NYCU Introduction to Machine Learning, Homework 3

110550014, 吳權祐

The screenshot and the figures we provided below are just examples. The results below are not guaranteed to be correct. Please make sure your answers are clear and readable, or no points will be given. Please also remember to convert it to a pdf file before submission. You should use English to answer the questions. After reading this paragraph, you can delete this paragraph.

Part. 1, Coding (50%):

(30%) Decision Tree

1. (5%) Compute the gini index and the entropy of the array [0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 1].

```
gini of [0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 1]: 0.5371900826446281
entropy of [0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 1]: 0.9456603046006401
```

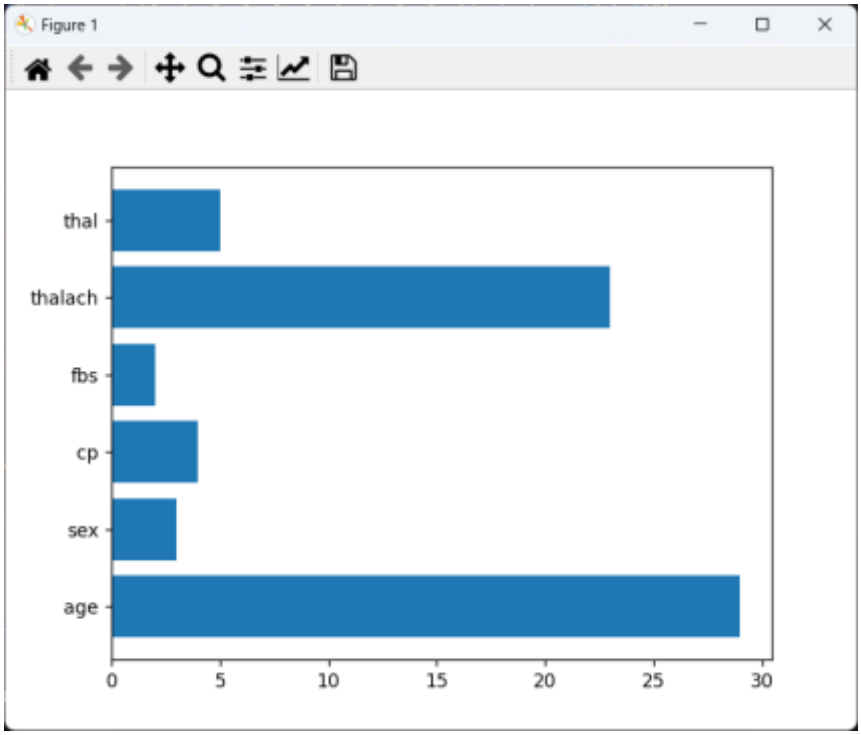
2. (10%) Show the accuracy score of the testing data using criterion="gini" and max_depth=7. Your accuracy score should be higher than 0.7.

```
Accuracy (gini with max_depth=7): 0.7049180327868853
```

3. (10%) Show the accuracy score of the testing data using criterion="entropy" and max_depth=7. Your accuracy score should be higher than 0.7.

```
Accuracy (entropy with max_depth=7): 0.7213114754098361
```

4. (5%) Train your model using criterion="gini", max_depth=15. Plot the feature importance of your decision tree model by simply counting the number of times each feature is used to split the data. Your answer should look like the plot below:



(20%) Adaboost

1. (20%) Tune the arguments of AdaBoost to achieve higher accuracy than your Decision Tre es.

```
Accuracy: 0.8032786885245902
```

Part. 2, Questions (50%):

1. (10%) True or False. If your answer is false, please explain.
    a. (5%) In an iteration of AdaBoost, the weights of misclassified examples are increa sed by adding the same additive factor to emphasize their importance in subsequen t iterations.

       False
       The weights of misclassified samples are increased by multiplying a ratio depending on error of current iteration and then train next iteration.

    b. (5%) AdaBoost can use various classification methods as its weak classifiers, such as linear classifiers, decision trees, etc.

       True

2. (10%) How does the number of weak classifiers in AdaBoost influence the model's perfor mance? Please discuss the potential impact on overfitting, underfitting, computational cost, memory for saving the model, and other relevant factors when the number of weak classifi ers is too small or too large.

   When the number of weak classifiers in the model is too small, it may have lower computational cost and memory usage, but this cold lead to poor performance or accuracy due to underfitting. Conversely, if the number is too large, the model may incur a higher computational cost and require more memory for storing the model. Although such a model might perform well on training data, its accuracy on testing data could be poor, indicating overfitting.

3. (15%) A student claims to have a brilliant idea to make random forests more powerful: since random forests prefer trees which are diverse, i.e., not strongly correlated, the student proposes setting m = 1, where m is the number of random features used in each node of each decision tree. The student claims that this will improve accuracy while reducing variance. Do you agree with the student's claims? Clearly explain your answer.

   I don't agree with the student. Here is my explanation. First, if m is too small, like 1, the trees may become very similar with each other. And this results in low effectiveness of ensemble. Another reason is that if m is too small, the model may be very keen to some outlier or noise in the data, this may cause to overfitting and make the get bad performance.

4. (15%) The formula on the left is the forward process of a standard neural network while the formula on the right is the forward process of a modified model with a specific technique.
   a. (5%) According to the two formulas, describe what is the main difference between the two models and what is the technique applied to the model on the right side.

      The left formula shows a standard neural network model, and the right one with dropout technique which multiplying y with Bernoulli distribution before entrering to the next layer.
   b. (10%) This technique was used to deal with overfitting and has many different explanations; according to what you learned from the lecture, try to explain it with respect to the ensemble method.

      Ensemble method combines several weak classifiers, each of them could classify part of data precisely. This could prevent the model from overemphasizing some nodes from training, which reduces overfitting to improve performance for the whole model. And dropout could be seen as a method to ensemble the subnetworks that makes each of them predict part of data precisely, and get better prformance for the model since the same reason as ensemble method.

$$z^{(l+1)} = w^{(l+1)} y^l + b^{(l+1)}$$

$$y^{(l+1)} = f(z^{(l+1)})$$

$$r^l = Bernoulli(p)$$

$$\tilde{y}^l = r^l y^l$$

$$z^{(l+1)} = w^{(l+1)} \tilde{y}^l + b^{(l+1)}$$

$$y^{(l+1)} = f(z^{(l+1)})$$