Linear Regression Analysis

Revanth Reddy Bandaru **Date:04-30-21**

Table of Contents

Summary of the report

This report consists of implementation of linear regression models from scratch in python. The given data is regularized using L2 Regularization is also called Ridge Regression. The bias–variance trade off is the property of a model that the variance of the parameter estimates across samples can be reduced by increasing the bias in the estimated parameters. The bias–variance dilemma or bias–variance problem is the conflict in trying to simultaneously minimize these two sources of error that prevent supervised learning algorithms from generalizing beyond their training set.

Problem Definition:

Linear Regression is a method used to define a relationship between a dependent variable (Y) and independent variable (X). Which is simply written as:

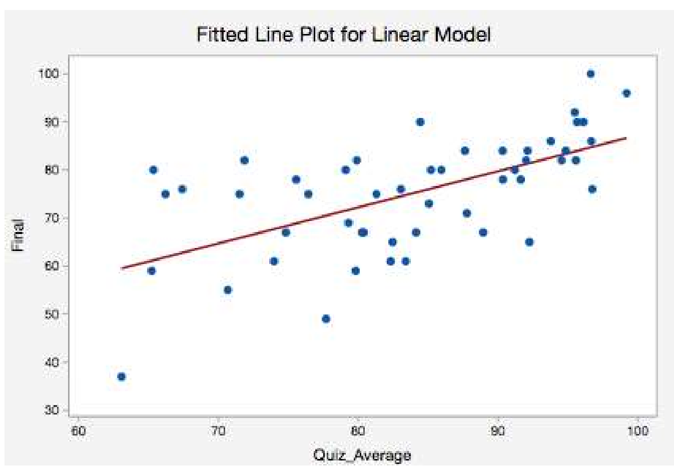
$$Y = aX$$

Where y is the dependent variable, m is the scale factor or coefficient, X being the independent variable. The bias coefficient gives an extra degree of freedom to this model. The goal is to draw the line of best fit between X and Y which estimates the relationship between X and Y.

But how do we find these coefficients, we can find these using different approaches. One is the Ordinary Least Mean Square Method approach and the Gradient Descent approach. We will be implementing the Ordinary Least Mean Square Method.

Linear Regression using Gradient Descent:

Earlier we discussed estimating the relationship between X and Y to a line. For example, we get sample inputs and outputs and we plot these scatter point on a 2d graph, we something similar to the graph below:



The line seen in the graph is the actual relationship we going to accomplish, and we want to minimize the error of our model. This line is the best fit that passes through most of the scatter points and also reduces error which is the distance from the point to the line itself as illustrated below.

Gradient descent is an iterative optimization algorithm to find the minimum of a function. Here that function is our Loss Function.

Analysis approach:

Given X and y are the data sets, where X is the Input and y is the output, To model the relationship between inputs and outputs linearly. Let us consider the following equation y = aX which can further classified as

$$\hat{y}=h(X)=Xa$$

where a: n x 1 vector X: m x n matrix y: m x 1 vector

Input file has 5 features and output file has 3 different outputs. Load the input using Pandas (easy tool for data manipulation and pre-processing). Split the Data into training data and test data (i.e 80 and 20). The updated Cost function and Gradient descent with the regularization parameters are

$$J(a) = \frac{1}{2m}\left[\sum_{i=1}^{m}(h_a(x_i) - y_i)^2 + \lambda\sum_{j=1}^{n} a_j^2\right]$$

$$a_j = a_j - \alpha\frac{1}{m}\left[\sum_{i=1}^{m}(h_a(x_i) - y_i)x_{ij} + \lambda \cdot a_j\right]$$

Where J(a) is the cost function and a is the parameter and λ is regularization parameter

Regularization and Bias/Variance:

A large lambda heavily penalizes all the parameters, which greatly simplifies the line of our resulting function, so causes underfitting.

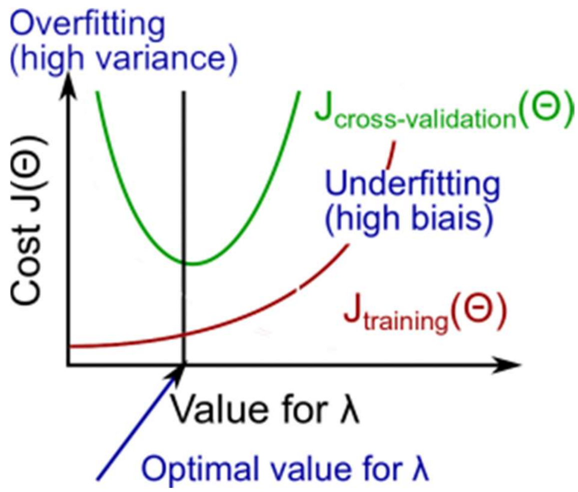
The relationship of λ to the training set and the variance set is as follows:

Low λ: J_train is low and JCV is high (high variance/overfitting).

Intermediate λ: J_train(Θ) and JCV are somewhat low and J_train≈JCV.

Large λ: both J_train(Θ) and JCV will be high (underfitting /high bias)

The figure below illustrates the relationship between lambda and the hypothesis:



In order to choose the model and the regularization λ, we need:

Create a list of lambdas (i.e. λ∈0,0.01,0.02,0.04,0.08,0.16,0.32,0.64,1.28,2.56,5.12,10.24), Create a set of models with different degrees or any other variants. Iterate through the λs and for each λ go through all the models to get get cost function and error.

Compute the cross-validation error using the learned (computed with λ) on the JCV without regularization or λ = 0. Select the best combo that produces the lowest error on the cross-validation set.

Using the best combo, apply it on J_test to see if it has a good generalization of the problem.

Implement cost function, optimization function (gradient descent) and predict method by predicting Regularization parameter lambda by plotting a graph between lambda vs error for both training data and test data.

Results:

To find the best fit model for the given data the value of λ(regularization parameter) is 0.075 and plotting the learning curves by varying the training size the following graphs are shown below.

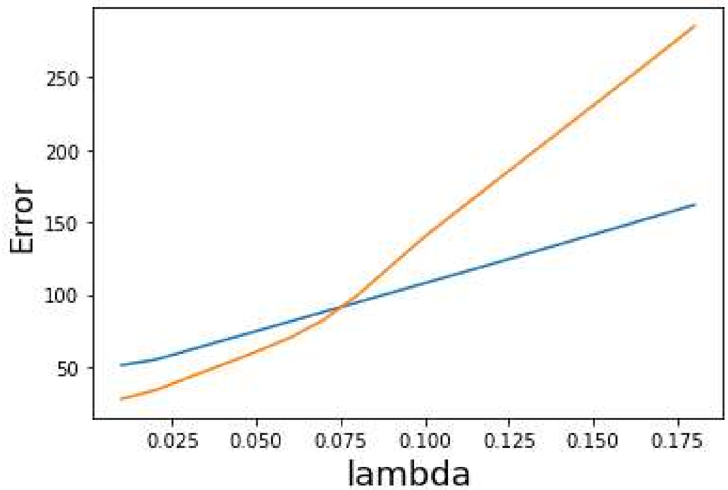
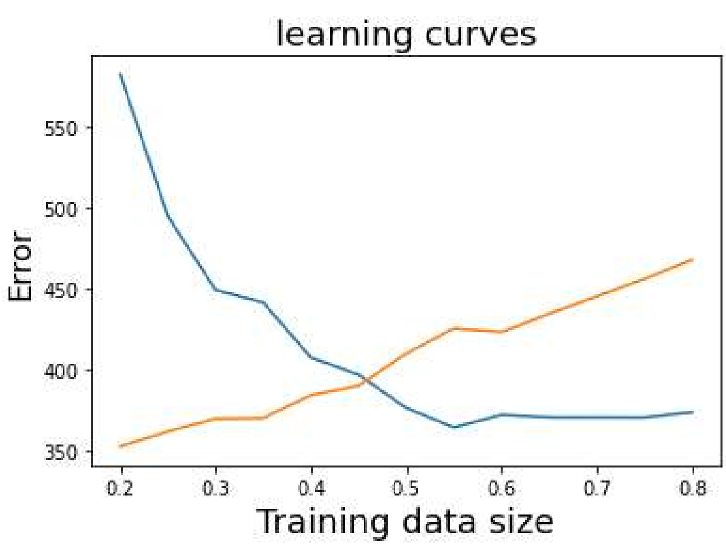


Figure 1

The Figure 1 is Error versus Lambda which is plotted by varying lambda with different values and selecting the λ as 0.075 and with this lambda value plot the learning curves for training data and test data.



Conclusions:

Regression Analysis is used in the broader sense; however, primarily it is based on quantifying the changes in the dependent variable (regressed variable) due to the changes in the independent variable by using the data on the dependent variables. This is because all the regression models whether linear or non-linear, simple or multiples relate the dependent variable with the independent variables.