Programming Assignment #4

The coding component of this assignment should be <u>submitted through e-Learning</u> before **11:59 pm Central Daylight Time on Sunday, April 9, 2023**. The report component of this assignment <u>should be submitted along with the code</u> **on Sunday, April 9, 2023**. These deadlines are without exceptions unless permission was obtained from the Professor **in advance**. You may collaborate with other students, discuss the problems and work through solutions together. However, you **<u>must write up your code (a team of two is allowed)</u>** and solutions, without copying another team's work or letting another team copy your work. In your solution for each problem, you must write down the names of any person with whom you discussed it. <u>This will **not** affect your grade.</u>

Problem: Implement Bagging and AdaBoost based on the decision tree code that you developed in Programming Assignment 1. This code must be modified to work with ensemble methods. Ensure that your code is modified and organized as follows:

a. For this assignment, we will restrict ourselves to binary trees. This means that we will restrict ourselves to two splits instead of multiple possible splits at each node. More precisely, each node is restricted to be of the form $x_i = v_k$?, where it tests if the feature x_i takes the value v_k.

b. Implement three functions: bagging(x, y, max depth, num trees), boosting(x, y, max_depth, num_stumps) and predict example(x, h_ens), where h_ens is an ensemble of weighted hypotheses. The ensemble is represented as an array of pairs [(alpha_i, h_i)], where each hypothesis and weight are represented by the pair: (alpha_i, h_i).

Data Sets: We will use the Mushroom Data Set[1] from the UCI Repository for this assignment. There are 22 attributes in this data set, of which we have dropped the attribute (stalk-root) as it contains too many missing values. The data set has been converted from string to integer, with the unique feature values being assigned indices starting from 0 in alphabetical order. Also note that rather than perform the classical task of predicting whether the mushroom is poisonous or edible, our classification task is to predict (bruises?).

Experiments: Once you have debugged and tested your code, run the following experiments and write a brief report answering the following questions:

a. (**Bagging**, 20 points) Construct four models for each combination of maximum depth d = 3,5 and bag size (k = 10,20). Report the confusion matrix for these four settings.

b. (**Boosting**, 20 points) Construct four models for each combination of maximum depth d = 1,2 and bag size (k = 20,40). Report the confusion matrix for these four settings.

c. (**scikit-learn**, 40 points) Use scikit-learn's bagging and AdaBoost learners and repeat the experiments as described in parts (a) and (b) above. Report the confusion matrices for these sets of settings. What can you say about the quality of your implementation's performance versus scikit's performance?

Upload: Make sure all your code, including the (modified) decision tree functionality, is in a single file. Also ensure that your code can be it can be executed by calling the main function. Upload your file through e-Learning before the deadline.

<u>**Acknowledgement:**</u> The assignment is developed in collaboration with Dr. Gautam Kunapuli.

[1] https://archive.ics.uci.edu/ml/datasets/mushroom