Report PA 2: Classification - Decision Tree

Student Details

Student Name and ID: **Bhogal, Gurvir Singh Tarlok Singh, UTA ID: 1001769871**
Student Name and ID: **Rohith Rajagopalan Ramesh Babu, UTA ID: 1001518031**

1) **Describe the Decision Tree methods, and Naive Bayes classifier.**

DECISION TREE:

Decision Tree Mining is a type of data mining technique that is used to build Classification Models. It belongs to a class of supervised learning. The classification models which is obtained by this technique is in form of a tree. First, we build a classification model based on a training set data (learning phase). Then the model is tested for accuracy against a test set data (classification phase). The accuracy of the classifier is obtained by the percentage of the tests that are predicted correctly.

A decision tree is a classification method which gives a tree and a set of rules representing the model where,

- Every internal node denotes a test on an attribute.
- Every branch denotes outcome of the test.
- The topmost node in a tree is the root node.
- Every leaf node holds a class label.

The root node is selected based on a value called Gain value, the formula to calculate gain value is

Gain value =P-M where,

- P is impurity measure before splitting the data.
- M is impurity measure after splitting the data.
- Gini Index, Entropy and misclassification Error are few methods used to calculate the impurity measures

$$\text{Gini Index} = 1 - \sum_{i=1}^{n} (P_i)^2$$

here Pi denotes the probability of an element being classified for a distinct class.

- $$\text{Entropy} = \sum_{i=1}^{n} -p(c_i)\log_2(p(c_i))$$

NAÏVE BAYES CLASSIFIER:

Naive Bayes classifier is a machine learning model that primarily uses probability and is based on the Bayes theorem

BAYES THEOREM:

Let B be a data tuple.Let A be a hypothesis such that the data tuple B belongs to a specified class C .
The goal is to find the value of P(A|B) which is the probability that the hypothesis X holds, this means that the tuple B belongs to class C, given the attribute description of B.

For Example P(A/B) is the probability that customer A will buy a laptop given that we know the income and age of the customer

$$P(A|B)=\frac{P(B|A)P(A)}{P(B)}$$

2) Describe the datasets and code.

The train_cl dataset consists of 29 columns. The first column is an unnamed column, which has been renamed to ID. It has various details such as fare rate, class, and also data as to whether a passenger survived the journey or not.

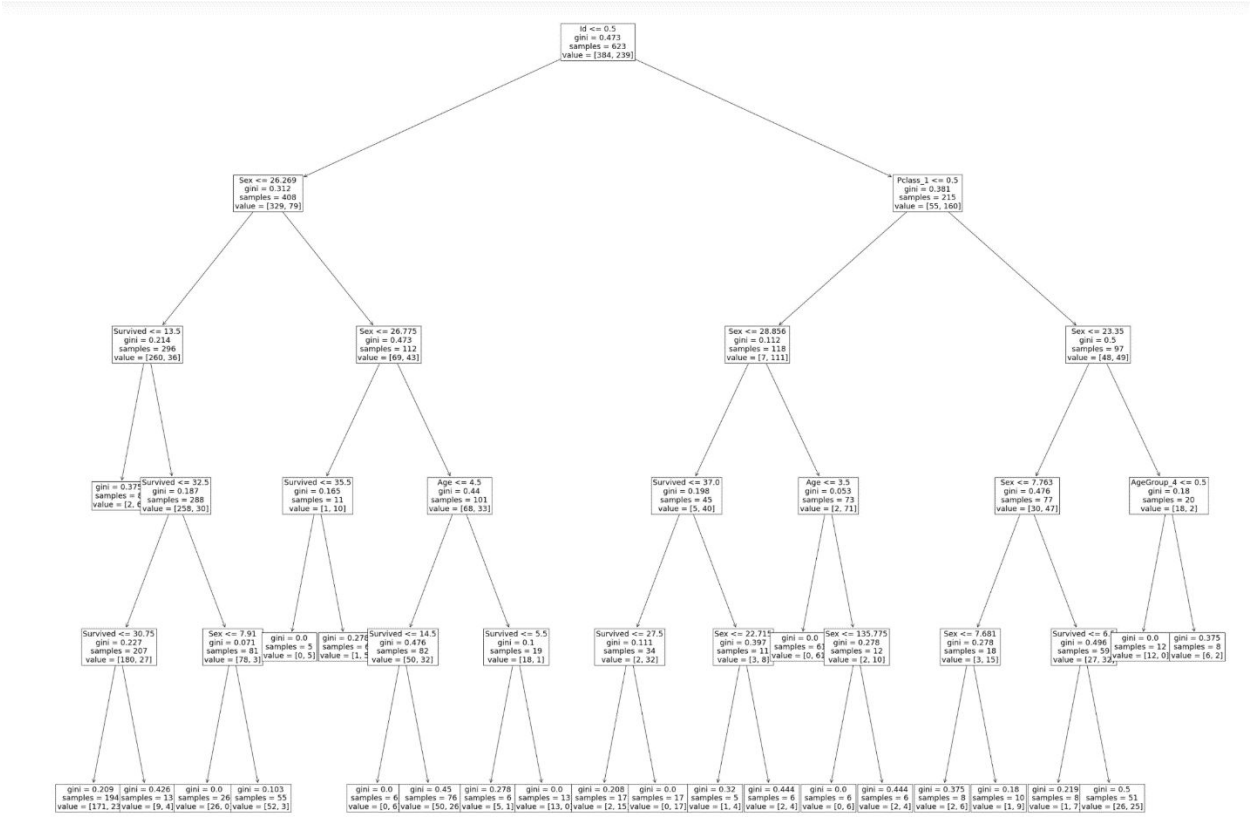
Preprocessing- we have renamed the unnamed column to ID.

Code Explanation-The Target variable selected here is survived. The code predicts whether a given passenger has survived the journey or not. We have used two methods using the decision tree classifier. The first one is gini and the second is entropy. We split the dataset into 70% for training and 30% for testing. We predict the result and then measure the accuracy for the code. We also print out the confusion matrix and the classification report.

We use the Naive Bayes(Gaussian) classifier to predict the data. We split the dataset into 70% for training and 30% for testing. We predict the result and then measure the accuracy for the code. We also print out the confusion matrix and the classification report.

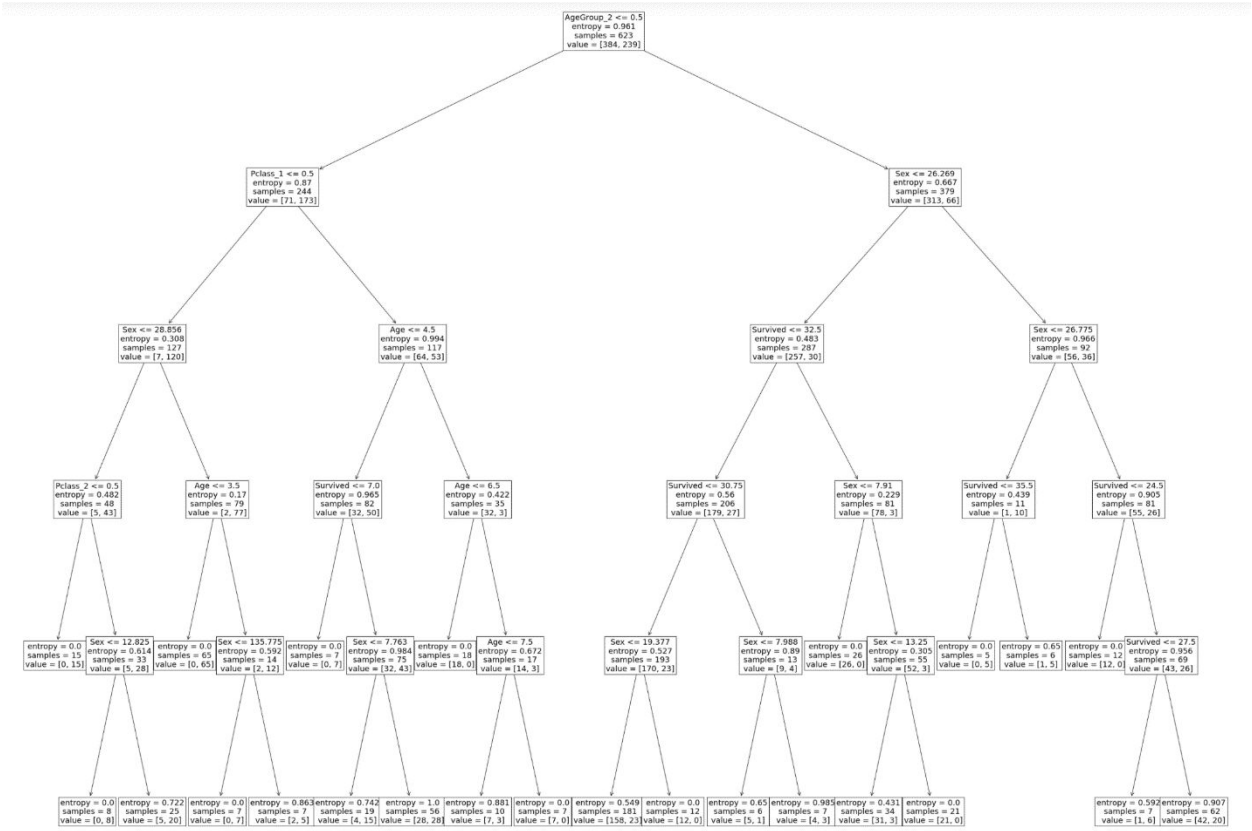
3) Visualization of the decision tree for Gini and Entropy.

GINI-



The Visualization using Gini is as shown. The max depth of the tree is 6

Entropy-



The visualization of the decision using entropy tree is shown. The depth of the tree is 6, similar to GINI

4) Interpret your results, and do not forget to compare Gini and entropy.

Gini-

```
Accuracy: 0.8171641791044776
[[151  14]
 [ 35  68]]
              precision    recall  f1-score   support

           0       0.81      0.92      0.86       165
           1       0.83      0.66      0.74       103

    accuracy                           0.82       268
   macro avg       0.82      0.79      0.80       268
weighted avg       0.82      0.82      0.81       268
```

Confusion Matrix

	Predicted: NO	Predicted: Yes
Actual: NO	True Negative: 151	False Positive: 14
Actual: Yes	False Negative: 35	True Positive: 68

Columns represent predicted values and rows represent actual values. Here in the 1st row, there are 165 values, among which 151 are correctly classified Negative and 14 are wrongly classified as Positive.
In the 2nd row there are 103 values among which 35 are correctly classified as Positive and 68 are wrongly classified as Negative

Entropy-

```
Accuracy: 0.8171641791044776
[[150  15]
 [ 34  69]]
              precision    recall  f1-score   support

           0       0.82      0.91      0.86       165
           1       0.82      0.67      0.74       103

    accuracy                           0.82       268
   macro avg       0.82      0.79      0.80       268
weighted avg       0.82      0.82      0.81       268
```

	Predicted: NO	Predicted: Yes

Actual: NO	True Negative: 150	False Positive: 15
Actual: Yes	False Negative: 34	True Positive: 69

Columns represent predicted values and rows represent actual values. Here in the 1st row, there are 165 values, among which 150 are correctly classified Negative and 15 are wrongly classified as Positive.
In the 2nd row there are 103 values among which 34 are correctly classified as Positive and 69 are wrongly classified as Negative

Comparing both the results we can see that the accuracy for both Entropy and Gini are the same 0.817

Naive Bayes Classifier:

```
Accuracy: 0.4253731343283582
[[ 10 153]
 [  1 104]]
              precision    recall  f1-score   support

           0       0.91      0.06      0.11       163
           1       0.40      0.99      0.57       105

    accuracy                           0.43       268
   macro avg       0.66      0.53      0.34       268
weighted avg       0.71      0.43      0.30       268
```

Confusion Matrix:

	Predicted: NO	Predicted: Yes
Actual: NO	True Negative: 10	False Positive: 153
Actual: Yes	False Negative: 1	True Positive: 104

Columns represent predicted values and rows represent actual values. Here in the 1st row, there are 163 values, among which 10 are correctly classified Negative and 153 are wrongly classified as Positive.
In the 2nd row there are 105 Yes values among which 1 are correctly classified as Positive and 104 are wrongly classified as Negative.

References:

- https://www.datacamp.com/community/tutorials/decision-tree-classification-python
- https://scikit-learn.org/stable/modules/tree.html#tree
- https://scikit-learn.org/stable/modules/generated/sklearn.tree.export_graphviz.html#sklearn.tree.export_graphviz