CS434 Assignment 1
Due: Oct 11th in class

Submission instruction:

Implementation assignment: Please submit the implementation assignment through TEACH. Your submission should include both the source code (a readme file for how to run) and your report.

Written assignment: The solution should be typed. You can either submit a printed version in class, or via email to both the TA and the instructor.

Part I. (50 pts) Implementation assignment. You need to form a team of up to two students to work on this part of the assignment. The task is to implement the online perceptron algorithm. There is no specific requirement on which programming language you use, but your code should provide comments and be easily understandable. Writing clean and easy to understand code will be rewarded. Your implementation should contain the following functions:

1. a \texttt{p\_train} function, which will take the following inputs: the training data, \# of epochs, and output a weight vector for the learned decision boundary.
2. a \texttt{p\_classify} function, which takes the following puts: an example \(x\) and weight vector \(w\), and outputs the prediction of \(y\) for the example \(x\).

Test your algorithm on the dataset that is posted online. Two data sets will be provided, one for training, one for testing. In this data, each example is described by three features, and there are two classes \({+1, -1}\).

You will need to write up a report to summarize your results on this data set. The report should contain the following:

1. A plot of the classification accuracy of the learned classifier on both the training set and test set respectively as a function of the number of epochs (up to 100 epochs).

2. When training the perceptron algorithm, the ordering that the examples are presented to the learner can make a big difference. In this experiment, you will examine two ways of handling the ordering. First, you will feed the examples to the algorithm in the exact same order they are read from the file. Alternatively, in each epoch, randomly reshuffle the order in which the examples are processed. Please plot the results (as requested in #1 and #2) for both methods. What difference do you observe? Provide an explanation for this.

3. If a magic crystal ball tells you that the data only has two useful features, which of the three features do you think is not useful? What is your strategy for identifying it?
Part II For this part, you need to submit your own individual solution.

(30 pts) See the class website for the training and testing data for this problem. Please complete the following tasks using the provided data.

a. (8 pts) Learn a linear regression model using only the knee height to predict the height. You can compute your solution using excel, by hand, or writing a program for it. Please report the optimal $w$ and $b$, and the resulting SSE (Sum of squared error) on the training data and test data respectively. Please also provide a plot of the training and testing data (in different colors) along with the learned regression line.

b. (10 pts) Learn using the training data a linear regression model using both knee height and arm span to predict the height. Report the learned $w$ and $b$ and the resulting SSEs on the training and testing data.

c. (12 pts) Compare the SSEs achieved above, and answer the following questions.
   1) What is your observation regarding the comparison between training data SSE and test data SSE? Do you generally expect one to be smaller than the other? In which order? Why?
   2) What is your observation regarding the comparison between the SSEs achieved using one feature and using two features? Do you generally expect one to be smaller than the other? In what order? Why? Provide your answer for training SSE and testing SSE separately.