CMPE 462 - Spring 2021
Assignment 1

Introduction

This assignment consists of 2 parts.
The first one is about percentrons and the second part is about linear regression.

Part1

In Part 1 of this assignment, you will implement **Perceptron Learning Algorithm** (PLA) from scratch (details on page 15 of Lec02-Perceptron notes) and work on 2D data.

Given the target separating function f as below:

y = -3x+1

you will generate 2D data points (x,y) in two classes {0,1} defined as:

if y < -3x+1, then c=0 if y > -3x+1, then c=1

You will run your PLA in order to classify the generated points and plot the resulting decision boundary.

The steps of Part1 are below:

- Step1: Generate total 50 points in two classes and apply PLA.
- Step2: Generate total 100 points in two classes and apply PLA.
- Step3: Generate total 5000 points in two classes and apply PLA.

For each step, plot the target separating function f (in green) and the generated points (use red for class 0, blue for class 1) and your decision boundary (in purple). Use labels for both axis and the target separating function f and your decision boundary. Save your plot as a png file named as part1_stepn.png, example part1_step1.png.

You can use mathplot library. But special functions or libraries for decision boundary plotting is forbidden.

In your project report, include results of your runs for each step, 50,100 and 5000. Place the plots and discuss over them. Compare your resulting boundaries and the target function. Also discuss the number of iterations you need for each step.

Part2

In Part 2 of this assignment, you will implement **Multiple Linear Regression** from scratch (closed form solution is on page 41 of Lec03-LinearRegression notes). In Linear Regression, the relation between an input variable (1D) and an outcome (1D) is modelled. In Multiple Linear Regression, the input variable is multidimensional and the modelled relation is as below:

$y=c_1 * x_1 + ... + c_n * x_n + e$

where y is the outcome (dependent variable), x_i are the input parameters (independent variables), c_i are the coefficients and e is the error term.

For Part2, two input datasets (DS1 and DS2) will be given to you. Each dataset file is a csv file with no header, no index. Each row represents a different sample. Each column in a sample represents a variable. The last column keeps the dependent variable. Below is an example csv file. There are 3 lines, it means that there are 3 samples. There are 6 columns in each line, it means there are 5 independent variables (first 5 columns in a row) and 1 dependent variable (last column). In this example all dependent variables are 6.

```
1, 2, 3, 4, 5, 6
1, 2, 3, 4, 5, 6
1, 2, 3, 4, 5, 6
```

The steps of Part2 are below:

- Step1: Run Multiple Linear Regression on DS1.
- Step2: Run Multiple Linear Regression on DS2.
- Step3: Implement Multiple Linear Regression with l_2 regularization (details on page 50 of Lec03-LinearRegression notes). Run Multiple Linear Regression with l_2 on DS2.

For each step, plot the loss over iterations. Save your plot as a png file named as part2_stepn.png, example part2_step1.png. If you implement closed form solution, you can print the time taken for your run in miliseconds as: "Time to complete stepn: XX msec", example "Time to complete step1: 550 msec"

In your project report, include results of your runs for each step. Place loss over iterations plots and discuss over them. Also discuss the number of iterations you need for each step. If you implement closed form solution, you can discuss the durations for your runs.

Base Environment

You will be implementing your code with Python 3.6.
You need to create a python virtual environment with Anaconda for your project. After installing Anaconda, a base environment can be created with below commands:

```
conda create -n 462assignment python=3.6
conda activate 462assignment
```

While you keep working on your models, you will need to import additional libraries. List these libraries in a requirements.txt file. State any special versions if needed. A sample requirements file can be as below:

```
scikit-learn >= 0.22.2
scipy
pandas
sentencepiece==0.1.91
```

For grading, we will load your requirements with the command below:

```
python3 -m pip install -r requirements.txt
```

Before submission, test your code on a clear new conda environment by installing additional libraries from your requirements file. Because, there will be penalty if your code doesn't run like this.

Grading Details

The assignment will be graded over 100 points. You will be graded for your code and report.

- 10 points for report
- 90 points for code
  - 45 points for Part1
    - 15 points for step 1
    - 15 points for step 2
    - 15 points for step 3
  - 45 points for Part2
    - 15 points for step 1
    - 15 points for step 2
    - 15 points for step 3

We will run your code on a clear new conda environment. First we will load your requirements.txt file. Then we will test your code with below commands:

- Part1

```
python3 assignment1.py part1 step1
python3 assignment1.py part1 step2
python3 assignment1.py part1 step3
```

Consider first command, you will generate 50 2D data points, run PLA, generate your plot and save it as png file.

- Part2

```
python3 assignment1.py part2 step1
python3 assignment1.py part2 step2
python3 assignment1.py part2 step3
```

Consider first command, you will run linear regression on DS1, generate your plot and save it as png file.

Submission Details

This is an individual assignment. Your code should be original. Any similarity between submitted assignments or to a source from the web will be accepted as cheating.

If you have any further questions, send an e-mail to the course page on Piazza.

- The deadline for submitting Assignment 1 is **April 27, 2021 - 23:59**.
- There will be 2 submissions open for this assignment.
- Submission 1:
  - You should submit 3 items:
    - your Python script, assignment1.py
    - your requirements.txt file, blank file if no additional library is needed
    - your assigment report in pdf, 462_assignment1_<studentid>_report.pdf, example 462_assignment1_20181123456_report.pdf
  - You should compress all submission items in a zip file with name as 462_assignment1_<studentid>.zip, example 462_assignment1_20181123456.zip
  - The zip will be submitted on Moodle.
- Submission 2:
  - You should also submit your reports in Turnitin submission on Moodle.