Assignment 1

Q1. To obtain the the least squares solution W_{ML} to the regression problem using the analytical solution.

The Least squares solution to the regression problem using the Analytical solution :

$$W_{ML} = (XX^T)^{-1}Xy$$

In the above equation multiplication of X and Y is not possible due to mismatch of the order of multiplication X is (1000,2) and Y is (1000,1). The equation is rewritten in the form of Standard least squares solution to obtain W.

$$W_{ML} = (X^TX)^{-1}X^Ty$$

To implement the above equation:

1. X is a 2D numpy array of dimension 1000 x 2 which contains first two columns of the FMLA1Q1Data_train.csv file corresponding to the features.
2. Y is a 2D numpy array of dimension 1000 x 1 which contains the third column of the FMLA1Q1Data_train.csv file which corresponds to the labels.
3. W_{ML} is realized using numpy functionalities such as dot, transpose, inverse and matrix multiplication operations.
    a. The numerator term X^Ty is computed by transpose of X followed by a dot operation with y.
    b. The denominator term X^TX is is computed by transpose of X followed by a dot operation with X.
    c. The inverse of the denominator term is computed by numpy inverse function(numpy.linalg.inv).
    d. W_{ML} is obtained by the matrix multiplication of inverse of denominator term and the numerator term.
4. The Y values are predicted using the least squares solution and the SSE is computed for the train set.

Q2. The gradient descent algorithm with suitable step size to solve the least squares algorithms and to plot $\|w_t - w_{ML}\|^2$ as a function of t.

1. W_{ML} is computed by following steps as mentioned in Question 1.
2. Since we have two features x1 and x2, then y_pred = w0 + w1x1 + w2x2.
3. w0 is the bias (i.e., the intercept) and w1, w2 is the weights for features respectively.
4. The weights W are randomly initialized (set to high values) which is sampled from a uniform distribution in range (50, 100). The bias is initially set to 0.
5. In each iteration:
    a. Predict y based on current wi and b
    b. Compute the error
    c. Compute the gradients
    d. Update of parameters

6. It is observed that based on the learning parameters, the weights gradually reduce towards the optimum value which minimizes the error function $||\mathbf{X}^T\mathbf{W}-\mathbf{Y}||^2$
7. At every time instance t, the value $||\mathbf{W}_t-\mathbf{W}_{ML}||^2$ is computed and it depicts a gradual decrease as weights w approaches towards the Least squares solution.
8. From the above plots, it is clear that smaller the learning parameter, the gradient descent algorithm would require more no of iterations to reach the optimal W*.

Q3. Stochastic gradient descent:

1. W_{ML} is computed by following steps as mentioned in Question 1.
2. Since we have two features x1 and x2, then y_pred = w0 + w1x1 + w2x2.
3. w0 is the bias (i.e., the intercept) and w1, w2 is the weights for features respectively.
4. The weights W are randomly initialized (set to high values) which is sampled from a uniform distribution in range (50, 100). The bias is initially set to 0.
5. In each iteration:
    a. Sample 100 data points at random from the dataset
    b. Predict y based on current w_i and b
    c. Compute the error
    d. Compute the gradients
    e. Update of parameters
    f. Append the current w_i to the W_t list (The array which stores the weights computed during each iteration, it is used to compute the final weights by averaging over all w_i's.
6. It is observed that the weights at each iteration updates towards the optimal value which results in the error function $||\mathbf{X}^T\mathbf{W}-\mathbf{Y}||^2$ approaching the minimum value, even though its trained on batches of 100 samples.
7. The final resulting weights w1 and w2 is obtained by computing the mean of each column (i.e., each component of W_t), resulting in an array of the average values.
8. The final resulting weights obtained by stochastic gradient descent after T iterations is almost similar to the weights obtained by the standard gradient descent algorithm. It indicates that the stochastic gradient descent's performance is at par with the standard gradient descent algorithm's performance. Stochastic gradient descent would be a suitable choice if the compute power is limited.

Q4. Gradient descent algorithm for ridge regression

1. The Weights W_{ML} is computed by following steps as mentioned in Question1.
2. The Learning parameter is set to 0.01, based on previous tasks 0.01 value helps in reaching optimal weights at a faster rate.
3. The regularization parameter is searched in the space (10^{-4} - 10^4) using Kfold cross validation along with gradient descent technique.
4. The optimal lambda is the one which produces minimal validation error.
5. For the optimal value of regularization parameter, the weights and the bias term is computed using gradient descent.
6. Predictions are computed on the test data from FMLA1Q1Data_test.csv .

7. The predictions made using the (weights + bias) computed through ridge regression, result in lower error compared to those computed with the weights obtained from the least squares solution. This indicates that the weights from ridge regression, denoted as $\mathbf{W}_R$, are more tuned towards the optimal weights than the weights from least squares regression $\mathbf{W}_{LS}$. The bias computed in ridge regression also plays a key role in minimizing the error.
8. The regularization term mitigates the effects of multicollinearity and stabilizes the estimates by penalizing weights making the solution more stable and less sensitive to correlations among features.
9. Ridge regression produces a model with low variance compared to the Least squares fit. This approach of penalizing weights for terms which doesn't contribute to the prediction/ involved in multicollinearity prevents overfitting and model generalizes better on the unseen test data. This results in the test error being lower compared to the Least squares fit.

Q5. Kernel Regression

1. The pair plots depict a non linear relationship between the features and the target variable.
2. Gaussian kernel is a suitable to model the non linear relationship as it maps the features into an infinite-dimensional space, where the relationship between the features and the labels/ target may become linear. There could possibly be a hyperplane which linearly separates the data points in higher dimension, which is equivalent to learning a non linear function in the lower dimension. Gaussian kernel gives higher weight to points that are close to the target point and less weight to points that are farther away. It implies a smoothness assumption, meaning that points closer to each other in the input space will have similar outputs.
3. The Gaussian Kernel is chosen to perform the Kernel regression.
4. The Kernel regression using the Gaussian kernel performs better than the standard least squares regression and SSE is much lower compared to SSE of the least squares regression.
5. The parameters lambda and sigma which is utilized during the computation of the kernel values and the corresponding alpha is finalized by cross validation. The final values of lambda and sigma is utilized in the code.
6. The Gaussian kernel is a non linear model, it is capable of modelling complex relationships between input features and the target variables. The kernel operates in the higher dimension space without explicitly computing the features (kernel trick), exploring the possibility of a hyperplane which linearly separates the data. The kernel regression is robust towards multicollinearity with the regularization term introduced providing more control on the model's performance with fine tuning the term. The regularization term balances the bias – variance trade off, the kernel regression model tries to achieve lower variance by trading off the bias to certain extent which would help in generalization on unseen data.
7. The least squares regression is a linear model, it will not capture the true relationship if the relationship of data/ features is non linear with the target output. The OLS is more prone to overfitting on the training data when multicollinearity exists or if data has noise. It does not involve a regularization term to reduce the

effects to multicollinearity. The OLS model tries to minimize the bias resulting in high variance, leading to overfitting.

8. For the given dataset, Kernel regression using the Gaussian kernel is better than the standard least squares regression.