CSE 151: Introduction to Machine Learning Summer 2019, Session 1

Programming Assignment 4

Instructor: Joseph Geumlek **Due on:** Sep 5, 2019

Instructions

- This is a 20 point assignment, to be done individually.
- For this assignment, you are free to use any programming language you wish.
- The programming assignment should be submitted as a single pdf file, containing the answers to the questions first, followed by the code. Please submit the assignment through Gradescope.

Problem 1: 20 points

In this assignment, we will look at the task of spam classification using boosting. Our raw data is a set of emails, which were collected from a liguistics mailing list; the emails are labeled as spam or not spam. For your benefit, we have already preprocessed the emails to remove stop-words, punctuation, and to do some preliminary preprocessing that lemmatises the words (for example, that maps words such as *include*, *includes* and *included* to the same word), and converted them to vectors of features.

Download files `pa4train.txt`, `pa4test.txt` and `pa4dictionary.txt` from the class website. The first two files contain your training and test datasets respectively. The third file is a dictionary and contains a list of words. Each line in the files `pa4train.txt` and `pa4test.txt` correspond to an email followed a label which can be 1 or -1. An email is represented by a feature vector of length 4003; a label 1 indicates that the email is not a spam message, and a label -1 indicates that it is spam. Coordinate i of the feature vector of an email is 1 when word i in `pa4dictionary.txt` is present in the email and 0 otherwise.

1. Write down the training and test errors of the classifiers obtained after $t = 3, 7, 10, 15, 20$ rounds of boosting. Use the following weak learning procedure. Each weak learner will be a simple threshold on a single feature. Each candidate weak learner corresponds to a classifier $h_{i,+}$ or $h_{i,-}$, where i is a word in the dictionary and the classifier $h_{i,+}$ is the rule:

$$\begin{aligned} h_{i,+}(x) = &\quad 1, \quad \text{if word } i \text{ occurs in email x} \\ = &\quad -1, \quad \text{otherwise} \end{aligned}$$

   Similarly, the classifier $h_{i,-}$ is the rule:

$$\begin{aligned} h_{i,-}(x) = &\quad 1, \quad \text{if word } i \text{ does not occur in email x} \\ = &\quad -1, \quad \text{otherwise} \end{aligned}$$

   The set of weak learners C is the collection of such classifiers for all i, and your weak learning procedure should select the weak learner which has the *highest accuracy* in C with respect to the current weighted set of examples. Thus, in each round of boosting, you will have 8006 candidate weak learners, and you will select the best in each round. If there are ties, break them randomly.

2. Based on the dictionary file, write down the words corresponding to the weak learners chosen in the first 10 rounds of boosting.

[Hint: If your code is correct, you should get a training error of 0.051 and a test error of 0.039 after 4 rounds of boosting.]