

# Homework1 / Linear Binary Perceptron

李豪章 (HW-Lee) ID 103061527

## Overview

---

The homework consists of two parts: 1) Implement a perceptron which is able to tell data with positive labels from those with negative labels; and 2) Calculate the weights, which can be simply regarded as correlation, between students features (namely gender, activities, tests, and participation) in a course and final consequences (pass or fail) of them.

In the first part, a matlab script `Hw1_LinBinPerc_DataGen.m` is needed to generate a random set of data with binary labels so that the data are known to be linearly separable. Then, the task is to pretend that the linear weight are unknown and train the perceptron with some algorithm iteratively.

In the second part, there is a given 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 an activity, and the task is to discover how these columns were linearly combined to determine whether a student has passed (P) or failed (F). The code developed for the first part can be used in this part, and the final weights and bias in the trained perceptron should be shown in the report.

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

## Implementation

---

### 1. Functions

- **`Hw1PerceptronClassifier.train(X, y)`**

Return a reference of a perceptron classifier trained with training set  $X$  and corresponded ground truths  $y$ .

- **`Hw1PerceptronClassifier.predict(X)`**

Return a binary vector contains elements with value of either -1 or 1, which refers to predicted results with unknown testing set  $X$ .

- **`Hw1PerceptronClassifier.visualInfo()`**

Show the property of a classifier in a visual way, including parameters  $w$ ,  $b$ , and all  $w$ 's and  $b$ 's during training process.

## 2. Process

- **Generate training set  $X$  and corresponded ground truths  $y$**

Combine two matrices `class0` and `class1` into a matrix vertically, and generate a vector which contains the 'class label' indicating which class each row instance belongs to. (-1 refers to `class0` and 1 to `class1`)

- **Initialize parameters of the perceptron classifier**

In default, the parameters, namely  $w$  and  $b$ , will be zero vector and zero respectively.

- **Iterate algorithm with  $N$  times**

Predict labels with current parameters :  $z = Xw + b$

Find the set consisting of all

incorrectly predicted instances :  $errIdx = \{i | z_i \neq y_i \ \forall i = 1, \dots, N\}$

Update parameters : 
$$\begin{pmatrix} w_{\text{new}} \\ b_{\text{new}} \end{pmatrix} = \begin{pmatrix} w \\ b \end{pmatrix} + \sum_{i \in errIdx} y_i \begin{pmatrix} x^{(i)} \\ 1 \end{pmatrix}$$

## Results

---

## Discussion

---

1. In the first part, does the 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`, line16-17, the data are randomly sorted. What is the purpose of this, or does it matter?
5. In the second part, does the gender information help predicting whether a student passed or failed?