ISSN 2250-3153

Machine Learning (CSL 603)

Lab-2 Report

Linear and Logistic Regression

Piyush Pilaniya
2016csb1049

September 2017

Abstract- Linear and Logistic Regression are the most basic form of regression which are commonly used. The following report depicts the analysis done by performing different experiments and by plotting the results obtained. Further effect of Regularization is analyzed in details.

Index Terms- Machine Learning,Linear Regression, Logistic Classification , Regularization.

I. Introduction

This report discuss the results of two experiments. One in which the age of Abalone (a type of snail) is predicted using Linear Regression and experimented with varying fraction and regularization parameter. In another one , a classifier is designed which predict whether credit card can be issued or not.

II. Dataset Preprocessing

In the first part the data given is in the file named 'linregdata'. In this file there are 8 features are given and then the number of rings is given. So from this data is read by replacing the first feature which is gender as the vector.

In the second part the data set is very small and consists of merely 100 datapoint with just 2 features and one output as 0 or 1. Data is simply read and stored without any preprocessing.

III. Lab 2 - Part 1

The goal here is to implement Linear Regression. I have implemented Analytical Solution to find the weights which minimizes the cost function.

A. Experiment 1

In this experiment we were required to simply convert the first feature as 3 length vector. [1,0,0] for female, [0,1,0] for infants and [0,0,1] for males.It increases the length of feature vector .

Here I also added 1 as feature to every datapoint. It is done because we have -> x0w0 + x1w1…… So x0=1.

B. Experiment 2

In this we were required to Standardize the data by finding the mean and standard deviation and then for each data point subtract mean and divide by Standard Deviation.

C. Experiment 3

In this experiment we were required to make two functions namely mylinridgereg(X,Y,lambda) and mylinridgeregval(x,weights). Out of which the first one returns the weights using the analytical solution and the second one finds the outputs using the weights obtained using previous function.

D. Experiment 4

For this I have first extracted the test data (20%) out of total data and then for remaining data I have created a function named Sampling(data,frac) which takes two input data and fraction and using that function it samples the remaining data in train and self validation data.

E. Experiment 5

We are required to implement meansquarederr(T,Tdash) which return the mean square error of desired output and obtained output.

F. Experiment 6

Here comes the most important part where we executed all the functions built above. A set of different values of lambdas and fraction have been chosen and for each 100 repetition have been done.

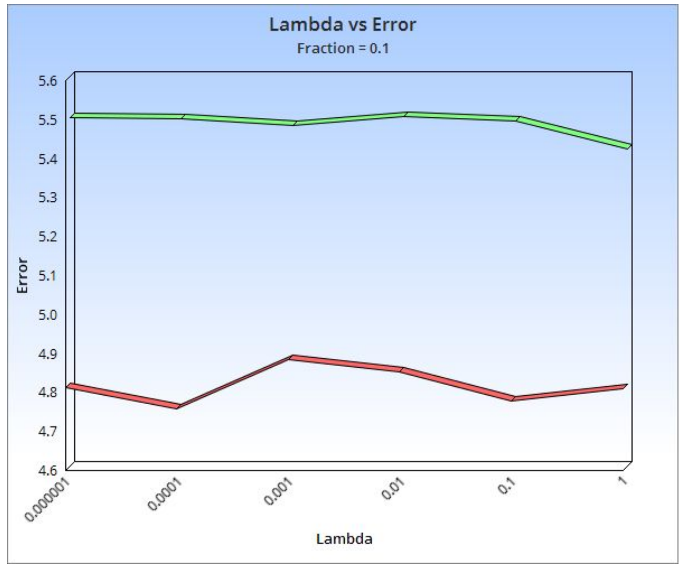
G. Experiment 7

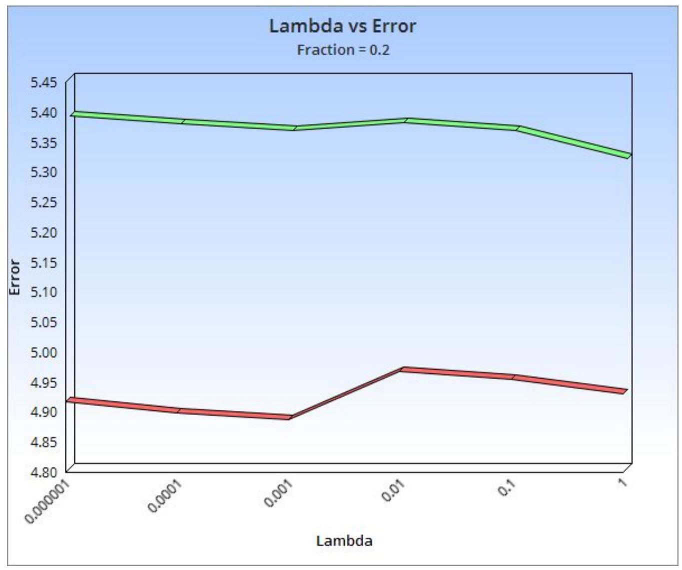
I have chosen 5 different values for fraction and 6 different values for lambda and implemented the function of finding weights then finding output and then error 100 times and stored the average error and also displayed the average error and standard deviation over 100 run. The below table shows the results obtained for different set of fractions and lambdas.

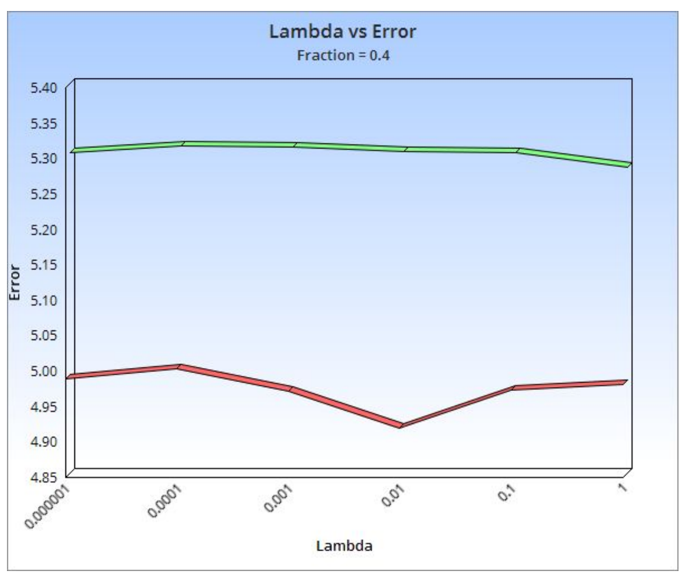
	1e-06	0.0001	0.001	0.01	0.1	1
0.1	4.811	4.755	4.883	4.850	4.775	4.808
0.2	4.917	4.898	4.887	4.966	4.954	4.929
0.4	4.988	5.002	4.971	4.919	4.972	4.981
0.7	5.025	4.978	5.002	5.023	5.011	5.013
0.9	5.012	5.016	5.017	5.020	5.024	5.017

1st row represents different set of lambda values and 1st column represents different set of fraction of data.

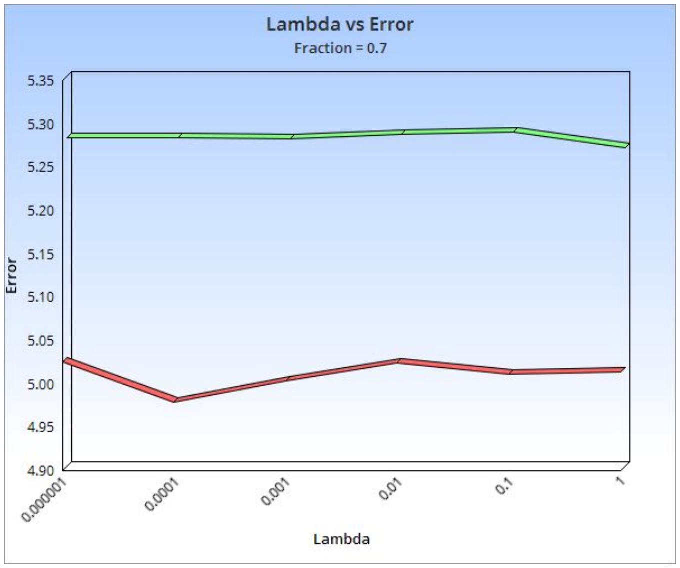
Below are the graphs each for a specific fraction.

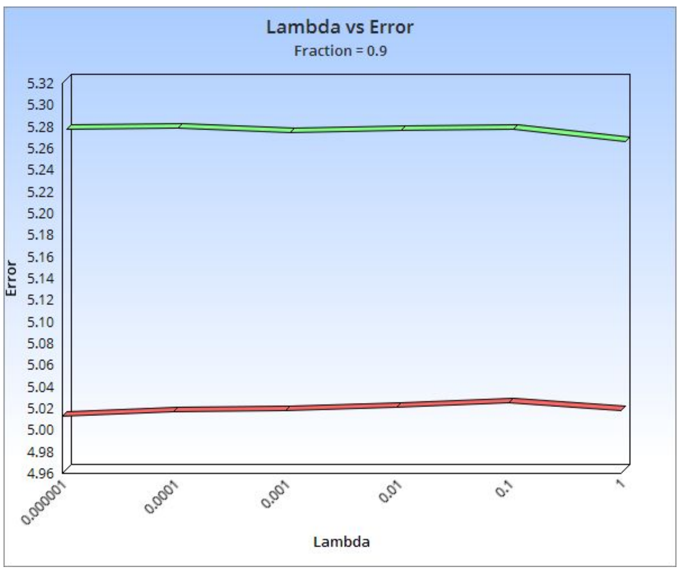






ISSN 2250-3153

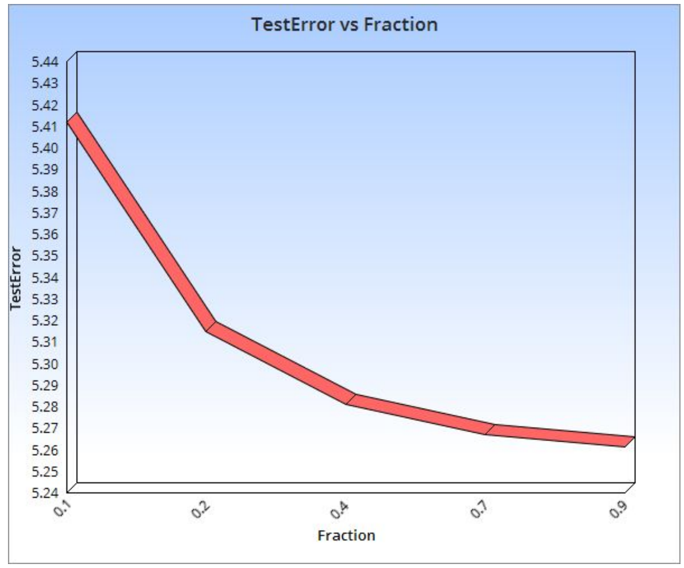




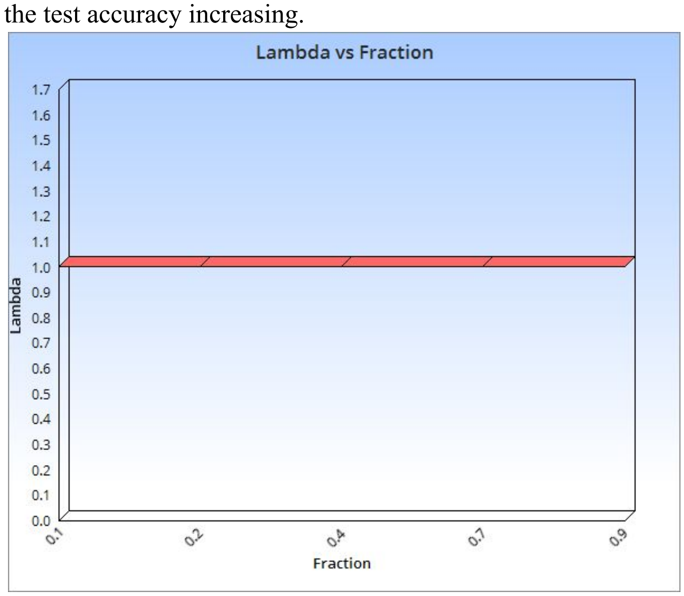
OBSERVATION: We can observe that all the error are nearly close independent of fraction and lambda . The difference between train and test error is approximately constant around 0.26.

H. Experiment 8

Below are two more graphs where the first one represents plot of average minimum test error vs fraction and the second one represents plot of lambda value for which we obtained average minimum test error vs fraction.



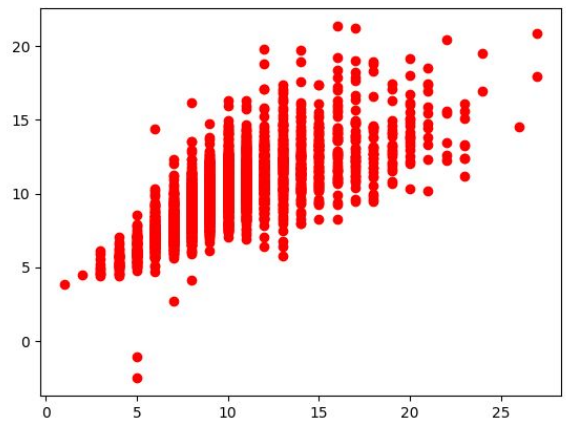
Observation : As the fraction of data is increasing the test error is reducing. It is happening because if we have less data point then the linear regressor won't be able to get trained properly and as the number of training data points increasing , the test accuracy increasing.



I. Experiment 9

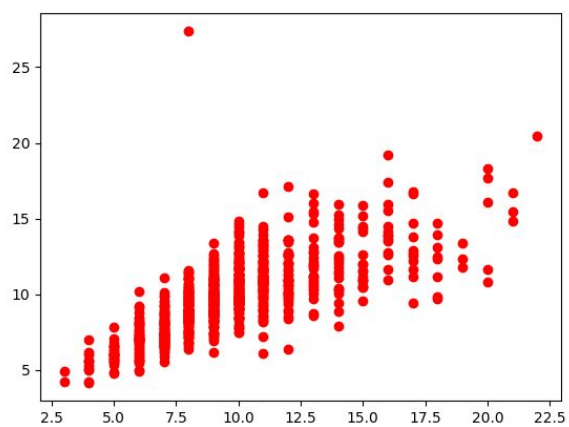
In this experiment I have plotted a graph between actual output vs predicted output obtained with minimum average error both for train and test data.

Below is the plot on Train data



X- axis represents Actual output Y-axis represents Predicted output .

Below is the plot for Test data



X-Axis represents Actual output and Y-axis represents predicted output.

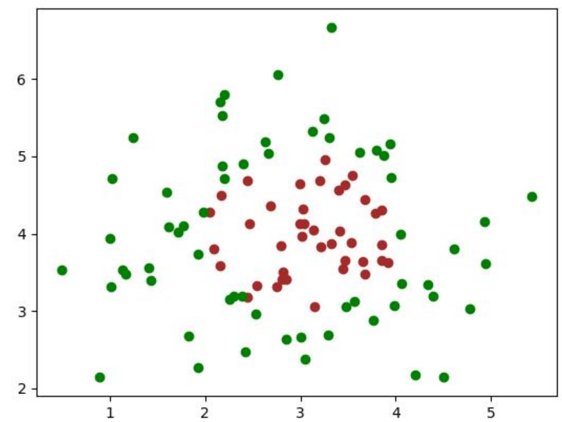
Observation: It can be seen from plot that Actual output and predicted output are nearly close for all data points as the graph is depicting almost 45 degree (slope=1).

IV. LAB2 - PART2

In this part I have implemented a logistic regressor which outputs '1' if sigmoid function returns value>0.5 and else '0'. Regularization is also done and it is observed that its effect is approximately negligible.

A. Experiment 1:

We are simply required to plot the dataset.



Red Nodes are positive data points while green points are negative data points.

B. Experiment 2:

In this part I have randomized the selection of the weights between -0.1 and 0.1. To reproduce its result please set the value of lambda at line 121 in q2.py =0 as I have implemented regularized and normal at one go . So put the regularization parameter 0 to reproduce this parts results.

The performance observed is very poor . The number of mismatching point ranges from 40 to 60 out of 100 which means quiet a poor performance.

Observations:

Below are the observations for three different weight set . To Verify these modify the value of degree in line 86 = 1.

w0	w1	w2	Error
0.0813736	0.0837171	0.0182391	49%
-0.076409	-0.094325	-0.063153	41%
0.0645611	0.0052762	-0.085509	54%

Analysis: Since the degree is initially set to 1 that's why we are not able to produce a separable boundary using degree 1. We would require at least degree 2 features to separate the data. Furthur in below experiments I have dealt with higher degree classifiers.

C. Experiment 3:

Analysis:

No the data is not linearly separable. See the experiment 1 of this part. We can easily see that there is no line which can separate these data points. It can be seen that there will exist a circle which can separate both , so degree 2 might give us better result with error close to 0.

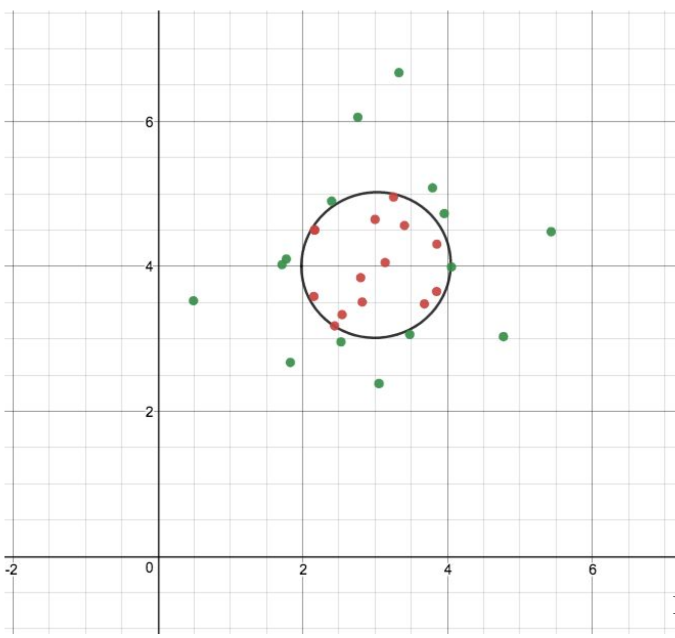
D. Experiment 4:

As seen from the experiment , the data is not linearly separable so we would require higher degree curves to separate the data.

When using degree 2 features also. We would get total 6 features which can easily separate the data points given.

Below is the plot for one such curve of degree 2:

Equation of curve = -0.28852955*x^2 + 0.00950507*x*y + 1.70015289*x - 0.30419406*y^2 + 2.4163829*y - 7.10908923 = 0



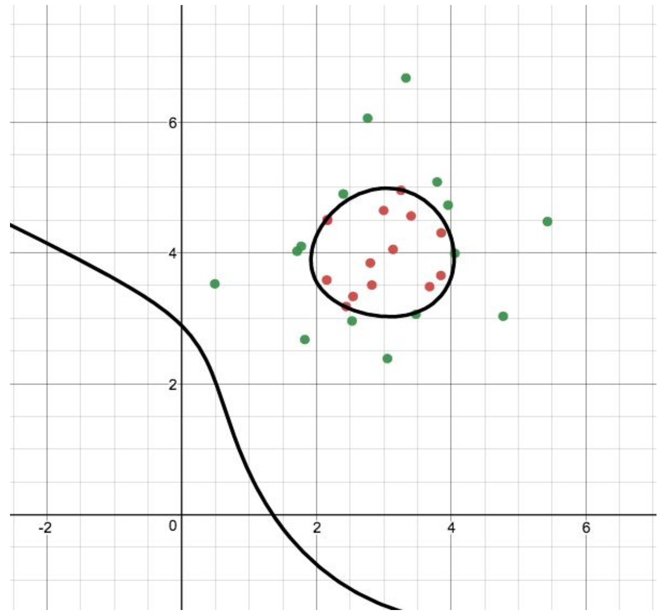
Here I have plotted the curve along with a few data points at random. You can see in the figure that the curve is separating the data points very effectively.

Observation : The Error obtained is zero. To reproduce the result set the value of degree in line 86 to 2.

Now when using 3 degree curve , I have got total 10 features.

Below is the plot for degree 3:

Equation of curve = 16.5586523 - 10.96709792*y + 2.85433069*y*y - 0.36114275*y*y*y - 17.15219449*x + 5.12436786*x*y - 0.15393811*x*y*y + 4.10054712*x*x - 0.64851226*x*x*y - 0.35472182*x*x*x = 0



Here also the data is getting seperated.

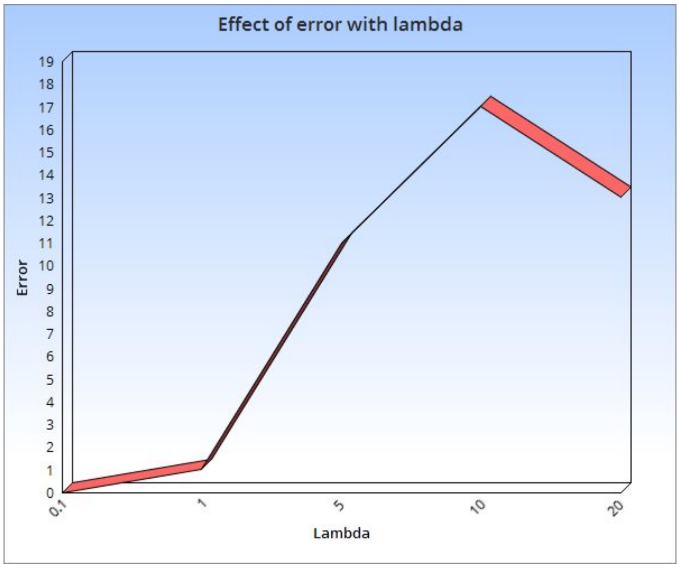
Analysis : Thus we can infer that sometime to obtain better result we have to analyze the data points to find what type or degree of curve will separate these data points, the same happened with me here.

E. Experiment 5:

Graph plotted in previous part itself.

F. Experiment 6:

Below is the graph of variation of lambda and error.



Analysis : When the regularization parameter is less than 1 then there is no effect of it on accuracy. But when the lambda is increases very higher then the error increases.