## Programming Assignment 1

## Linear Regression using numpy

We will be using ipython notebooks for most of our assignments. You can refer <u>http://cs231n.github.io/ipython-tutorial/</u> for installing the required software for working with ipython notebooks and getting started with it. We will be using python **2.7**, so, please make sure that the version you are using aligns with the that.

This assignment consist of a python notebook that depicts the outline of using linear regression to learn a linear mapping function to map input data to the output. The function should be learned with the objective to minimize the Mean Square Error (MSE) loss between the predicted output and the ground truth data. The dataset for training and testing is being generated randomly using a fixed seed. You have to fill all the missing parts which is indicated using question marks (?) and hints wherever required (Use numpy operations wherever possible). Please don't declare any function other than given in the notebook. The functions have their usual meanings and is described below in brief. Some of the functions are partially implemented and some of them are left for you to complete it.

- **generate\_dataset()**: It generates the dataset on the fly, to be used for training and testing purposes.
- **LinearRegression::forward()**: It computes the output using parameters of the network and the input data. This is also called a forward pass of the linear regression network.
- LinearRegression::backward(): It computes the gradients for every parameter of the network and update the corresponding parameter using gradient descent step. This is also called a backward pass as a prediction has been made and now the network is updated accordingly, so as to minimize the loss function.
- **MSELoss()**: It computes the Mean Squared Loss between the predicted output and ground truth data. The formula for calculating the same has been discussed in the lecture.

The algorithm to implement is mentioned in the comments, you have to use **only** numpy and basic python functions and **no** other imports are allowed.

**Submission:** You have to submit a ipython notebook with the **.ipynb** extension. Please make sure that all the cells are running. Please use the given **Assignment\_1.ipynb** as a starting point by uploading it to the jupyter notebook. You can complete the assignment gradually by completing every cell and execute it in the order given. After completing the assignment, please make sure that you are able to:

- visualize a plot between the loss function and training epochs.
- get training and test loss.
- visualize a learned function on the test data. This will be a plot where the test data will be plotted as blue points and the red line indicates the learned function.

Outline of the results mentioned above has been implemented in the notebook, you just have to complete the blanks. Please submit the completed file named as 'Assignment1\_<Roll No>.ipynb'.