# CSE 417T: Homework 4

Due: April 1 (Friday), 2022

**Notes:**

- Please submit your homework via Gradescope and check the submission instructions.
- Please download the following files for this homework.
  `http://chienjuho.com/courses/cse417t/hw4/hw4.html`
- Homework is due **by 11:59 PM on the due date.** Remember that you may not use more than 2 late days on this homework, and you only have a budget of 5 in total.
- Please keep in mind the collaboration policy as specified in the course syllabus. If you discuss questions with others you **must** write their names on your submission, and if you use any outside resources you **must** reference them. **Do not look at each others' writeups, including code.**
- Please comment your code properly.
- There are 5 problems on 3 pages in this homework.

**Problems:**

1. (50 points) The purpose of this problem is to write code for bagging decision trees and computing the out-of-bag error. You may use `sklearn.tree.DecisionTreeClassifier` function[1], which learns decision trees (read the documentation carefully), but do not use the inbuilt functions for producing bagged ensembles. You may assume that all the **x** vectors in the input are vectors of real numbers, and there are no categorical variables/features. You will compare the performance of the bagging method with plain decision trees on the handwritten digit recognition problem (the dataset is in `zip.train` and `zip.test`, available from `http://amlbook.com/support.html`. [2])

   We will focus on two specific binary classification problems – distinguishing between the digit one and the digit three, and distinguishing between the digit three and the digit five. You need to report the results for both problems (1 versus 3 and 3 versus 5).

   - Code (Complete and submit `hw4.py`)
     a Write code that creates training and testing datasets from `zip.train` and `zip.test` for the two binary classification problems. The first will only include digits classified as one or three. The second will only include digits classified as three or five.

[1] https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html
[2] Check the links to "training set" and "test set".

b Complete the implementation of `bagged_trees`. For each bag, please learn a fully grown tree and use information gain as the split criterion. You do not need to implement the random feature split. You can update the function headers or write new functions for your convenience. The requirement is that you need to be able to calculate and plot the out-of-bag error as a function of the number of bags from 1 to the number specified as input (`num_bags`),

c Learn a single decision tree model (again, fully grown with information gain as the split criterion) from the training dataset and calculate the test error on the test set.

- Report:
  For each of the problems (1 versus 3 and 3 versus 5):

  a Plot the OOB error for bagging decision trees with the number of bags from 1 to 200 (with x-axis being the number of bags, and y-axix being the OOB error). Make sure the axes are clearly labeled.

  b Report the OOB error of bagging decision trees (with 200 trees) and the test errors of (1) a single decision tree and (2) bagging decision trees (with 200 trees).

  Summarize and interpret your results in one or two concise paragraphs as part of your writeup. You should at least comment on 1) the differences between the one-vs-three and three-vs- five problems, 2) the effect of increasing the number of bags, and 3) the connection between OOB error and test error.

2. (20 points) You have been hired by a biologist to learn a decision tree to determine whether a mushroom is poisonous. You have been given the following data:

| Color | Stripes | Texture | Poisonous? |
|---|---|---|---|
| Purple | No | Smooth | No |
| Purple | No | Rough | No |
| Red | Yes | Smooth | No |
| Purple | Yes | Rough | Yes |
| Purple | Yes | Smooth | Yes |

   Use ID3 to learn a decision tree from the data (this is a written exercise – no need to code it up):

   (a) What is the root attribute of the tree? Show the computations.

   (b) Draw the decision tree obtained using ID3.

3. (10 points) Think about weak learners in AdaBoost for a 2-class classification problem. Suppose you're using depth 0 decision trees, which simply return the weighted majority class of the data points as the classification, as the weak learner. Imagine that, at the first iteration, 80% of the data points were positive and 20% of the data points were negative. What would the cumulative weight of positive points and the cumulative weight of negative data points be after one round of boosting)? From your result, do you think whether using depth 0 decision trees as weak learners is a good idea?

4. (5 points) Assume we want to use bagging to solve a regression problem. Argue why using linear regression as weak learners for bagging might be a bad idea.

5. (15 points) Read the article(s) in one of the topics below:
   https://docs.google.com/document/d/1QmKDwodF9gTZQzrzQuES6EXDDwmqNWOfRFqGqTP83k8
   edit?usp=sharing

   Answer the following questions:

   (a) Summarize the article(s) in one paragraph.

   (b) Rephrase the issues raised in the article using the language you learned in this course.

   (c) Propose potential approaches to mitigate the issues raised in the article.

   This question will be graded in a loose manner. The grading will focus on 1) whether you have put thoughts into your answer, and 2) whether your answers are logical (i.e., it's okay to give unpopular opinion, but make sure you provide your reasoning).

   Example grades (not an exhaustive list, but just an illustration):
   You will get full points if the summary is accurate, and the proposed approach makes sense (e.g., feasible if you are given the data and resource). You will get 10 points with an accurate summary but non-negligible flaws in the proposed approach. You will get 5 points for providing inaccurate summary (e.g., incorrect mapping to the language in this course).