TRINITY COLLEGE DUBLIN

School of Computer Science and Statistics

Week 1 Assignments CS7CS4/CSU44061 Machine Learning

Rules of the game:

- **Its ok to discuss with others, but we ask that you do not show any code you write to others. You must write answers in your own words and write code entirely yourself. All submissions will be checked for plagiarism.**
- **Reports must be typed (no handwritten answers please) and submitted as a pdf on Blackboard.**
- **When your answer includes a plot be sure to (i) label the axes, (ii) make sure all the text (including axes labels/ticks) is large enough to be clearly legible and (iii) explain in text what the plot shows.**
- **Include the cource of code written for the assignment as an appendix in your submitted pdf report and also include the code as a separate zip file (so we can easily execute it). Programs should be running code written in Python. Keep code brief and clean with meaningful variable names etc.**

Downloading Dataset

- Download the assignment dataset from `https://www.scss.tcd.ie/Doug.Leith/CSU44061/week1.php`. Important: You must fetch your own copy of the dataset, do not use the dataset downloaded by someone else.
- Please cut and paste the first line of the data file (which begins with a #) and include in your submission as it identifies your dataset.
- The data file consists of one column of data (plus the first header line).

.

Assignment

(a) Write a short python program that (i) reads in the data you downloaded, (ii) normalises it and then (iii) uses gradient descent to train a linear regression model. Do not use sklearn or other helper packages for (ii) and (iii), you should fully implement the data normalisation and gradient descent code yourself in vanilla python. To read in the data you can use, for example, pandas:

```
import numpy as np
import pandas as pd
df = pd.read_csv("week1.csv",comment='#')
print(df.head())
X=np.array(df.iloc[:,0]); X=X.reshape(-1, 1)
y=np.array(df.iloc[:,1]); y=y.reshape(-1, 1)
```

(b) Using your program train a linear regression model on the downloaded data.

(i) Try a range of learning rates α in your gradient descent algorithm, e.g. 0.001, 0.01, 0.1, and plot how the cost function $J(\theta)$ changes over time. Discuss. Hint: for v small learning rates the cost function should decrease v slowly, for v large learning rates the cost function may not converge.

(ii) Report the parameter values of the linear regression model after it has been trained on your downloaded data and plot the model predictions together with the training data.

(iii) Also report the value of the cost function for the trained model. Compare with the value of the cost function for a baseline prediction that always predicts a constant value regardless of the input value (pick a reasonable prediction value based on inspection of your data - explain your choice).

(iv) Now use sklearn to train a linear regression model on your data. Plot the predictions of the sklearn model and the model you trained above together with the training data and compare. How do the parameters of the two trained models compare.