Feature Engineering on Forest Cover Type Data with Ensemble of Decision Trees

Pruthvi H.R*, Nisha K.K†, Chandana T.L‡, Navami K§ and Biju R M¶
¶Faculty, Department Of Information Technology
National Institute of Technology Karnataka, India,
* phr.11it64@nitk.edu.in,†nishakk94@gmail.com,‡tlchandana@gmail.com,§knavami@gmail.com,¶bijurmohan@gmail.com

***Abstract*—The paper aims to determine the forest cover type of the dataset containing cartographic attributes evaluated over four wilderness areas of Roosevelt National Forest of Northern Colorado. The cover type data is provided by US Forest service inventory, while Geographic Information System (GIS) was used to derive cartographic attributes like elevation, slope, soil type etc. Dataset was analyzed, pre processed and feature engineering techniques were applied to derive relevant and non-redundant features. A comparative study of various decision tree algorithms namely, CART, C4.5, C5.0 was performed on the dataset. With the new dataset built by applying feature engineering techniques, Random Forest and C5.0 improved the accuracy by 9% compared to the raw dataset.**

I. INTRODUCTION

The four wilderness areas at Roosevelt National Forest under study are Rawah (73,213acres), Comanche Peak (67,680 acres), Neota (9,647 acres) and Cache la Poudre (9433 acres). These areas are located 70 miles to the north west of Denver, Colorado, as shown in Figure 1. The reason for selecting these wilderness areas is that they contain forest area that have very less direct human intervention. Hence, the present composition of the forest cover type is due to natural ecological process rather than the intervention of the forest management.

Blackard et al.[3] have compared two techniques for predicting forest cover types. The comparison of the two techniques indicated that model built on feedforward artificial neural network (70.58%) predicted the forest cover type with higher accuracy than a statistical model based on Gaussian discriminant analysis (58.38%). B. Chandra et al. [5] used the same dataset to evaluate the performance of the decision trees. Decision tree algorithm achieved a maximum classification accuracy of 88.14% as compared to that of 70.58% from artificial neural networks. Ragini Jain et al. suggested a hybridized rough set model that provides mechanism to trade-off between different performance parameters like - accuracy, complexity, number of rules and number of attributes in the resulting classifier for a large benchmarking dataset.

During the last decade, many supervised learning methods were introduced. Caruana et al.[4] took ten supervised learning algorithms into consideration and made an empirical comparison using eight performance criteria. Performance evaluation of Naïve Bayes, Logistic Regression, SVMs, Neural Network, Decision Trees, Random Forests, Bagged Trees, Boosted Trees, Boosted Stumps and Memory-based learning on eleven binary classification problems was done. Various performance metrics namely, ROC Area, accuracy, F-score, Lift were used. The conclusion of their study indicates that learning methods as follows: Decision Tree, Random Forests, Boosting, Bagging and SVMs achieve higher performance than others.

Entezari - Malecki et al. [6] have compared different classification techniques based on size of the sample and type of attributes against Area Under the Curve (AUC) of ROC. Their analysis showed that Decision Tree and C4.5 shows better performance on many datasets. In instances where the number of continuous attributes is higher than the discrete attributes, Decision tree, C4.5 and SVM show excellent accuracies.

In this paper, we propose to predict the forest cover type from strictly cartographic attributes. We intend to do so by applying preprocessing and feature engineering techniques on the dataset followed by decision tree models. We also make a performance comparison of different decision trees - C4.5, C5.0, and CART, based on different metrics - accuracy, area under roc curve and number of nodes on the current dataset.

The rest of the paper has been organized as follows: The proposed model has been explained in detail in Section II. Section III describes the results of the various experiments performed at every stage, using decision tree for classification. Section IV concludes the paper.

II. PROPOSED WORK

This section shall give a detailed description of all the steps followed in our model. Figure 2 shows all the steps undertaken for the prediction of the forest cover type using decision tree has a classifier.

A. Data Collection

In the data collection phase, data is collected from various sources and integrated. UCI Machine Learning Repository provides the current dataset. This dataset has been derived

978-1-4799-8047-5/15/$31.00

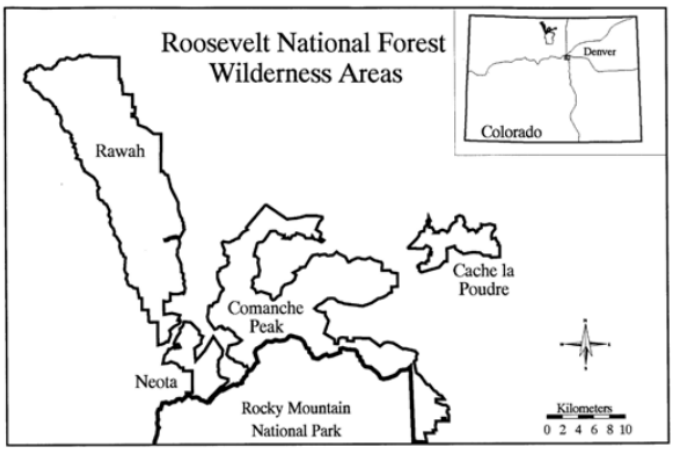

Fig. 1: Roosevelt National Forest Wilderness Area

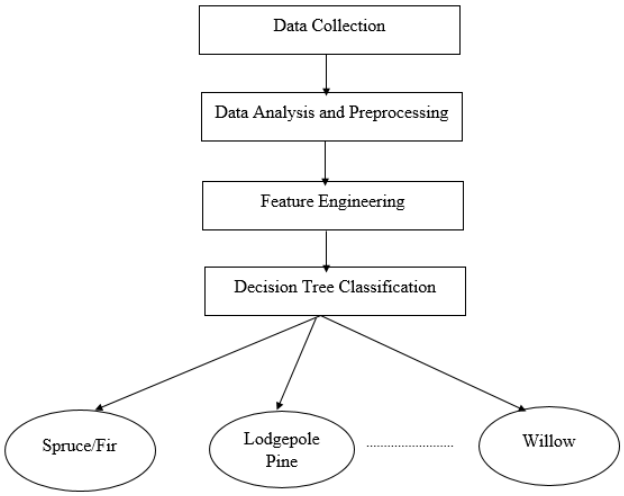

Fig. 2: Process followed in building our model.

from Region 2 Resource Information System (RIS) data of US Forest Service (USFS). Forest cover type dataset contains only cartographic variables (no remotely sensed data). Data is unscaled in the raw form and has binary columns for the variables like wilderness area and soil types which are qualitatively independent. The dataset for forest cover type prediction includes 54 features. There are forty soil types and four wilderness areas which are binary attributes and other attributes are numeric in nature. We intend to predict the forest cover type which is an integer, for every tuple in the dataset. The seven forest cover types are Lodgepole Pine, Spruce/Fir, Willow, Ponderosa Pine, Douglas/Fir, Aspen and Krummholz and are represented as integers. The dataset contains continuous, binary and nominal data. Table I gives the description of all the attributes in our dataset.

TABLE I: Description of attributes for forest cover dataset

Attribute Name	Data Type	Description
Elevation	Quantitative	Elevation in meters
Aspect	Quantitative	Aspect in degrees azimuth
Slope	Quantitative	Slope in degrees
Horizontal_Distance _To_Hydrology	Quantitative	Horizontal distance to nearest surface water features
Vertical_Distance _To_Hydrology	Quantitative	Vertical distance to nearest surface water features
Horizontal_Distance _To_Roadways	Quantitative	Horizontal distance to nearest roadways
Hillshade_9am	Quantitative	Hillshade index at 9am,summer solstice
Hillshade_Noon	Quantitative	Hillshade index at noon,summer solstice
Hillshade_3pm	Quantitative	Hillshade index at 3pm, summer solstice
Horizontal_Distance _To_Fire_Points	Quantitative	Horizontal distance to nearest wildfire ignition points
Wilderness_Area(4 binary columns)	Qualitative	Wilderness area designation
Soil_Type(40 binary columns)	Qualitative	Soil type designation

B. Data Analysis and preprocessing

After collecting the data, it is important to analyze the data and understand the relationships between various attributes. Such an analysis will help in data preprocessing to remove the redundant and irrelevant attributes. The changes made to the dataset after data analysis and preprocessing is as stated below:

1) According to [2], the wilderness areas which are more typical over the dataset are Rawah and Comanche Peak, due to their collection of tree species and range of predictive variable values (elevation, etc.) Cache la Poudre has relatively low elevation range and species composition and hence would probably be more unique than the others.
2) Elevation with Wilderness Area: Among the four wilderness areas, the area 2 - Neota has the highest mean elevation value probably. The second highest mean elevation value would be in the area of Rawah (area 1) and Comanche Peak (area 3) followed by Cache la Poudre (area 4), where it is found to have the lowest mean elevation value.
3) Cover type with wilderness area: Neota would have its primary tree species to be spruce/fir (type 1), while Rawah and Comanche Peak would probably have Lodgepole pine (type 2) as their primary species, the next important species is spruce/fir and aspen (type 5). Cache la Poudre usually tend to have Ponderosa pine (type 3),

Douglas-fir (type 6), and cottonwood/willow (type 4).

4) Conversion of binary attributes into categorical attributes: The 40 soil types and 4 wilderness areas which were binary attributes were converted into corresponding categorical attributes. The categorical attribute corresponding to soil types took values from s1 to s40, while the attribute corresponding to wilderness areas took values from w1 to w4.
5) Missing values of Hillshade_3pm: When Hillshade_3pm was plotted, it was found to have certain missing values, which were filled with the median of the values.

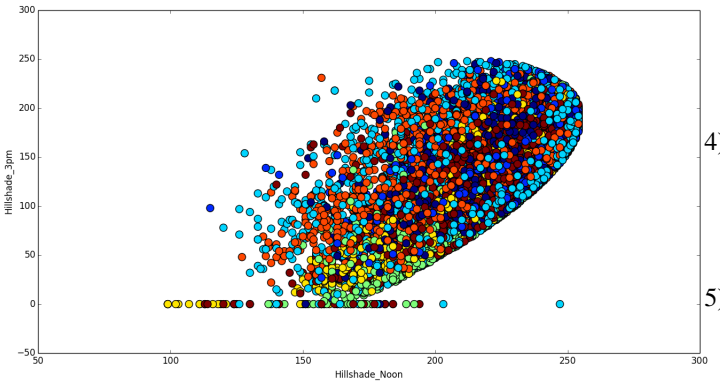

Fig. 3: Hillshade_3pm Vs Hillshade_Noon

C. Feature Engineering

1) Feature Extraction: Feature extraction refers to the process of deriving features from the initial set which are intended to be more relevant, non-redundant. This improves the following learning and generalization steps, also making the features more interpretable.

After a thorough data analysis and understanding the relationship between the features, new features were engineered from the existing ones. This section gives a detailed description of the feature engineering done on the data set.

1) Preprocessing based on Soil Type: Soil Types are numbered from 1 to 40 and are categorized based on the USFS Ecological Landtype Units (ELUs). The ELU code of each soil type is a four digit number, where the first and second digits refer to the climatic and geologic zone respectively. The third and the fourth digit refer to a certain mapping unit and does not have any relationship with the climatic or geologic zone. The two changes made to the dataset using the ELU codes are as follows:
   - Soil types which had same climatic(first digit) and geologic zone(second digit) were found to have similar characteristics and were grouped as one. There were 11 such groups and hence, 40 soil types were converted into 11 soil types.
   - The first digit and the second digit of the ELU code is used to create two categorical attributes called 'Climatic' and 'Geologic', which take eight distinct values [2]
2) Negative values of Vertical_Distance_To_Hydrology [1]: Looking at the distribution of Vertical_Distance_To_Hydrology in Figure 4, we found that it has some negative values. We created another variable called 'Hg_wter', which indicates whether this attribute has a positive or a negative value.
3) Relationship between Vertical_Distance_To_Hydrology and Elevation: Elevation and Vertical_Distance_To_Hydrology are correlated to each other. In the Figure 5, coloring each cover type in different colors, seem to reveal a pattern of the plotted points. Hence, we create a new feature called EV_DTH which gives a simpler relation seen in Figure 6. Here, EV_DTH is given by subtracting Vertical_Distance_To_Hydrology from Elevation.
4) Relationship between Elevation and Horizontal_Distance_To_Hydrology: From a similar graph, it has been observed that Elevation and Horizontal_Distance_To_Hydrology are correlated. Hence, we define a new attribute called EH_DTH.
5) Other new features: A new set features HyF_1, HyF_2, HyR_1, HyR_2, FiR_1, FiR_2 were derived from combining all distance based attributes. The final feature set is shown in Table II.

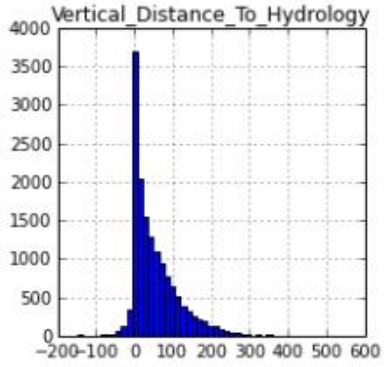

Fig. 4: Plot of Vertical_Distance_To_Hydrology

2) Feature selection: Feature selection aims to select relevant and non redundant subset of features from the initial set. Removal of features which are irrelevant and redundant is the primary motive of feature selection. Redundant features are the ones that provide no more information than the selected features selected at current time. The features which provide no useful information at all are irrelevant.

On plotting features against each other, we observed that the features are independent of each other as in Figure 7, hence not redundant.

Also, decision trees use attribute selection measures like Gain Ratio during tree construction. Thus, only the relevant attributes appear in the final model. This eliminates the need for using any other feature selection technique.

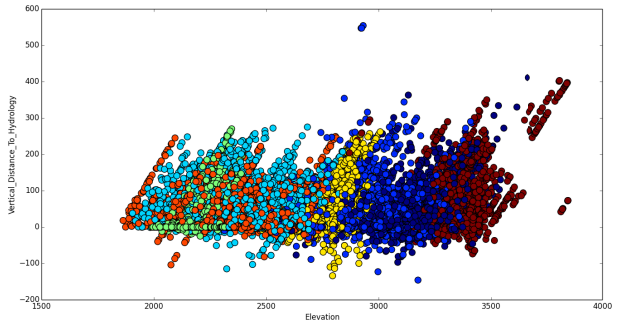

Fig. 5: Graph of Vertical_Distance_To_Hydrology Vs Elevation

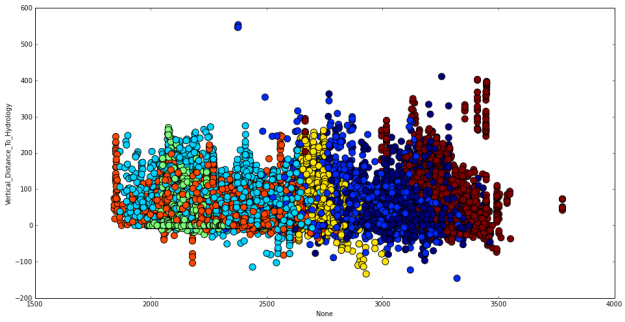

Fig. 6: Graph of EV_DTH Vs Elevation

TABLE II: Description of the new additional features for Forest Cover Dataset

Attribute Name	Data type	Description
Wilderness Area	Qualitative	Wilderness area designation
Soil type	Qualitative	Soil type designation
Aspect2	Qualitative	Aspect in degrees azimuth
Hg_wter	Qualitative	Indicative for positive or negative values to Vertical_Distance_To_Hydrology
EV_DTH	Qualitative	Elevation - Vertical_Distance_To_Hydrology
EH_DTH	Qualitative	Elevation - Horizontal_Distance_To_Hydrology*0.2
Dis_To_Hy	Qualitative	(Horizontal_Distance_To_Hydrology^2 + Vertical_Distance_ To_Hydrology^2)$^{1/2}$
HyF_1	Qualitative	Horizontal_Distance_To_Hydrology + Horizontal_Distance_To_Fire_Points
HyF_2	Qualitative	Horizontal_Distance_To_Hydrology - Horizontal_Distance_To_Fire_Points
HyR_1	Qualitative	Horizontal_Distance_To_Hydrology + Horizontal_Distance_To_Roadways
HyR_2	Qualitative	Horizontal_Distance_To_Hydrology - Horizontal_Distance_To_Roadways
FiR_1	Qualitative	Horizontal_Distance_To_Fire_Points + Horizontal_Distance_To_Roadways
FirR_2	Qualitative	Horizontal_Distance_To_Fire_Points- Horizontal_Distance_To_Roadways

D. Decision Trees

A decision tree is a simple model for classifying data objects. Decision tree learning is a popular technique used for

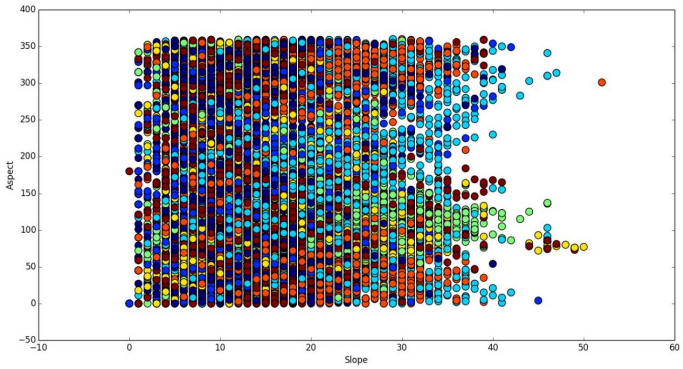

Fig. 7: Plot of Aspect vs Slope

classification. A test on attibute is performed at each internal node of the decision tree. The branches leading from the node represent the outcome of this condition. The leaves of the tree are labeled with class names. A tree is constructed by partitioning the initial data set into subsets. These subsets are got based on the test performed on attribute at current node. This process is repeated on every derived subset recursively and is called recursive partitioning. The recursion is terminated when the subset at a node has pure partions, or when partitioning is not significant to classifications. A number of decision trees have been proposed in the literature and they mainly differ by the measure used for attribute selection. For example, C4.5 and C5.0 use Gain Ratio whereas CART uses Ginni as the attribute selection measure, in the tree construction phase. One such important Decision tree is C5.0.

1) C5.0: C4.5 algorithm ensured better way of building a decision tree with the use of gain ratio has the spitting factor but it lacked many non-functional requirements to popularize among the applications. Hence, C5.0 [7] was developed as an improvement over the existing C4.5 algorithm. Many key aspects found in C5.0 makes it better than C4.5 Algorithm. Below are the list of its extended features:

- Rules Formation - Building the decision trees and using then for every test set prediction leads to high wastage of time. In order to overcome this, C5.0 has incorporated a ruleset formation instead of trees to save significant amount of time and space.
- Winnowing technique - used in C5.0 helps in reduction of memory by using lesser sample set, which was not available in C4.5.
- Smaller decision trees - C5.0 and C4.5 give comparative results. However, C5.0 produces smaller decision tree than C4.5.
- Boosting - Creation of multiple decision trees is supported by boosting. These trees formation is improvement over time till no more misclassification. Then based on

the decision from all the classifiers a better prediction is made leading to better accuracy.

- Weighting - C5.0 allows you to vary the importance of different cases by giving different weights. Minimizing the weighted predictive error rate is the way to handle weighting.
- Misclassification Costs - In C4.5 and other decision trees all errors are considered equal. In practice severity of some misclassifications could be more than the others. Hence, C5.0 provides a facility to define separate cost for each predicted versus actual class pair ratio. The decision tree is then constructed by C5.0 to reduce the expected misclassification costs instead of error rates.
- Data Types - Data types like times, dates, timestamps, ordered discrete attributes, and case labels are provided by C5.0, in addition to the data types provided in C4.5. C5.0 allows values to be missing, or even noted as not applicable. Also, new attributes which are functions of other attributes can be defined using C5.0.

III. Result and Analysis

This section shall give a detailed description and analysis of the results obtained from the various experiments conducted as stated in the proposed work.

All the experimentations were carried out in the R programming using the CRAN built in packages and the python codes were used for the feature engineering and data analysis.

A. Decision Trees

The decision trees was selected has the classifier for our forest cover type prediction from the literature survey. Experimental study for the selection of the right decision tree was conducted using the existing packages in R.

The Table V summarizes the comparison of decision trees under the selected metrics. The classification metrics taken in to consideration include:

- Accuracy: Gives a measure of the number of the samples are correctly predicted.
- Tree size: Indicates the number of nodes that appear in the constructed decision tree.
- AUC: Area Under Curve for Receiver Operating Characteristic

TABLE III: Performance Evaluation

Performance Metrics/Decision trees	C4.5	C5.0(No Boosting)	CART
Accuracy in %	80.78	91.11	79.45
Size of the Tree(nodes)	2111	772	951
AUC	0.92	0.88	0.94

Based on all the measures mentioned above, important point is to note that C5.0 has higher accuracy and less tree size due to the various built-in techniques like pruning, boosting and winnowing to improve tree performance and possibly reduce the memory requirements for building the model.

B. Feature engineering

As stated in the Section II, our dataset needed preprocessing and feature extraction. Table IV gives the modifications made to the features and the improvements to the model when compared to the base case.

Base case: Accuracy of the model when tested on real test set with no changes made to the raw data. This case gave an accuracy of 68.44%.

TABLE IV: Data Pre-Processing and Feature Extraction

Expt No.	Feature Description	Feature Modifications	Reason	Results
1	Missing Soil Types7 and Soil Types15 in the training set	Removal of s7 and s15 from test set (There were 105 samples with soil type s7).	The relevance of these features seemed to be very low according to the attribute usage obtained from C50 tree built from test set.	Improved to 69.13%
2	Soil Type (Qualitative) with 40 binary columns	Generalization of Soil Type to 11 Columns	Generalization is based on the ELU codes of soil types	Decreased to 67.32%
3	Soil Type (Qualitative) with 40 binary columns	Generalization based on geologic and climatic zones	Generalization is based on the ELU codes of soil types	Decreased further to 66.95%
4	Wilderness Area (Qualitative) with 4 binary columns	Generalization to one categorical feature	To lower this features used due to its lower relevance.	Improved to 68.89%
5	Missing values in Hillshade_3pm	Replaced those values with the mean of all Hillshade_3pm	To remove the outliers	Improved to 69.17%
6	Soil type (Qualitative) with 40 binary columns	Generalized to one categorical feature	To lower the features used due to its lower relevance	Improved to 69.13%

The experiments performed above showed very slight improvements over the primary dataset. Based on the accuracy improvement, the features were selected for the new dataset.

The detailed data analysis with decision trees also showed the need for enhancing the feature space with more relevant features. So, the further experimentations were performed in building new features from the given feature set. Table II shows the lists of new features obtained from the feature engineering:

The new feature set when tested on the C5.0 decision tree with no boosting gave an improvement in accuracy to 69.22%.

C. Feature selection

Feature Selection is a technique to select the best subset of features from the given set in order to maximize the accuracy. The decision trees are known to have the build in feature selection based on the splitting parameter. These are in fact

known as embedded feature selectors.
Hence, no additional feature selection was performed.

D. Pruning

Pruning is a method to reduce the size of the decision tree. It is used to reduce over fitting, where the model built achieves perfect accuracy on training data, but the model is too specific that it doesn't give good results for anything but training data. Confidence factor is a papmeter in C5.0 which controls the amount of pruning. In our case, reducing the value of confidence factor from 0.25 to 0.15, thereby increasing the amount of pruning found to improve the accuracy to 69.25%.

This will reduce the accuracy on the training data, but (in general) increase the accuracy on unseen data.

E. Ensemble Learning

The new features obtained from the previous steps were found to be more relevant. Yet the model showed only 1% improvement . Ensemble learning is a technique to enhance the learning of weak base learners like decision trees. Hence, ensemble techniques Random forest and C5.0 were used for further testing. Random forest gave an improvement in accuracy to 77.24% which was found to be better than that of single decision tree. Whereas C5.0 with boosting iterations set to 10 gave an improved accuracy to 76.02%.The Table V gives the detailed comparison of both the techniques.

TABLE V: Performance Evaluation for Ensemble Techniques

Performance Metrics/Ensemble Learning with 10 trials	C5.0	Random Forest
Accuracy in %	76.02	77.24
AUC	0.82	0.85

The above boosting and bagging techniques showed to increase the prediction performance. The detailed study of these ensemble techniques and their modification may help in improving the overall model for better prediction.

IV. Conclusions

The purpose of this study is to use the decision tree classification algorithms for predicting the forest cover type. The forest cover data of the Roosevelt National Forest of northern Colorado was used to evaluate the performance of various Decision Tree algorithms. Among the decision trees, C5.0 was found to give higher accuracy. Various feature engineering techniques performed on the dataset showed improvement over the primary data-set. The new feature set when tested with C5.0 decision tree with no boosting gave an improvement in accuracy to 69.22%. Random forest and C5.0 gave an improvement in accuracy to 77.24% and 76.02% respectively. These show that ensemble techniques can enhance the performance of decision trees considerably. These ensemble techniques can be improved further.

Acknowledgment

We would like to thank Mr. Biju R Mohan for guiding us and giving his valuable insights throughout the project.

References

[1] "Kaggle." [Online]. Available: https://www.kaggle.com/c/forest-cover-type-prediction/forums/t/10693/features-engineering-benchmark

[2] "Uci repository database." [Online]. Available: https://archive.ics.uci.edu/ml/machine-learning-databases/covtype/covtype.info

[3] J. A. Blackard and D. J. Dean, "Comparative accuracies of artificial neural networks and discriminant analysis in predicting forest cover types from cartographic variables," *Computers and electronics in agriculture*, vol. 24, no. 3, pp. 131–151, 1999.

[4] R. Caruana and A. Niculescu-Mizil, "An empirical comparison of supervised learning algorithms," in *Proceedings of the 23rd international conference on Machine learning*. ACM, 2006, pp. 161–168.

[5] B. Chandra and V. Pallath Paul, "Prediction of forest cover using decision trees," *J. Ind. Soc. Agril. Statist*, vol. 61, no. 2, pp. 192–198, 2007.

[6] R. Entezari-Maleki, A. Rezaei, and B. Minaei-Bidgoli, "Comparison of classification methods based on the type of attributes and sample size," *Journal of Convergence Information Technology*, vol. 4, no. 3, 2009.

[7] S.-l. Pang and J.-z. Gong, "C5.0 Classification Algorithm and Application on Individual Credit Evaluation of Banks," *Systems Engineering - Theory & Practice*, vol. 29, no. 12, pp. 94–104, Dec. 2009. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S1874865110600920