CSE 40625 — Machine Learning
Assignment 1 (10 points)
Due Date: Feb. 7, 2017 (Sakai Dropbox)

The Perceptron

Overview

The purpose of this assignment is for you to implement the perceptron, one of the most basic and useful machine learning models. The perceptron is a supervised linear classifier that makes predictions by combining the feature vector with a set of weights. Materials for the assignment, including the dataset, expected output, and template code can be found [here](#) on GitHub.

You may use Python libraries for handling data preprocessing and visualization, including but not limited to NumPy, SciPy, pandas, and Matplotlib, but *you may NOT use any Python libraries that employ machine learning models, including but not limited to scikit-learn, StatsModels, TensorFlow, or Orange*. Your solution to the assignment should be individually submitted.

Dataset

You will use a modified version of the "digits" dataset on handwritten digit classification (more details on the digits dataset [here](#)) for this assignment. The modified dataset is a classification problem with 2 classes (labeled +1 and -1). The data is provided in comma-separated (CSV) file format. For all rows, the last column designates the class (y) and the remaining columns designate features (X). The first row consists of the feature and class names.

Procedure

Use the modified digits dataset discussed above as the input to a perceptron. Insert the threshold into the weight vector as a negative bias term (i.e., feature column of -1). Initialize the weight vector with random samples from a uniform distribution over [0, 1]. Iterate through each instance in order, correcting each misclassification. Use the sign of the dot product of the weight vector and feature vector as the class prediction. Repeatedly iterate through the entire dataset until no misclassified instances remain. Use the provided code template to guide your thinking.

Output

Your code should output a new line for each correction, showing the total number of corrected misclassifications (starting from 0), a space, and the current classification accuracy to three decimal places. Note that the format of your output should match the example output provided.

Example output:
        0 0.494
        1 0.978
        …

Submission

Please submit a Python executable (perceptron.py) file of your code to your Sakai Drop Box.

Should you run into any problems, please feel free to email or meet with the instructor.