Assignment 1
Linear Regression

1 Overview

In this assignment, you are required to implement the linear regression explained in the lectures with only one variable. The data provided is the statistic of a restaurant chain comparing the number of customers with the total profit.

2 Plotting The Data

Before starting the linear regression implementation it will be of benefit to understand the data by visualizing it, the best visualization technique for this type of data is the scatter plot. Useful libraries that can be used are :

1. Pandas, for reading and handling the dataset.
2. matplotlib, for plotting and visualizing the dataset.

3 Gradient Descent

As explained in the lecture, gradient descent is used to fit the linear regression parameters to the dataset, in this part you are required to fit the linear regression model to our dataset.

3.1 Update Equations

The aim of the linear regression is to minimize the cost function.

$$\mathcal{J}(\theta) = \frac{1}{M}\sum_{i=1}^{M}(\hat{y}_i - y_i)^2 \tag{1}$$

Where the linear model is represented by

$$\hat{\boldsymbol{y}} = \theta^T \boldsymbol{x} = \theta_1.\boldsymbol{x} + \theta_0 \tag{2}$$

So the equation can be expanded to be

$$\mathcal{J}(y, x, \theta) = \frac{1}{M}\sum_{i=1}^{M}\left((\theta_1 x_i + \theta_0) - y_i\right)^2 \tag{3}$$

Recalling that the parameters of your model are the θ_j , these are the values that should be adjusted to minimize the cost of $J(\theta)$.
One way to do so is to use the batch gradient descent algorithm, where each iteration performs an update

$$\theta_0 = \theta_0 - \alpha \frac{\partial \mathcal{J}(y, x, \theta)}{\partial \theta_0} \tag{4}$$

$$\theta_1 = \theta_1 - \alpha \frac{\partial \mathcal{J}(y, x, \theta)}{\partial \theta_1} \tag{5}$$

Where

$$\frac{\partial \mathcal{J}(y, x, \theta)}{\partial \theta_0} = -\frac{1}{M} \sum_{i=1}^{M} 2\big((\theta_1 x_i + \theta_0) - y_i\big) \tag{6}$$

$$\frac{\partial \mathcal{J}(y, x, \theta)}{\partial \theta_1} = -\frac{1}{M} \sum_{i=1}^{M} 2x_i\big((\theta_1 x_i + \theta_0) - y_i\big) \tag{7}$$

With each step, the θ value comes closer to the optimal value that will achieve the lowest cost $J(\theta)$.

4 Implementation details

In this section, you will be provided with some helpful notes to help you with your code

1. The number of iterations should be = 1500
2. Alpha (α) in the update step should be = 0.01
3. Parameters (θ) are all initialized to 0

4.1 Computing the cost $J(\theta)$

As you apply gradient descent to minimize the cost function, it will be helpful to monitor the convergence by computing the cost, in this section, you will implement the function to calculate J(θ) so you can check the convergence of your gradient descent implementation.
You are required to have a function called computeCost(x, y, theta) where it takes as input:

1. Data points X
2. Data pints Y
3. Theta (linear regression parameters)

And return:

1. the J variable containing the cost $J(\theta)$

After that, you are required to use the function computeCost to calculate the J() with your parameters initialized as zeros, you should expect to have a cost equal to 32.07

4.2 Implementing Gradient Descent

Next, you will implement the gradient descent, as your program you have to make sure you understand what you are trying to optimize and what is being updated. That $J(\theta)$ is being minimized by changing the value of θ not by changing X and Y.

You should have a function called gradientDescent(X, y, theta, alpha, num_iters) where it takes as an input :

1. Data points X
2. Data pints Y
3. Theta (linear regression parameters)
4. Alpha (the learning rate)
5. Num_iters (number of iterations)

You will also need to keep track of each cost update in a variable called J_history , which get filled the cost in each iteration
"The cost should be calculated using your function computeCost"

The function should in the end return the updated theta and the J_history

4.3 Final Steps

Finally , after using both functions to fit your linear regression model you should

1. Use the model it to predict the profit on days where the customers are 35,000 and 70,000
2. Plot the J_history

5 Notes

You will be asked to predict the profit on any given number of customers

5.1 Deliverables

- Your well commented code.
- A report showing your work.

5.2 Further Notes

- You should use Python for your implementation.
- You can use Jupyter or Colab notebooks.
- Copied assignments will be given zero, both the copier and the copied from.
- You are required to work individually.

6 Bonus

As a bonus, you can implement the multivariate linear regression, it will be the same steps as the one variable linear regression with few differences in the implementation of each function. A different dataset will be included for the bonus part.
i.e. Number of Thetas will be 3 instead of 2.

Good Luck