Improving Classification Accuracy in Random Forest by Using Feature Impurity and Bayesian Probability

Cuong-Nguyen[1], HaNam-Nguyen[2], Wang Yong[1]
[1] College of Economy and Business Administration, Chongqing University, Chongqing , PR China
[2]College of Technology, Vietnam National University, Hanoi, Vietnam

Abstract - *Improved accuracy in data mining tasks is one of the important issues that have been being seized a great attention of many researchers in recent years. The dense forest of the algorithm in data mining generally and data classification specifically, Random Forest seems to be a promising method to implement classification tasks for high-dimensional dataset. In this paper, we use Random, feature impurity and estimation of Bayesian probability as the cardinal elements to build feature ranking formula. After that, we gradually eliminate the feature of lowest position in feature ranking list and compare classification accuracy before and after this elimination. In this way, we build up a best feature subset for the classifier. We conducted the experiments on two public datasets. The results of those experiments show that our proposed method is better than original method as well as some other popular methods, both in classification accuracy and stability.*

Keywords: Data mining, classification, random forest, accuracy, improve

1 Introduction

Improving accuracy in classification tasks is one of interesting topics in data mining. In last several years, the topic has been grasping a lot of attentions from many researchers all over the world. In 2001, Leo Breiman [1] proposed a new algorithm called Random Forest, this algorithm is new approach to data exploration, data analysis, and predictive modeling. This algorithm is combination of three components [2]: (1) CART, (2) Learning ensembles, committees of experts, combining models and (3) Bootstrap Aggregation. Experiments prove that Random Forest's performance is better other previous methods such as: AdaBoost, SVM, Neural Network, C45 [3], etc. Especially, many researchers experimental work [4-5] has proved that Random Forest seems to be very effective in dealing with high-dimensional dataset. The experimental results inspired other researchers, [3, 6-8] tried to improve Random Forest to higher level of classification accuracy as well as to eliminate redundant and noisy features in the classifier and they achieved some remarkable successes.

In this paper, we propose a model by the combination of random forest algorithm, feature impurity (GINI index) and Bayesian probability to improve classification accuracy of the classifier in Random Forest. At the first glance, the method seems akin to method DEF-RF proposed by HaNam-Nguyen et all [3]. Actually, the proposed method is an improvement of DEF-RF and RF to get increase classification accuracy of algorithm, especially in case of imbalance classes which is not in the scope DEF-RF algorithm.

The paper is sectioned as follow: section 2 and section 3 briefly introduce Random Forest algorithm and Bayesian probability, respectively. The proposed method will be presented in section 4, the experimental results will be discussed in section 5. The last section is the conclusion.

2 Random Forest

As mentioned above Random Forest is combination of three components: CART, Learning ensembles, committees of experts, combining models and Bootstrap Aggregation. How does Random Forest deals with classification tasks? To classify a new object from an input vector, put the input vector down each of the CARTs in the forest. Each CART gives a classification, and Random Forest asks the trees "votes" for that class. The forest chooses the classification having the majority votes [1, 9].

In Random Forest each CART is grow as follows:

- If the number of cases in the training set is N, sample N cases at random - but with replacement, from the original data. This sample will be the training set for growing the tree.
- If there are M input variables, a number m<<M is specified in such way that at each node, m variables are selected at random out of the M and the best split is used to split the node. The value of m is held constant during the forest growing. For example if we have a 200 column of predictors, typically we select square root (200), it means we will select only 14 predictors, then we split our node with the best variable among the 23, not the best variable among the 200

- Each tree is grown to the largest possible extent. There is no pruning.

The notable thing here in Random Forest is GINI index, in Random Forest GINI index is used as the splitting criterion and defined as squared probabilities of membership for each target category in the node.

$$\mathrm{GINI}\left(\mathrm{N}\right)=\frac{1}{2}\left(1-\sum_{j}\mathrm{p}(\mathrm{w}_{j})^{2}\right) \qquad (1)$$

Where $p(\omega_j)$ is the relative frequency of class ω_j at node N. It means if all the samples are on the same category, the impurity is zero, otherwise it is positive value. In this paper we will use GINI index as a first key element to build the features ranking formula that will be discussed in chapter 4.

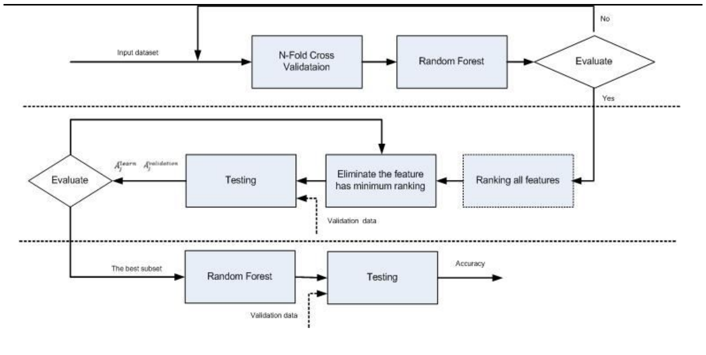
3 Bayesian Probability

Bayesian probability [10] is named after English scientist, Thomas Bayes, who did early work in probability and decision theory during the 18th century. Assume X is an entity and X is described by measurements made on a set of n attributes. H is any hypothesis, such as X belongs to a specified class A. For classification tasks, we want to determine P(H|X), the probability that X belongs to class A, given that we know the attribute description of X. In Bayesian terms, P(H|X) is called the posterior probability of H conditioned on X. In contrast, P(H) is the prior probability of H. The posterior probability, P(H|X), is dependent of X whereas prior probability, P(H), which is independent of X.

Similarly, P(X|H) is the posterior probability of X conditioned on H and P(X) is the prior probability of X. The answerable question is “How can we estimate these probabilities?”. Bayes provides an effective method to estimate the probabilities. In practical P(H), P(X|H), and P(X) could be estimated from the given data and the estimation of P(H|X) depends on P(H), P(X|H), and P(X) as follow.

$$\mathrm{P}\left(\mathrm{H|X}\right)=\frac{\mathrm{P}\left(\mathrm{X|H}\right)\mathrm{P(H)}}{\mathrm{P(X)}} \qquad (2)$$

4 Proposed Method



Our proposed method comprises of four steps akin to DEF_RF algorithm:

1. *Train data by Random Forest with the cross validation*
2. *Calculate the ranking criterion for all features F_i, $i=1...n$.*
3. *Remove a feature by using Dynamic Feature Elimination function.*
4. *Back to step 1 until reach the desired criteria.*

In step 1, we use Random Forest with n-fold cross validation to train the classifier. In jth cross validation, we obtain a set of (F_j , $A_{j,k=1..m}^{learn}$, $A_{j,k=1..m}^{validation}$). In which, F_j, $A_{j,k=1..m}^{learn}$ and $A_{j,k=1..m}^{validation}$ is feature importance, the learning accuracy of class kth and the validation accuracy of class kth respectively. For example if we need to classify a dataset into 2 classes, using Random Forest with n-fold cross validation in jth cross validation we will obtain a set of (F_j, $A_{j,1}^{learn}$, $A_{j,2}^{learn}$, $A_{j,1}^{validation}$, $A_{j,2}^{validation}$). The classification accuracy of the classifier on class kth is calculated as follow:

$$A_{j,k}=\frac{Number\ of\ \ features\ of\ \ j^{th}\ cross\ validation\ classified\ correctly\ on\ class\ k}{Number\ of\ \ features\ of\ \ class\ k} \qquad (3)$$

In step 2, we will setup a feature ranking formula that is use to rank all features in the dataset. This step is the most important step in our algorithm. It is indispensable to mention that our proposed method uses feature ranking formula as key factor to determine as which feature should be eliminated firstly. In other words, the feature ranking formula will help us in determining which feature may be a noisy/redundancy feature. If a feature has high ranking in the dataset then it will be a useful feature for classifier and otherwise. The weakness of feature ranking formula will lead to the weakness of proposed algorithm because this problem will lead time-consuming of algorithm and other related issues. This problem will be discussed in step 3 in detail.

In reality, a simply method usually will use when we judge whether the feature is useful to the classifier or not (that is classification accuracy of the classifier). The method can best be understood as follow: we add a feature into the classifier and assess classification accuracy of the classifier before and after (add the feature). However, in our situation the question is that how can we have a good estimation of classification accuracy? Especially, in case of high-dimensional dataset, the dataset is classified into many classes and the number of features in each class is very different. In other words, we need to deal with a difficult case in classification tasks then with imbalance classes. In order to deal with this issue, within the scope of this paper we will use Bayesian probability to estimate classification accuracy of the classifier.

Now assume that there are m classes, C_1, C_2, ..., C_m. Given an entity X, X is depicted by m features. According to Bayesian probability, the probability that X belongs to the class C_i is estimated as follow:

$$P\left(C_i|X\right) = \frac{P\left(X|C_i\right)P(C_i)}{P(X)} \qquad (4)$$

P(X) is constant for all classes because we know that the probability of an entity can be classified in to a class are the same, so that only $P(X|C_i)P(C_i)$ need to be estimated. According to Bayes' suggestion in case the prior probabilities of the class are unknown, then it is commonly assumed that prior probabilities of all the classes are equally or in other word we have P(C$_1$) = P(C$_2$) =…= P(C$_m$), and we therefore only need estimate $P(X|C_i)$.

We know that with the given dataset of many attributes, it would be extremely computationally expensive to estimates $P(X|C_i)$. In order to reduce computation in evaluating P$(X|C_i)$, the naive assumption of conditional independence of class is made. This presumes that the values of the attributes are conditionally independent of one another. Thus,

$$P\left(X|C_i\right) = \prod_{k=1}^{n} P\left(X_k|C_i\right) \qquad (5)$$

Or $$P\left(X|C_i\right) = P\left(X_1|C_i\right).P\left(X_2|C_i\right)...P\left(X_n|C_i\right) \qquad (6)$$

$$P(C_i \mid X) = \prod_{k=1}^{n} P(X_k \mid C_i) = P\left(X_1|C_i\right).P\left(X_2|C_i\right)...P\left(X_n|C_i\right) \qquad (7)$$

From (4) and (8) we propose a way to estimate classification accuracy of the classifier on learning set as follow:

$$\overline{A}_j^{learn} = \prod_{k=1}^{n} A_{j,k}^{learn} \qquad (8)$$

Similarly, classification accuracy of the classifier on validation set:

$$\overline{A}_j^{validation} = \prod_{k=1}^{n} A_{j,k}^{validation} \qquad (9)$$

We propose a new feature ranking formula for feature ith base upon the calculations above

$$F_i^{rank} = \sum_{j=1}^{n} F_{i,j}\, x \frac{1}{\left(\overline{A}_j^{learn} - \overline{A}_j^{validation}\right)^2 + \varepsilon} \qquad (10)$$

Where:

- j=1,.., n is the number of cross validation folders,
- F$_{i,j}$ is GINI index,
- $\overline{A}_j^{learn}$, $\overline{A}_j^{validation}$ is general accuracy of classifier on learning set and validation set,
- ε is the real number with very small value.

The feature ranking formula includes two elements: (1) the first element is GINI index, the element decreases for each feature over all trees in the forest when we train data by Random Forest; (2) the second element is fraction, nominator of the fraction is constant, equals to 1, denominator of the fraction equals $\left(\overline{A}_j^{learn} - \overline{A}_j^{validation}\right)^2 + \varepsilon$, presents the variance between classification accuracy of classifier on learning set and validation set. That means the smaller variance, the better features we have. The combination between GINI index and the fraction presents our expectation: higher ranking feature are better feature. The ε is used to deal the case $\overline{A}_j^{learn}$ equals $\overline{A}_j^{validation}$, in this case $F_i^{rank} = \sum_{j=1}^{n} F_{i,j}$.

After finishing the step 2, we have an ordered list of ranking features. The list will use in step 3 to determine optimal features of the classifier. One should be noted that feature assessing procedure is the correlation among features. We know that a feature may have a low position in feature ranking list but when it is use concurrently with other features they will bring a great contribution to classification accuracy of the classifier. One feasible way to deal with this issue is to use feature elimination strategy which is the next step (step 3) of our proposed method.

In step 3, same as DFE-RF, we also use dynamic feature elimination strategy to eliminate noisy/redundant features. In this step, we will use feature ranking list as a standard criterion to determine which feature should be eliminate first. In other words, the feature of lowest position in feature ranking list will eliminate first. At each step in feature eliminating procedure we will validate the classification accuracy of the classifier. Purpose of the validation is to determine whether the feature to be eliminated is actually redundancy/noisy feature or not. We can perform the validation by comparing the classification accuracy of the classifier before and after eliminating the feature. If classification accuracy of the classifier before eliminating feature is greater than classification accuracy of the classifier after eliminating feature then feature will be kept or otherwise. This iteration will terminate whenever classification accuracy of new subset is higher than classification accuracy of previous subset. Our algorithm will stop when we cannot find out better classification accuracy or no feature to eliminate. In this case the current subset is the best subset we can have. Otherwise, in term of n-fold cross validation the procedure will jump back to step 1 (step 4).

5 Experimental Results

We use R-language[11] as programming language and Random Forest package[12] to validate our proposed method. We validate our proposed method on two public datasets: Medalone dataset, and Colon cancer dataset. In our experiment each dataset was randomly divided into two subsets called learning set and validation set. Random Forest and our proposed method (RF_CT) is executed on both two subset, the achievement we have after execute Random Forest and RF_CT will be used to evaluate classification performance of each method.

5.1 Madelone

MADELON is an artificial dataset containing data points grouped in 32 clusters placed on the vertices of a five dimensional hypercube and randomly labeled +1 or -1 [13]. This dataset is one of five datasets use in the NIPS 2003 feature selection challenge. Actually, Madelon is matrix of 4000 rows x 500 columns that equally a dataset of 4000 instances in which each instance includes 500 attributes.

Table 1: The comparison of some statistical parameters between RF and RF_CT on learning set and validation set of Medalon dataset through 50 testing times with number of trees in RF n=100,150, 200 and 250.

	Mean (%)	Standard Deviation	Min (%)	Max (%)
n=100				
RF	71.24	1.79	66.50	76.50
RF_CT	87.29	0.56	86.00	88.67
n=150				
RF	72.26	1.29	69.67	75.83
RF_CT	87.47	0.05	86.33	88.33
n=200				
RF	72.50	1.32	68.67	74.83
RF_CT	87.70	0.06	86.17	89.00
n=250				
RF	73.11	1.28	70.33	75.83
RF_CT	87.55	0.06	86.00	88.83

Table 1 shows experimental results after 50 testing times with replacements of Random Forest parameter, n, number of trees in Random Forest on Medalon dataset. Generally, we see that proposed method shows better performance than original method both in classification accuracy and stability. In the best case, under framework of our experiment, our method reaches classification accuracy of 87.70±0.06 that seem to be much better than original Random Forest (72.50±1.32).

Figure 1 shows the comparison of classification accuracy between Random Forest and RF_CT on Medalon dataset through 50 testing times and number of trees in Random Forest n=100, 150, 200 and 250 respectively.

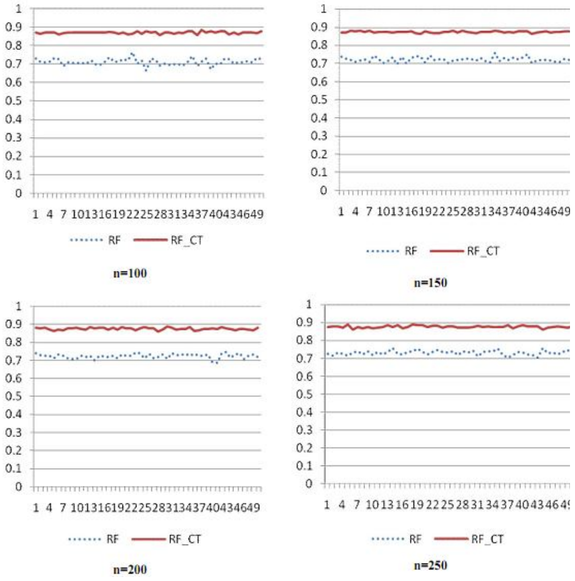


Figure 1: The comparison of classification accuracy between RF and RF_CT on Medalone dataset through 50 testing times and number of trees in RF n=100, 150, 200 and 250.

In last couple of years, some researchers have tested their proposed methods on Madelone dataset [14] and they have achieved some successes but our proposed method seem to be superior on Madelon dataset if the methods are evaluate based on two criterion: classification accuracy and stability.

Table 2: The comparison of classification accuracy among some methods on Madelone dataset

Method	Classification Accuracy (%)	Standard Deviation
Naïve Bayes	58,3	1,5
C45	69,8	4,7
GOV	71,2	2,9
DOG	71,4	2,6
RF_CT	87,7	0,6

5.2 Colon Cancer

Colon Cancer is also a public dataset [15]. The data set contains 62 samples collected from colon-cancer patients. Among them, 40 tumor biopsies are from tumors (labeled as “negative”) and 22 normal (labeled as “positive”) biopsies are from healthy parts of the colons of the same patients. Table 3 shows experimental results of Random Forest and RF_CT on the dataset.

Table 3: The comparison of some statistical parameters between RF and RF_CT on learning set and validation set of Colon cancer dataset through 20 testing times with number of trees in RF n=800,1100, 1400 and 1700.

	Mean (%)	Standard Deviation	Min (%)	Max (%)
n=800				
RF	76.17	7.03	56.67	90.00
RF_CT	87.17	5.44	76.67	96.67
n=1100				
RF	78.67	5.76	63.33	86.67
RF_CT	87.17	4.98	76.67	93.33
n=1400				
RF	76.33	8.30	56.67	86.67
RF_CT	86.83	4.52	80.00	93.33
n=1700				
RF	78.17	6.71	66.67	93.33
RF_CT	88.17	3.82	8.00	93.33

Through 20 testing times on Colon cancer dataset our proposed method also presents an impressing achievement in comparison with Random Forest. In case of 1700 of number of trees in Random Forest our method archives the classification accuracy of 88.17±8.82 meanwhile Random Forest is only 78.17±6.71. In other cases that number of trees in Random Forest are 800, 1100 and 1400 our method shows better classification accuracy than Random Forest: 87.17±4.98 and 76.17±7.03, 87.17±4.98 and 78.67±5.76 and 86.83±4.52 and 76.33±8.30. Figure 2 graphs the experimental results of our proposed method on Colon cancer dataset.

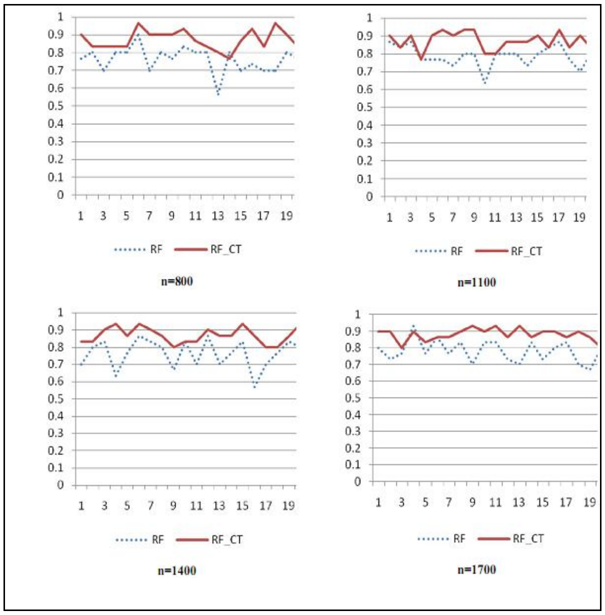


Figure 2: The comparison of classification accuracy between RF and RF_CT on Colon cancer dataset through 20 testing times and number of trees in RF n=800, 11000, 1400 and 1700

Colon cancer dataset is public dataset that widely used as basic dataset to validate new proposed methods in data mining. In fact, many data mining researchers have executed their proposed methods on Conlon cancer data. Table below summarizes some results of some other data mining methods on Colon cancer dataset.

Table 4: The comparison of classification accuracy among some methods on Colon Turmo data

Method	Classification Accuracy (%)	Standard Deviation
GA\SMV	84,7	9,1
Bootstrapped GA\SVM	80	
Combined Kernel for SVM	75,33	7,0
DFE-RF	85,5	4,5
RF_CT	**88,17**	**3,82**

6 Conclusions

Our proposed method presents an improvement of Random Forest and DFE-RF algorithm. In proposed method we took advantages of Bayesian probability and feature impurity to improve classification accuracy in classified tasks. Especially, proposed method also proposes a new approach to improve classification accuracy in case of classification of imbalance classes. Experimental results show our method is better than original method as well as some other popular methods both in classification accuracy and stability.

References:

[1] L. Breiman, "Random Forests," *Machine Learning Journal Paper*, vol. 45, 2001.

[2] X. Su, *Bagging and Random Forests*. [online]. Available: http://pegasus.cc.ucf.edu/~xsu/CLASS/STA5703/notes11.pdf

[3] T. N. V. Ha-Nam Nguyen, S. Y. Ohn,Y. M. Park, M. Y. Han, and C. W. Kim, "Feature Elimination Approach Based on Random Forest for Cancer Diagnosis," *MICAI 2006: Advances in Artificial Intelligence*, (2006).

[4] W. A. M. S. Kally, "An Optimum Random Forest Model for Prediction of Genetic Susceptibility to Complex Diseases," *Advances in Knowledge Discovery and Data Mining*, vol. 4426/2007, ed: Springer Berlin / Heidelberg, pp. 193-204, (2007).

[5] M. W. A. X. C. Heping Zhang, (2009) Software Willows: a memory efficient tree and forest construction package. *BMC Bioinformatics*. [online]. Available: http://www.biomedcentral.com/content/pdf/1471-2105-10-130.pdf

[6] C. Dahinden, Ed., *An Improved Random Forests Approach with Application to the Performance Prediction Challenge Datasets*.

[7] C. V. Anneleen Van Assche, H. Blockeel,S. S. D. Zeroski, "First order random forests: Learning relational classifiers with complex aggregates," *Machine Learning*, pp.149-182, 2006.

[8] S. G. Isabelle Guyon, Ed., *Feature Selection*. Springer, 2006.

[9] L. Breiman. (2002), *Manual On Setting Up, Using, And Understanding Random Forests V3. {online]*. Available: http://oz.berkeley.edu/users/breiman/Using_random_forests_V3.1.pdf

[10] M. K. Jiawei Han, *Data Mining:Concepts and Techniques*, Second Edition ed.: Diane Cerra, 2006.

[11] *R-Language*. [online]. Available: http://www.r-project.org/

[12] *Random Forest package*. [online]. Available: http://cran.r-project.org/web/packages/randomForest/

[13] *Medalone Dataset*. [online]. Available: http://archive.ics.uci.edu/ml/datasets.html

[14] L. Rokach. (2008)Genetic Algorithm-based Feature Set Partitioning for Classification Problems. *Pattern Recognition*. 1693-1717 . [online]. Available: http://portal.acm.org/citation.cfm?id=1340831

[15] *Colon Cancer Dataset*. [online]. Available: http://www.inf.ed.ac.uk/teaching/courses/dme/html/datasets0405.html