Decision Tree

To run the Decision Tree code, use the following command,

- Python decision_tree.py [options]

The valid options can be seen using -h or –help.

The options are,

- --algorithm: Used algorithm in classification. Valid options are “decision_tree” and “random_forest”. Default is “decision_tree”.
- --generate_tree: If present visualizes the tree trained in decision tree or the first the tree trained in random forest.
- --criterion: Used criterion in the decision tree classifier. Valid options are “gini”, “entropy” and “log_loss”. Default is “gini”.
- --seed: Seed used in all random related parts. Default is None.
- --test_size: The proportion of samples which will be used in testing. Default is 0.2. (%20 test, %80 train)
- --cv_enabled: If present cross validation is done for best parameter selection. Note that cross validation is being done prior to other parts which means it can affect visualized tree, most significant features.
- --obtain_most_significant: If present it obtains the most significant features and runs logistic regression on [5, 10, 15, 20] sized subsets.
- --plot: If present it plots the performance of the random forest with varying number of estimators (tree count).
- --repeat: Repeat count of the experiment. Default is 1.

Accuracies

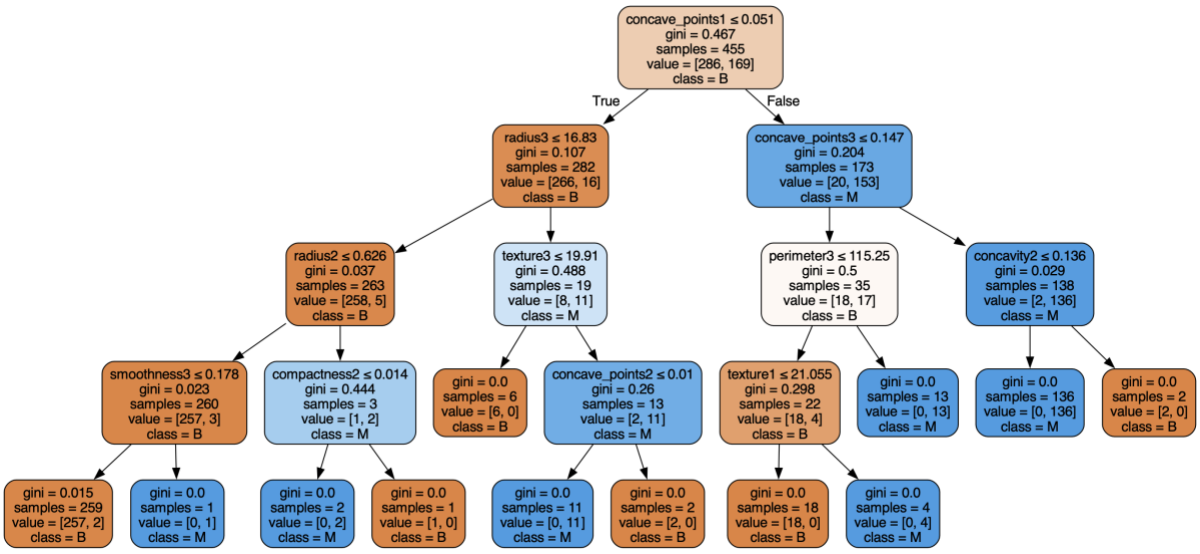
For the accuracy calculations, we have used the following options,

- Test size: 0.2
- Seed: 42
- Algorithm: “decision_tree”
- Cross validation is enabled and 5-Fold is used.
- Generate tree option is used for generating tree representation.
- Repeat count is 1000.

With these options max depth is decided as 4 with cross validation and we have observed following accuracies,

- Train accuracy: 99.56 %
- Test accuracy: 94.13 %

Tree visualization of latest tree generated in experiment:



While we can get higher accuracies with certain seeds and splits, we chose to stick with one seed.

Naïve Bayes Comparison

In the first assignment our naïve bayes classifier demonstrated the following average accuracies over 100 runs,

- Train accuracy: 93.84 %
- Test accuracy: 93.48 %

Since decision tree can capture the relationships between features, they generally achieve better results than the Naïve Bayes classifier. The most important problem with decision trees is that it they can be overfit to data. However, we have used cross validation with depth parameters, we prevented the overfitting to some degree.

Most Significant Features

We have used to the following parameters to train the decision tree which we will obtain the most important features from,

- Algorithm: “decision_tree”
- Cross validation enabled (Sets the max depth to 4)
- Seed: 42
- Obtain most significant tag is present.

Performance of the decision tree:

- Train Accuracy: 99.56 %
- Test Accuracy: 94.73 %

Logistic regression with 1000 iteration is used for all comparisons.

For 5 features:

- Train Accuracy: 94.28 %
- Test Accuracy: 95.61 %

For 10 features:

- Train Accuracy: 93.84 %
- Test Accuracy: 96.49 %

For 15 features:

- Train Accuracy: 93.62 %
- Test Accuracy: 96.49 %

For 20 features:

- Train Accuracy: 95.16 %
- Test Accuracy: 96.49 %

These results suggest that utilizing smaller set of predictive features can lead to improved test accuracy. Moreover, this experiment shows robustness of the decision tree to select most important features.

Random Forest

We have used to the following parameters to train the random forest,

- Algorithm: “random_forest”
- Cross validation enabled (Sets the max depth to 5, Estimators to 23)
- Seed: 42
- Repeat: 1000
- Plot tag is present.

Performance of the random forest:

- Train Accuracy: 99.26 %
- Test Accuracy: 96.27 %

While the random forest shows slight decrease in training it shows significant improvement in testing accuracy. This supports that the random forest, with ensemble approach and number of estimators is better at generalizing the model comparing to decision tree. Using multiple trees with different feature set allows algorithm to reduce overfitting.

Change in performance with varying number of trees(estimators) in the forest: