Decision Tree & Ensemble Methods

with

Indian Liver Patients Records Data-Set

Overview :

In this we are going to implement decision tree and Ensemble methods. To predict the test data set and print out both the training accuracy results and testing accuracy results. It breaks down a Data-set into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed. The final result is a tree with decision nodes and leaf nodes, training and testing accuracy score. Solving with both a logistic regression problem and classification problem using Decision Tree, Ensemble, Voting, Bagging and Random Forest.

1. What are we going to do ?

- We clean the data and select features and labels to apply the following:
- To build Decision Tree and three ensemble classifiers: Voting Classifier, Bagging Classifier, and Random Forest Classifier using training and testing data.
- We are going to build a decision tree method for a Liver Patients to predict the gender ratio for different samples using different classifications methods for both regression and classification problem with our training data.
- Implementation of whole data set split into training part (80%) and testing part(20%).
- ID3 algorithms for continuous with support for inconsistent data-set.
- Graphviz component to visualize the learned tree.
- Matplotlib component to visualize the Horizontal bar plot.
- Support for multiple, and symbolic outputs and graphing of continuous trees.
- We will get to learn how training and testing tuning helps in model performance.

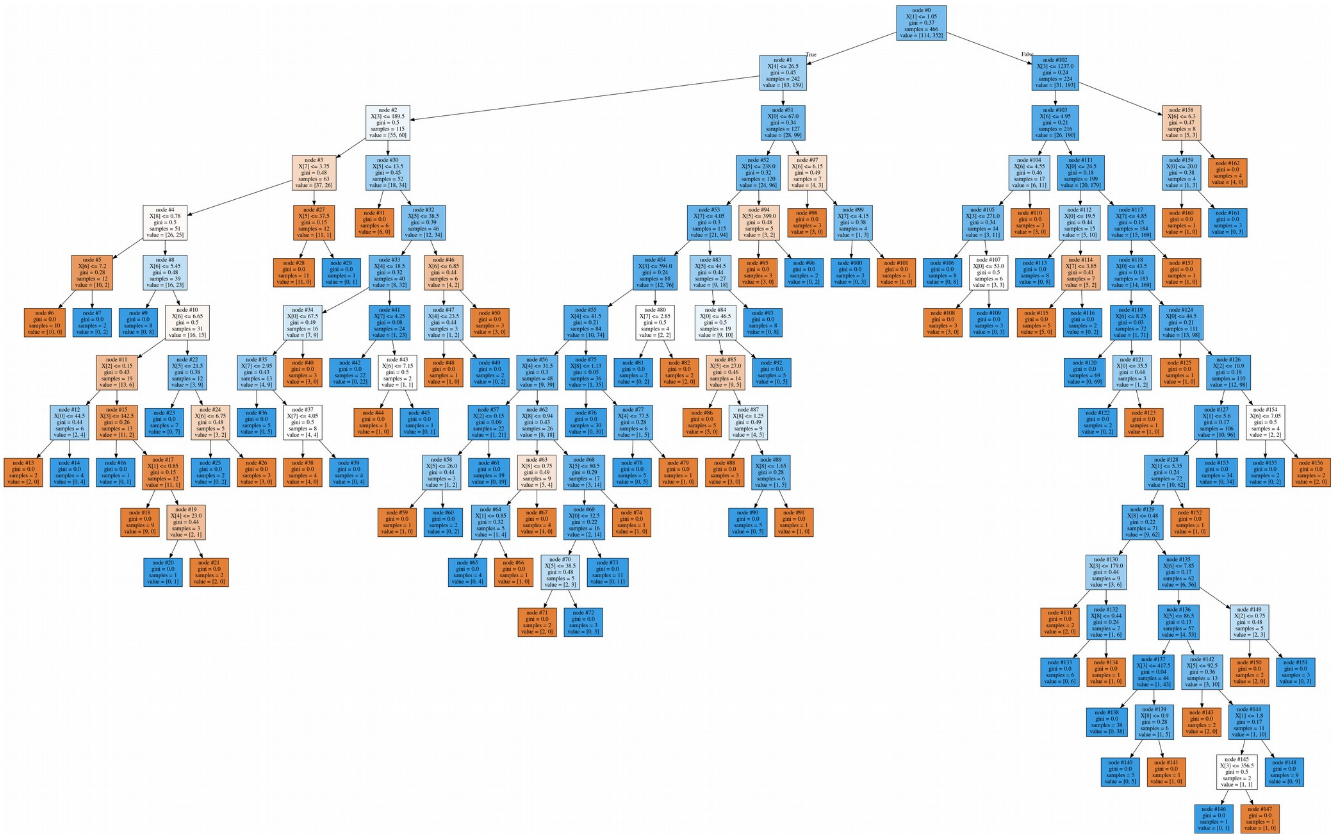
2. Describe your data set :

- Observations : **583**
- Features : **10** (Including “Gender” **11** features)
- Females : **142**
- Males : **441**

3. Decision Tree: how decision tree works?

- Place the best attribute of the data-set at the root of the tree. Calculate entropy of the target.
- The training data-set is then split on the different attributes. The gini or entropy for each branch is calculated. Then it is added proportionally, to get total entropy for the split. The resulting gini or entropy is subtracted from the entropy before the split. The result is the information gain or decrease in entropy.
- Choose attribute with the largest information gain as the decision node, divide the data-set by its branches and repeat the same process on every branch until we find leaf nodes in all the branches of the tree.
- For predicting a class label for a record, we start from the root of the tree. We compare the values of the root attribute with record’s attribute. On the basis of comparison, we follow the branch corresponding to that value and jump to the next node.
- We continue comparing our record’s attribute values with other internal nodes of the tree until we reach a leaf node with predicted class value. The modeled decision tree can be used to predict until all data is classified.
- So, from given assignment the blue boxes represent criteria for true is number of males is greater than females.
- The orange boxes represent criteria for false is number of females is greater than males.
- When the condition sample #0 is True its split into split into another sample #1 which also results in True. Splitting the sample this way the condition is False in #5 , #6 , #27 etc.

4. Please display your PNG D-Tree figure generated from your training data.



5. Voting Classifier: How voting classifier works ?

- We can prepare our model utilizing different algorithms and after gathering them to predict the final output. Say, we utilize a Random Forest Classifier, SVM Classifier, Linear Regression and so forth.; models are hollowed against one another and chose upon best execution by casting a voting utilizing the Voting Classifier Class from sklearn.ensemble.

- Hard Voting is the place a model is chosen from an ensemble to make the last prediction by a straightforward vote in favor of accuracy.

- Soft Voting must be done only when every one of your classifiers can ascertain probabilities for the results. Soft Voting land at the best outcome by averaging out the probabilities computed by individual algorithms.

- The accuracy of the Voting Classifier is by and large higher than the individual classifiers. Make a point to include various classifiers with the goal that models which fall prey to comparative kinds of errors don't total the mistakes.

6. Bagging Classifier: How bagging classifier works ?

- Bagging uses a straightforward methodology that appears in statistical analyses again and again improve the estimate of one by consolidating the estimates of many. Bagging builds n classification trees utilizing bootstrap sampling of the training data and then combines their expectations to create a last meta-prediction.

- Let’s say we have 600 observations and 100 components. A bagging approach will make a several models with a subset of observations and a subset of variables. i.e we may make 300 trees with 300 random observations and 20 random factors in each tree. we will then average the consequences of all the 300 tree's (models) to get final prediction.

- Bagging can be utilized with any technique , yet usually is utilized with trees. Random forest is a bagging approach.

- We are reducing the variance and bringing stability into your predictions by building several models and then averaging them.

7. Random Forest Classifier: How random forest classifier works

- Ensemble of Decision trees is a Random Forest. Random Forests performs Bagging inside. Random Forest creates several trees, sometimes thousands, and calculates the best possible model for a given data-set. Rather considering all features while splitting a node, Random Forest calculation chooses the best component out of a subset all features considered. This exchanges a higher inclination for lower fluctuation, which yields a greatly improved model.

- Randomly select “a” features from total “b” features.

- Where a << b

- Among the “a” feature, calculate the node “c” using the best split point.

- Split the node into daughter nodes using the best split.

- Repeat 1 to 3 steps until “l” number of nodes has been reached.

- Build forest by repeating steps 1 to 4 for “n” number times to create “n” number of trees.

8. Comparison Table for the training and testing accuracy results for all four classifiers

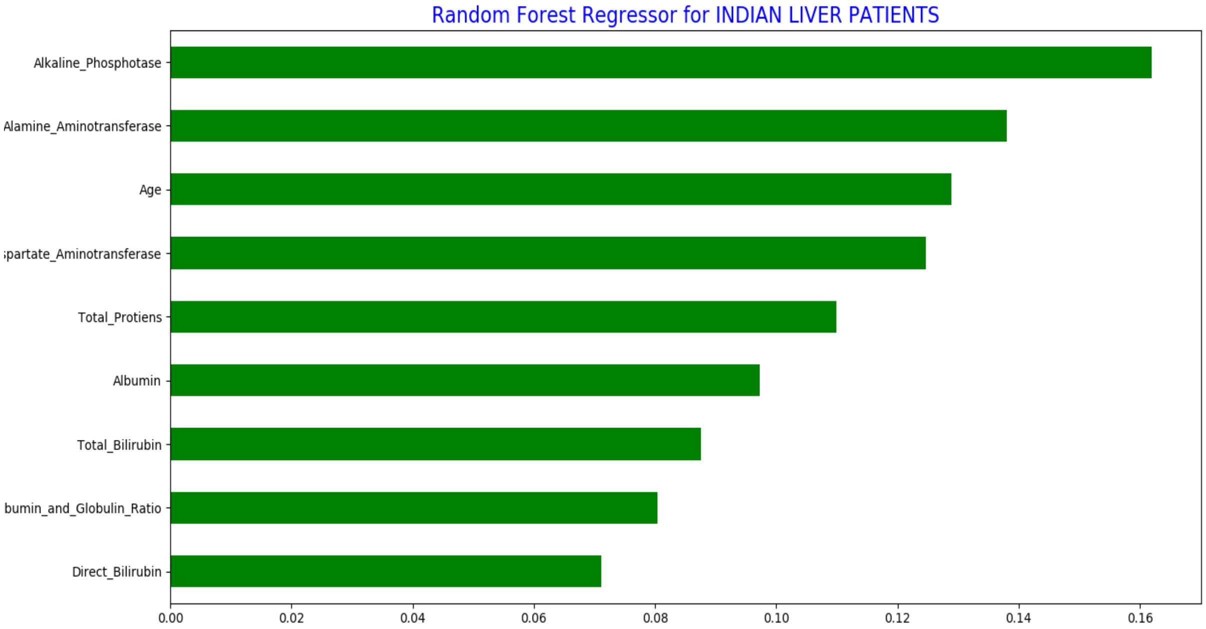
Decision Tree result :-
Training Accuracy: 1.0
Testing Accuracy: 0.7777777777777778

Voting Classifier result :-
Training Accuracy: 0.841
Testing Accuracy: 0.752

Bagging Classifier result :-
Training Accuracy: 1.000
Testing Accuracy: 0.803

Random Forest result :-
Training Accuracy: 1.000
Testing Accuracy: 0.812

Random Forest Horizontal Bar-Plot :-


Random Forest Regressor for INDIAN LIVER PATIENTS
Alkaline_Phosphotase
Alamine_Aminotransferase
Age
spartate_Aminotransferase
Total_Protiens
Albumin
Total_Bilirubin
bumin_and_Globulin_Ratio
Direct_Bilirubin