

## Machine Learning HW2

### Logistics

Due date: 10/9 (Sun) 23:59

Submission

- Via LMS (no email submission)

- **TWO FILES**

1. Zip of your code. (compress your code into **one file**)
2. Report. (any format)

### Problem Description: Logistic Regression

The main purpose of this assignment is to code a binary classifier using logistic regression. You are given two files (train.csv and test.csv) for a dataset which is characterized by two features, denoted by  $x_1$  and  $x_2$ , and each data point belongs to class 0 or 1 denoted by  $y$ . Your task is 1) to train a classifier using train.csv, and 2) to classify the data in test.csv using the trained model:

#### Task 1. Train a classifier with train.csv.

The file train.csv has three columns,  $x_1$ ,  $x_2$  and  $y$ . The first two columns indicate the two features of the data, and the last column  $y$  means the label. There are no restrictions on the decision boundary model; you can use any, but do not forget to wrap it in the sigmoid. To train your classifier, use a gradient descent algorithm as you did in the last assignment.

As a result of this task, you must give 1) the parameters of your model, 2) the cost.

#### Task 2. Classify the data in test.csv using the trained model.

The file test.csv contains a set of data in the same format as train.csv. Use the trained classifier in Task 1 to classify the given dataset in test.csv.

This task requires measuring two metrics: 1) the cost and 2) the accuracy which is defined as the number of correct classifications for the total number of data points.

### Evaluation Policy (10 pts in total)

Score (10pts) = Report (7pts) + Implementation (3pts)

Penalties

1. Unable to build or run → Implementation = 0
2. **Plagiarism → Score = 0 (will affect your overall grade)**
3. Late Submission → -1 for each day of delay.
4. Missing files and wrong formats (not compressed code, ...) → -1

### Report

Any format, no template. But the followings must be included:

1. Program description (comments on important code lines)
2. How to run (so that I can test your program)
3. A set of snapshots (of progress, final result, and etc.)

#### 4. Results

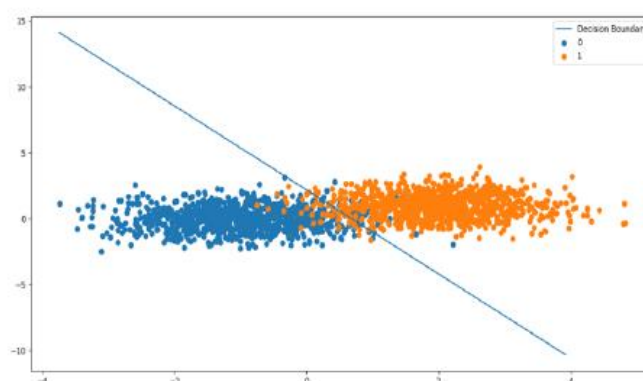
- A. The parameters of your classifier (Below is an example)

|       | a     | b     | c |
|-------|-------|-------|---|
| Task1 | 0.001 | -0.02 | 0 |

- B. Total Cost for each task (Below is an example)

|       | Cost  |
|-------|-------|
| Task1 | 0.001 |
| Task2 | 0.002 |

- C. Plots of the classification results. (Below is an example)



- D. Accuracy: For example) 90%.

5. Conclusion.

\* The top 5 students with high accuracy will receive extra points.