Applied Soft Computing 70 (2018) 1000–1009

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

A principle component analysis-based random forest with the potential nearest neighbor method for automobile insurance fraud identification

Yaqi Li[a], Chun Yan[a], Wei Liu[b,*], Maozhen Li[c]

[a] College of Mathematics and Systems Science, Shandong University of Science and Technology, Shandong, Qingdao, China
[b] College of Computer Science and Engineering, Shandong University of Science and Technology, Shandong, Qingdao, China
[c] Department of Electronic and Computer Engineering, Brunel University London Uxbridge, UB8 3PH, UK

ARTICLE INFO

Article history:
Received 21 September 2016
Received in revised form 6 July 2017
Accepted 14 July 2017
Available online 19 July 2017

Keywords:
Ensemble
Random forest
Principle component analysis
Potential nearest neighbors
Voting mechanism
Automobile insurance fraud

ABSTRACT

As a successful ensemble method, Random Forest has attracted much attention. In this paper, individual classifiers are appropriately combined and a multiple classifier system with an increase in classification accuracy is presented. According to Breiman's methodology, we propose a multiple classifier system based on the Random Forest, Principle Component Analysis and Potential Nearest Neighbor methods As Breiman suggested, the performance of the Random Forest depends on the strength of the weak learners in the forests and diversity among them. The Principle Component Analysis method is applied to transform data at each node to another space when computing the best split at this node. This process increases the diversity of each tree in the forest and thereby improves the overall accuracy. The Random Forest is studied through the perspective of the Adaptive Nearest Neighbor. We introduce the concept of monotone distance measures and potential nearest neighbors and show that the Random Forest can be viewed as an adaptive learning mechanism of k Potential Nearest Neighbors. Considering the information loss caused by out-of-bag samples, a new voting mechanism based on Potential Nearest Neighbor is also presented to replace the traditional majority vote. The proposed algorithm improves the classification accuracy of the ensemble classifier by improving the difference of the base classifiers. The performance of the proposed method is compared with those of the Oblique Decision Tree Ensemble, Rotation Forest and basic Random Forest on the data sets. The experimental results show that the proposed method produces a better classification accuracy and lower variance. The proposed method is also applied to detect automobile insurance fraud, and the fraud rules are obtained.

1. Introduction

In the current era of rapid information growth, many methods for obtaining information exist, but tools are lacking for data analysis and data mining. The vigorous development of pattern classification and machine learning also reflects the needs of industry and daily life. Machine learning can be used to estimate new test samples, but it is difficult to construct a single classifier with high accuracy. Additionally, the single classifier cannot be perfectly applied to all data sets. All of these problems lead to the generation of ensemble learning.

Ensemble learning is a new machine-learning framework that treats multiple learners as different modules to solve the same problem. This approach can significantly improve the generalization ability of learning systems [1] and originates from the working hypothesis of the nerve cell [2]. Currently, most ensemble learning algorithms use a large number of individual learners to achieve better performance and reduce generalization error, among which the Boosting [3] and Bagging [4] methods are the most influential. With the development of ensemble learning technology, Tin Kam Ho first proposed the concept of Random Decision Forests [5] and the Random Subspace method [6]. These methods have significant effects in practical application.

Recently, many ensemble-learning approaches have been introduced in the literature and have suggested new research directions for classification and regression problems [1,7,8]. Classifier ensemble learning makes full use of the complementary relationship between the classifiers and can successfully solve classification problems that the single classifier cannot. This approach has been widely used in the fields of artificial intelligence, machine learning, pattern recognition and other engineering applications. In this

* Corresponding author.
E-mail address: liuwei_doctor@yeah.net (W. Liu).

https://doi.org/10.1016/j.asoc.2017.07.027
1568-4946/

paper, the classifier ensemble learning approach is used in fraud identification and fraud law mining.

Fraud occurs in many fields, such as tax fraud [9] and telecommunications fraud [10], among others. Many different forms of fraud exist, from traditional fraud (e.g., simple tax evasion) to more complicated forms (gang crime), and fraud events can also be found in automobile insurance. With the current development of the insurance industry, insurance fraud has continuously emerged. According to the Federal Bureau of Investigation Financial Crimes Report to the Public [11], an upward trend has been noted among many types of crimes, automobile insurance fraud included. Most analyses show that approximately 20–35% of automobile insurance claims are fraudulent to a certain extent [12,13]. Insurance fraud has caused economic and reputation losses to insurance companies, and an effective approach to identifying insurance fraud is thus needed. Traditionally, insurance fraud detection relies heavily on auditing and expert inspection. Manual detection of fraud cases is costly and inefficient, and data mining technology is increasingly considered a key tool in fraud detection. Data mining and machine learning techniques have the potential to detect suspicious fraud cases in a timely manner and can significantly reduce economic losses to the insurance company and policy-holder. Indeed, a great demand has emerged for effective predictive methods that can maximize the true positive detection rate, minimize the false positive rate, and quickly identify new fraud cases. Many experts worldwide have studied insurance fraud from a theoretical and empirical perspective using various data mining methods.

Several identification methods are available for insurance fraud such as Expert System [14], Binary Choice models (Logit model and Probit model) [15], Decision Tree (DT) [16], Support Vector Machine (SVM) [16,17], Artificial Neural Network (ANN) [16] and other data mining techniques. These methods offer a new approach to the identification of insurance fraud, and they all display distinct advantages. Researchers conducted a detailed comparative experiment on Random Forest (RF) and other classification algorithms, and the results showed that RF had better classification and regression performance than other algorithms [18–20]. Certain studies have shown that RF is more suitable for working with imbalanced data and large data sets [16,17,21–24]. Random Forest, proposed by Leo Breiman [25], combines the Bagging [4] and Random Subspace [6] methods and exhibits high performances in many applications. The following advantages make RF particularly useful for identification of automobile insurance fraud [25]: (i) suitable for high-dimensional small-sample data, (ii) automatically selects features and is not sensitive to irrelevant features, (iii) considers the interaction between features, (iv) applicable to two classification and multiple classification problems, and (v) does not require complex parameter selection process. We applied RF to mining of automobile insurance fraud data and analysed the importance of various fraud variables [41]. However, the Random Forest method is not perfect, and it is still necessary to further improve the accuracy and reduce the generalization error. Until now, many improved methods for RF have been proposed. Certain methods are used to pre-process data with specific characteristics [26]. Other methods use a combination of ensemble classifiers [27], and still others offer improved feature selection [18,28–30]. Furthermore, many other improved methods for RF (e.g., Oblique Random Forest [31], Deep Learning and Random Forest [42–44]) have been proposed but are not introduced in this work.

The performance of RF depends on the classification ability of each tree and the diversity and correlation among them. In this paper, Principle Component Analysis (PCA) was used in feature selection to transform the data at each node into another space when computing the best split at this node. This process increases the diversity of each tree in the forest and thereby improves the overall accuracy. The complexity of the base classifier was also improved. Random Forest is studied from the perspective of the Potential Nearest Neighbor algorithm and can be viewed as an adaptive learning mechanism of k Potential Nearest Neighbors. Considering the information loss caused by out-of-bag samples (OOB samples), a new voting mechanism based on Potential Nearest Neighbor was also proposed. Different from Rotation Forest [18], the proposed method uses cross-validation to evaluate the classification performance. This approach can improve the classification performance of the algorithm but is expected to increase the complexity of the algorithm to a certain extent.

This section introduces the development of the ensemble classifier and Random Forest as well as the research methods and development of fraud identification. The remainder of the paper is organized as follows. Section 2 briefly reviews the Classification and Regression Tree (CART), Bagging, and Random Forest algorithms. Section 3 analyses the PCA method for feature extraction and the relationship between Random Forest and Adaptive Nearest Neighbors. The information loss of Random Forest in the prediction process is considered. Based on this information, a new voting mechanism is proposed. Section 4 selects 12 benchmark data sets from various fields to evaluate the performance of the proposed method. The proposed method is also compared with the Oblique Decision Tree Ensemble, Rotation Forest and traditional Random Forest methods applied to the same data sets. The results show that the proposed method displays better performance. In Section 4, the proposed method is also used in automobile insurance fraud identification and fraud law mining. Section 5 summarizes our conclusions and the direction of future work.

2. Preliminaries

This section presents the theories of CART, Bagging, and Random Forest. Before introduction of the theory, we first explain the symbols. We assume that $S = \left\{(x_i, y_i), i = 1, 2, \cdots, n\right\}$ is an observation set of independent and identically distributed random variables (X, Y), where $X = \left(X^1, X^2, \cdots, X^d\right) \in R^d$ is the input variable, $Y \in R$ is the dependent variable, R represents the real number set, N_{tree} is the number of trees in the forest, $T_i\,(i = 1, 2, \cdots, N_{tree})$ is the random tree in the random forest, and $mtry$ is the number of candidate variables randomly selected to split in each non-leaf node.

2.1. CART

CART was proposed by Breiman et al. [32] in 1984 and is a collective name for classification and regression trees. The CART algorithm uses the technique of two-point recursive segmentation to divide the current sample set into two subsample sets. Each generated non-leaf node has two branches, and therefore, the decision tree generated by the CART algorithm is a binary tree with a simple structure. The CART algorithm primarily focuses on two major problems: tree growth and tree pruning. For tree growth, CART generates the decision tree based on the training set. For tree pruning, the generated tree is pruned using the testing data set, and the optimal sub-tree is selected. The pruning criterion is the minimal loss function.

The CART algorithm uses the Gini index as the split criterion and can select the attribute that reduces the degree of disorder of the data. In establishing the model, the split attribute is selected by the classification degree of the sample data under different predictions. Given the sample set S, the construction of CART is composed of the following three steps:

- Construct the maximum tree $T_{\max}$ based on S. In the maximum tree $T_{\max}$, the sample number of each leaf node is less than a given threshold.
- Construct a finite ordered sub-tree sequence with node number decreased by the pruning algorithm.

- Select the best pruning tree from the sub-tree sequence as the final tree.

2.2. Bagging

The Bagging algorithm, proposed by Breiman [4] in 1996, generates mutually different sub-classifiers by operating on the training sample set. The core of this approach is the Bootstrap Sampling method that is combined with the single prediction model and generates a set of combination forecasting models. The Bagging technique consists of three phases: modelling phase, evaluation phase and prediction phase.

1) Modelling phase:

- Randomly generate a sample S_1 (known as the bootstrap sample) by sampling N times from the original sample set S with replacement.
- Treat the bootstrap sample S_1 as the training set and construct the classification or regression tree.
- Repeat the above two steps N_{tree} times to obtain the bootstrap samples $S_1, S_2, \cdots, S_{N_{tree}}$ and the corresponding prediction models $T_1, T_2, \cdots, T_{N_{tree}}$.

2) Evaluation phase:

For each tree $T_i\,(i = 1, 2, \cdots, N_{tree})$ grown on a bootstrap sample S_i, the error rate for the observations left out of the bootstrap sample is monitored. Taking the classification problem as an example, if the i_{th} observation is treated as an OOB observation for q times, all of the q prediction models should give one vote on the i_{th} observation. In this case, the predicted label of the sample is determined by the sample label with the most votes. The ratio of the false observation number to the total sample number is the OOB error rate.

3) Prediction phase

The prediction of a new observation consists of the N_{tree} prediction model produced in the first phase. For the classification problem (as mentioned in the second phase), the predicted label of the sample is determined by the one with the most votes. For the regression problem, the final value is the average value of the N_{tree} predicted models.

2.3. Random forest

Ensemble classifiers can significantly improve the classification accuracy and reduce the variance of the classifier [25]. From the accuracy and variance point of view, the ensemble classifier should show much better performance if the base classifier has high variance. Therefore, it is easy to understand why the randomized algorithms (such as the randomized neural network [38]) naturally work well with ensemble methods, whereas other methods (such as SVM) might need to be combined with a much more complicated algorithm [39].

In ensemble learning, Random Forest is a randomized ensemble classifier that consists of many decision trees [19], and it outputs the class that is the mode of the output classes by individual trees. The algorithm for inducing a random forest was developed by Leo Breiman [25] and Adele Cutler, and "Random Forest" is their trademark. The term originated from the random decision forest first proposed by Tin Kam Ho [5] of Bell Labs in 1995. The method combines Breiman's "Bagging" [4] idea and the random selection of features introduced independently by Ho [5] and Amit and Geman [33]. This approach can construct a collection of decision trees with controlled variation.

The Random Forest f is the set of decision trees $\left\{T\left(X, \theta_k\right), k = 1, 2, \cdots\right\}$, where the meta-classifier $T\left(X, \theta_k\right)$ is the classification and regression tree constructed by the CART algorithm without pruning. The parameter θ_k is an independent and identically distributed random vector that determines the growth process of a single tree. The final output prediction value is obtained using the majority voting method (classification) or average method (regression).

In the CART algorithm, each internal node is a subset of the original data set, and the root node contains all of the original data. The CART algorithm splits using the best attribute splitting method at each internal node and finally minimizes the test error by pruning. However, each single tree in the Random Forest exists without pruning. The sub-classifiers in Random Forest are significantly different such that it has excellent classification performance.

The Random Forest algorithm is described as follows:

Input: 1.Training set $S = \{(x_i, y_i), i = 1,2,\cdots,n\}, (X, Y) \in R^d \times R$

2. Testing sample $x_t \in R^d$

For $i = 1,2,\cdots,N_{tree}$

1) Generate the training set S_i by Bootstrap sampling from the original training set S.

2) Generate a tree T_i without pruning by S_i:

a. Randomly select $mtry$ features from d features

b. Select the optimal features according to the Gini index from $mtry$ features of each node.

c. Split until the tree grows to the maximum.

End

Output: 1. A collection of trees: $\{T_i, i = 1,2,\cdots,N_{tree}\}$

2. For the testing sample x_t, the decision tree T_i outputs $T_i(x_t)$

Regression: $f(x_t) = \frac{1}{N_{tree}} \sum_{i=1}^{N_{tree}} T_i(x_t)$

Classification: $f(x_t) = majority\ \ vote\{T_i(x_t)\}_{i=1}^{N_{tree}}$

Using CART as a meta-learning algorithm, the Random Forest algorithm can address continuous attributes and class attributes and combines Bagging [4] and random selection of features [6], which makes more tolerant to noise. Random Forest can also effectively solve the imbalanced classification problem with higher classification accuracy and without overfitting. The automobile insurance claims data are imbalanced because the data contain continuous attributes and class attributes. Random Forest is thus suitable for identification of automobile insurance fraud. The best characteristic of Random Forest is the estimation function of OOB. When generating the training set by Bootstrap, for each sub-classifier, approximately 37% samples of the original sample set S do not appear in the training set. These samples are known as OOB samples. OOB samples can be used to estimate the generalization error of the forest and the variable importance.

Theorem 1. The upper bound of the generalization error of Random Forest is given by the following formula [25]:

$$PE^* \le \bar{\rho}\left(1 - s^2\right)/s^2 \tag{1}$$

where $\bar{\rho}$ is the average value of the correlation degree ρ between the sub-classifier of Random Forest, and s is the classification efficiency of sub-classifier $h\left(X, \theta_k\right)$.

2.4. Principle component analysis

The global insurance industry has rapidly developed in recent years. Many insurance claims occur every year, and thus the amount of insurance claims data is notably large. Generally, among the numerous claims, the information for each applicant is not the same. For automobile insurance, many factors affect insurance fraud, such as insurance maturity, type of automobile, insured amount, repair shop type, gender of policy-holder, historical claims, etc., which is why insurance claims data have high dimensionality. In addition, a certain correlation exists between selected indicators (variables). Taking the automobile type and insured amount as an example, the insured amount of high-end automobiles is generally higher. Moreover, the repair shop type and repair costs of different automobile are quite different. These factors all increase the diffi-

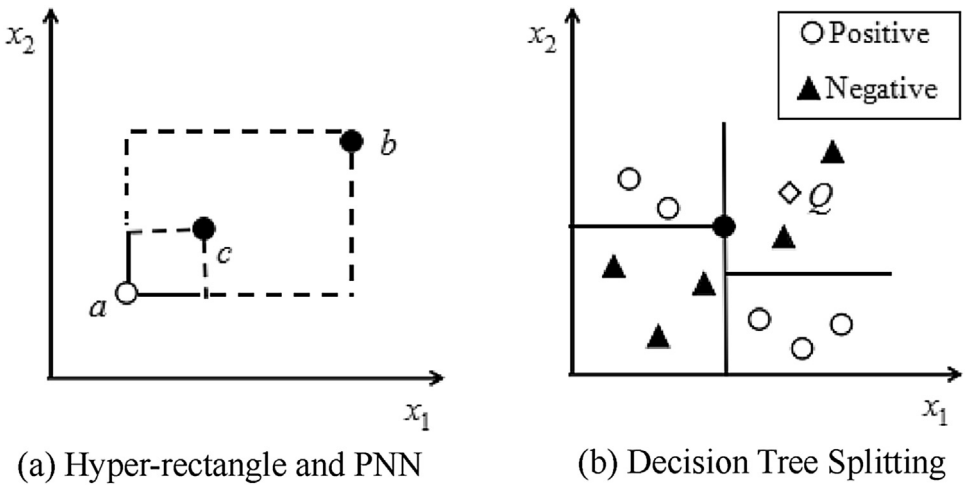

Fig. 1. PNN and Decision Tree.

culty of the study of insurance fraud. Although certain indicators are relevant, they are all highly important for insurance fraud identification, and therefore, all of the indicators should be retained. The Principal Component Analysis method, first proposed by Karl Pearson in 1901 and extended by Hotelling, is commonly used in dimensionality reduction. For PCA, because only a few components are retained, there is a chance that the most relevant discriminatory components that correspond to small variance will be discarded. In this situation, certain important variables that affect fraud identification might be discarded, which could have an impact on the accuracy of the result. Therefore, in this work, we retain all of the components, and the discriminatory information is preserved. PCA is well studied in the field of ensemble learning, and empirical studies show that it can yield high accuracy while improving diversity [18].

As shown in formula (1), to improve the prediction accuracy of Random Forest, the correlation degree between the trees should be reduced. At the same time, the classification efficiency of the single tree should be increased. We are likely to choose a different subspace of features at each node while building a CART, and the transformation at each node can be completely different, which can improve the diversity of the individual tree. In this work, the Principle Component Analysis method is applied to transform the data at each node to another space when computing the best split at this node. Furthermore, we treat the transformation-based CART trees as an ensemble to construct the Random Forest, a process known as PCARF.

The PCARF algorithm is described as follows:

Input: 1.Training set $S=\{(x_i,y_i),i=1,2,\cdots,n\},(X,Y)\in R^d\times R$

2. Testing sample $x_t\in R^d$

For $i=1,2,\cdots,N_{tree}$

1) Generate the training set S_i by Bootstrap sampling from the original training set S.

2) Generate a tree T_i without pruning by S_i:

a. Randomly select $mtry$ features from d features.

b. Select the optimal features according to the Gini index from $mtry$ features of each node.

c. Calculate the scatter matrix V_i of the $mtry$ features at this node, and calculate the eigenroots λ and the corresponding eigenvectors U of the V_i.

d. Transform the data into the PCA space, i.e., principal component transformation $X_{PCA}=U'X$.

e. Split until the tree grows to the maximum.

End

Output: 1. A collection of trees: $\{T_i,i=1,2,\cdots,N_{tree}\}$

2. For the testing sample x_t, the decision tree T_i outputs $T_i(x_t)$

Regression: $f(x_t)=\frac{1}{N_{tree}}\sum_{i=1}^{N_{tree}}T_i(x_t)$

Classification: $f(x_t)=majority\ \ vote\{T_i(x_t)\}_{i=1}^{N_{tree}}$

3. Proposed method

3.1. Relationship between random forests and adaptive nearest neighbors [34]

This section analyses the relationship between Random Forest and Adaptive Nearest Neighbors. We first introduce the concept of monotone distance measures. A hyper-rectangle $\otimes_{j=1}^{d}\left[a^j,b^j\right]$ is defined by two points a and b with the two points as opposing vertices, as shown in Fig. 1(a). Any distance metric should satisfy the positivity condition and the triangle inequality. In addition, in Euclidean space, it is reasonable to require distance measures to satisfy the following monotonicity condition: for any two points a and b and any point c in the hyper-rectangle defined by a and b, the distance from c to b is less than or equal to the distance from a to b. This statement is intuitively clear because c is closer to b than a in every dimension j, $j=1,\ldots,d$. We refer to any distance measure in the Euclidean space satisfying this monotonicity property as a monotonic distance measure. At this point, we are ready to introduce the potential nearest neighbors.

Definition 1. k Potential Nearest Neighbors (k-PNN): The sample x_k is known as the k-PNN of target sample x_t if and only if there exists a monotonic distance measure L under which x_k is among the k nearest neighbors to x_t among all the samples.

Naturally, a sample x_k is the k-PNN of x_t when and only when there are fewer than k sample points other than x_k in the hyper-rectangle $\otimes_{j=1}^{d}\left[x_k^j,x_t^j\right]$ determined by x_k and x_t under a distance measure L. For example, in Fig. 1(a), b and c are the 2-PNNs of a, and c is 1-PNN (adaptive nearest neighbor) of a.

As shown in Fig. 1(b), a decision tree without pruning divides two types of samples in the 2-dimensional space. The entire space is divided into four rectangular regions. Each region corresponds to a leaf node and contains only one class of data. When the sample to be tested Q is placed into the tree, it is divided into the right upper rectangle area corresponding to the leaf node. This process determines that the sample Q belongs to the positive sample. A decision tree T chooses a feature at each node to split. The feature makes the Gini coefficient of the node a minimum. The feature space is recursively divided into several hyper-rectangle regions, which defines the PNN of sample Q under a monotone metric function L. The class of sample Q is determined by the class label of PNN. For the classification problem, the number of nearest neighbors on the same leaf node with Q is uncertain.

Obviously, the metric function L defined by tree T is different from the Euclidean distance. In the 2-dimensional Euclidean space, positive and negative samples are not linearly separable. However, the metric function L can maximize the similarity of the same type and minimize the similarity of the different type. The decision tree defines an adaptive sample similarity measure, and thus, we can obtain the following conclusion:

Theorem 2. If the sample number of the original training set S is n, and the training sample $(X, Y) \in R^d \times R$, then a random forest $\left\{T\left(X, \theta_k\right), k = 1, 2, \cdots\right\}$ corresponds to a set of monotonic metric functions $\left\{L_1, L_2, \cdots\right\}$.

If N_{tree} decision trees are generated in the random forest, then the metric function $\left\{L_1, L_2, \cdots, L_{N_{tree}}\right\}$ defines a large number of k-PNN for the testing sample x_t. The predicted value of x_t is determined by the value of the potential neighbors. This metric function has been separated from the original Euclidean space and is generated by the decision tree through the learning of the training data, and therefore, it can be treated as an adaptive learning mechanism of k nearest neighbors.

We set PN_i as the set of potential neighbors defined for x_t by T_i in Random Forest.

For the regression problem,

$$T_i(x_t) = \frac{1}{|PN_i|}\sum PN_i \tag{2}$$

where $|PN_i|$ is the number of element in the set is PN_i, and $\sum PN_i$ is the sum of elements in the set.

Furthermore, the prediction of x_t is expressed in formula (3):

$$f(x_t) = \frac{1}{N_{tree}}\sum_{i=1}^{N_{tree}} T_i(x_t) = \frac{1}{N_{tree}}\sum_{i=1}^{N_{tree}}\left(\frac{1}{|PN_i|}\Sigma PN_i\right) \tag{3}$$

For the classification problem, T_i is not pruned, each leaf node is 'pure', and the predictive value of x_t is actually the class label of the training sample on the node. Thus $T_i(x_t)$ is equivalent to formula (4):

$$T_i(x_t) = majorityvote\,(PN_i) \tag{4}$$

Furthermore, the prediction of x_t is expressed in formula (5):

$$f(x_t) = majorityvote(T_i(x_t))_{i=1}^{N_{tree}} \tag{5}$$

3.2. Random forest-based potential nearest neighbor (RF-PNN)

Random Forest uses the Bagging algorithm to randomly select the training sample subset with replacement. This process can effectively increase the differences in the decision tree and enhance the performance of the combination classifier. However, from the angle of k-PNN, only the partial training data S_i is used in the voting process of each decision tree T_i. Therefore, the OOB sample of $T_i$is not included in the potential nearest neighbor defined by T_i (corresponding to the monotone metric L_i) for x_t. Certain information losses exist in formula (2), (3), (4) and (5). Although this portion of the information is scarce, it has a certain impact on the classification and regression result.

Considering these issues, this paper proposes a method that takes the information of the OOB sample into consideration in the prediction process. For the regression problem, formula (2) and formula (3) do not change, but PN_i in the formula contains the OOB sample. For the classification problem, in formula (4), most samples are 'pure' training samples with the same label and fall within the sample PN_i in the same one leaf node. Generally, only a few OOB samples exist. If the simple majority voting method is adopted, the OOB sample information will still be overwhelmed by the training sample information. This situation is verified by preliminary experiments. Therefore, this paper designs a new voting mechanism, as shown in formula (6):

$$f(x_t) = vote\left(arg\ \max_{x_i \in S} Fr(x_i)\right) \tag{6}$$

where $Fr(x_i) = \sum_{j=1}^{N_{tree}} fr_j(x_i)$ represents the frequency that x_i occurs in the potential nearest neighbor sets $\left\{PN_1, PN_2, \cdots, PN_{N_{tree}}\right\}$, and $fr_j(x_i)$ is the number of times that x_i occurs in PN_i. The voting points for a target point x_i belong to the k-PNN set of x_i. The terminal nodes of each tree define rectangular areas. If a sample point x_t is not a k-PNN of x_i, then there are more than k sample points (including x_t) in the hyper-rectangle $\otimes_{j=1}^{d}\left[x_i^j, x_t^j\right]$ determined by x_i and x_t. Therefore, only k-PNN can become a voting point. All k-PNN of the target point have positive probabilities of being located in the same terminal node as the target point. The sample x with the highest frequency is defined as the potential nearest neighbor of x_t in the forest. The class label of the sample determines the class of x_t, and the algorithm is referred to as the Random Forest-based Potential Nearest Neighbor (RF-PNN) algorithm.

The advantages of RF-PNN can be summarized as follows:

a In the decision tree T_i, only the training set S_i in RF is used to predict the sample x_t. For the PNN algorithm based on transductive learning, the RF-PNN algorithm considers the information loss caused by the OOB sample set.
b All decision trees are used to vote in RF in the process of prediction. The process selects partial PNN samples (the samples on the same leaf node with the sample x_t in the training set S_i) to vote. The RF-PNN algorithm selects the samples that are nearest to x_t in the training set S_i. Under the monotonic distance metric $\left\{L_1, L_2, \cdots, L_{N_{tree}}\right\}$, as the potential neighbor of x_t, the occurrence frequency of the sample is the highest. This approach is more in line with the nearest neighbor rule.

As mentioned above, we first use Principle Component Analysis to transform the data into PCA space for dimensionality reduction. We subsequently combine the Random Forest with Potential Nearest Neighbor and the new voting mechanism to reduce the information loss caused by the OOB samples. We refer to the Random Forest based Principle Component Analysis introducing the Potential Nearest Neighbor algorithm as PCARF-PNN for short.

The PCARF-PNN algorithm is described as follows:

Input: 1. Training set $S = \{(x_i, y_i), i = 1,2,\cdots,n\}, (X, Y) \in R^d \times R$

2. Testing sample $x_t \in R^d$

1) Invoke algorithm 2 and generate the Random Forest model $f = \{T_i, i = 1,2,\cdots,N_{tree}\}$.

2) Simultaneously import the entire training set S and testing sample x_t into Random Forest:

For $i = 1,2,\cdots,N_{tree}$

Search the sample set PN_i that has the same leaf node as x_t in the decision tree T_i, i.e., the potential nearest neighbor of x_t.

End

Output: 1. A collection of trees: $\{T_i, i = 1,2,\cdots,N_{tree}\}$

2. For the testing sample x_t, the decision tree T_i outputs PN_i

Regression: $f(x_t) = \frac{1}{N_{tree}}\sum_{i=1}^{N_{tree}}\left(\frac{1}{|PN_i|}\sum PN_i\right)$

Classification: $f(x_t) = vote\left(\arg\max_{x_i \in S} Fr(x_i)\right)$

Compared with the traditional RF, PCARF-PNN first applies PCA to further improve the diversity of the individual trees while maintaining the high accuracy. Keeping all of the components does not mean that the classification is easier in the new space of the extracted features. However, even if the rotation does not contribute significantly to finding good discriminatory directions, it

Table 1
Data set descriptions.

Data Sets	Patterns	Features	Classes
AFP	9974	19	2
Letter	20000	16	26
Nursery	12960	8	5
Page Blocks	5473	10	5
Lung Cancer	12600	203	5
Car Evaluation	1728	6	4
Thyroid	215	5	3
Wine	178	13	3
Cleveland Heart	303	13	2
Australian	690	14	2
Breast Cancer	699	9	2
Parkinson	195	22	2

is valuable in this context as a diversifying heuristic. PCARF-PNN selects the samples that are nearest to the sample to be tested in the training set based on the principle of PNN. This step avoids information loss of the OOB sample. The data set is always high dimensional and relevant. The proposed method keeps all of the components and considers the OOB samples. All of these factors might require the algorithm to process a larger amount of data, and thus it can increase the complexity of the algorithm to a certain extent.

4. Experimental validation

4.1. Performance evaluation of proposed algorithm

An experiment is set up to compare the standard Random Forest, Oblique Decision Tree Ensemble (ODT), Rotation Forest (ROF) and PCARF-PNN. All simulations are performed in the MATLAB environment. Considering the computational complexity, we set the number of decision trees to 100 for all experiments. The other parameter is set to the default value (the number of candidate features randomly selected at each node is the square root of the number of features). Table 1 shows the characteristics of the 12 benchmark data sets used in the study and selected from various fields.

For each data set and ensemble method, ten 10-fold cross validations were performed. The accuracies were averaged over all 100 testing accuracies per method and data set. In practical applications, cross-validation is the simplest and most widely used method for estimating the prediction error and the generalization performance of the model. Currently, in most literature in the classification field, cross-validation is used to evaluate the performance of the classifier [35,36]. The 10-fold cross-validation method randomly divides the sample into 10 disjoint groups. Nine groups is are used as the training set for model construction and the remaining group is the testing set used to calculate the prediction error cv_i of the model. The average accuracy of the 10 tests was recorded as a result of the 10-fold cross-validation. The cross-validation is estimated as follows:

$$CV = \frac{1}{k}\sum_{i=1}^{k} cv_i \tag{7}$$

The classification accuracy of each data set is presented in Table 2. The bias and variance of each data set and each algorithm are presented in Table 3. The training time is shown in Table 4.

As shown in Table 2, the PCARF-PNN algorithm achieves better classification accuracy in the six classification data sets and therefore, the PCA and PNN algorithm can be used to optimize RF to achieve better classification performance. Compared with the traditional RF, the classification accuracy of Oblique Decision Tree Ensemble and Rotation Forest are also greatly improved. As mentioned previously, ensemble classifiers should be able to strongly reduce the variance. Hence, in Table 3, the performance of the three

Table 2
Classification accuracy and standard deviation of each algorithm.

Data sets	RF	ODT	ROF	PCARF-PNN
AFP	86.15 ± 5.79	87.59 ± 4.70	87.08 ± 5.70	89.11 ± 3.02
Letter	87.35 ± 3.20	88.03 ± 2.80	88.04 ± 2.31	89.38 ± 1.61
Nursery	87.50 ± 1.09	89.37 ± 1.25	89.51 ± 1.61	92.28 ± 1.11
Page Blocks	88.28 ± 2.10	89.66 ± 3.32	89.17 ± 1.96	91.12 ± 2.08
Lung Cancer	85.60 ± 4.32	87.01 ± 3.25	87.75 ± 3.99	88.94 ± 2.38
Car Evaluation	88.65 ± 3.65	90.08 ± 1.95	90.32 ± 2.14	91.58 ± 2.56
Thyroid	91.74 ± 3.02	92.93 ± 3.98	94.14 ± 3.67	94.97 ± 1.44
Wine	92.10 ± 1.93	93.23 ± 1.64	92.28 ± 1.60	94.82 ± 1.23
Cleveland Heart	90.19 ± 6.00	91.45 ± 3.99	91.47 ± 4.32	93.22 ± 2.10
Australian	85.35 ± 3.20	89.51 ± 2.80	89.37 ± 2.31	90.74 ± 1.61
Breast Cancer	92.05 ± 1.02	94.31 ± 1.25	95.45 ± 1.67	96.97 ± 1.09
Parkinson	91.28 ± 5.79	93.66 ± 5.70	94.17 ± 4.70	95.12 ± 3.02

Table 3
Biases and variances of each algorithm.

Data sets	RF	ODT	ROF	PCARF-PNN
AFP	(3.82, 1.12)	(3.68, 2.96)	(3.53, 2.63)	(3.34, 2.01)
Letter	(6.10, 3.01)	(5.81, 2.96)	(5.21, 2.94)	(5.24, 2.92)
Nursery	(6.65, 2.21)	(5.98, 1.82)	(5.81, 1.17)	(5.66, 1.06)
Page Blocks	(4.55, 2.21)	(4.28, 1.92)	(4.31, 1.57)	(3.96, 1.06)
Lung Cancer	(6.22, 2.35)	(5.21, 2.10)	(5.50, 1.95)	(5.26, 1.56)
Car Evaluation	(4.32, 1.11)	(3.25, 1.05)	(3.54, 1.25)	(3.21, 1.22)
Thyroid	(3.34, 2.96)	(2.84, 2.69)	(3.22, 2.88)	(2.67, 2.18)
Wine	(4.56, 0.70)	(4.55, 0.55)	(4.49, 0.42)	(4.33, 0.34)
Cleveland Heart	(3.26, 1.02)	(2.53, 0.62)	(2.13, 0.59)	(1.39, 0.42)
Australian	(5.65, 2.21)	(4.98, 1.82)	(3.81, 1.17)	(3.66, 1.06)
Breast Cancer	(11.22, 3.05)	(10.21, 2.10)	(9.50, 2.95)	(7.86, 2.56)
Parkinson	(6.65, 3.90)	(6.75, 1.82)	(6.60, 1.92)	(6.85, 1.19)

Table 4
Training time of each algorithm CPU/s.

Data sets	RF	ODT	ROF	PCARF-PNN
AFP	2.74	2.72	2.72	2.60
Letter	1.88	1.78	1.74	1.42
Nursery	2.56	1.95	1.95	1.76
Page Blocks	3.29	2.96	3.19	2.86
Lung Cancer	2.24	2.22	2.22	1.60
Car Evaluation	4.69	3.27	3.01	2.65
Thyroid	10.1	7.4	7.9	6.1
Wine	12.7	14.8	13.5	10.9
Cleveland Heart	4.6	5.7	5.9	9.5
Australian	5.8	5.3	4.7	3.8
Breast Cancer	6.5	7.5	6.9	7.2
Parkinson	11.1	11.8	12.6	14.2

algorithms is better than RF in most cases. Moreover, we also find that the variance of PCARF-PNN is smaller than that of ODT and ROF. Furthermore, Table 4 shows the training time of each algorithm. For the smaller data sets, RF is faster than PCARF-PNN. However, PCAPF-PNN is faster on data sets with large scale and many features and classes. We conclude that PCAPF-PNN is more scalable in feature selection. This observation is important, especially in addressing the computational burden involved in high-dimensional data sets. All of these factors indicate that the proposed method is more stable and thus the optimization of RF is feasible. The proposed method can be applied for better identification of automobile insurance fraud.

4.2. Identification of automobile insurance fraud

As previously mentioned, automobile insurance fraud accidents generally have the following features. These accidents generally occur during the late hours and in suburban areas without any witnesses. The drivers are usually younger males, and the accident cars are usually private cars. The police are always called to the scene to make the subsequent claims easier and more reasonable. Many

Table 5
Description of variable attributes.

Variables	Types	Variables	Types
Insurance Maturity	Discrete variable	Investigation	Categorical variable
Vehicle Type	Categorical variable	Garage	Categorical variable
Underwriting Type	Boolean variable	Damage photo Number	Discrete variable
Car-testing	Categorical variable	Historical times	Discrete variable
Site Report	Boolean variable	Judgment result	Boolean variable
Driver Gender	Boolean variable	–	–

Table 6
Layers of categorical variables.

Feature	Layered
Vehicle Property	Business car is set to 1 0 0; Private car is set to 0 1 0; Agency car is set to 0 0 1.
Car-testing	Not tested is set to 1 0 0; Have been tested is set to 0 1 0; The exemption is set to 0 0 1.
Survey type	Not surveyed is set to 1 0 0; The first scene is set to 0 1 0; Filled the survey site set to 0 0 1.
Repair Shop	The first type of factory is set to 1 0 0 0; The second type is set to 0 1 0 0; The third type is set to 0 0 1 0; The special service station is set to 0 0 0 1.

other suspicious characteristics exist, such as the income and profession of the claimant, which are not introduced in this work. Due to the existence of the above characteristics, to verify whether the claims with these characteristics are fraudulent or not, the appropriate indicators are selected to establish the model.

Although the characteristics mentioned above have a certain impact on fraud, there only are eleven variables. The insurance company's customer data are not shared. The information consists of the claim data for a Chinese automobile insurance company in 2015. The insured year of the vehicle is 2015, and the insurance period is one year. We established the automobile insurance fraud identification model based on the Random Forest and PCARF-PNN algorithms, and their performances are also compared. The two methods are implemented in SPSS and MATLAB.

The claims data set contains 1000 samples with 11 types of variables. The variables are insurance maturity (V1), vehicle type (V2), underwriting type (V3), car testing (V4), site report (V5), driver gender (V6), investigation (V7), garage (V8), damage photos number (V9), historical times (V10) and judgment result (V11). A description of the variable attributes is presented in Table 5.

As shown in Table 5, certain variables with categorical features exist in the claims data set. Therefore, we unify the encoding to ensure that PCA can be performed. For the binary features, each time, all feature values in the subset selected are encoded as 1 and the remainder are encoded as 0. The missing value is replaced with the median value of the feature. For other categorical features, we layer these indices according to the number of categories. The results are shown in Table 6.

Because the data partitioning of the claims data set was not already performed, the 10-fold cross-validation method was selected instead of OOB accuracy to measure the performance of the model. Before establishing the two models, we must first determine the two main parameters, *mtry* and *ntree* (N_{tree} mentioned above). The *mtry* parameter represents the number of candidate features randomly selected at each node during the growth phase of each tree and is generally the square root of the number of features. The automobile insurance claims data set is selected to evaluate the performance of the models with different *mtry* parameters. The *ntree* parameter denotes the number of decision trees in the forest and is determined by the specific data set. When the *ntree* parameter is rather small, the classification error is larger. Because the Random Forest is not prone to over-fitting, we can attempt to increase the *ntree* parameter to ensure the diversity of the ensemble classifier. However, the construction of a model with an *ntree* parameter that is too large is time-consuming to a certain extent. In this work, we first determine the *mtry* parameter with *ntree* = 100 and evaluate the specific performance of the two models. The result is shown in Fig. 2. The vertical axis displays the classification accuracy of specific *mtry* for the two models, and the horizontal axis denotes *mtry*.

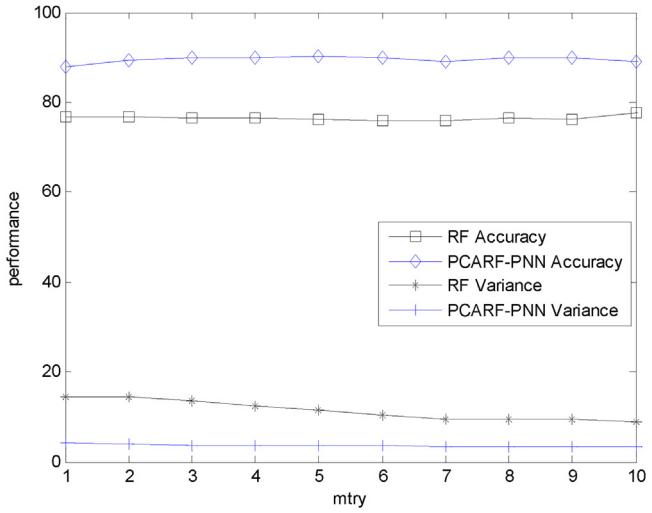

Fig. 2. Performance of the Two Models with Different "*mtry*" Parameters.

Fig. 2 shows the performance of the two methods with different *mtry* for the automobile insurance claims data set. The "RF Accuracy" in the figure indicates the classification accuracy of RF model, the "PCARF-PNN Accuracy" is the classification accuracy of the PCARF-PNN model, the "RF Variance" is the variance of the RF model, and the "PCARF-PNN Variance" is the variance of the PCARF-PNN model. As shown in Fig. 2, after the parameter *mtry* reaches 3 (the default value, which is the square root of the number of features in the Random Forest), the results of the two models tend to be stable (although certain fluctuations occur). This observation indicates that the two methods are quite robust to the parameter *mtry*, although a certain amount of fluctuation of the curve is noted. The parameter *mtry* determines the randomization of the algorithm. For RF, larger *mtry* leads to better performance with lower variance. For PCARF-PNN, the performance is quite stable when it lies in the middle of its range.

As mentioned above, we set the *mtry* parameter to 3 and the decision tree number to *ntree* = {10, 20, . . ., 100, 150, 200, . . ., 500, 1000, 2000, . . ., 5000}. The 10-fold cross-validation method is conducted 100 times. Fig. 3 shows the comparison of the 100 10-fold cross validation accuracy of the two algorithms. The vertical axis shows the average accuracy rate, and the horizontal axis represents the number of decision trees.

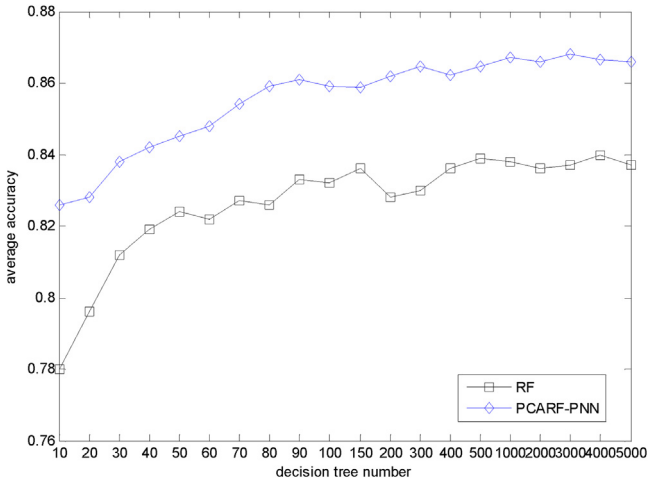

Fig. 3. Accuracy Comparison (with different "*ntree*" parameters).

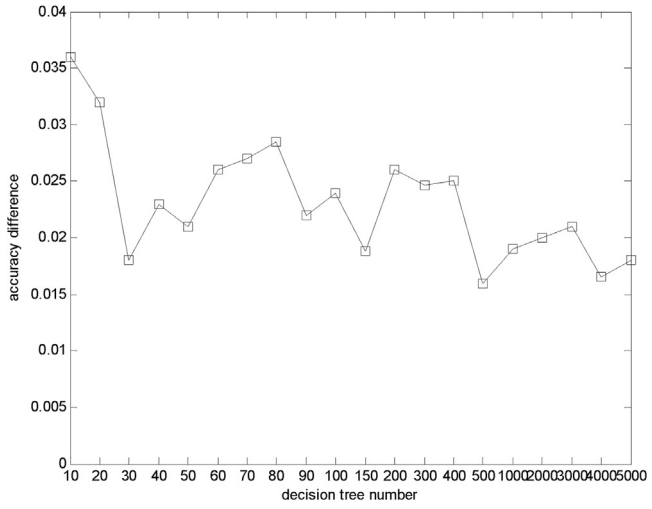

Fig. 4. Accuracy Difference of the Two Models (with various decision trees).

We set Acc_1 as the average accuracy rate of RF and Acc_2 as the average accuracy rate of PCARF-PNN. Fig. 4 shows the accuracy difference of the two models when the forest size is increased from 10 to 5000 on the claims data set, i.e., $Acc_2 - Acc_1$.

The following conclusions can be obtained from the above experimental results:

a The performance of the two models fluctuates with the change in forest scale. When the number of decision trees is small, the effect of the two algorithms is not ideal. With the increase in the forest size, the performance of the algorithm is improved and tends to be stable (when the number of decision tree approaches 100). When the scale of the forest reaches a thousand, no over-fitting phenomenon is observed. A number of decision trees that is too large does not lead to a decline in the performance of the combined classifier. This characteristic indicates that the two models are not sensitive to the *ntree* parameters of the model.

b The average classification performance of PCARF-PNN is better than that of RF. From the effect of parameter *mtry* on the classification accuracy and variance of the model, Fig. 2 indicates that PCARF-PNN is more stable than RF. As shown in Fig. 3, the average classification accuracy of PCARF-PNN is higher than that of RF under different forest scales. Fig. 4 shows the improvement of PCARF-PNN for RF under different forest scales. The result indicates that the performance of all forest scales was improved.

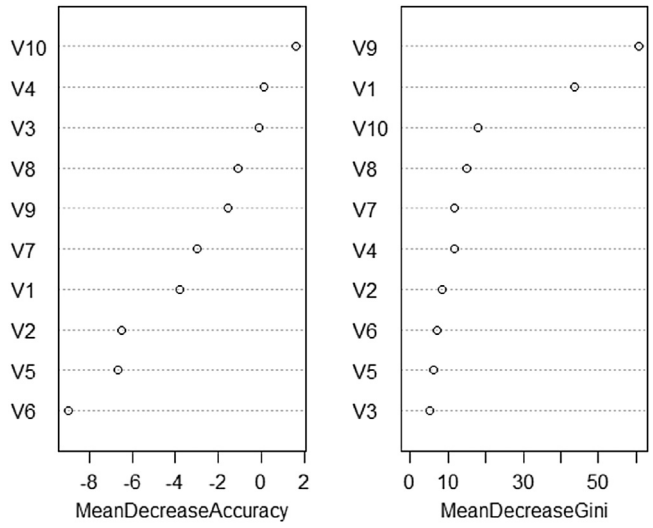

Fig. 5. Importance Ranking of Variables.

Compared with RF, PCARF-PNN is improved for selected data sets on the OOB sample. If there are fewer or even no OOB samples on certain nodes, the effect of PCARF-PNN is similar to that of RF, which is the reason why the promotion effect is not obvious in certain data sets. In addition, when the algorithm has achieved notably good classification results, it is difficult to improve the performance of the classifier. For example, in Fig. 3, the accuracy rate of RF has reached the 90% level, and thus the improvement effect of PCARF-PNN is quite small.

As observed from the variation trend in Fig. 3, the two models are similar in fluctuation of the random forest scale, which indicates that the classification performance improvement of PCARF-PNN is still subject to RF, although the classification performance of PCARF-PNN is better than that of RF. The improvement was based on RF, but with the change in the number of decision trees, the performance of the two models has changed. However, the performance of PCARF-PNN remains better than that of RF. Even when the size of the forest is increasing and the effect is stable, the advantage of PCARF-PNN still exists. This observation fully proves that the improvement effect of PCARF-PNN is remarkable but not caused by random factors.

After the two main parameters are determined, the automobile insurance fraud identification model is constructed to mine the data for the fraud law. For the 10 variables that influence fraud, we can detect which variables are more important such that the fraud law can be obtained. We use the parameter Mean Decrease Accuracy (MDA) and the parameter Mean Decrease Gini (MDG) to evaluate the importance of variables. The parameter MDA is used to measure the reduction of prediction accuracy when a variable value is changed to a random value. The MDA method is based on OOB error. The parameter MDG uses the Gini index to calculate the heterogeneity reduction of the output variable. The MDG method is based on Gini impurity. The larger the two parameters are, the more important the variable will be [25,37]. According to the two parameters, the scatter diagram of the importance ranking of the input variables is shown in Fig. 5.

Fig. 5 shows the importance ranking of each input variable. From the left figure of the parameter MDA, historical times (V10), car testing (V4), underwriting type (V3), garage (V8) and damage photo number (V9) are more important. From the right figure of the parameter MDG, damage photo number (V9), insurance maturity (V1), historical times (V10), garage (V8) and investigation (V7) are more important. The importance of each input variable for fraud is slightly different. Although the difference is not large, the more

important variables should receive more attention in addressing the actual fraud incidents.

In reality, the information accuracy of car testing directly affects the accuracy of fraud identification. The different survey types lead to different time intervals between accident occurrence and investigation. The longer the interval, the more prone to fraud the event will be. The type of garage is related to repair cost. A high repair cost increases the probability of fraud. The number of damage photos represents the sufficiency of evidence at the accident scene. Insufficient evidence increases the possibility that the claimant has exaggerated the accident. The more historical times might be caused by multiple claims for one accident. This factor is also the primary means of fraud. Therefore, in the actual processing of claims, the insurer should fully analyse the characteristics of the policy-holder to make the correct decision.

5. Discussion

The example in Section 4 shows that using the PCA to transform the features increases diversity, thereby resulting in decision trees with low correlation. According to the Theorem 1, the prediction accuracy of Random Forest is improved. If we must choose t features from all T features, the number of exhaustively searched best-split features is at most $T + 1\text{-}t$. If we rank all features according to the splitting rule, the last t-1 features can never be selected to split. Indeed, in practice, certain features can seldom be selected to split because according to the splitting rule, the top features with better discrimination power are dominant. In most cases, each rotation matrix generates one unique split feature. We can use more choices to generate less correlated decision trees. The tree structure is highly sensitive to small changes in the data, as evidenced by Ye et al. [40].

A larger number of decision trees exists in the ensemble feature spaces of PCARF-PNN. This type of ensemble tends to select a tree with higher accuracy. The diversity of the ensemble classifier can be improved because different types of feature spaces are involved. Thus, the PCARF-PNN with an ensemble of feature spaces is observed to perform better. Furthermore, the PCARF-PNN considers the information loss of OOB samples, which improves the accuracy of the ensemble classifier to an extent.

These reasons, although informal, are quite intuitive and aid us in understanding that involving rotation inside each node of a decision tree might have a beneficial effect on diversity among the ensembles. These reasons also explain why different types of decision trees are involved in the ensemble in this study. We also learn that considering the information loss might improve the accuracy and stability of the model.

6. Conclusion

The random forest model established in this paper offers a new method for mining of automobile insurance fraud data, which has a certain reference value. In this paper, the classification mechanism of Random Forest is analysed from the perspective of Potential Nearest Neighbor. The majority voting mechanism is replaced to avoid information loss caused by OOB samples. Furthermore, the Principle Component Analysis transformation method is proposed to transform the data into the PCA space to improve the diversity of the individual classifier. The proposed method of Random Forest is known as PCARF-PNN. The PCARF-PNN method is compared with the traditional Random Forest, Oblique Decision Tree Ensemble and Rotation Forest methods based on 12 data sets selected from various fields. The experimental results show that the proposed method produces better classification accuracy and lower variance. The proposed method is also more stable. Finally, the actual automobile insurance claims data set is used to construct a fraud identification model, and the most important variables are obtained, which can greatly aid the insurance company in detection of fraud.

Acknowledgments

The authors are grateful to Prof. Ponnuthurai Nagaratnam Suganthan for valuable suggestions and the use of the codes available at the homepage address (http://www.ntu.edu.sg/home/epnsugan/).

This work was financially supported by the Project of the National Natural Science Foundation of China (No. 61502280, No. 61472228), the Project of Qingdao Applied Basic Research of Qingdao (special youth project, No. 14-2-4-55-jch), the Natural Science Foundation of Shandong province (No. ZR2014FM009), and the Graduate Education Innovation Program Project of Shandong University of Science and Technology (No. KDYC14016).

Appendix A.

Theorem 2. Suppose that the sample number of the original training set S is n, the training sample $(X, Y) \in R^d \times R$, and a random forest $\left\{T\left(X, \theta_k\right), k = 1, 2, \cdots\right\}$ corresponds to a set of monotonic metric functions $\left\{L_1, L_2, \cdots\right\}$.

Prove: In RF, a new training set S' is generated by randomly selecting n samples from the original training set S with replacement. The classification tree h is trained by the new training set S'. Given a positive integer $M_{try} \ll d$, at each internal node, we randomly select M_{try} features from all d features as the candidate features. We subsequently select the feature that makes the Gini coefficient of the node minimum split. The division of the hyper-rectangle is realized in the M_{try} dimensional feature space. Therefore, the classification tree h corresponds to a monotonic metric function L.

References

[1] Y. Ren, L. Zhang, P.N. Suganthan, Ensemble classification and regression-Recent developments, applications and future directions [Review article], IEEE Comput. Intell. Mag. 11 (1) (2016) 41–53.
[2] D.O. Hebb, The organization of behavior, a neuropsychological theory, in: The Organization of Behavior: A Neuropsychological Theory, John Wiley Chapman & Hall, 1949.
[3] R.E. Sehapire, The strength of weak learnability, Mach. Learn. 5 (2) (1990) 197–227.
[4] L. Breiman, Bagging predictors, Mach. Learn. 24 (2) (1996) 123–140.
[5] T.K. Ho, Random decision forests, in: International Conference on Document Analysis and Recognition, IEEE, 1995, pp. 278–282 (1).
[6] T.K. Ho, Random subspace method for constructing decision trees, IEEE Trans. Pattern Anal. Mach. Intell. 20 (8) (1998) 832–844.
[7] R. Mallipeddi, S. Mallipeddi, P.N. Suganthan, Ensemble strategies with adaptive evolutionary programming, Inf. Sci. Int. J. 180 (9) (2010) 1571–1581.
[8] R. Ye, P.N. Suganthan, Empirical comparison of bagging-based ensemble classifiers, International Conference on Information Fusion (2012) 917–924.
[9] D. DeBarr, Z. Eyler-Walker, Closing the gap: automated screening of tax returns to identify egregious tax shelters, SIGKDD Explor. 8 (1) (2016) 11–16.
[10] R.A. Becker, C. Volinsky, A.R. Wilks, Fraud detection in telecommunications: history and lessons learned, Technometrics 56 (1) (2010) 143–144.
[11] Federal Bureau of Investigation Financial Crimes Report to the Public, Fiscal Years 2010–2011, 2017 (http://www.fbi.gov/stats-services/publications/financial-crimes-report-2010-2011, last Accessed 18 March 2013).
[12] S. Tennyson, P. Salsas-Forn, Claims auditing in automobile insurance: fraud detection and deterrence objectives, J. Risk Insur. 69 (3) (2002) 289–308.
[13] R.A. Derrig, Insurance fraud, J. Risk Insur. 69 (3) (2002) 271–287.
[14] L. Šubelj, M. Bajec, et al., An expert system for detecting automobile insurance fraud using social network analysis, Expert Syst. Appl. 38 (1) (2011) 1039–1052.
[15] Y. Jin, B. Bertis, et al., Binary choice models for rare events data: a crop insurance fraud application, Appl. Econ. 37 (7) (2005) 841–848.
[16] A.K.I. Hassan, A. Abraham, Modeling insurance fraud detection using imbalanced data classification, in: Advances in Nature and Biologically Inspired Computing, Springer International Publishing, 2016.
[17] H.L. Sithic, T. Balasubramanian, Survey of insurance fraud detection using data mining techniques, Int. J. Innov. Technol. Explor. Eng. 2 (3) (2013) 62–65.

[18] L. Zhang, P.N. Suganthan, Random forests with ensemble of feature spaces, Pattern Recogn. 47 (47) (2014) 3429–3437.
[19] L. Zhang, Y. Ren, P.N. Suganthan, Towards generating random forests via extremely randomized trees, in: International Joint Conference on Neural Networks, IEEE, 2014, pp. 2645–2652.
[20] A.M. Prasad, L.R. Iverson, A. Liaw, Newer classification and regression tree techniques: bagging and random forests for ecological prediction, Ecosystems 9 (2) (2006) 181–199.
[21] S.D. Río, V. López, J.M. Benítez, et al., On the use of MapReduce for imbalanced big data using Random Forest, Inf. Sci. 285 (2014) 112–137.
[22] S. Nikumbh, S. Ghosh, V.K. Jayaraman, Biogeography-based informative gene selection and cancer classification using SVM and random forests, in: IEEE Congress on Evolutionary Computation, IEEE, 2012, pp. 1–6.
[23] W. Yoo, B.A. Ference, A comparison of logistic regression, logic regression, classification tree, and random forests to identify effective gene-gene and gene-Environmental interactions, Int. J. Appl. Sci. Technol. 2 (7) (2012) 268.
[24] J.O. Ogutu, H.P. Piepho, T. Schulz-Streeck, A comparison of random forests, boosting and support vector machines for genomic selection, BMC Proc. 5 (3) (2011) 1–5.
[25] L. Breiman, Random forests, Mach. Learn. 45 (1) (2001) 5–32.
[26] P.P. Bonissone, J.M. Cadenas, M.D.C. Garrido, et al., Fundamentals for design and construction of a fuzzy random forest, Stud. Fuzziness Soft Comput. 249 (7) (2010) 23–42.
[27] Y. Ren, L. Zhang, P.N. Suganthan, K-nearest neighbor based bagging SVM pruning, in: Computational Intelligence and Ensemble Learning, IEEE, 2013, pp. 25–30.
[28] B.H. Menze, B.M. Kelm, R. Masuch, et al., A comparison of random forest and its Gini importance with standard chemometric methods for the feature selection and classification of spectral data, BMC Bioinf. 10 (1) (2009) 213.
[29] S. Li, E.J. Harner, D.A. Adjeroh, Random KNN feature selection – a fast and stable alternative to Random Forests, BMC Bioinf. 12 (1) (2011) 1–11.
[30] J. Kittler, P.C. Young, A new approach to feature selection based on the Karhunen-Loeve expansion, Pattern Recogn. 5 (4) (1973) 335–352.
[31] L. Zhang, P.N. Suganthan, Oblique decision tree ensemble via multisurface proximal support vector machine, IEEE Trans. Cybern. 45 (10) (2014) 2165–2176.
[32] L.I. Breiman, J.H. Friedman, R.A. Olshen, et al., Classification and regression trees (CART), in: Classification and Regression Trees, Wadsworth International Group, 1984, pp. 17–23.
[33] Y. Amit, D. Geman, Shape quantization and recognition with randomized trees, Neural Comput. 9 (7) (1997) 1545–1588.
[34] Y. Lin, Y. Jeon, Random forests and adaptive nearest neighbors, J. Am. Stat. Assoc. 101 (474) (2006) 578–590.
[35] D. Laha, R. Ye, P.N. Suganthan, Modeling of steelmaking process with effective machine learning techniques, Expert Syst. Appl. 42 (10) (2015) 4687–4696.
[36] Y. Isler, A. Narin, M. Ozer, Comparison of the effects of cross-validation methods on determining performances of classifiers used in diagnosing congestive heart failure, Measurement Sci. Rev. 15 (4) (2015) 196–201.
[37] A. Cutler, D.R. Cutler, J.R. Stevens, Random forests, Mach. Learn. 45 (1) (2011) 157–176.
[38] L. Zhang, P.N. Suganthan, A survey of randomized algorithms for training neural networks, Inf. Sci. 364-365 (2016) 146–155.
[39] G. Valentini, T.G. Dietterich, Bias-variance analysis of support vector machines for the development of SVM-based ensemble methods, J. Mach. Learn. Res. 5 (3) (2004) 725–775.
[40] Y. Ye, et al., Stratified sampling for feature subspace selection in random forests for high dimensional data, Pattern Recogn. 46 (3) (2013) 769–787.
[41] Y. Li, C. Yan, W. Liu, et al., Research and application of random forest model in mining automobile insurance fraud, International Conference on Natural Computation And, Fuzzy Systems and Knowledge Discovery (2016) 1756–1761.
[42] Z.H. Zhou, J. Feng, Deep Forest Towards An Alternative to Deep Neural Networks, 2017.
[43] N. Dhungel, G. Carneiro, A.P. Bradley, Automated mass detection in mammograms using cascaded deep learning and random forests, in: International Conference on Digital Image Computing: Techniques and Applications, IEEE, 2016, pp. 1–8.
[44] S. Amiri, I. Rekik, M.A. Mahjoub, Deep random forest-based learning transfer to SVM for brain tumor segmentation, International Conference on Advanced Technologies for Signal and Image Processing (2016) 297–302.