CO544 - Machine Learning and Data Mining - Lab05

E/17/219 Nawarathna K.G.I.S

Task 01

Separating labels and features

```
X = wine_df.drop(['class'], axis=1) # Drop the target variable
y = wine_df['class']
X, y
```

Gini decision tree model with the max depth of 4.

```
[43] clf_gini = DecisionTreeClassifier(criterion='gini', max_depth=4, random_state=0)
     clf_gini.fit(X_train, y_train)      # Train the classifier
```

Confusion matrix related to the gini model

```
array([[18,  0,  0],
       [ 1, 15,  1],
       [ 0,  0, 10]])
```

Entropy decision tree model with max depth of 4

```
clf_entropy = DecisionTreeClassifier(criterion='entropy', max_depth=4, random_state=0)
clf_entropy.fit(X_train, y_train)      # Train the classifier
```

Confusion matrix related to the entropy model

```
array([[17,  1,  0],
       [ 1, 16,  0],
       [ 0,  0, 10]])
```

- It was observed that the accuracy of the decision tree model was significantly low when the feature engineering step is applied. Without parameter encoding, the accuracy of the model rises up to 95.5% therefore that step is not applied.

Task 02

x1	x2	x3	y
10	10	5	45
20	?	4	49
10	25	7	75
10	45	8	65
10	?	9	41
30	4	4	74

Consider this dataset. It can be clearly observed that x2 has some missing values in their rows. There are few ways to deal with these.

1. Delete the rows with missing values.
- Since the decision tree get stuck when it finds a missing value, the dataset should be preprocessed before in order to create a dataset without missing values.

- One way of doing this would be to remove rows with missing values. This is one of the most simplest way of dealing with missing values but one of the major issue with this method is that this will result in a smaller dataset compared to the original one. When the dataset is larger it will cause the accuracy of the model to be improved, therefore removing the rows with missing values would indirectly result in lower accuracy of the model.

2. Delete the columns with missing values
- If the majority of some column contains missing values, the whole column can be dropped. But as in the previous method, this method is also not very efficient since this will result in a smaller dataset.

3. Substitution of missing values
- Since data is so valuable, removing rows of the dataset is not that efficient. Instead , the missing values can be substituted by some values. There are few common ways to substitute missing values.
    - Substitute missing value by the most common value in the column.
    - Substitute the missing value with the average value of the column.
- While this will result in non-reduction of the original dataset, since the values are not the actual ones, the accuracy of the model might be reduced.

4. Implementing a learning algorithm to find missing values.
- This will be considered as the best method to deal with missing values. Here, the machine learning model will be implemented to find the missing values of the rows.

Task 03

- Below is the confusion matrix of the decision tree related to the **gini index.**

```
array([[18,  0,  0],
       [ 1, 15,  1],
       [ 0,  0, 10]])
```

- It can be seen that the main diagonal of the confusion matrix contains the correct predictions of the decision tree model.

    Therefore TP values = 18 + 15 + 10 = 43
    Accuracy of the model = (43 / 45) x 100% = 95.56%
    Precision of (class=1) = (18/19)x100% = 94.73%
    Precision of (class=2) = (15/15)x100% = 100%
    Precision of (class=3) = (10/11)x100% = 90.90%

- We can find recall and f1-score parameters as well using the confusion matrix.
- All of these evaluation parameter details can be obtained from the classification report of the decision tree model.

	precision	recall	f1-score	support
1	0.95	1.00	0.97	18
2	1.00	0.88	0.94	17
3	0.91	1.00	0.95	10
accuracy			0.96	45
macro avg	0.95	0.96	0.95	45
weighted avg	0.96	0.96	0.95	45

Task 04

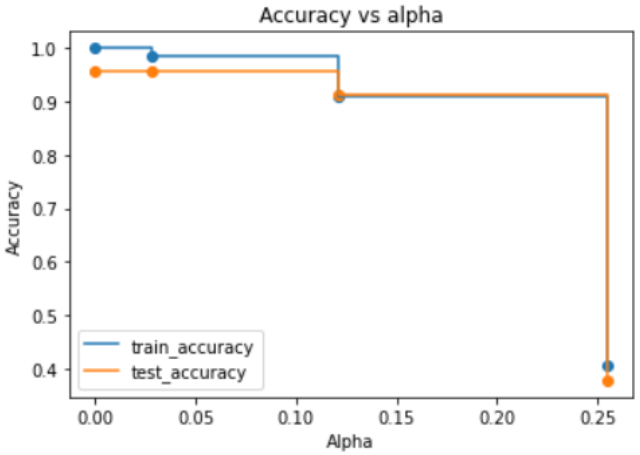
Pruning is a method of removing the overfitting of decision trees by reducing the size(depth) of the machine learning model. There are two types of pruning which are pre-pruning and post-pruning methods.

When applying the pruning, first the alpha values or the smoothness parameter value of the gini model is checked. It was able to observe that there are 4 alpha values related to the gini indexed decision tree model.Below are those alpha values.

```
array([0.        , 0.02857143, 0.12089046, 0.25451322])
```

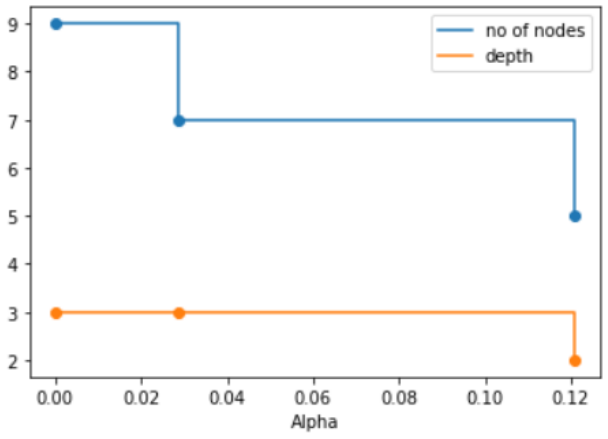
We need to figure out the most optimal alpha value, which will give us the best accuracy of the best non-overfitted decision tree model.

Then for each alpha value, a decision tree model is implemented and it is trained on the training dataset. Then for all of those models, training accuracy and the test accuracy was checked.



We can see that the accuracy of the training and test datasets are reducing just after alpha = 0.1 . This is because , when alpha increases, the smoothness of the machine learning model also increases and hence the model tends to overfit. Due to this overfitting, accuracy of the machine learning model decreases rapidly. Therefore the optimal value for the alpha can be taken as 0.1 so that the model will not overfit.

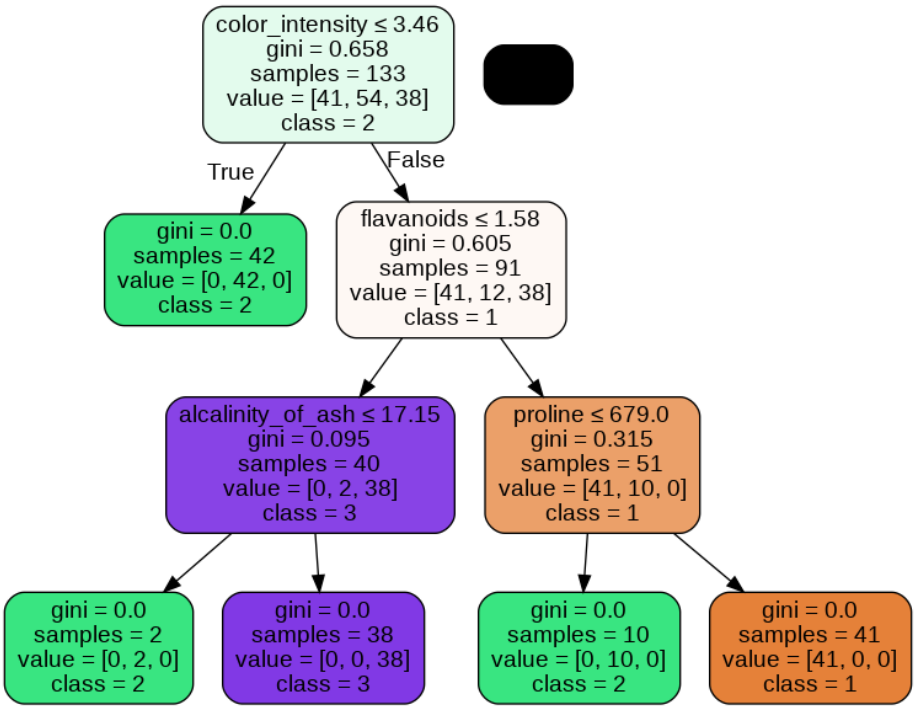
We can also see how the depth and the number of nodes changes with alpha.



As we have seen we can take alpha to be 0.1 as its optimal value. For that value we can take the depth of the decision tree to be 3 , so that the model will not overfit and it will give the best performance.

Task 05

Gini as the index



Entropy as index