Read Me

Files:

Question 1 : Normal Equation

<https://www.kaggle.com/muditr97/assignment-1-normal-equation?scriptVersionId=11518225>

Question 1 : Gradient Descent <https://www.kaggle.com/muditr97/assignment-1?scriptVersionId=11518227>

Data Set: Housing Price Data set Provided

The Assignment where done on Jupyter Notebook  
FileName for Normal Equation : Assignment\_1\_NormalEquation  
FileName for Gradient Descent : Assignment\_1\_GradientDescent

Open the file which ( is public ) add this to jupyter Notebook, Data set Name : “ Housing Price data set “ and run the code.  
It will produce the error for Normal Equation(mean squared error) and for Gradient Descent Algorithm(mean squared error)

Libraries used :

1. NumPy
2. SciPy
3. Pandas
4. Os
5. Sklearn

Data Set Highlights:

1. 546 Training Examples
2. 12 Features Independent
3. Price as dependent Variables

Analysis for Assignment 1 for Linear Regression

The primary technique that we have to use in this assignment was Linear Regression, we have used Normal Equation and Gradient Descent techniques in order to solve this

We have considered the hypothesis in the linear variables and with x1,x2,..., xn considering there are n number of features. We have used parameters θ0,θ1 ,......,θn for the hypothesis.

The general approach that we have considered, in the Normal Equation technique is that we have to draw a curve using the features and parameters, for the independent variable and predict the dependent variable. And we have to minimize the actual value difference with the predicted value of the hypothesis. Thus we have considered the following:

Hypothesis: hθ( xi) = θ0 + θ1 x1+..... + θnxn

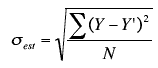
Cost Function : J(θ0,θ1 ,......,θn) = (Actual value - predicted value)  
**{since we don’t have to do anything with the sign of the value but only the magnitude is what we have to minimize, we will take the square of this and multiplying with 1/m where m is the training examples won’t have any effect)}.**

Thus,

Cost Function : J(θ0,θ1 ,......,θn) = (1/2m)  i = 0 to m (hθ( xi) - yi)2

Now, after this we know that we have to minimize the cost thus we will take partial derivative with respect to the parameters of this and equate to zero.

This will give us the values of the parameters and getting the values of the parameters we will be able to get the hypothesis and thus putting the independent variable values we will be able to predict the dependent variables.

Error:where y is the actual score and y’ is the predicted score and n is the number of training examples.

Above we have described how we have proceeded and the results were

Q1 :

We have applied feature scaling to the independent variable with x - 2007

hθ( x) = θ0 + θ1 x1 Where θ0 = 58.36 and θ1= 3.456

Expected Revenue in 2019 = Rs. 99.84 billion

Error = 15.43

Q2: In this question we have first considered the marks of SOC as independent variable and HUR as dependent and vice versa in order to predict the marks in various subjects.

Q3: In this questions we been given a equation and we need to predict the parameters but the equation needs to be in a certain form in order to apply the normal equation, so we have preprocessed the given equation by applying log and the comparing this with the generalised equation and prediction.

Q4: In this we have considered the hypothesis and the parabolic equation.

hθ( x) = θ0 + θ1 x1 + θ2 x12

And we solved for the parameters and found the equation.

Q5:

In this we have to use the Vector form for the weights/ parameters, Vector form for the features including the x0=1 feature, the design Matrix formed by the transpose of the feature matrices for training examples.  
Also, In the other part of the question, we implemented gradient descent algorithm using repetitive approach for over 1000 iterations and keeping the learning rate 0.4

Observations:

1.Feature Scaling can really increase the efficiency of the gradient descent algorithms.

2. Feature scaling reduces the number of iterations required for the algorithm to work significantly.

3. Normal equation gives the not invertible matrix but the gradient descent always work for the regular fitting or convex curves

4. Value of the slope reduces as we move towards the local optima and doesn’t change any further once it touches the local optima (or very close to the local optima). As the slope becomes zero in the curve on the point where the value is minimum.

5. The Mean squared error for this particular example is lower for the Gradient Descent Algorithm than the Normal Equation