Forest Cover Type Prediction using Cartographic Variables

Tejas Anant Wagh
Kogentix technologies,
Hyderabad, India.

R. Bhargavi
Associate Professor
Schools of computer Science and Engineering,
Vellore Institute of Technology,
Chennai Campus,
Chennai, India

Tanmay Anant Wagh
Information Technology,
NBN Sinhgad School of Engineering,
Pune, India

R. M. Samant
Assistant Professor
Department of Information Technology,
NBN Sinhgad School of Engineering,
Pune, India

ABSTRACT
Information regarding forest land is highly required for developing ecosystem management. This paper provides an analysis related to classification and prediction estimation using machine learning techniques. The approach is to predict the forest cover type using the cartographic variables like aspect, slope, soil type, wilderness area etc. Various Data mining techniques such as decision tress, random forest, regression trees, and gradient boosting machines are used for prediction of the forest cover type. Using these machine learning methods models have been developed and tested for accuracy ranging from 19.4% to 74.8%. Kaggle dataset which is the standard benchmarking dataset, is taken for comparison studies. The comparisons of these models are done to identify a better model for predicting the forest cover type with better accuracy. For performance comparison, metrics like accuracy and error rate are used. An important aspect of the study is the use of different performance measures to evaluate the learning methods.

Keywords
Machine learning, classification and regression, decision trees, random forest, gradient boosting machines.

1. INTRODUCTION
Machine learning is a scientific discipline that explores the construction and study of algorithms that can learn from data. Machine learning effectively learns from train dataset and apply transformation on the test data to get the estimated results. In this new learning algorithms have emerged (e.g. Random forest, bagging and boosting, neural networks) that has given an excellent performance, accuracy and results.

The machine learning algorithms are now used in many domains and different performance metrics are appropriate for the particular domain [12]. The different performance metrics measure different tradeoffs in the predictions made by the classifier and this is important to evaluate the results which are predicted on dataset using this algorithms. Machine learning is improving by acquisition of

knowledge from experience. The industrial perspective has also turned towards the domain specific and knowledge engineering in machine learning. It consumes less time and human activity with automatic techniques that improves performance and accuracy and efficiency by discovering and exploiting regularities in the dataset using training and testing. The main challenge machine learning has its ability to produce system that are used in industry, education etc. Most experiments done in machine learning on the separate test set in more than one domain i.e. better than performing on the data without learning.

Machine learning is mostly based on the statistics. There are various algorithms which have connections with the statistics. For example, the boosting which is one of the prediction algorithm used for forest cover type is now widely thought to be in stage wise regression using loss function. There are various applications of machine learning including medical, bioinformatics, search engines, credit card fraud, natural language processing, speech and handwriting recognition, robotics and game playing.

The forest cover dataset which consist of predominant kind of tree cover and various attributes which are cartographic and we used different algorithms for predicting the cover type using cartographic variables. The algorithms used are firstly regression. The Linear regression is a tool in statistics and machine learning [3]. It functionality is to create a relationship between the sets of attributes or variables i.e. dependent and independent variables and predict the values of dependent variables given new values of independent variables. Secondly decision tree are used which are produced by algorithm that identify various ways of splitting a data into branch like segments [4] [14]. The output is predicted by each and every iteration in the tree. Thirdly Random forest algorithm which are used for classification and predication in which multiple decision tree are made by averaging at every iteration and the values are classified and predicted [5][8]. Fourthly, Gradient boosting machines are newly developed algorithm which gives much good accuracy. These are the ensemble or committee classifiers. Popular approaches for ensemble classifiers are bagging and boosting which are also a type of a decision tree but it mainly focusses on the loss function or error prone areas by which it can predict accurate results[9][10].

2. LITERATURE SURVEY
The attempt to explore the space of parameters and common variations for each learning algorithm as thoroughly as it is computationally feasible. This section summarizes the parameters used for each learning algorithm.

2.1 Methodologies
In [3], it tells about the linear and multiple and multivariate regression which is an important analysis tool in statistics and machine learning. It aims to create the linear relationship between two sets of variables, namely the dependent/response and the independent/predictor variables, and then to predict the values of the dependent variables given new values of independent variables. In recent decades, regression analysis has been widely used for prediction and forecasting, and intersects much with the field of pattern recognition and machine learning.

The multivariate linear regression is another important because this model models the relationship between the multiple dependent variables and set of independent variables.

In [4][14], Decision tree is another approach for classification. Decision trees are created by algorithms that distinguish various ways of dividing a data set into branch-like sections. These sections form an inverted decision tree that starts with a root node at the tip of the tree. The aim of analysis is reflected in this root node as a simple, linear display in the decision tree interface. The epithet of the field of data that is the object of analysis is usually exposed, along with the spread or distribution of the values that are contained in that area.

Decision trees are the form which consist of multiple variable with multiple effects analyses. This multiple variables allow us to predict, explain, describe, or classify a target or outcome. Decision trees attempt to see a solid relationship between input values and the target values in a group of observations that form a data set. When a set of input values is identified as possessing a solid relationship with a target value, then all of these values are grouped in a bin that becomes a branch of the decision tree. These groupings are defined by the observed shape of the kinship between the bin values and the object [13].

In [5], it gives importance of applying random forest on dataset. Random forest is statistical method for classification and decision–tree based supervised learning algorithm. Its algorithm is an ensemble classification which is unsurpassable in accuracy among current data mining algorithms. It has been used for Microarray cancer [6], android malware [7], and intrusion detection.

The random forest consists of many individual trees [5]. Each tree votes on an overall classification for the given set of data and the random forest algorithm chooses the individual classification with the most ballots. There are two different sources of randomness in Random Forests: random training set (bootstrap) and random selection of properties.

In [9][10], Gradient boosting machines (GBM) is another methodology discussed. Boosting is used to bring down the prediction error of any weak learner that consistently generates classifiers only a little safer than random guessing. Boosting works by repeatedly passing a given weak learner on various distributions over the training data, and then combining the classifiers produced by the weak learner into a single composite classifier. In each iteration, the distribution of the training data depends on the operation of the classifier trained in the former iteration. The method to calculate the distribution of the training data and to combine the predictions from each classifier is different for various boosting methods.

GBM is mainly used for the binary classification but now a days it known for good prediction, but the drawback is it requires a huge training dataset then the test data [11].

3. PROPOSED METHODOLOGY

The methodology applied is essential step for any data analysis for any dataset and applying models.

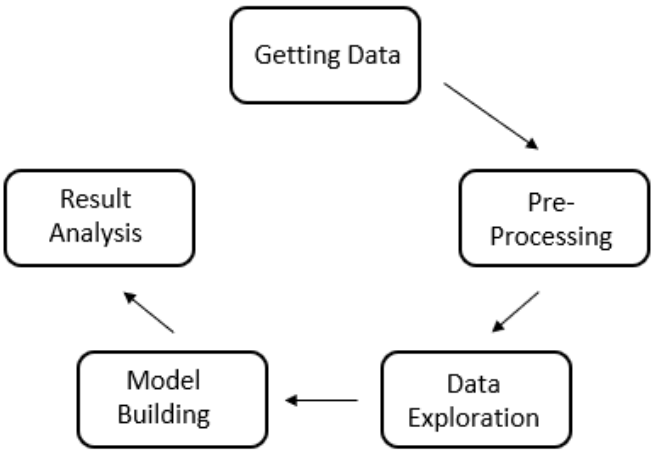

Fig. 1 Workflow Design for Data Analysis and Model Building

The Proposed methodology is explained in Fig 1. Where it shows the complete workflow design starting from taking input dataset to pre-process and Explore the data for better understanding. After that model building and Result Analysis.

As shown in Fig 1, below is explanation of each block consisting in the Workflow Design

3.1 Getting Data

The data has been taken from Kaggle competition 'Forest Cover Type Prediction. This data is obtained from the US Geological Survey (USGS) and the US Forest Service (USFS) which is in open domain and includes four wilderness areas located in Roosevelt National Forest of northern Colorado, and provided by Machine Learning Laboratory of University of California Irvine. Dataset contains 581012 entries with 54 attributes each. However, there are only 12 real features because two of them are represented as a vector opposed to number notation [1].

Each entry is observation on 30 X 30 m^2 patch of forest land and goal for this dataset is to predict cover type of this patch. Training set is chosen in such a way so that each class has the same number of observations.

3.2 Pre- Processing

There is a need of data pre-processing because the data may be incomplete or inconsistent or noisy [2]. There are many ways to deal with un-processed data via:

3.2.1 Data Cleaning

By this term we mean to fill the missing values in data, identifying and removing outliers in data, smoothing data. In this data there are no missing values but when we eliminate sum attributes it is require levelling it .

3.2.2 Data Transformation

In this stage operations like normalization and aggregation are performed. It is the most important step as we use normalization in our data to aggregate 4 wilderness area attributes and 40 soil type into a single Normalized forming the dataset of forest cover type.

Standard Normalization= $x' = (x-\mu)/\sigma$ (1)

Where x is value of the element, μ is the Mean and σ represents the standard Deviation.

Min/Max Normalization= $x' = (x-min)/(max-min)$ (2)

Where min and max are the minimum and maximum values in x given its range.

Table 1. Dataset Description

1	Data_field	Description
2	Elevation	Elevation in meters
3	Aspect	Aspect in degrees azimuth
4	Slope	Slope in degrees
5	Horz_ hydro	Horz Dist to nearest surface water feature
6	Vert_hydro	Vert Dist to nearest surface water features
7	Horz_road	Horz Dist to nearest roadway
8	Hillshade_9am	Hillshade index at 9am, summer solstice (0 to 255 index)
9	Hillshade_noon	Hillshade index at noon, summer solstice (0 to 255 index)
10	hillshade _3pm	Hillshade index at 3pm, summer solstice (0 to 255 index)
11	horz _fire	Horz Dist to nearest wildre ignition points
12	Wild	Wilderness area designation (4 binary columns)
13	Soil_ type	Soil Type designation (40 binary columns)

3.2.3 *Data Reduction*

In this stage the data set is modified such that the results produced by the model are almost the same but unnecessary values in dataset are removed. The attribute soil type 7 and 15 are reduced as it contains zero values in it. The data is to be reduced to increase its performance after applying the models on the dataset.

3.2.4 Data Integration

In this stage data is merged from different sources if needed, again redundancies are removed too. Sometimes it is required to aggregate the training and testing data to apply models and again separate it.

3.3 Exploratory Data Analysis

Exploratory data analysis is a statistical way of understanding the data which is usually done in a visual way. The graphs plotted in exploratory data analysis are for better understanding of data to the analyst. The graphs and plots plotted for the data are essentially use for understanding the data and values consisting in the attribute. It helps in understating the co-relation between the variables and how their values differ with each other.

Exploratory data analysis (EDA) is the critical first step in analyzing the data from experiment. The main reasons we use EDA:

- Detection of mistake
- Checking of assumptions
- Preliminary selection of appropriate model
- determining relationships among the explanatory variables, and
- Assessing the direction and rough size of relationships between explanatory and outcome variables.

3.4 Models

3.4.1 Logistic Regression

The very first model that was applied was a logistic regression in one vs. others mode. For each class $y^{(K)}$ we train Regression on items that belong to class vs. all other items. Logistic regression assumes that probability of observation

X belonging to class y is given by

$$P(x/y) = \sigma(x) = 1/ 1+e^{-\Theta.x} \quad \ldots(3)$$

$L (data) = \log P (data/\Theta)$

$$= \sum_{i=1}^{N} (y(i) \log \sigma x^{(i)}) + (1- y (i)) \log (1-\sigma(x^{(i)})) \ldots(4)$$

Maximizing it with respect to Θ will give as classier for class

yk Then, Bayes optimal classier h(x) for observation x is:
Maximizing it with respect to Θ will give as classier for class yk Then, Bayes optimal classier h(x) for observation x is:

$H(x) = \arg\max P (x/y^{(k)}) \quad \ldots(5)$

Unfortunately, performance of Logistic regression on this train set is not satisfactorily. But more importantly, analyzing the learning curve suggests that even if we had more labeled data, it would not improve performance of this Model. One of the advantages of this model is it works very fast and actually for simple models, it classifies with good accuracy.

3.4.2 Decision Trees

A decision tree is a flow chart structure which consist of internal node which represents a test in attribute and that comes with each branch out i.e. the result of the test and each leaf represents a category label (a decision taken after testing all attributes in the path from the beginning to the leaf). Each path from the source to a leaf can also be interpreted as a sorting rule.

When establishing a supervised classification model, the frequency distribution of attribute values is a potentially

significant component in deciding the proportional importance of each attribute at various levels in the model construction procedure.

In data modeling, we can use frequency distributions to compute entropy. We calculate the entropy of multiplying the proportion of cases with each category label by the log of that proportion, and then getting the negative essence of those conditions.

Entropy (S) =−p1log2 (p1) − p2log2 (p2)
Where pi is proportion (relative frequency) of class *i* within the set *S*.

A decision tree is constructed algorithm that selects the best attribute, splits the data into subsets based on the values of that attribute present in the dataset and repeats the process on each of these subsets until a stopping condition is met.

Information gain measures the decrease in entropy that results from splitting a set of instances based on an attribute.
IG (S, a) = entropy (S) – [p (s1) x entropy (s1) + p (s2) x entropy (s2)… +p (sn) x entropy (sn)].
Where n is the number of distinct values of attribute a, and si is the subset of S where all instances have the ith value of a.

3.4.3 Gradient boosting machines

In gradient boosting machines, or simply, GBMs, the learning procedure consecutively fits new models to provide a more accurate estimate of the response variable. The principle idea behind this algorithm is to construct the new base-learners to be maximally correlated with the negative gradient of the loss function, associated with the whole ensemble. The loss functions applied can be arbitrary, but to give a better intuition, if the error function is the classic squared-error loss, the learning procedure would result in consecutive error-fitting. In general, the choice of the loss function is up to the researcher, with both a rich variety of loss functions derived so far and with the possibility of implementing one's own task-specific loss.

The algorithm is implemented in R studio under the Gbm package which fits generalized boosting regression models. The formula consists of the train data, the type of distribution, shrinkage factor no. of trees etc. The number of tress varies. The algorithm has been tried with 500 and 1000 trees and found to give good performance with 1000 trees. The models work best when the training data is huge and small test set. But for our dataset it is totally opposite of it though it gives good result but not better than random forest and decision tree.

3.4.4 Random Forest

Random forest is an ensemble classifier. An ensemble consists of a set of individually trained classifiers whose predictions are combines for classifying new instances.

Random Forest is a classifier consisting of a collection of tree-structured classifiers {h(x, Θk) k=1, 2, ….}, where the {Θk} are independent identically distributed random vectors and each tree casts a unit vote for the most popular class at input x.

For building a decision tree in random forest the steps have to be followed. If the number of records in the training set is N, then N records are sampled at random but with replacement, from the original data, this is bootstrap sample. This sample will be the training set for growing the tree. If there are M input variables, a number m << M is selected such that at each node, m variables are selected at random out of M and the best split on these m attributes is used to split the node. The value of m is held constant during forest growing. Each tree is grown to the largest extent possible. There is no pruning. In this way, multiple trees are induced in the forest; the number of trees is pre-decided by the parameter Ntree. The number of variables (m) selected at each node is also referred to as mtry or k in the literature. The depth of the tree can be controlled by a parameter node size (i.e. number of instances in the leaf node) which is usually set to one.

For random forest python libraries like pandas and numpy are used which consist of set random forest classifier function in that tree estimators have to add to get good results.

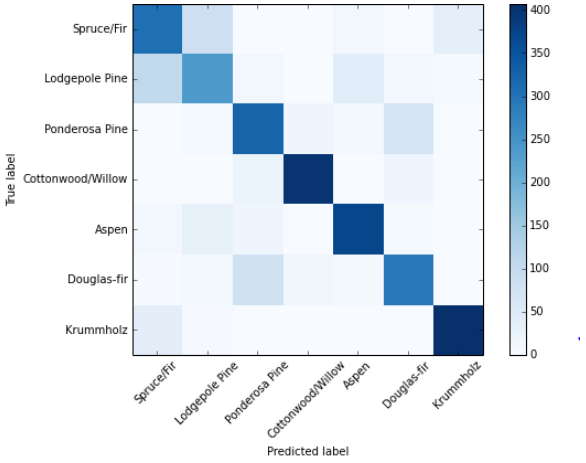

Fig. 2 Confusion Matrix for Random Forest

Confusion matrix is used to check the performance of a classification model. The diagonal elements are the number of points for which the predicted label is equal to the true label, while off-diagonal elements are those that are mislabeled by the classifier. The higher the diagonal values of the confusion matrix the better, indicating many correct predictions. Fig 2. shows the visualization of confusion matrix for the random forest model. As we can infer from Fig 2. Random forest is giving the best predictions**.**

4. RESULTS

The four algorithms are applied on the forest cover dataset for predicting the cover type. We got some good results and accuracy with decision tree, random forest and gradient boosting machines.

The accuracy and performance have been compared between the models using Root Mean Squared Logarithmic Error (RMSLE). R is used for implementing the models and RMLSE for computing the performance.

It is given by:

$$\sqrt{\frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{log}(\boldsymbol{p_i}+\mathbf{1})-\mathbf{log}(\boldsymbol{a_i}+\mathbf{1}))^2} \qquad \ldots(6)$$

Where:

- n is the number of hours in the test set
- pi is your predicted count
- ai is the actual count
- log(x) is the natural logarithm

And we use Cross-Validation to estimate $\mathbf{R^2}$ and reduce over fitting. These are mainly used to compare accuracy for two or more models and select the most accurate model.

Table 2. Results

Algorithms Used	Accuracy %
1) Logistic Regression	19.4%
2) Decision Trees	35.6%
3) GBM	60%
4) Random Forest	74.8%

From the analysis it is observed that **Random forest** gives the most accurate predictions from all four models.

5. CONCLUSION

The objective of this paper is to predict the forest cover type based on cartographic variables. To predict forest cover type four different techniques are applied Regression, Random Forest, Decision tree and GBM and their accuracy and performance has been compared. It is obtained that Random Forest gives better prediction with 74.8% accuracy.

6. ACKNOWLEDGMENTS

We greatly acknowledge Machine Learning Laboratory of University of California Irvine for making the data of the US Geological Survey(USGS) and the US Forest Service(USFS) openly available.

7. REFERENCES

[1] D.J. Newman A. Asuncion. UCI machine learning repository, 2007.

[2] Jiawei Han and Micheline Kamber, "Data Mining Concepts and Techniques",second edition Morgan Kaufmann publisher.

[3] Ya Su, Xinbo Gao, Xuelong Li,and Dacheng Tao. "Multivariate Multilinear Regression",IEEE transactions on systems, man and cybernetics-Part B: cybernetics, vol 42.No.42 .

[4] Decision Trees for Business Intelligence and Data Mining: Using SAS Enterprise Miner.

[5] Simon Bernard, Laurent Heutte, Sebastian Adam. "On the selection of decision trees in Random Forests". International Joint Conference on Neural Networks IEEE, Jun 2009, France.

[6] Myungsook Klassen," Learning microarray cancer datasets by random forests and support vector machines",IEEE, 2010.

[7] Mohammed S. Alam and Son T. Vuong, "Random Forest Classification for Detecting Android Malware", 2013 IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing.

[8] Vrushali Y Kulkarni and Dr Pradeep K Sinha, " Random Forest Classifiers :A Survey and Future Research Directions", International Journal of Advanced Computing, ISSN:2051-0845, Vol.36, Issue.1.

[9] Yasser Ganjisaffar, Rich Caruana, Cristina Videira Lopes, "Bagging Gradient-Boosted Trees for High Precision, Low Variance Ranking Models", ACM SIGIR'11, July 24–28, 2011, Beijing, China.

[10] Chun-Xia Zhang, Jiang-She Zhang, Gai-Ying Zhang, " An efficient modified boosting method for solving classification problems", Science Direct, An efficient modified boosting method for solving classification problems.

[11] Chun-Xia Zhang, Jiang-She Zhang, "A local boosting algorithm for solving classification problems", Science Direct, Computational Statistics & Data Analysis 52 (2008) 1928 – 1941.

[12] Uyen Nguyen Thi Van, and Tae Choong Chung, "An Efficient Decision Tree Construction for Large Datasets", IEEE journal 2008.

[13] Christopher M. Bishop. Pattern Recognition and Machine Learning (Information Science and Statistics).Springer-Verlag New York, Inc., Secaucus, NJ, USA,2006.1

[14] Thangaparvathi.B and Anandhavalli.D, "An Improved Algorithm of Decision Tree for Classifying Large Data Set Based on RainForest Framework",IEEE journal 2010.