METHODOLOY

Entropy Calculation and Feature Selection

The entropy for each characteristic or feature at the root node was determined in the first phase. The metric used to evaluate impurity was Information Gain. The goal was to determine which feature, when employed as the root node of the decision tree, maximizes Information Gain.

Decision Tree Construction and Depth

A Decision Tree was constructed using DecisionTreeClassifier. The model was trained with the data provided. The accuracy of this model was evaluated on the training set, and the depth of the decision tree was determined using the model's get\_depth() function.

Visualization of Decision Tree

The decision tree graph was plotted using the plot\_tree() function in conjunction with Matplotlib and scikit-learn's tree module. The resulting visualization helped in comprehending the structure of the decision tree.

Decision Tree Construction and Accuracy

A Decision Tree classifier was created and then fitted with the training data. The model's accuracy on the training set was determined, and subsequently, the accuracy on the test set was also evaluated. Also, a graphical representation of the decision tree was plotted.

Decision Tree with Max Depth Constraint

In this, we implemented a maximum depth constraint when constructing the Decision Tree classifier. This constraint helped to limit the tree's depth also preventing it from becoming overly complex. After constructing the model with this constraint, we proceed to evaluate its performance by measuring its accuracies on both the training as well as test datasets. We also visualized the resulting Decision tree to see how the imposed constraint influenced the tree's structure and complexity.

Criterion Comparison

In this, the criterion for the Decision Tree was studied. Initially, the model was built using the default criterion. Later, the criterion was changed to "Entropy". The differences in the models and graphical representation between the default and entropy criterion were investigated.

Random Forest Classifier

In this, we constructed the Random Forest classifier on the data set. A comparison was made between the performance metrics of the Decision Tree and Random Forest classifiers. This evaluation helped to highlight the differences in accuracy, precision, recall, and other relevant metrics.

Random Forest Parameters and Attributes

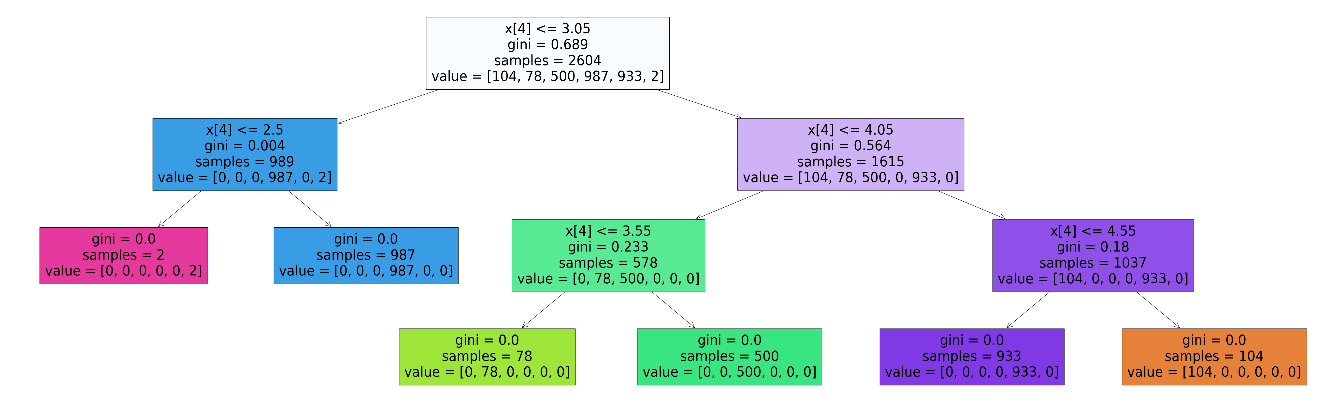
Finally, a thorough study of the various parameters and model attributes of the Random Forest classifier was conducted. The importance of these parameters and attributes in shaping the model's behavior was explored. Key parameters, such as the number of trees, maximum depth, and minimum samples per leaf, were noted, along with attributes like feature importance and out-of-bag score. The analysis provided valuable insights into how different settings impact the Random Forest model.

**Result**:

1. Results obtained from A4 question
2. Training set accuracy: 1.0

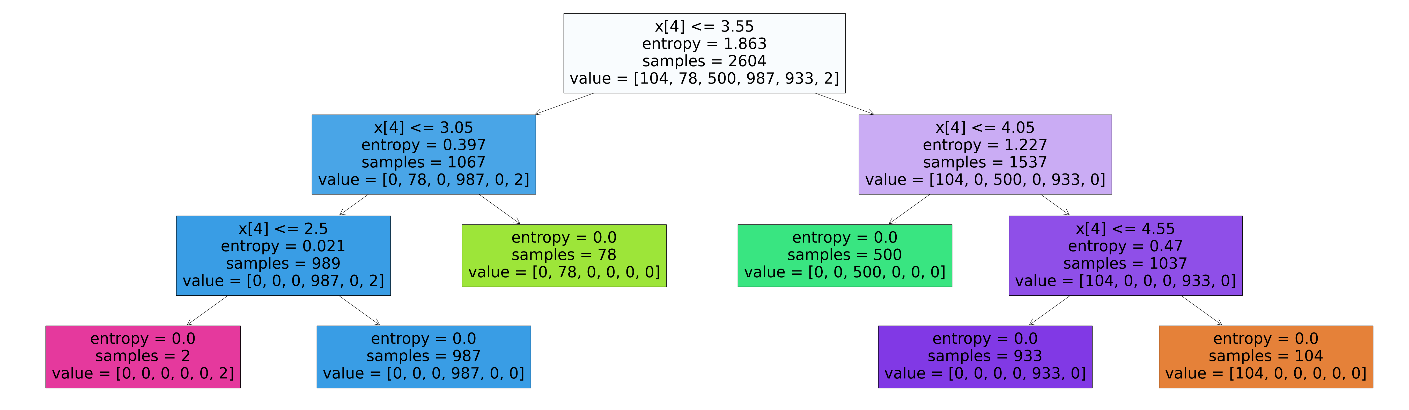
Tree depth : 3

1. Decision Tree diagram



1. Results obtained from A6
2. Training set accuracy: 1.0

Tree depth : 3

1. Decision Tree diagram with criterion of Entropy
2. Results obtained from A7
3. Accuracy of Random Forest Classifier: 0.999012508229098
4. Accuracy of Decision Tree Classifier : 0.999835418038183
5. Classification report matrices of Random Forest Classification

precision recall f1-score support

0 1.00 1.00 1.00 225

1 1.00 1.00 1.00 203

2 1.00 1.00 1.00 1089

3 1.00 1.00 1.00 2286

4 1.00 1.00 1.00 2267

5 1.00 0.00 0.00 6

accuracy 1.00 6076

macro avg 1.00 0.83 0.83 6076

weighted avg 1.00 1.00 1.00 6076

1. Confusion matrix of Random Forest Classifier

[[ 225 0 0 0 0 0]

[ 0 203 0 0 0 0]

[ 0 0 1089 0 0 0]

[ 0 0 0 2286 0 0]

[ 0 0 0 0 2267 0]

[ 0 0 0 6 0 0]]

1. Classification report of matrices of Decision Tree classifier

precision recall f1-score support

0 1.00 1.00 1.00 220

1 1.00 1.00 1.00 199

2 1.00 1.00 1.00 1083

3 1.00 1.00 1.00 2346

4 1.00 1.00 1.00 2224

5 0.80 1.00 0.89 4

...

accuracy 1.00 6076

macro avg 0.97 1.00 0.98 6076

weighted avg 1.00 1.00 1.00 6076

1. Confusion matrix Decision Tree classifier

[[220 0 0 0 0 0]

[ 0 199 0 0 0 0]

[ 0 0 1083 0 0 0]

[ 0 0 0 2345 0 1]

[ 0 0 0 0 2224 0]

[ 0 0 0 0 0 4]]

Model performance: The performance of model on general basis is same in both methods (Random Forest Classifier and Decision Tree Classifier). The model got the same accuracy in both methods.

*Precision*: Based on precision Random Forest classification model perform better than Decision Tree model. In the precision of RFC, the accuracy to measure 6th label is 1.0 whether in Decision Tree classification it is 0.8. That meant there is 20% that Decision Tree classification will fail to make true positive decision.

*Recall*: Decision Tree Classification outperformed Random Forest Classification in terms of recall. For label 6th Decision tree has got 1.0 value, which means the model is able identify the true positive cases. However, in case of Random Forest classification recall value for 6th label is 0.0, which means that model is not able to identify true positive instances. In other words, the system failed to identify the instances for 6th label.

*F1-Score*: Decision Tree classification performs better compare to Random Forest Classification. Since F1-Score is combination of both precision and recall, it gives an overall idea about the model’s ability to identify and capture all the true positive instances for particular label. Random Forest Classifier has f1-score as 0.00 and Decision Tree Classification has the f1-score as 0.89.

*Confusion Matrix*: We use confusion matrix to check how well the model performs to on identifying the correct labels out of given samples. On observing confusion matrix of both classifiers, Random Forest classifier has six false positive values on the other hand Decision Tree classifier has one false negative value. Confusion matrix of Random Forest Classifier completely fails to identify the instances for 6th class, which is not good for model. Therefore, compare to Random Forest classifier Decision Tree Classifier performs better as from confusion matrix we can see it is successfully identifying all the classes with slight error.

*Overfit and Underfit* : Random Forest Classifier performs overfit because it has accuracy 0.999 yet it is unable to predict the last label. Decision Tree Classifier is able to predict the instances corresponding to all the classes. Thus, Decision Tree Classifier has underfit condition.

**Discuss on the usage of pruning to avoid overfitting in decision trees. Relate that to your data and performance from experiment in A5.**

Pruning is a technique that is used when there a high level of complexity found in model. It can be used with any classifier techniques such as Decision Tree classification, Random Forest Classification, etc. It avoids overfitting by cancelling all the branches that are not significant in tree model. As per our dataset, pruning is required to be applied on Decision Tree classifier. In problem statement A5, the tree classifier used criterion as ‘max\_depth=5’ and produces a tree of height 3. In that tree every branch is leading to a valid class label, which means all of them are significant for the model. So, in this case we are avoid using pruning in the classifier. However, if we use then the performance of the model would increase for sure.