CS 4340/5340
Project #1
Points 100, Due: 11:59 pm, Sep 15

Implement the Perceptron Learning Algorithm (PLA) for binary classification. Any one variant of the PLA/Pocket will do. Python is the recommended language (if you do not know Python, please let me know ASAP). Please do not use an off-the-shelf implementation of PLA from any package (for example, do NOT use perceptron_learning.Perceptron). The goal is for students to develop a solid understanding of the working of the PLA as a vehicle for demonstration of the connection between training error and test error. We will use the PLA as a running example in illustrating many of the recurring themes in machine learning, namely hypothesis set complexity, VC dimension, regularization, etc. Choose appropriate (typically randomly generated) training and test data sets of two dimensions (points on a plane). Use 50 data points for training (25 for each class) and 30 for testing. Consider two separate cases for the training data (the test data set remains the same for the two different training sets):

- Training data points are linearly separable, and
- Training data points are not linearly separable.

Please submit on Canvas a single doc/docx/pdf file (no other file type, please) containing the source code, the training and test data, and a numbered list providing brief notes on:

1. how (that is, following what logic) you generated the data points (training and test),
2. whether the training data points are linearly separable,
3. whether the test points are linearly separable,
4. your initial choice of the weights and constants,
5. the final solution equation of the line (decision boundary),
6. the total number of weight vector updates that your algorithm made,
7. the total number of iterations made over the training set (an iteration involves checking each of the training points and determining its class as given by the line), and
8. the final misclassification error, if any (expressed as a percentage), on the training data as well as on the test data.

Also, re-run your code (for both the above cases: linearly separable training set and not linearly separable training set) by varying the following and describe the effect, if any, that each of the three variations had on the final solution equation (test each of the three variations separately and summarize your results in a numbered list):

I. the initial choice of the weights,
II. the initial choice of the step size constant (we call it c or η in the notes),
III. the order in which you consider the points in the training set.

Add notes for any other special issues/techniques that you think might be important in your implementation.

Important: This is a group project, with up to four students per group. Please form your own group by interacting with fellow students. While working in groups is highly recommended, it is not mandatory (individual submission is OK). A group must not mix graduate and undergraduate

students. Please write all group-members' names at the top of the very first page of the submission. Feel free to discuss your project with classmates not in your group but your final submission should be your group's own work. Only one group-member should upload the file on Canvas; the other members of the group should upload nothing.

Optional: Feel free to provide an (optional) acknowledgments section at the end of your report where you list any sources (books, research articles, web sites, individuals, etc.) that you may have drawn upon in your work. Do not acknowledge this course's material (textbook, notes, etc.) or your own group's members.

Mandatory: Please add the following statement at the end of your file and hand-write or type your full name (or names in the case of groups): "I/we certify that this report is my/our own, independent work and that it does not plagiarize, in part or in full, any other work."