NCTU Pattern Recognition, Homework 3

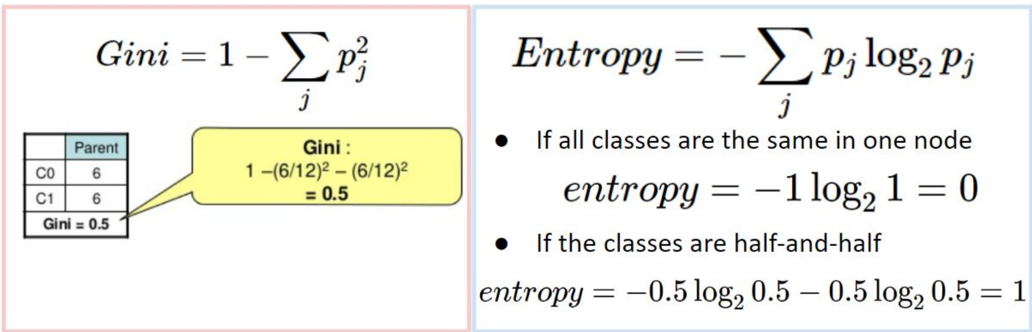
Deadline: May 22, 23:59

Part. 1, Coding (80%):

In this coding assignment, you need to implement the Decision Tree and Random Forest algorithm by using only NumPy, then train your implemented model by the provided dataset and test the performance with testing data. Find the sample code and data on the GitHub page https://github.com/NCTU-VRDL/CS_DCP3121/tree/master/HW3

Please note that only NumPy can be used to implement your model, you will get no points by simply calling sklearn.tree.DecsionTreeClassifier.

1. (10%) Gini Index or Entropy is often used for measuring the "best" splitting of the data. Please compute the Entropy and Gini Index of the provided data by the formula below. (More details on page 7 of the hw3 slides)



   Gini of data is 0.4628099173553719
   Entropy of data is 0.8299157956468823

2. (30%) Implement the Decision Tree algorithm (CART, Classification and Regression Trees) and trained the model by the given arguments, and print the accuracy score on the test data. You should implement **two arguments** for the Decision Tree algorithm,
   1) **Criterion**: The function to measure the quality of a split. Your model should support "gini" for the Gini impurity and "entropy" for the information gain.
   2) **Max_depth**: The maximum depth of the tree. If Max_depth=None, then nodes are expanded until all leaves are pure. Max_depth=1 equals to split data once

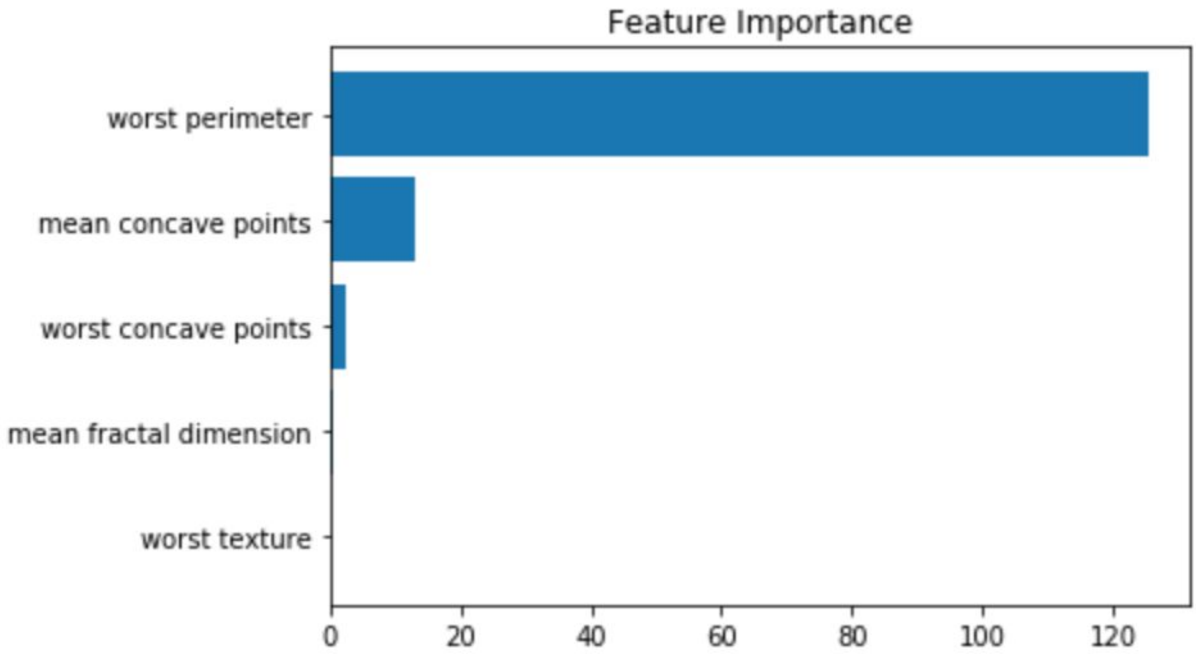
   2.1. Using Criterion='gini', showing the accuracy score of test data by Max_depth=3 and Max_depth=10, respectively.
   accuracy (Max_depth=3): 0.9178403755868545
   accuracy (Max_depth=10): 1.0

   2.2. Using Max_depth=3, showing the accuracy score of test data by Criterion='gini' and Criterion='entropy', respectively.
   accuracy (Criterion='gini'): 0.9178403755868545
   accuracy (Criterion='entropy'): 0.9178403755868545

Note: All of the accuracy scores should over 0.9

Note: You should get the same results when re-building the model with the same arguments, ***no need to prune the trees***
Hint: You can use the recursive method to build the nodes

3. (10%) Plot the feature importance of your Decision Tree model. You can use the model from question 2.1, max_depth=10. (matplotlib is allowed to used)



4. (30%) Implement the Random Forest algorithm by using the CART you just implemented from question 2. You should implement **three arguments** for the Random Forest.
1) **N_estimators**: The number of trees in the forest.
2) **Max_features**: The number of features to consider when looking for the best split
3) **Bootstrap**: Whether bootstrap samples are used when building trees

4.1. Using Criterion='gini', Max_depth=None, Max_features=sqrt(n_features), Bootstrap=True, showing the accuracy score of test data by n_estimators=10 and n_estimators=100, respectively.
accuracy (n_estimators=10): 0.9436619718309859
accuracy (n_estimators=100): 0.9436619718309859

4.2. Using Criterion='gini', Max_depth=None, N_estimators=10, Bootstrap=True, showing the accuracy score of test data by Max_features=sqrt(n_features) and Max_features=n_features, respectively.
accuracy (Max_features=sqrt(n_features)): 0.9436619718309859
accuracy (Max_features=n_features): 0.9366197183098591

Note: Use majority votes to get the final prediction, you may get different results when re-building the random forest model

Part. 2, Questions (20%):

1. (20%) Consider a data set comprising 400 data points from class C_1 and 400 data points from class C_2. Suppose that a tree model A splits these into (300, 100) at the first leaf node and (100, 300) at the second leaf node, where (n, m) denotes that n points are assigned to C_1 and m points are assigned to C_2. Similarly, suppose that a second tree model B splits them into (200, 400) and (200, 0). **Evaluate the misclassification rates for the two trees and hence show that they are equal**. Similarly, **evaluate the cross-entropy** $Entropy = -\sum_{k=1}^{K} p_k \log_2 p_k$ and **Gini index** $Gini = 1 - \sum_{k=1}^{K} p_k^2$ **for the two trees** and show that they are both lower for tree B than for tree A. Define p_k to be the proportion of data points in region R assigned to class k, where k = 1, . . . , K

A: 400 C_1, 400 C_2 → A_1: 300 C_1, 100 C_2; A_2: 100 C_1, 300 C_2

B: 400 C_1, 400 C_2 → 200 C_1, 400 C_2; 200 C_1, 0 C_2

misclassification rates of A

$= \frac{100+100}{400+400} = \frac{1}{4}$

misclassification rates of B

$= \frac{200}{400+400} = \frac{1}{4}$

$\text{Gini } B_1 = 1-\left(\frac{400}{600}\right)^2-\left(\frac{200}{600}\right)^2$

$\text{Gini } B_2 = 0$

$B_G = \frac{600}{800} \times 0.44 + \frac{200}{800} \times 0 = \underline{0.33}$

$\text{Entropy } B_1 = -\frac{200}{600}\log_2\frac{200}{600} - \frac{400}{600}\log_2\frac{400}{600} = 0.92$

$\text{Entopy } B_2 = 0$

$B_E = \frac{600}{800} \times 0.92 + \frac{200}{800} \times 0 = \underline{0.69}$

$\text{Gini } A_1 = 1-\left(\frac{300}{400}\right)^2-\left(\frac{100}{400}\right)^2 = 0.375$

$\text{Gini } A_2 = 1-\left(\frac{300}{400}\right)^2-\left(\frac{100}{400}\right)^2 = 0.375$

$A_G = \frac{400}{800} \times 0.375 + \frac{400}{800} \times 0.375 = \underline{0.375}$

$\text{Entropy } A_1 = -\frac{300}{400}\log_2\frac{300}{400} - \frac{100}{400}\log_2\frac{100}{400} = 0.81$

$\text{Entropy } A_2 = -\frac{100}{400}\log_2\frac{100}{400} - \frac{300}{400}\log_2\frac{300}{400} = 0.81$

$A_E = \frac{400}{800} \times 0.81 + \frac{400}{800} \times 0.81 = \underline{0.81}$

$\boxed{\begin{array}{l} A_G = 0.375 > 0.33 = B_G \\ A_E = 0.81 > 0.69 = B_E \end{array}}$