CSE 417T: Homework 4
Due: Saturday, November 9th, 2019 by 9 PM

Notes:

• Please check the submission instructions for Gradescope provided on the course website. You must follow those instructions exactly. For this assignment, you may work in groups of up to two people. Each group only needs to submit a single copy of the homework. The following video contains instructions on submitting group submissions:
  
  - https://www.youtube.com/watch?v=rue7p_kATLA

  It is your responsibility to ensure that group members are assigned correctly; **regrade requests for incorrect groupings will not be entertained.**

• Please download these files (three Matlab stub files, one completed Matlab script and two data sets):
  
  - https://classes.cec.wustl.edu/~SEAS-SVC-CSE417T/AdaBoost.m
  - https://classes.cec.wustl.edu/~SEAS-SVC-CSE417T/BaggedTrees.m
  - https://classes.cec.wustl.edu/~SEAS-SVC-CSE417T/RandomForest.m
  - https://classes.cec.wustl.edu/~SEAS-SVC-CSE417T/OneThreeFive.m
  - https://classes.cec.wustl.edu/~SEAS-SVC-CSE417T/zip_train.csv
  - https://classes.cec.wustl.edu/~SEAS-SVC-CSE417T/zip_test.csv

• Your score on coding questions will be based exclusively on your written report. The code you submit is only used for checking correctness (i.e. resolving discrepancies in your writeup) and for plagiarism checking. **Results included in the code submission will not be graded.**

• Homework is due by 9 PM on the due date. Remember that you may not use more than 2 late days on any one homework. The group has access to as many late days as the member with the most remaining late days but all group members will be charged those late days. For example, if one group member has two late days and the other only has one, then this group may submit the assignment two days late, after which neither member will have any late days remaining.

• 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. You do not need to include your group members. Do not look at each others’ writeups, including code.

• There are 3 problems and 1 bonus problem on 3 pages in this homework.

Problems:

1. (50 points) In this problem, you will write code to implement bagged decision trees and random forests. In order to do this, you should complete the stub files BaggedTrees.m and RandomForest.m. You may use Matlab’s fitctree function, which learns decision trees using the CART algorithm (be sure to read the documentation carefully), but **do not use any**
functions for producing bagged ensembles or random forests. You may assume that all inputs are vectors of real numbers; there are no categorical features. You will compare the performance of these methods with plain decision trees on the handwritten digit recognition problem (you can read more about the data set at http://amlbook.com/data/zip/zip.info).

We will focus on two specific problems: distinguishing between the digit one vs. the digit three, and distinguishing between the digit three vs. the digit five.

(a) Complete the implementation of BaggedTrees.m. Include the plots (with clearly labeled axes) of the out-of-bag error as a function of the number of bags for both the one-vs-three and three-vs-five problems in your writeup and submit your code.

(b) Complete the implementation of RandomForest.m. Include the plots (with clearly labeled axes) of the out-of-bag error as a function of the number of bags for both the one-vs-three and three-vs-five problems in your writeup and submit your code.

(c) Run the provided OneThreeFive.m script, which creates training data sets based on the one-vs-three and three-vs-five cases we are interested in, then calls both the in-built decision tree routine and your bagging/random forest functions, prints out the cross-validation error for decision trees, the OOB error for your implementations and the test error for each method on the test data set. Report these results in your writeup.

(d) Summarize and interpret all of the findings above in two or three concise paragraphs. Make sure to comment on 1) the differences between the three machine learning models, 2) the differences between the one-vs-three and three-vs-five problems and 3) the effect of increasing the number of bags. Discuss any other interesting trends you observe in your results.

2. (30 points) In this problem, you will write code to implement AdaBoost using decision stumps as the weak learners (you may use the fitctree function to implement the weak learners). We will again focus on the one-vs-three and three-vs-five problems (as described in Problem 1) using zip_train.csv and zip_test.csv.

Complete the implementation of AdaBoost.m. Submit your code and include two plots in your writeup (both with clearly labeled axes and a legend):

(a) The training error and test error (both on one plot) as a function of the number of weak learners for the one-vs-three problem.

(b) The training error and test error (both on one plot) as a function of the number of weak learners for the three-vs-five problem.

(c) Summarize and interpret these figures in one or two concise paragraphs. Make sure to comment on 1) the differences between the one-vs-three and three-vs-five problems, 2) the effect of increasing the number of weak learners and 3) the generalization of AdaBoost as the number of weak learners increases. Discuss any other interesting trends you observe in your results.

3. (20 points) Consider the following two-dimensional classification data set:

\[ D = \{(2, 0, +1), (5, 2, +1), (6, 3, +1), (0, 1, -1), (2, 3, -1), (4, 4, -1)\} \]

(a) Draw the k-NN decision boundary for \( k = 1 \) given the above data set. You do not need to submit your code for this problem.
(b) Now modify the data set by multiplying the $x_2$ feature of each observation by 5. Draw the new $k$-NN decision boundary for $k = 1$ for this scaled data set. You do not need to submit your code for this problem.

(c) Comment on the effect of scaling the $x_2$ feature. Is this effect good or bad? Why?

4. (Bonus 5 points) Write a multiple choice question related to the content of this homework (Decision trees, bagging, boosting and $k$-NN). Be sure to indicate the correct answer! See Lecture 2, Slides 3 and 4 for the rubric and guidelines to writing a good multiple choice question. If you write a great question, there’s a chance it will be included on the next exam!