Improving Prediction Accuracy Based On Optimized Random Forest Model with Weighted Sampling for Regression Trees

S. Bharathidason[#1], *C. Jothi Venkataeswaran*[*2]

[#]*Assistant Professor, Department of Computer Science, Loyola College, Chennai, India.*
[*]*Dean, Department of Computer Science, Presidency College, Chennai, India.*

***Abstract*- Random Forest (RF) is an ensemble, supervised machine learning technique useful for regression and classification problems. Random forest algorithms tend to use a simple random sampling of observations in building their decision trees. In random forest, random selection has the chance for noisy and outlier data to take place during the construction of trees. This leads to inappropriate and poor ensemble prediction decision. Appropriately handling noise and outliers is an important issue in data mining. This paper aims to optimize, the sample selection through probability proportional to size sampling (weighted sampling) in which the noisy and outlier data points are down weighted to improve the prediction performance by minimizing the error rate in the model. Experimental results have shown that, the random forest can be further enhanced in terms of minimizing the prediction error with weighted sampling.**

***Keywords*— Random Forest, Weighted sampling, Decision trees, Noisy data, Outlier.**

I. Introduction

It is common that noise and outliers exist in real world datasets due to errors such as, typographical errors or measurement errors. When the data is modeled using machine learning algorithms, the presence of noise and outliers can affect the model that is generated. Improving how learning algorithms handle noise and outliers can produce better models [1].

Outlier problem could be traced to its origin in the middle of the eighteenth century, when the main discussion is about justification to reject or retain an observation. “It is rather because of the loss in the accuracy of the experiment caused by throwing away a couple of good values is small compared to the loss caused by keeping even one bad value” [2]. Handling noise and outliers has been addressed in a number of different ways, beginning with preventing overfit. A common approach to prevent overfit is adhering to Occam’s razor which states that the simplest hypothesis that fits the data tends to be the best one. Using Occam’s razor, a trade off is made between accuracy on the training set and the complexity of the model, preferring a simpler model that will not overfit the training set. Another technique to prevent overfit is to use a validation set during training to ensure that noise and outliers are not learned [3].

In Data Mining there are mainly two techniques are available for the data analysis and those techniques are known as the Data Classification and the Data Prediction [4]. Where classification techniques are mainly used to predict the discrete class labels for the new observation or new data on the basis of training dataset provided to the classifier algorithm and prediction techniques generally works with the continuous valued functions.

Random Forest (RF) is an ensemble, supervised machine learning algorithm applied in the domain of Data Mining [4]. Random Forest [5] uses decision tree as base classifier and generates multiple decision trees. In random forest, the randomization is present in two ways: first random sampling of data for bootstrap samples, and second random selection of input attributes for generating individual base decision trees. Strength of individual decision tree and correlation among base trees are key issues which decide generalization error of Random Forest [5].

In random forest, random selection has the chance for noisy and outlier data to take place during the construction of trees. This will decrease the prediction performance of the individual tree in the forest. This paper aims to optimize, the sample selection through probability proportional to size sampling (weighted sampling) in which the noisy and outlier data points are down weighted, to improve the prediction performance by decreasing the error rate in the model.

II. RANDOM FOREST ALGORITHM

Random forest is an ensemble prediction method by aggregating the result of individual decision trees. In the past decade, various methods have been proposed to grow a random forest [5], [6], [7], [8]. Among these methods, Breiman's method [5] has gained increasing popularity because it has higher performance against other methods [9].

Let D be a training dataset in an *M*-dimensional space X, and let Y be a continuous dependent variable. The method for building a random forest [5] follows the process including three steps [6]:

Step 1: Training data sampling: use the bagging method to generate *K* subsets of training data $\{D_1, D_2, ..., D_K\}$ by randomly sampling D with replacement;

Step 2: Feature subspace sampling and constructing regression tree: for each training dataset D_i $(1 \leq i \leq K)$, use a decision tree algorithm to grow a tree. At each node, randomly sample a subspace X_i of F features (F << M), compute all splits in subspace X_i, and select the best split as the splitting feature to generate a child node. Repeat this process until the stopping criteria is met, and a tree $h_i(D_i, X_i)$ built by training data D_i under subspace X_i is thus obtained;

Step 3: Prediction aggregation: ensemble the K trees $\{h_1(D_1, X_1), h_2(D_2, X_2), \ldots , h_K(D_K, X_K)\}$ to form a random forest and use the aggregated prediction of these trees to make an ensemble prediction decision.

The algorithm has two key parameters, *i.e.*, the number of *K* trees to form a random forest and the number of *F* randomly sampled features for building a decision tree. According to Breiman [5], parameter *K* is set to 100 and parameter *F* is computed by $F= [\ \log_2 M + 1]$. For large and high dimensional data, a large *K* and *F* should be used.

III. WEIGHT CALCULATION OF TRAINING SAMPLES BASED ON THE INFLUENCE AND PREDICTION ERROR

In the proposed approach, before constructing a random forest with many trees, a single regression tree is used to measure the influence and the prediction error of each data point, which will be used to train the Random Forest model.

The weights of each data point is determined in two aspects, which are *(i) finding each data point influence on the model through Leave-One-Out method (ii) measuring the prediction error of each data point* using a single regression tree. The mean absolute error is used to measure the performance.

If a data point has high negative influence (degrade the performance) on the model (a regression tree) and has high prediction error rate, then it will be treated as a noisy or outlier data point. These, data points will be down weighted to minimize the overall prediction error during the construction of Random Forest model.

A. Measuring the Influence of Training Samples using Leave-One- Out Method

Leave-one-out is a method where in each iteration, all the data except for a single observation are used for training the model. Using this method each observation's influence on the model can be measured. A single regression tree is used to measure the influence of each data point. The model (a tree) trained without a single observation is called Reduced Model and a model (a tree) trained with full set of training observations is called Full model. The influence of a data point is the difference between these two models performance, which is as follows

$$Influence_i = \eta_{\mathrm{Re}\,duced} - \eta_{Full}$$

Where, $\eta_{\mathrm{Re}\,duced}$ is the Mean absolute error of the reduced model and η_{Full} is the Mean absolute error of the full model

Likewise, each data point's influence on the model is estimated. The estimated influence of each data point is normalized using minmax normalization and it is used as a part of weight calculation to perform the probability proportional to size sampling (weighted sampling) in random forest construction.

B. Measuring the Prediction Error Rate of Training Samples

A regression tree is used to measure the prediction error of each data point. In regression, the dependent variable denoted as *y*, is a continuous value. So, the prediction error is calculated directly by finding the absolute difference between the observed (original) *y* value and the predicted *y* value.

$$Error_i = (\varepsilon_i - \min(\varepsilon)) / (\max(\varepsilon) - \min(\varepsilon)) \;, where$$

$$\varepsilon_i = abs(y_i - \hat{y}_i) \;, \mathrm{i=1,2,3,\ldots,n}$$

Similarly, each data point's prediction error is estimated. The absolute prediction error of each data point is normalized and used as a part of weight calculation to perform the probability proportional to size sampling (weighted sampling) in building the random forest.

C. Combining the Weights

The measured Influence and the prediction error are combined as a weight for each data point in the training sample and these are used to carry out the probability proportional to size sampling for building a random forest.

$$Weight_i = Influence_i * (1 - Error_i)^2 \quad , \mathrm{i=1,2,3,\ldots,n}$$

Thus, the combined weight of each data point in the training sample is calculated and the same is used for weighted sampling to train the Random Forest.

Based on the range of Influence and prediction error the weights may vary for each data point. If a data point has high negative Influence and also has high prediction error, then it is highly down weighted to optimize the Random Forest through Weighted sampling.

IV. OPTIMIZED RANDOM FOREST ALGORITHM

Let D be a training dataset in an *M*-dimensional space X, and let Y be a continuous dependent variable. The method to build an Optimized Random Forest from *X* with *probability proportional to size sampling* (weighted sampling) based on the weight calculated for each data point mentioned in section3 follows the following steps.

Step 0: Weight Initialization: Assign the weight for each Training sample based on the Influence and Prediction Error of the sample;

Step 1: Training data sampling: use the bagging method to generate *K* subsets of training data {D_1, D_2, ..., D_K} by Probability Proportional to size sampling (weighted sampling) D with replacement;

Step 2: Feature subspace sampling and constructing regression trees: for each training dataset D_i ($1 \leq i \leq$ K), use a decision tree algorithm to grow a tree. At each node, randomly sample a subspace X_i of F features (F << M), compute all splits in subspace X_i, and select the best split as the splitting feature to generate a child node. Repeat this process until the stopping criteria is met, and a tree $h_i(D_i, X_i)$ built by training data D_i under subspace X_i is thus obtained;

Step 3: Prediction aggregation: ensemble the K trees {$h_1(D_1, X_1)$, $h_2(D_2, X_2)$, ... , $h_K(D_K, X_K)$} to form a random forest and use the aggregated prediction of these trees to make an ensemble prediction decision.

The algorithm has two key parameters, *i.e.*, the number of *K* trees to form a random forest and the number of *F* randomly sampled features for building a decision tree. For large and high dimensional data, a large *K* and *F* should be used.

V. DATA SOURCE

Detailed information of the Boston housing UCI dataset is obtained from the UCI Machine Learning Repository [10]. The Concrete Compressive Strength dataset information is also available in UCI Machine Learning Repository [11]. The Lung Cancer dataset is acquired from R Datasets [12]. The Fetal Weight dataset is also used to compare the prediction performance of the Random Forest with the proposed method [13]. In all the dataset 70% of the data used as a training sample, remaining 30% of the sample used for testing the model.

VI. RESULTS AND DISCUSSIONS

A series of experiments were conducted on four datasets such as, house, concrete, fetal weight and lung datasets. All datasets used are diverse in nature. In each dataset, it is concluded that the proposed Optimized Random Forest (ORF) performs consistently better than the conventional Random Forest (RF). The mean absolute error (MAE) is used as a metric to evaluate the performance of the algorithms.

A. Performance Analysis

The proposed optimized random forest method is compared with Breiman’s method, the average accuracy of 10 results were computed by performing 10 rounds of experiments on each dataset. The weight of each data point of the training sample is calculated based on the influence and prediction error of the same. In each round, probability proportional to size sampling (weighted sampling) is performed to construct the Optimized Random Forest. The random forest also builds by Breiman’s method by selecting the training samples randomly. The average prediction error of different random forest consisting different number of trees (ranging from 20 to 200 trees with increments 20) generated by the optimized random forest method (corresponding to column ORF) and Breiman’s method (corresponding to column RF) from four datasets are shown in Table1.The proposed method achieves high prediction accuracy by minimizing mean absolute error on the four datasets.

Table 1: Comparison of Prediction Error between Random Forest (RF) and Optimized Random Forest (ORF)

Datasets Trees	HOUSE		CONCRETE		FETAL		LUNG	
	RF	ORF	RF	ORF	RF	ORF	RF	ORF
20	3.827266	3.482816	12.02989	10.88024	585.3801	553.7072	10.37073	8.243092
40	3.679618	3.305167	11.32567	10.5513	582.283	555.1851	9.580592	8.750348
60	3.681222	3.383124	11.46533	11.11292	567.5031	551.6958	9.323817	8.492283
80	3.940995	3.340577	11.20255	10.64099	579.4054	560.4909	9.613958	8.376604
100	3.702792	3.544448	11.01249	10.65593	578.0222	559.7349	9.792745	8.7984
120	3.717286	3.449991	11.09625	10.40803	574.7413	558.3922	9.550382	8.875355
140	3.922508	3.493982	11.32668	10.64081	577.3055	562.3364	9.590688	8.900649
160	3.587278	3.437472	11.13735	10.69423	579.0673	558.7606	9.591132	8.469495
180	3.662153	3.450999	11.28508	10.71857	581.9025	568.1113	9.487939	8.700192
200	3.636611	3.496257	11.20968	10.6245	578.7739	559.2162	9.404361	8.227197

B. Comparison of Error Rate

The preceding section has shown that the Optimized Random Forest (ORF) outperforms the original random forest. The mean absolute error of the random forest is minimized by performing probability proportional to size sampling (weighted sampling) based on the weights calculated for each data point in the training samples. In the above mentioned four datasets, minimizing the prediction error (Mean Absolute Error) ranging from 5% to 12% has achieved with the optimized random forest than the original random forest.

Based on the complexity pattern of the dataset in terms of noise and outlier, the percentage of minimizing the prediction error (Mean Absolute Error) may vary. The proposed optimized random forest method minimized the prediction error rate on the four datasets is shown in Fig.1. The dotted blue curves represent the prediction error obtained with random forest and the red curves represent the prediction error obtained with Optimized Random Forest.

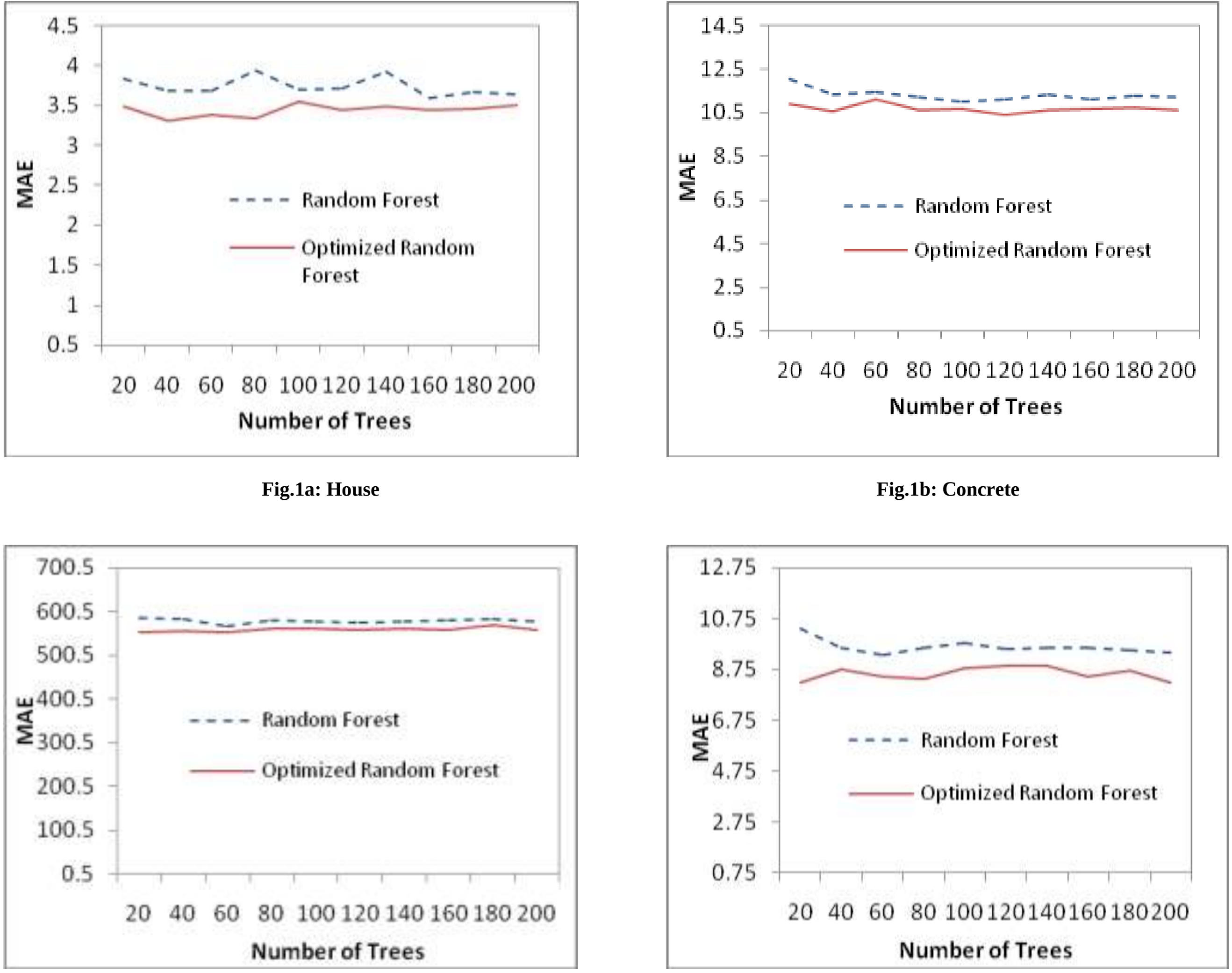

Fig.1a: House

Fig.1b: Concrete

Fig.1c: Fetal

Fig.1d: Lung

Fig 1: Comparison of Prediction Error between Random Forest (RF) and Optimized Random Forest (ORF)

VII. CONCLUSION

This paper presents an evaluation method to identify the noisy and outlier data points in the training sample, and proposed an optimized random forest algorithm which replaces the existing random sampling with probability proportional to size sampling (weighted sampling) in the construction of random forest model. This work aims to minimize the prediction error (Mean Absolute Error) of the random forest through down weighting the data points which increases the prediction error and negatively influence the model. Experimental results on various datasets have shown that the prediction error has been minimized when a random forest is composed with probability proportional to size sampling (weighted sampling). As a result, the prediction accuracy of the random forest is improved in regression analysis.

ACKNOWLEDGEMENT

We are grateful to Prof. Syluvai Antony, Assistant Professor, Dept. of Statistics, Loyola College and Dr. M. Raja, Assistant Professor, Dept. of Advanced Zoology and Biotechnology, Loyola College, Chennai for their constant support and valuable suggestions to complete this research work.

REFERENCES

[1] Michael R. Smith and Tony Martinez, “Improving classification accuracy by identifying and removing instances that should be misclassified”, in *Proceedings of the, The 2011 International Joint Conference on neural networks, IEEE*, 2011, pp. 2690 – 2697.

[2] Barnett, V. and T. Lewis, *Outliers in statistical data*, John Wiley & Sons, pp.1, 1978.

[3] Quinlan, J. R., *C4.5: Programs for Machine Learning*. Morgan Kaufmann, San Mateo, CA, USA, 1993.

[4] Han J and Kamber M, Data Mining: Concepts and Techniques, (2nd Edition), Morgan Kaufmann Publisher. pp. 258. 2006.

[5] Breiman, L, “Random Forests”. *Machine Learning*, Vol. 45 Issue 1, pp. 5-32, 2001.

[6] Baoxun Xu, Junjie Li, Qiang Wang, Xiaojun Chen, “A Tree Selection Model for Improved Random Forest”, *Bulletin of advanced technology research*, vol.6(2), 2012.

[7] Dietterich, T.G. “An Experimental Comparison of Three Methods for Constructing Ensembles of Decision Trees: Bagging, Boosting, and Randomization,” *Machine Learning*, vol. 40(2):139–157, 2000.

[8] Ho, T. “The random subspace method for constructing decision forests”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20(8):832–844, 1998.

[9] Banfield, R.E., L.O. Hall, K.W. Bowyer and W.P. Kegelmeyer, “A Comparison of Decision Tree Ensemble Creation Techniques”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29(1): 173–180, 2007.

[10] Harrison, D. and Rubinfeld, D.L, “Hedonic prices and the demand for clean air”, *J. Environ. Economics & Management*, vol. 5: 81-102, 1978.

[11] Cheng I. Yeh, “Modeling of strength of high performance concrete using artificial neural networks”, *Cement and Concrete Research*, vol. 28 (12):1797-1808, 1998.

[12] Loprinzi CL. Laurie JA. Wieand HS. Krook JE. Novotny PJ. Kugler JW. Bartel J. Law   M.Bateman M. Klatt NE, “Prospective evaluation of prognostic variables from patient-completed questionnaires”. North Central Cancer Treatment Group. *Journal of Clinical Oncology*, vol. 12(3):601-7, 1994.

[13] Sereno, F. *et al.*, “The Application of Radial Basis Functions and Support Vector Machines to the Foetal Weight Prediction”. *Intell Eng Syst Through Artif Neural Networks*, vol. 10: 801-806, 2000.