experiments including 10 UCI datasets are described in Table 1 [4]. The full experimental protocol is described as follows. Each original dataset was randomly split into two subsets, with one third of the samples used for training and the remaining third for the test. This splitting procedure has been repeated 20 times; thus, 20 different training and testing sets are available. The datasets were denoted as $X_j = (X_{rj}, X_{sj})$, such a split, with $j \in [1, \ldots, 20]$, where X_{rj} and X_{sj} stand for the training set and the testing set, respectively. For each X_j, a SRF is grown from X_{rj}, with the number of trees fixed to 100, 200, 300, 400, and 500 respectively. The value of the hyperparameter k can be fixed as $\sqrt{m}$, a default value commonly used in literature [5].

Table 2 presents the classification accuracy obtained by the original RF and SRF on each dataset with different number of trees. Table 2 also shows that SRF can achieve better performance on most of the datasets when $N \geq 300$. This observation indicates that SRF can improve the accuracy of multi-class classification. The second observation based on

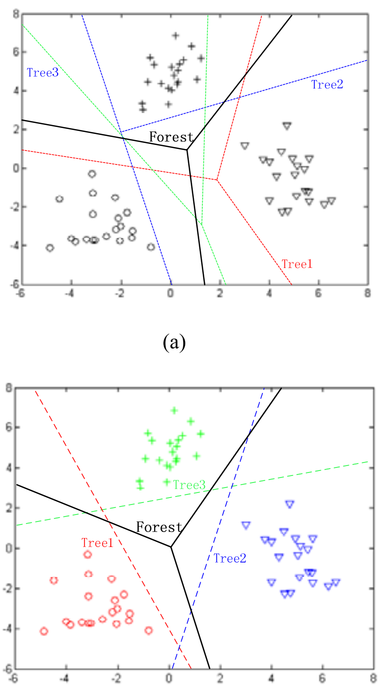

Fig. 1 Illustration for the difference between SRF and original RF. ∘, +, and ∇ represent the data in different classes respectively. (a) Operation of the original RF; (b) Operation of the SRF.

Table 1 Properties of the datasets for experiments.

Dataset	#Instances	#Attributes	#Classes
anneal	898	38	6
balance-scale	625	4	3
heart-h	294	13	5
iris	150	4	3
lymph	148	18	4
machine	209	7	8
segment	2310	19	7
vehicle	848	18	4
waveform-5000	5000	40	3
zoo	101	17	7

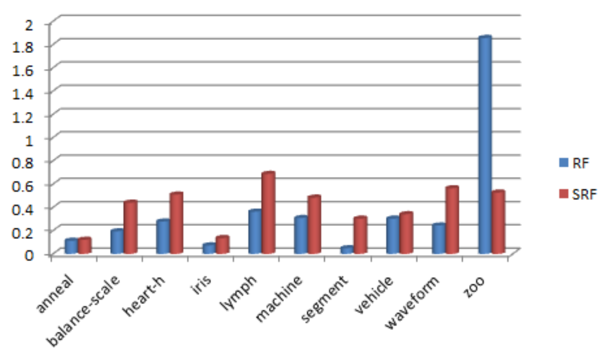

Fig. 2 Standard deviation bar of the accuracy rate of data classification with RF and SRF with the 10 different datasets.

Table 2 Accuracy rate of data classification with RF and SRF.

Dataset	#Trees=100		#Trees=200		#Trees=300		#Trees=400		#Trees=500	
	RF	SRF	RF	SRF	RF	SRF	RF	SRF	RF	SRF
anneal	99.22	**99.35**	99.28	99.25	99.35	**99.45**	99.47	**99.51**	99.52	**99.60**
balance-scale	80.49	**80.77**	80.16	**81.81**	79.97	**81.81**	80.01	**81.89**	80.32	**81.94**
heart-h	78.72	78.32	79.32	79.02	79.32	**79.49**	79.39	**79.51**	79.53	**79.76**
iris	95.10	**95.29**	95.29	**95.49**	95.10	95.10	95.18	**95.39**	95.22	**95.43**
lymph	80.76	**82.30**	81.13	**83.56**	81.57	**83.93**	81.63	**84.04**	81.72	**84.21**
machine	89.04	88.46	89.88	89.59	89.60	89.31	89.72	89.68	89.87	89.81
segment	97.99	97.48	97.92	**97.97**	97.99	**98.19**	98.04	**98.25**	98.06	**98.31**
vehicle	74.01	**75.26**	74.74	**75.71**	74.63	**74.94**	74.65	**74.98**	74.77	**74.99**
waveform	84.65	84.24	84.87	**85.16**	85.11	**85.28**	85.24	**85.98**	85.31	**85.44**
zoo	79.71	**93.86**	84.44	**93.35**	83.57	**94.45**	84.41	**94.51**	84.62	**94.83**

Table 2 indicates that the accuracy rate exhibited an increasing tendency with the number of trees using our method. Table 2 also shows that the increase in the number of trees for the original random forests does not always mean that the performance of the forest is significantly better than previous forests (fewer trees) [6]. In addition, the statistical error bar (standard deviation) with the 10 different datasets is presented (shown in Fig. 2).

4. Conclusion

SRF was proposed in this study to improve the performance of multi-class classification with the concept of random forest. As the main feature of our method, in multi-class classification, each tree of the random forest was used as a binary classifier for a certain class, but not as a multi-class classifier used in the original RF. The experimental results demonstrated that our method outperforms the original RF, especially on largescale datasets with more trees. Moreover, using SRT in these experiments has highlighted interests in designing forest method by considering the trees in a more independent and specific way than it is done in "classical" RF induction methods.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (No.60902068), and Independent Innovation Foundation of Shandong University, IIFSDU, (No.2012JC015).

References

[1] L. Breiman, "Random forests," Mach. Learn., vol.45, pp.5–32, 2001.
[2] A. Verikas, A. Gelzinis, and M. Bacauskiene, "Mining data with random forests: A survey and results of new tests," Pattern Recognit., vol.44, pp.330–349, 2011.
[3] L. Breiman, "Bagging predictors," Mach. Learn., vo.24, pp.123–140, 1996.
[4] A. Asuncion and D. Newman, "UCI machine learning repository," 2007. [Online]. Available: http://archive.ics.uci.edu/ml/
[5] S. Bernard, Laurent Heutte, and S. Adam, "On the selection of decision trees in random forests," IEEE International Conference on Neural Networks (ICNN), pp.302–307, 2009.
[6] T.M. Oshiro, P.S. Perez, and J.A. Baranauskas, "How many trees in a random forest?," Machine Learning and Data Mining in Pattern Recognition, Lect. Notes Comput. Sci., vol.7376, pp.154–168, 2012.