Anshul Shivhare MTech AI 17828

Pattern Recognition and Neural Networks

Homework 2

Problem 1

2-class classification problem with two dimensional feature vector.
Samples of size 10, 50, 100, 500, 1000. Comparing a linear least squares and logistic regression on the data, with 2000 training sample and 1000 test samples. so here we have 3 Sub problems.The performance of both the classifiers were observed by testing against the full test set.
Linear least square is trained with LMS algorithm. Logistic regression uses the sigmoid function.

Sub-problem 1

2D Data, features independent Gamma distributed.

Class-I: Gamma(shape=0.5, scale=1);

Class-II: Gamma(shape=2.0, scale=2).

epoch 250, learning rate=0.1

I have also calculated the F-score for both the methods. it is a measure which involves both precision and recall. Precision and recall are combined together into the F-score to get the single numerical measurement of a system's performance .

0-1 Loss function was used for these methods.In implementation of Linear regression, first augmentation is done with column of vector all 1, To calculate 'W' as $W = (A^TA)^{-1}A^Ty$, where ‘A’ is augmented training data and a data point is stored in row wise also ‘y’ represent the class label with y = 1 as class 1 and y = -1 as class 2.

Number of examples in training data	Accuracy of Linear least square	Accuracy of Logistic regression
10	0.723	0.6207
50	0.903	0.806
100	0.889	0.928
500	0.8976	0.945
1000	0.912	0.94

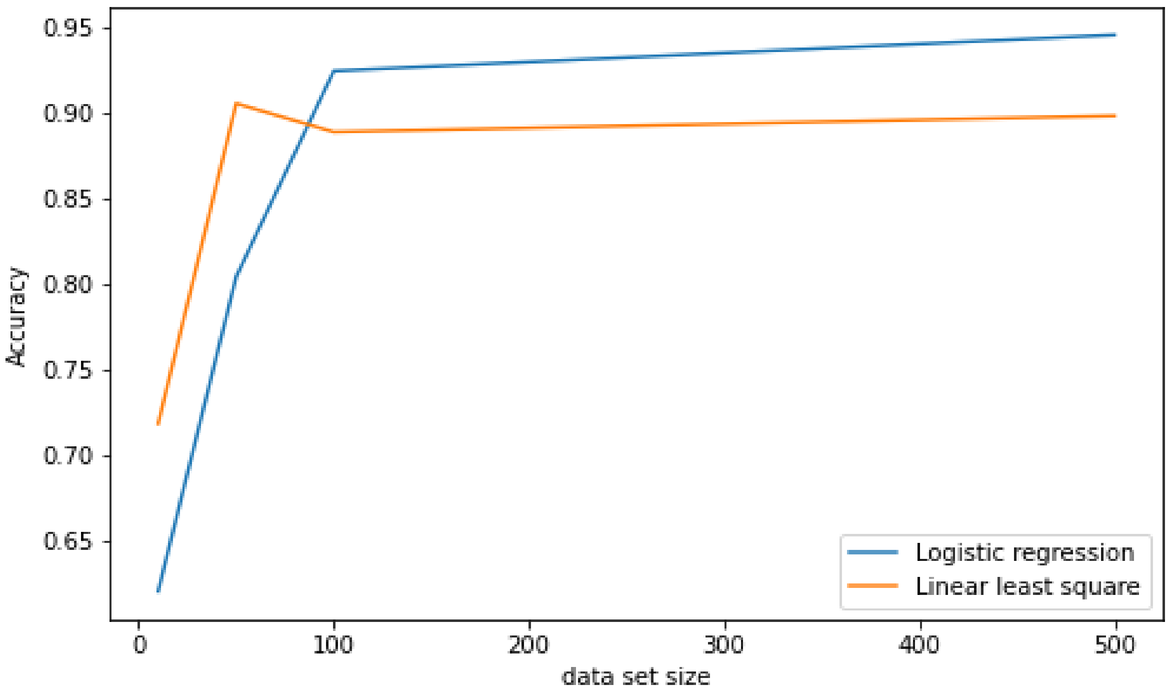
For data in Gamma train.txt and Gamma test.txt

Observation :
For less sample size, Linear least square classifier has higher accuracy but we can neglect it as for small sample size the accuracy can vary a lot . For large sample size, Logistic regression works better. Also as we increase the sample size it is intuitively clear that accuracy increases ,as it leads to better estimation of parameters.

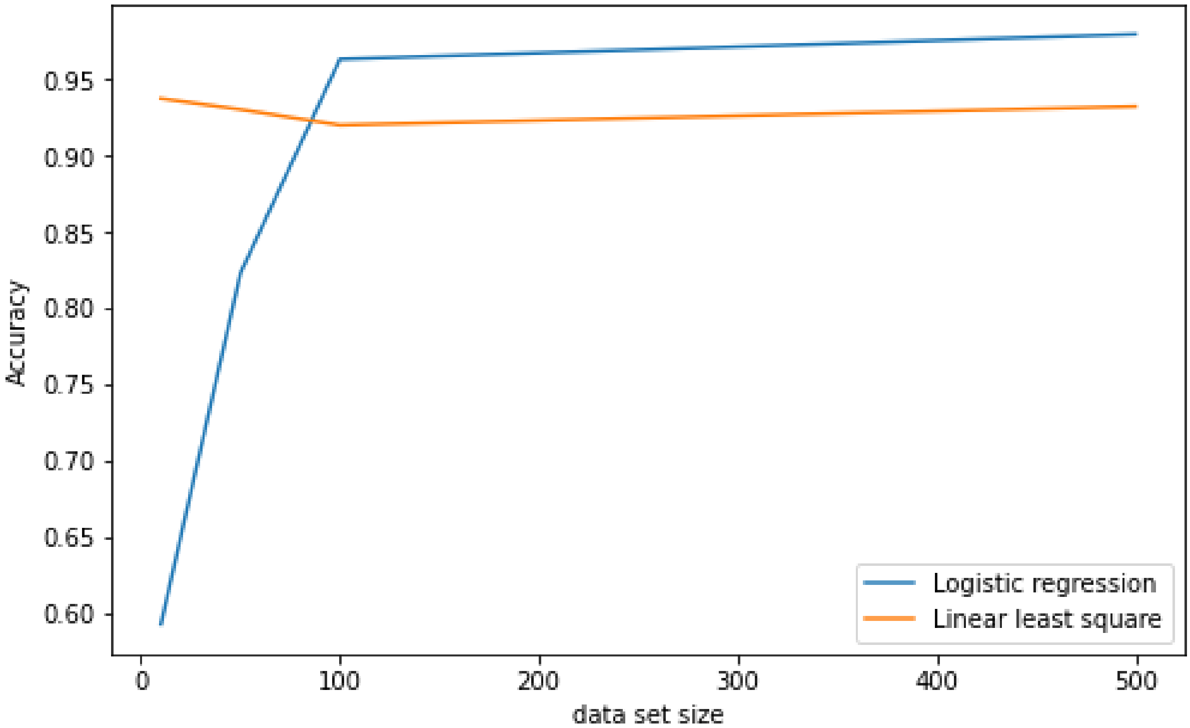
We can also see that in general logistic regression outperforms linear regression,which is shown to be almost always the case, empirically (for classification).

Linear least square had higher recall, but low precision ,where as Precision score for Logistic regression was better in comparision to linear least square.
Linear Regression F Score : 0.930059240058527
Linear least square F Score : 0.900022399015701

Logistic Regression is estimated using Maximum Likelihood Estimation (MLE) approach. with the help of MLE we estimate the parameters and get the mean and variance as the parameters for a particular model. also sigmoid function is used for determining the underlying probability of a data belonging to some class. the sigmoid function maps the number line to the desired range [0, 1].



Accuracy curve on data set Gamma test.txt



Accuracy curve on data set uniform_ test.txt

Sub-problem 2

we are given 2D data.
Class-I: Uniform over [0.5,0.6]x[0.5,0.6]
Class-II: Uniform over [0,1] x [0,1]

Number of examples in training data	Accuracy of Linear least square	Accuracy of Logistic regression
10	0.9299	0.5879
50	0.9311	0.8297
100	0.9244	0.9677
500	0.9312	0.9796
1000	0.944	0.9802

For data in Uniform_train.txt and Uniform_test.txt

Epoch 250, lr=0.1

Same procedure and algorithm is followed as above for linear least squares and logistic regression.The only difference here is class conditional densities are uniformly distributed.

Observation :
For less sample size, Linear least square classifier has higher accuracy(As already mentioned that accuracy can vary a lot when sample size is small). For large sample size, Logistic regression outclasses linear least square.

The models performed well for the class II samples, as compared to class I samples, reason can be attributed to the class density in some region. While Linear least square had higher recall, but low precision. Logistic regression is the better method for the provided data.

Linear Regression F Score : 0.9769509324765227
Linear least square F Score : 0.9262314229044735

As size of sample increases the accuracy increases(we can ignore the initial spikes and dips,as initially data samples are less).

Here accuracy of logistic regression classifier is best on full test data,This is because there is less overlap(relatively) between the class conditional densities (one class is uniform over [0.5, 6.0] x [0.5, 6.0] with mean [3.25,3.25] and other class is uniform over [0,1] x [0,1] with mean [0.5, 0.5]). As here means are far apart(relatively)(i.e m1=[3.25,3.25] and m2=[0.5,0.5]),hence the classification accuracy is better.

Also note that ,Linear regression classifies class 2 with 100% accuracy , while Logistic regression classifies class 1 with 100% accuracy. so for class 2 case linear regression is performing better than logistic regression. while for class 1 case logistic regression is performing better than linear regression.This can be good in case where we want to trade between 2 types of error.

The Accuracy graph of the above data which is uniformly distributed is shown in page 2.

Sub-problem 3

we have 10D Guassian data. Both class conditional densities have I as covariance matrix. The mean vector for class-I is all-zeros and that for class-II is all-ones.
epoch 1000, learning rate=0.7

Number of examples in training data	Accuracy of Linear least square	Accuracy of Logistic regression
10	0.6846	0.8444
50	0.9240	0.93199
100	0.9312	0.9337
500	0.9481	0.9429
1000	0.9469	0.9471

For data in Normal_train_10d.txt and Normal_test_10d.txt

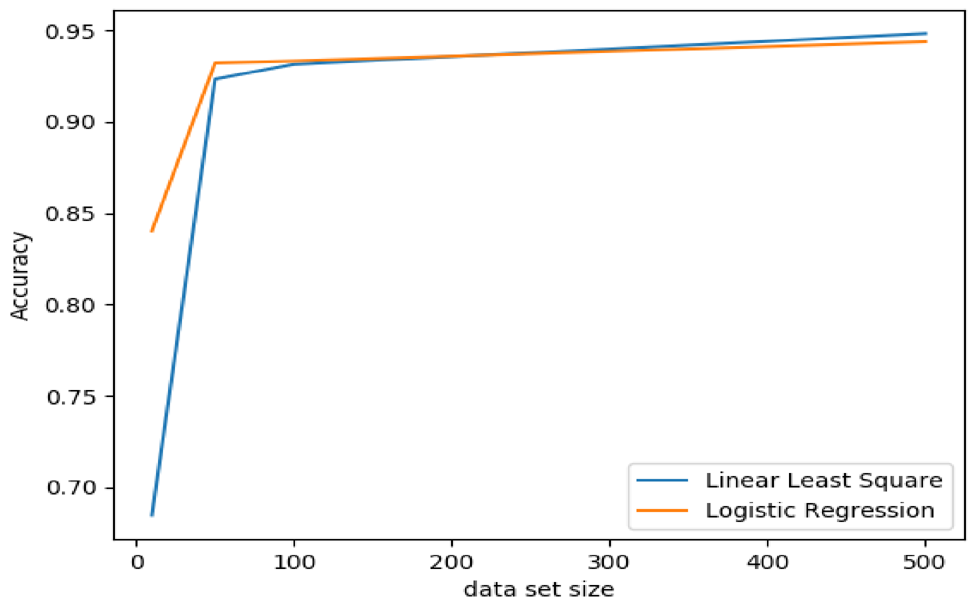
The implementation is done is the same way as before, first the data is loaded in a matrix 'A' of dimension nx10, (where n is the number of examples taken for training the model) then augmentation of data is done. Then the classes 1 or -1 is loaded in matrix 'y' of dimension nx1. The 'W' vector of Linear regression is calculated as $W=(A^TA)^{-1}A^Ty$. For Logistic regression the implementation is done in sklearn module.

Observations :

Initially, the linear least square method was working very poorly for low sample size, but when the sample size was increased to a higher value, the performance was way better, due to high dimension of data. Logistic regression works well for low as well as high sample sizes.

Also from the result above we can see that accuracy of both classifiers after 100 data samples are almost same. This shows that for Normal distribution with high dimentionality, both the classifiers are giving almost same accuracy for larger sample sizes.

So ,intotal accuracy of both the classifiers increases with increase in sample size,and as stated above, we can ignore the rare spikes and dips. Also as this is in higher dimensional with different means, it is intutively clear that classification will be more accurate.



Problem 2

Iris data, Multiclass classification problem
Sample size as 10, 20,50,80,100,120,140 is taken randomly from training data, making sure uniform
data is collected from each class.

First the data is split into test and train data. so taking 70% of total data as training and 30% as testing the models are trained(the below obseravation is on different split ratios). Also to ensure randomness the whole data is shuffled first and divided into training and testing data . The training data was loaded into a matrix 'A' . then Augmentation is done by adding 1 to matrix.

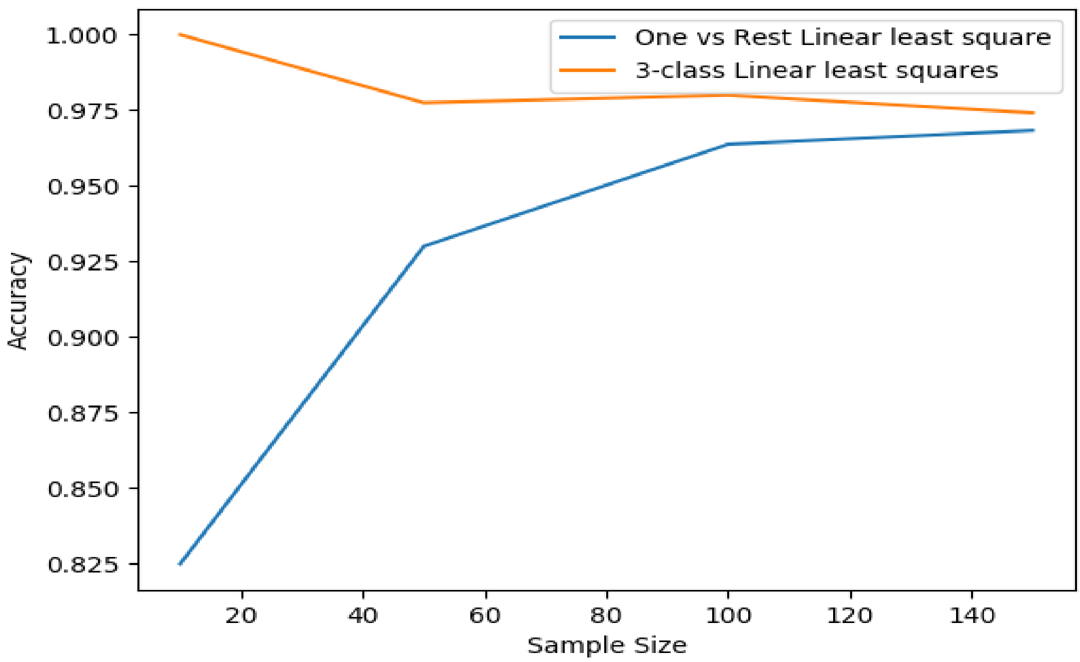
So implemented the 2 classifiers and there observation table is shown below:

(i) One vs rest linear least squares
(ii) 3-class Linear least squares(one hot)

In one vs rest approach , we first form a vector 'y1' in which if training example belongs to class 1 the value at that index is 1 while on other index it's -1("Virginica vs. rest","Sentosa vs. rest" and "Versicolor vs. rest".),For examples taking sample belong to class 1 then row vector of 'y' corresponding to that sample data will belike [1, -1, -1]. using these two matrices we learned the weight vector as Weight_1=$(A^TA)^{-1}A^Ty1$. Similarly we form the vector Weight_2 and Weight_3. the dimension of Weight matrix is 5 *3 .For each test data we find the dot product of W 's and test sample data.Whichever poduct of Weight_i results the maximum value the test sample is classified in that class.
In one hot vector approach, the dimension of matrix y of dimension nx3 .

Sample size	One vs Rest classifier Accuracy	3-class one hot classifier accuracy
10	0.825	0.99
20	0.83	0.98
50	0.93	0.977
80	0.93	0.975
100	0.96	0.98
120	0.961	0..9741
140	0.965	0.9739

For data in iris_dataset train.txt



Accuracy plot for both classifiers

Anshul Shivhare MTech AI 17828

Observations :

3-class linear classifier performs way better than one vs rest classifier. The one vs rest classifier changes a multi class problem into a binary classification(0/1) class.whereas the 3-class classifier is assigned only a single label . With only 150 datasize, 3-class classifier is better. But still as the dataset is small, then we should not directly jump to the conclusion. Also the split of 80:20 or 70:30 is best split for the above data(according to results).though not much difference is there.

NOTE: Both these methods treat a particular class as positive class and treat the other two classes together as a negative class.

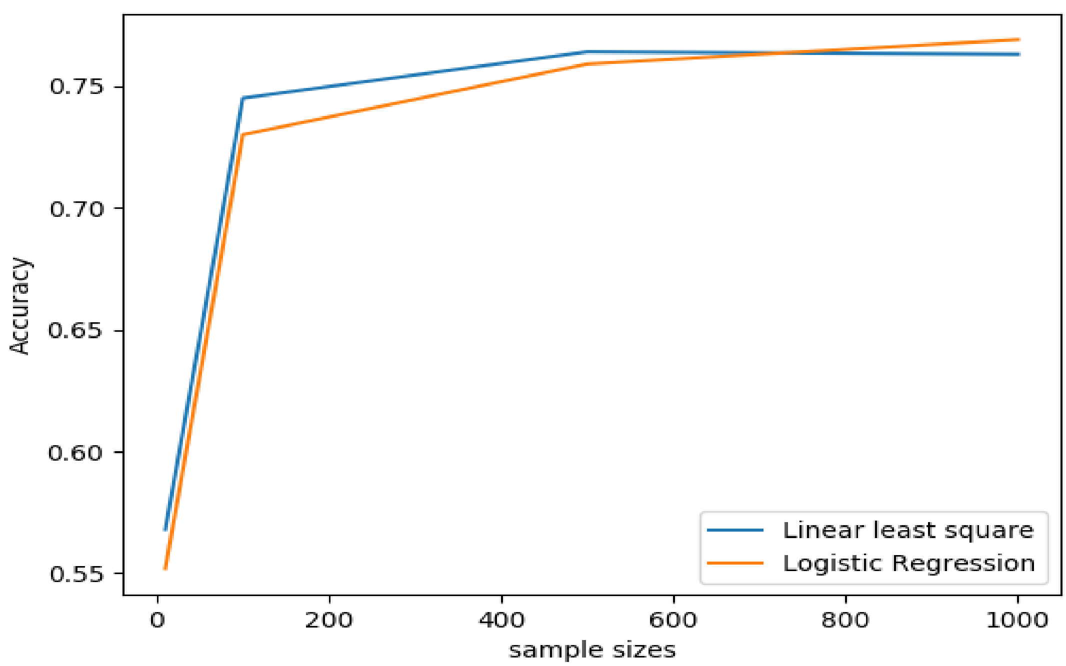
It was noticed that in one vs rest approach the class 1 is always classified truly, while class 2 is sometimes classified into class 3,and in some cases class 3 is classified to class 2.
and same in one hot approach(3- class linear least square), here also test data belonging to class 1 are always classified truly, while there is confusion between classifying class 2 and class 3.

Problem 3

we have 2-class classification problem ,24 dimensional feature vector, high dimension.
A sample size of 10,50,100, 500, 1000 were taken randomly from training data, making sure uniform data is collected from each class .Also our total input data is about 1000, so again we need to split it in training and testing data ,for different training split .The same procedure is used for implementation of Linear regression and Logistic regression as the previous question. The performance of respective classifier is shown in the below table by varying the sample sizes.

Number of examples in training data	Accuracy of Linear least square	Accuracy of Logistic regression
10	0.568	0.549
50	0.657	0.65
100	0.738	0.734
500	0.760	0.756
1000	0.763	0.767

For data in german.data-numeric



Accuracy curve on data set

The accuracy result for both of them were nearly same. The data is well shuffled.
The accuracy of both the classifier is in between 70% - 80%(when our sample size becomes sufficiently large) , and both of them have nearly sample accuracy. Also as the size of sample size increases the accuracy of classifiers increases(as it is intuitively clear, increase in train data result in better estimation of parameters).

NOTE :Estimated parameters converge in probability to true parameters when the number of samples is sufficiently large.

Observation:

Although both of the classes have similar accuracy for the whole sample size from 0 to 1000, linear least square gives better results when it comes to classification of class 2, and logistic regression gave better result when data is of class 2 .By changing train and test data again and again, both the classifier classified class 1 with about 90-100% accuracy, and class 2 is missclassified . For this dataset, both linear regression and logistic regression model performs more or less the same. The (relatively) low performance of the classifiers maybe due to the improper feature selections used. we can also optimize it by dropping the irrelevent features.

Problem 4

1D Regression Function y : y = $0.25x^3 + 1.25x^2 - 3x - 3$

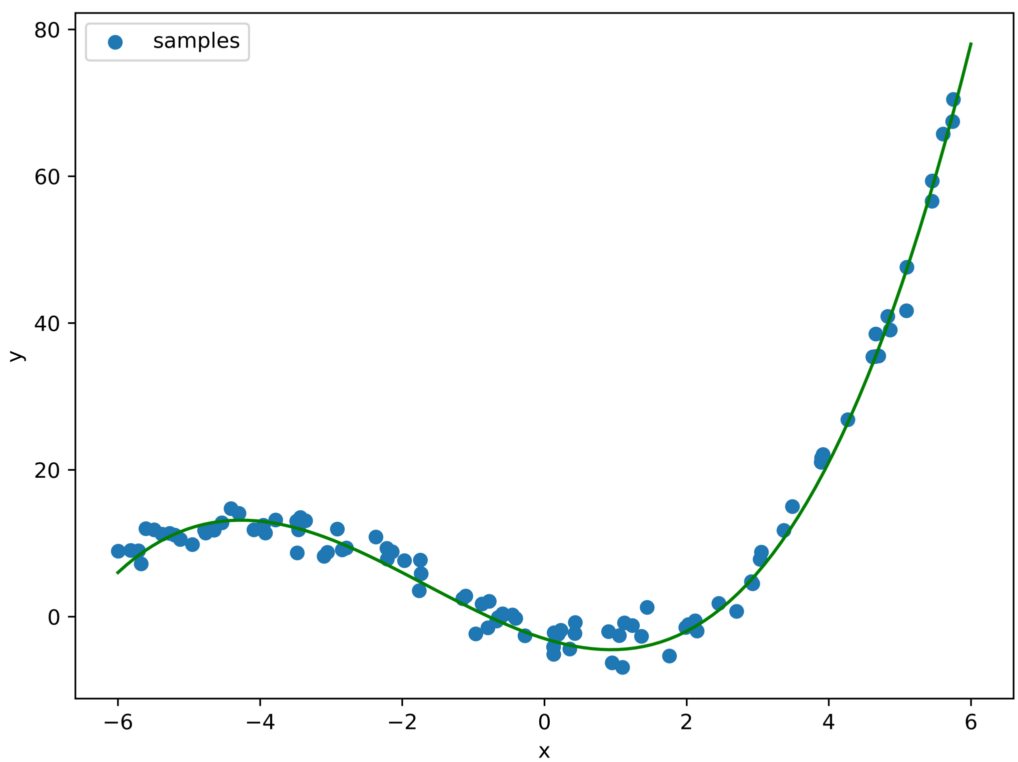
Our Objective is to find the best fitting curve for the function, by minimizing the sum of the squares of the offsets(Linear least squares Method) of points from the curve.Here also data Augmentation is performed first and then using least square solution weight vector is calculated.

a kth order polynomials fitted to the data(with gaussian noise) . so here the order of polynomials play a very import role for generalization purpose.

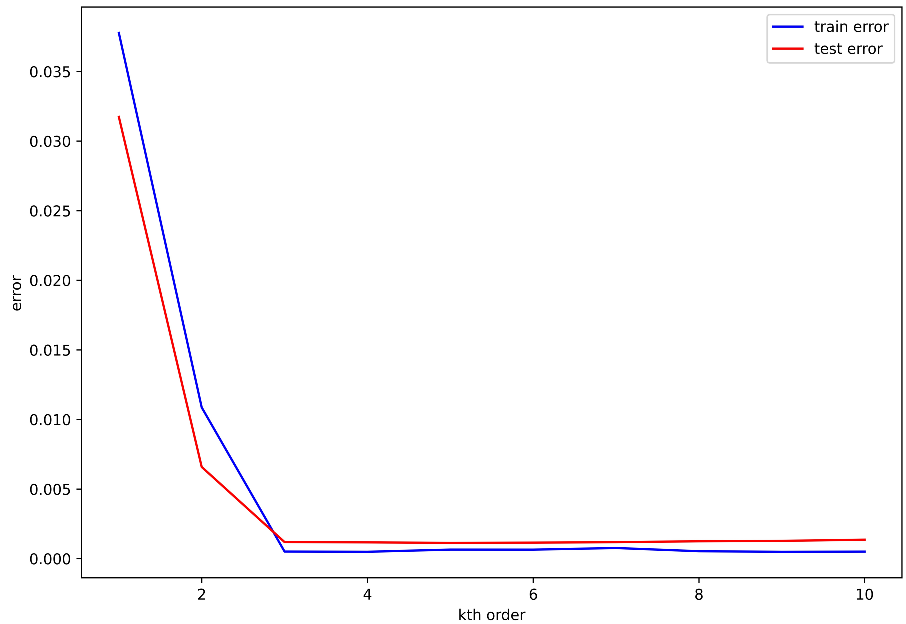
if we observe the mean square error, as the degree of polynomial increases the mean square error decreases(which is intutively clear),but after a particular degree the generalization ability of model decreases, as it result in overfitting of data.

After polynomial of order 3 and 4, there is overfitting for the given data. Also if considering R2 Score as evaluation metrics ,then as the degree increases there is increases in R2 score. and after a certain degree our results will not vary much as there will be overfitting.

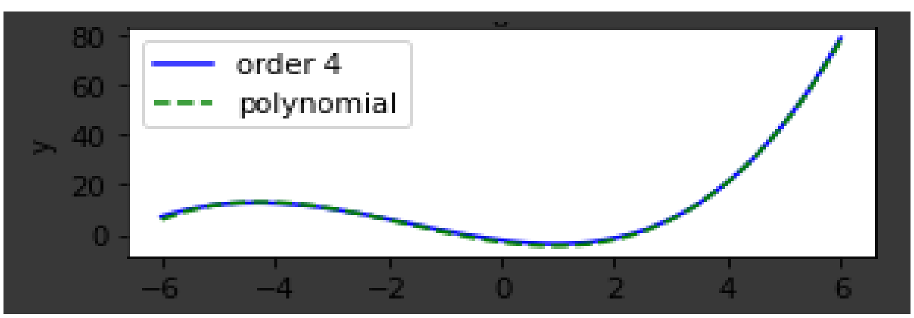
Also note that for 1 degree polynomial you will see underfitting.



Plot of train data.



Plotting the train and test error for kth orders



Plotting a polynomial of order 4 (best fit)

you can see in fig , as the order of polynomial increases both kind of error decreases ,later is becomes stable.

Note :Polynomial regression fits a nonlinear relationship between the value of x and the corresponding conditional mean of y, denoted E(y |x). Although polynomial regression fits a nonlinear model to the data, as a statistical estimation problem it is linear, in the sense that the regression function E(y | x) is linear in the unknown parameters that are estimated from the data. hence is is also called as a special case of multiple Linear Regression.