CAP- 5610 Machine Learning

Homework 2

Name: Nabila Shahnaz Khan

NID: 5067496

Using Titanic dataset, guessing whether the individuals from the test dataset had survived or not

Task 1:

1) The following preprocessing was done on the Titanic dataset:

- **a)** Checked which columns had null values in both training and testing data. In training data, columns "Age" and "Embarked" had null values. And in testing data, columns "Age" and "Fare" had null values.
- **b)** Handled missing values by using different methods. Used KNN for feature "Age", mode for "Embarked" and median for "Fare".
- **c)** For feature "Sex" and "Embarked", converted strings to numerical values.
  Sex = {'female' : 1, 'male' : 0}
  Embarked = {'S' : 0, 'C' : 1, 'Q' : 2}
- **d)** Banded the features "Age" and "Fare" using 7-way split and 4-way split respectively.
- **e)** Created a new feature "Family" where "Family" = "SibSp" + "Parch"
- **f)** Created another new feature "Alone" where it holds value 1 if the passenger had no other family members in the ship ("Family" = 0), otherwise it holds the value 0.
- **g)** From Cabin, created a new feature called "Deck" based on the first Alpha character of the "Cabin" feature's value and filled up the missing values in "Deck" with 0.
  deck = {"A" : 1, "B" : 2, "C" : 3, "D" : 4, "E" : 5, "F" : 6, "G" : 7, "X" : 8}

2) The selected features are: **Pclass, Sex, Age, Fare, Embarked, Deck, Family** and **Alone**. Feature "Name" is excluded as it has unique values for each of the data points, so it will be complex to categorize it. Also, it has no significant correlation with the survival rate. Similarly, "Ticket" is also discarded as it has too many unique values and no significant correlation with "Survival". Finally, new features were generated from "Cabin", "SibSp", "Parch" and after that, they were dropped from the table.

3) A decision tree model with the Titanic training data using Gini index has been learned and plotted. A figure named "**Decision_tree_plot**" has been added to the GitHub directory mentioned below. The accuracy of the decision tree is **91.36%.**

4) Applied the five-fold cross validation of the decision tree learning algorithm to the Titanic training data to extract average classification accuracy. After applying 5-fold cross validation, the mean accuracy of the decision tree is **80.92%.**

5) Applied the five-fold cross validation of the random forest learning algorithm to the Titanic training data to extract average classification accuracy . After applying 5-fold cross validation, the mean accuracy of the random forest is **81.93%.**

6) In this case, the Random forest algorithm performed slightly better compared to the Decision tree algorithm.

7) The performance and accuracy of these machine learning algorithms highly depends on data preprocessing and feature selection. Also, though Random forest performed better than Decision tree algorithm in this particular case, but the difference isn't that significantly high. On the other hand, decision tree is lighter, less complex and requires less runtime. So, considering all these factors along with accuracy, the decision tree algorithm might be a good choice as well.

Task 2:

a) Training Error Rate = $\frac{\#\text{ misclassified data points}}{Total\ data\ points} * 100$

$= \frac{5+6+2+6+5+5}{19+13+12+14+22+20} * 100$

$= \frac{29}{100} * 100$

= 0.29 * 100

= 29%

b) Given a test instance T={A=0, B=1, C=1, D=1, E=0}, the given decision tree will assign this instance to class "-". Here, A = 0 -> B = 1 -> E = 0 takes the instance to a leaf node where there are 2 members of class "+" and 10 members of class "-". As the majority number of members belong to class "-", the instance will be assigned to this class.

Task 3:

#Q1: The overall entropy before splitting is given below:

Entropy = $-\frac{4}{10} * \log\frac{4}{10} - \frac{6}{10} * \log\frac{6}{10}$

= 0.971

#Q2: The gain in entropy after splitting on A is given below:

Entropy(A = 'T') = $-\frac{4}{7} * \log\frac{4}{7} - \frac{3}{7} * \log\frac{3}{7}$ = 0.985

Entropy(A = 'F') = $-\frac{0}{3} * \log\frac{0}{3} - \frac{3}{3} * \log\frac{3}{3}$ = 0

Entropy after split based on A = $\frac{7}{10} * 0.985 + \frac{3}{10} * 0 = 0.69$

Gain(A) = Entropy - Entropy after split based on A

= 0.971 – 0.69

= 0.281

#Q3: The gain in entropy after splitting on B is given below:

Entropy(B = 'T') = $-\frac{3}{4} * \log\frac{3}{4} - \frac{1}{4} * \log\frac{1}{4}$ = 0.8113

Entropy(B = 'F') = $-\frac{1}{6} * \log\frac{1}{6} - \frac{5}{6} * \log\frac{5}{6}$ = 0.65

Entropy after split based on B = $\frac{4}{10} * 0.8113 + \frac{6}{10} * 0.65 = 0.7145$

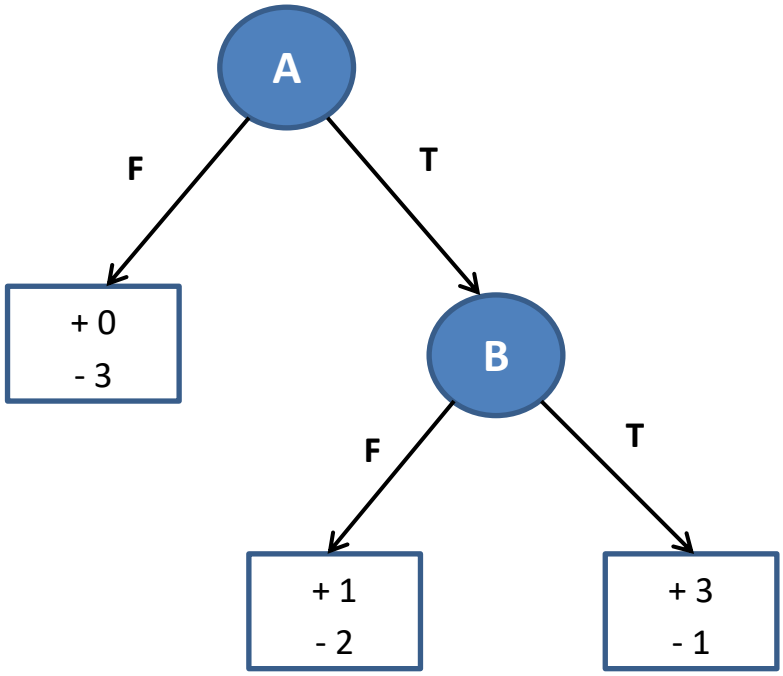
Gain(B) = Entropy - Entropy after split based on B

= 0.971 – 0.7145

= 0.2565

#Q4: The decision tree would choose attribute A because it provides more information gain compared to attribute B.

#Q5: The decision tree learned from this dataset is given below:



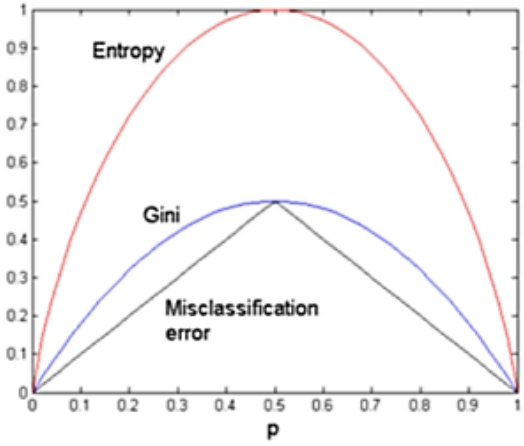
Task 4:

#Q1: Decision trees are not linear classifier. There is no linear function to relate the input with output.

#Q2: Some weaknesses of decision tree is given below:

1. Decision tree often requires higher time to train the model.
2. It tries to fragment the data as much as possible, as a result can end up having a single data point in a leaf node.
3. Decision tree is comparatively less stable. A small change in the data can cause a large change in the structure of the decision tree.
4. Decision tree algorithm doesn't work for continuous features.
5. Decision tree training is relatively expensive as the complexity and runtime can be higher compared to other models.
6. Decision tree might not work well if the number of features is very high.

#Q3: No, Gini index performs better than Misclassification Errors. The reason is Gini index has much higher sensitivity comapred to Misclassification Errors.



Additional Questions:

• It took me approximately 8-9 hours to complete the assignment. It took longer time cause I was trying different combinations of features to increase the accuracy of the models.

• Feature selection was the most challenging part for me. Specifically, while trying to convert continuous features to ordinal features, selecting the banding range was hard for me.

• I learned a lot while trying to preprocess the data and while practically implementing the algorithms, so that was the most interesting part for me. Also, questions based on the mathematical calculations helped me understand the algorithms better.

Code Link: https://github.com/NabilaKhan/CAP-5610-Machine-Learning-/blob/main/CAP-5610-HW2.ipynb

Image Link: https://github.com/NabilaKhan/CAP-5610-Machine-Learning-/blob/main/Decision_tree_plot.png

[Please let me know if you are having any issue to find the code]