

EE6530 Neural Networks Computer Assignment #1, brought to you by Prof. YWLi

Printed Report Due March 19 14:20 (in class). No late homework, please.

Task description: This homework consists of two parts. In the 1st part, you are asked to run a matlab script (Hw1_LinBinPerc_DataGen.m) to generate a random set of data with binary labels so that the data are known to be linearly separable. Then, you are supposed to pretend that the linear weights are unknown. You are asked to complete the starter code so as to obtain a set of weights that separate the data with the label 0 from those with the label 1.

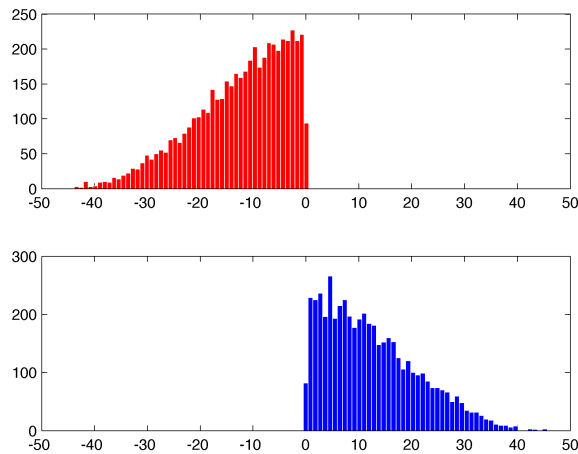
In the 2nd part, you are given an excel file (NN_RealDataForHW1.csv) that contains grade information from a real NTHU course (obfuscated to hide the identity of students). Each column contains the grades for one activity, and you are asked to discover how these columns were linearly combined to determine whether a student has passed (P) or failed (F). You can use the code you develop for Part I. You should report on the final weights and bias in your perceptron.

For part II, the following MATLAB functions may be useful: `csvread()`, `importdata()`.

Report guideline:

Your report should be no longer than 5 pages, including figures and texts. The report should include the following components (中英文不拘):

- A brief statement of the objective of this homework
- Clear description of the methods, using pseudo-codes, equations, or graphic illustration if you will. Completely doing it by texts would be less than ideal.
- Results with appropriate visualization. For instance, one way to visualize the results of Part I would be a histogram like the one shown below,



where the red plot (top panel) shows the distribution of the output $w^T x + b$ for the data that belong to class “0”, and blue (bottom panel) otherwise.

- Analysis of the performance, and discussion on how the performance varies if problems are solved in slightly different ways. Comparison against other methods can be included, too.
 - To analyze the performance, you can basically count the number of datapoints that are *misclassified* by your perceptron.
- A list of references, if any.

Following is a list of issues you might choose to investigate for this homework.

1. In part 1, does your final weight vector approximate the weights used for data generation (up to a scaling factor)?
2. Does the perceptron successfully (that is, with 100% accuracy) separate the data into two classes?
3. If not, does it help to repeatedly feed the whole set of data to your algorithm? (such as done in the for loop line 25)
4. In the starter code (Hw1_starter.m), line 16-17, the data are randomly sorted. What is the purpose of this, or does it matter?
5. In part 2, does the gender information help predicting whether a student passed or failed?