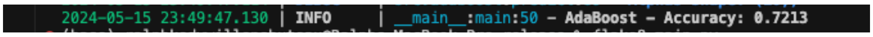
NYCU Pattern Recognition, Homework 3

109550198, 卜銳凱

Part. 1, Coding (60%):

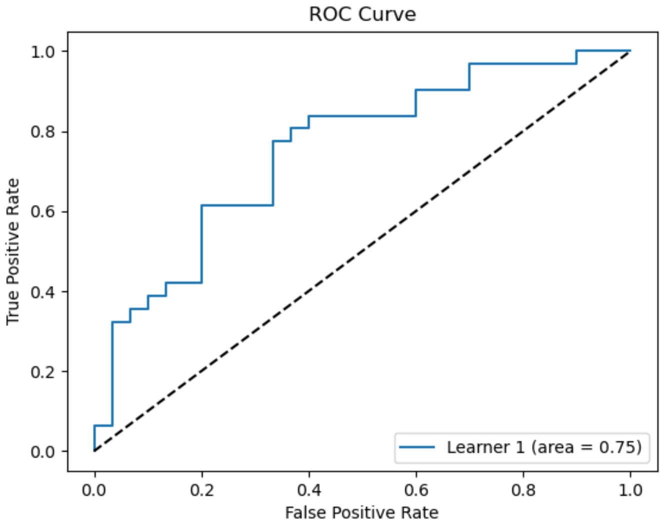
(20%) Adaboost

1. (10%) Show your accuracy of the testing data (n_estimators = 10)

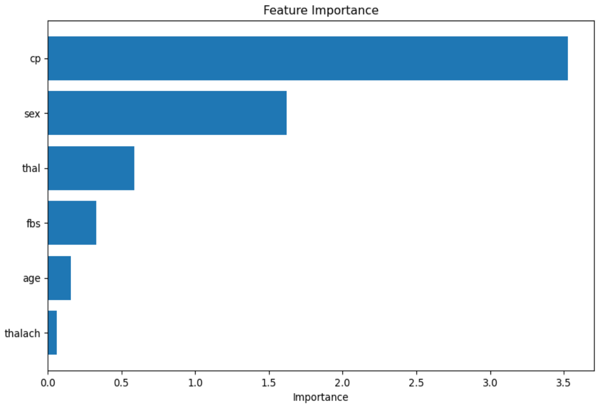


```
2024-05-15 23:49:47.130 | INFO     | __main__:main:50 - AdaBoost - Accuracy: 0.7213
```

2. (5%) Plot the AUC curves of each weak classifier.

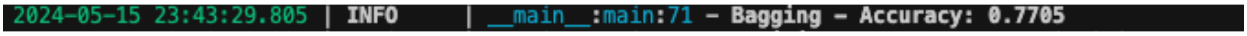


3. (5%) Plot the feature importance of the AdaBoost method. Also, you should snapshot the implementation to calculate the feature importance.



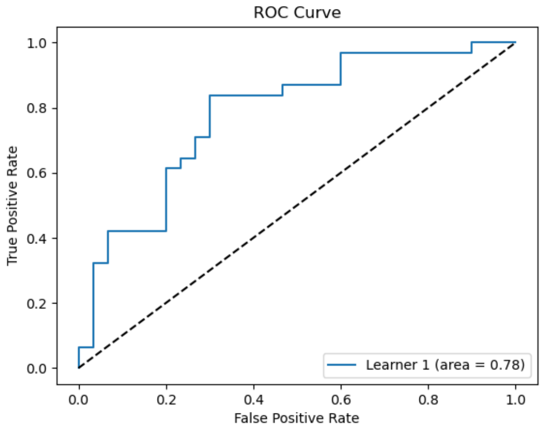
(20%) Bagging

4. (10%) Show your accuracy of the testing data with 10 estimators. (n_estimators=10)

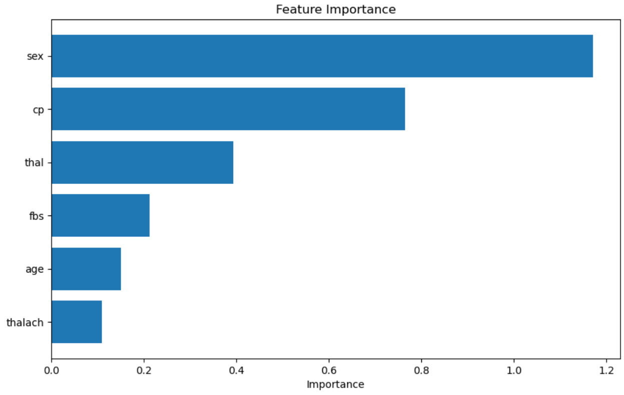


```
2024-05-15 23:43:29.805 | INFO     | __main__:main:71 - Bagging - Accuracy: 0.7705
```

5. (5%) Plot the AUC curves of each weak classifier.

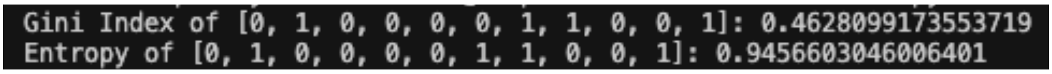


6. (5%) Plot the feature importance of the Bagging method. Also, you should snapshot the implementation to calculate the feature importance.



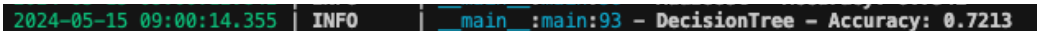
(15%) Decision Tree

7. (5%) Compute the gini index and the entropy of the array [0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 1].



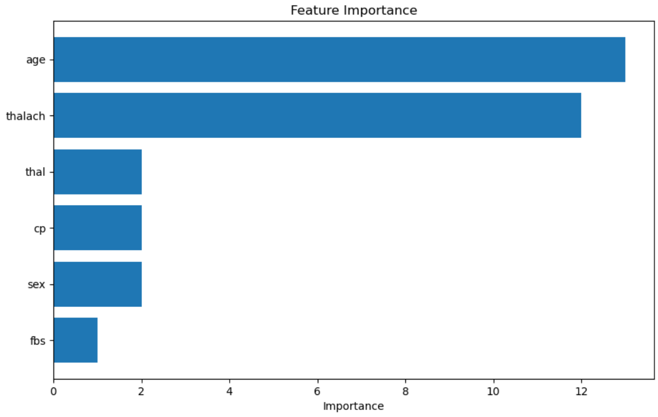
```
Gini Index of [0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 1]: 0.4628099173553719
Entropy of [0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 1]: 0.9456603046006401
```

8. (5%) Show your accuracy of the testing data with a max-depth = 7



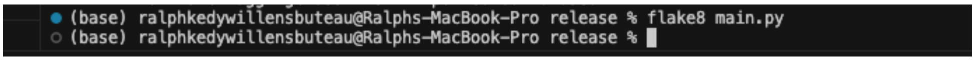
```
2024-05-15 09:00:14.355 | INFO     | __main__:main:93 - DecisionTree - Accuracy: 0.7213
```

9. (5%) Plot the feature importance of the decision tree.



(5%) Code Linting

10. Show the snapshot of the flake8 linting result.



```
● (base) ralphkedywillensbuteau@Ralphs-MacBook-Pro release % flake8 main.py
○ (base) ralphkedywillensbuteau@Ralphs-MacBook-Pro release % 
```

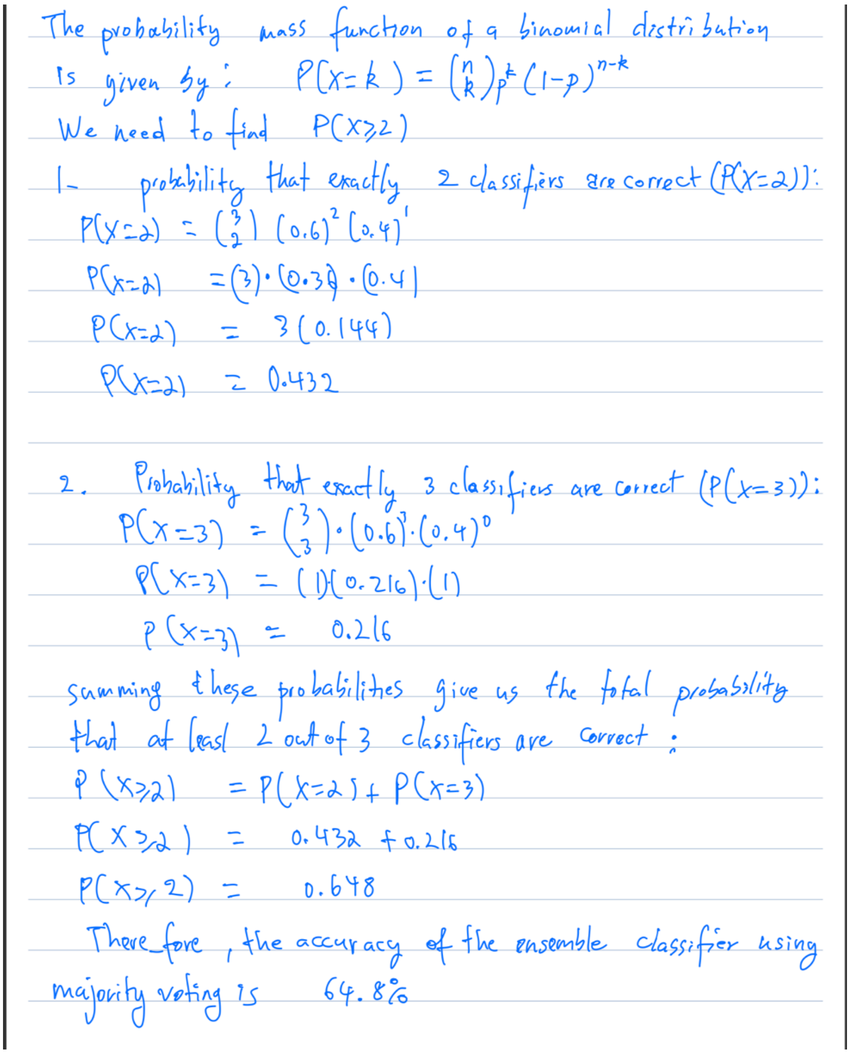
Part. 2, Questions (40%):

1. (10%) We have three distinct binary classifiers, and our goal is to leverage them in creating an ensemble classifier through the majority voting strategy to make decisions.
Assuming each individual binary classifier operates independently of the others with an accuracy of 60%, what would be the accuracy of the ensemble classifier?
Write or type your answer here.

Given that each classifier has an accuracy of 60%, the probability that a single classifier makes a correct decision is P(correct)=0.6 and the probability that it makes an incorrect decision is P(incorrect)=0.4.

In a majority voting system with three classifiers, the ensemble classifier will be correct if at least two out of the three classifiers make the correct decision. We can calculate this probability using the binomial distribution.

Let X be the random variable representing the number of classifiers that make the correct decision. X follows a binomial distribution with parameters n=3 (number of trials) and p=0.6 (probability of success).



The probability mass function of a binomial distribution is given by: $P(x=k) = \binom{n}{k} p^k (1-p)^{n-k}$

We need to find $P(X \geq 2)$

1. probability that exactly 2 classifiers are correct ($P(X=2)$):

$P(X=2) = \binom{3}{2} (0.6)^2 (0.4)^1$

$P(x=2) = (3)\cdot(0.36)\cdot(0.4)$

$P(x=2) = 3(0.144)$

$P(x=2) = 0.432$

2. Probability that exactly 3 classifiers are correct ($P(x=3)$):

$P(x=3) = \binom{3}{3}\cdot(0.6)^3\cdot(0.4)^0$

$P(x=3) = (1)(0.216)\cdot(1)$

$P(x=3) = 0.216$

summing these probabilities give us the total probability that at least 2 out of 3 classifiers are correct:

$P(x \geq 2) = P(x=2) + P(x=3)$

$P(X \geq 2) = 0.432 + 0.216$

$P(x \geq 2) = 0.648$

Therefore, the accuracy of the ensemble classifier using majority voting is 64.8%

2. (15%) For the decision tree algorithm, we can use the "pruning" technique to avoid overfitting. Does the random forest algorithm also need pruning?
Please explain in detail.
Write or type your answer here.

The random forest algorithm does not require pruning like individual decision trees. Here's a brief explanation:

Decision Trees and Pruning

- Overfitting: Decision trees can become too complex and overfit the training data.
- Pruning: Techniques like pre-pruning (early stopping) and post-pruning are used to simplify trees and improve generalization.

Random Forests and Pruning

- Random Forests: An ensemble method that builds multiple decision trees and combines their predictions.
- Reducing Overfitting: Uses bootstrapping (training each tree on random subsets of data) and random feature selection (considering random subsets of features for splits) to ensure diversity among trees.
- Ensemble Effect: The combination of many overfitted trees averages out errors, leading to better generalization.

Conclusion

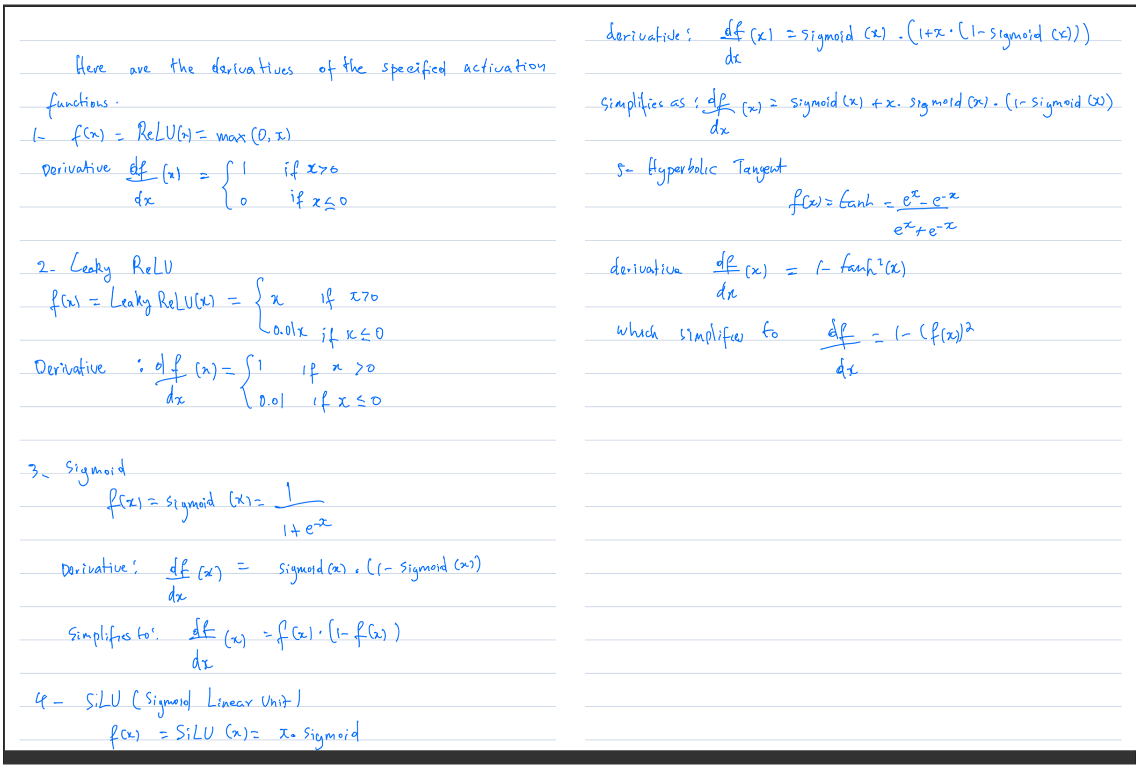
- Random forests do not require pruning because their structure and methods naturally prevent overfitting.
- Each tree is allowed to grow deep, capturing detailed patterns, while the ensemble averaging reduces overfitting.

In summary, random forests inherently manage overfitting and do not need the pruning applied to individual decision trees.

3. (15%) Activation functions are core components of neural networks. They need to be differentiable to ensure backpropagation works correctly. Please calculate the derivatives of the following commonly used activation functions.
(For questions 1. and 2., consider the cases where $x > 0$ and $x \leq 0$)

1. f(x) = relu(x),	df(x)/dx = ?
2. f(x) = leaky_relu(x) with negative_slope=0.01,	df(x)/dx = ?
3. f(x) = sigmoid(x),	df(x)/dx = ?
4. f(x) = silu(x),	df(x)/dx = ?
5. f(x) = tanh(x),	df(x)/dx = ?

Write or type your answer here.



Here are the derivatives of the specified activation functions.

1- $f(x) = \mathrm{ReLU}(x) = \max(0, x)$

Derivative $\frac{df}{dx}(x) = \begin{cases} 1 & \text{if } x > 0 \\ 0 & \text{if } x \leq 0 \end{cases}$

2- Leaky ReLU

$f(x) = \text{Leaky ReLU}(x) = \begin{cases} x & \text{if } x > 0 \\ 0.01x & \text{if } x \leq 0 \end{cases}$

Derivative : $\frac{df}{dx}(x) = \begin{cases} 1 & \text{if } x > 0 \\ 0.01 & \text{if } x \leq 0 \end{cases}$

3- Sigmoid

$f(x) = \text{sigmoid}(x) = \frac{1}{1+e^{-x}}$

Derivative: $\frac{df}{dx}(x) = \text{sigmoid}(x) \cdot (1 - \text{sigmoid}(x))$

Simplifies to: $\frac{df}{dx}(x) = f(x) \cdot (1 - f(x))$

4- SiLU (Sigmoid Linear Unit)

$f(x) = \text{SiLU}(x) = x \cdot \text{sigmoid}$

derivative: $\frac{df}{dx}(x) = \text{sigmoid}(x) \cdot (1 + x \cdot (1 - \text{sigmoid}(x)))$

simplifies as: $\frac{df}{dx}(x) = \text{sigmoid}(x) + x \cdot \text{sigmoid}(x) \cdot (1 - \text{sigmoid}(x))$

5- Hyperbolic Tangent

$f(x) = \tanh = \frac{e^x - e^{-x}}{e^x + e^{-x}}$

derivative $\frac{df}{dx}(x) = 1 - \tanh^2(x)$

which simplifies to $\frac{df}{dx} = 1 - (f(x))^2$