Machine Learning
Project-1 Linear Regression

Saachi Shivhare - 1001830083

Abstract

This report summarizes the analysis done for Linear Regression – Multivariant. Linear Regression is done on given set of training data points and the corresponding line function is found to predict the future values of the unseen data. Cross Validation is also performed to check the accuracy of the Model.

1 Introduction

Linear Regression is an approach to model the relationship between input variables and the output. We try to fit the datapoints using a line/plane in such a way that root mean square error (RMSE) calculated by the difference between target output and the predicted output is minimum.
Iris data is used which can be represented in matrix form. The data is a labeled data and contains 150 data points. The target variable has 3 labels.
The data contains 50 data points of each label and is organized as per the label. First, I shuffled the data so that data is divided proportionally. The shuffled data was splitted into two parts i.e. tarin data and test data. Target Label were also converted to integers. The train data was concerted to a matrix, transposed. Performed dot product of the original matrix and the transposed matrix, took inverse of the new matrix. Again I performed the dot product of new matrix, transposed matrix and the target matrix. Perform all transformations on the original data matrix to compute beta value and the training of the model is completed. I used these beta values to compute the predicted value of the test data and compared the predicted value with the original value for the test data. I computed the Root Mean Squared Error which shows the accuracy of the Model. Linear Regression is finding a model with less RMSE based on the input training data points. A model with less root mean square will help to predict the result of test data more accurately.

$$\widehat{\beta} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{Y}$$

The Beta values are calculated using this formula Beta = (XT X) 1XT Y. Once these values are estimated using the parameters model is built.

2 Iris Data

The data used for this project is as mentioned below. There are total 150 data points. 50 data points are of Iris-setosa, 50 data points are of Iris-Versicolor and 50 data points are of Iris-virginica. The first attribute is sepal length, second attribute is sepal width, third attribute is petal length and fourth attribute is petal width. All the attributes are in centimeter. I converted the target variable into integer values 0 for Iris-setosa, 1 for Tris-versicolor and 2 for Iris-virginica.

5.1,3.5,1.4,0.2, Iris-setosa
4.9,3.0,1.4,0.2, Iris-setosa
4.7,3.2,1.3,0.2, Iris-setosa
4.6,3.1,1.5,0.2, Iris-setosa
5.0,3.6,1.4,0.2, Iris-setosa
5.4,3.9,1.7,0.4, Iris-setosa
…
7.0,3.2,4.7,1.4, Iris-versicolor
6.4,3.2,4.5,1.5, Iris-versicolor
6.9,3.1,4.9,1.5, Iris-versicolor
5.5,2.3,4.0,1.3, Iris-versicolor
6.5,2.8,4.6,1.5, Iris-versicolor
…..
7.7,3.0,6.1,2.3, Iris-virginica
6.3,3.4,5.6,2.4, Iris-virginica
6.4,3.1,5.5,1.8, Iris-virginica
6.0,3.0,4.8,1.8, Iris-virginica
6.9,3.1,5.4,2.1, Iris-virginica
6.7,3.1,5.6,2.4, Iris-virginica

2.1 Observation

The data is listed in a order of the class label. To develop a fair model, I shuffled the data and splitted the data into train set and test set.

3 Model- Linear Regression and Classification

After performing all the transformation on the train data matrix, beta value is computed. I used these computed beta value to compute predicted value of target for unseen test data. I also calculated the RMSE to check the accuracy of the model. There is not much difference in RMSE of both Train data and Test data which indicates that there is no overfitting.

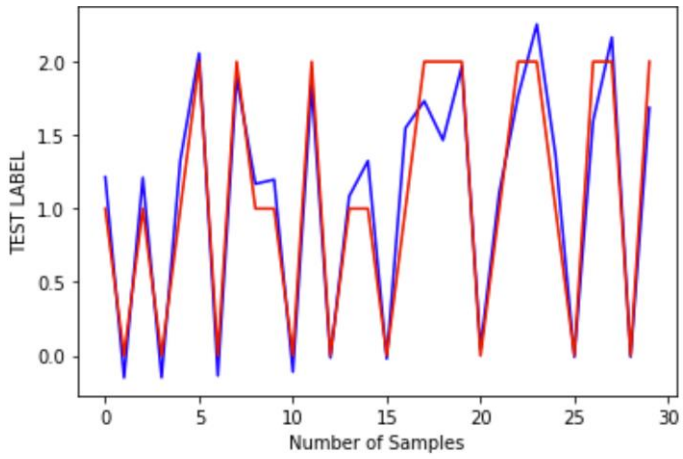

Figure 1

X axis: Number of Samples
Y axis: Test Labels

Blue Color indicates predicted value of Y
Red Color indicates actual value of Y

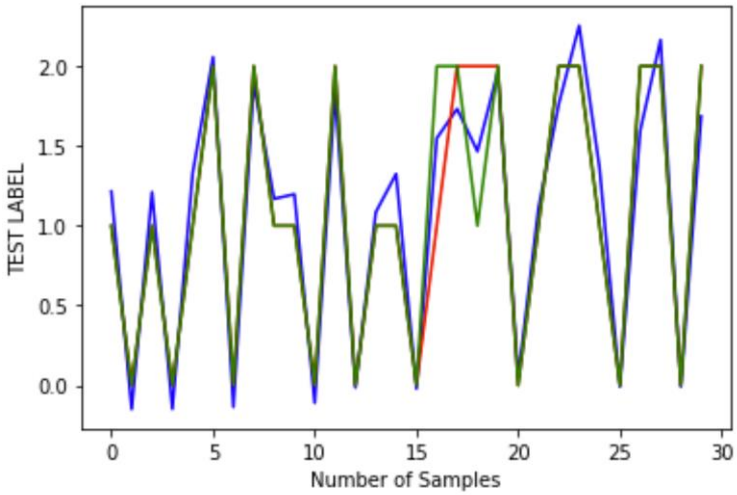

Figure 2

X axis: Number of Samples
Y axis: Test Labels

Blue color indicates the predicted value of Y (target variable).
Green color indicates the predicted value of Y after classification.
Red color indicates the actual value of Y.

In figure 2, it can be seen that red line and green line are almost similar and are not overlapping between sample 15 and 20. It is evident from the plot that the Model is predicting almost all the values correctly. Only one value got incorrectly classified.

4 Cross Validation

Some times the data can be biased. To reduced bias we perform Cross validation on the data. Error estimation is made after the model is trained. In other methods, by splitting the model into train data and test data, we sometimes lose some important patterns / trends in the data which in turn increases bias. I used K fold cross validation in this project. I divided that data into k subsets . Out of these one is used as a validation set and the other one is used as a training set. In this I picked one set to tarin the model and used remaining sets of data to validate the model. I also computed RMSE for both train data and validate data. The RMSE values of validate data are again averaged over all k trials to compute the effectiveness of the model. In this approach every dataset gets to be in validation set exactly once and gets to be in training set k-1 times. With this approach the bias will be reduced significantly. Interchanging the training and test sets also adds to the effectiveness of the model.

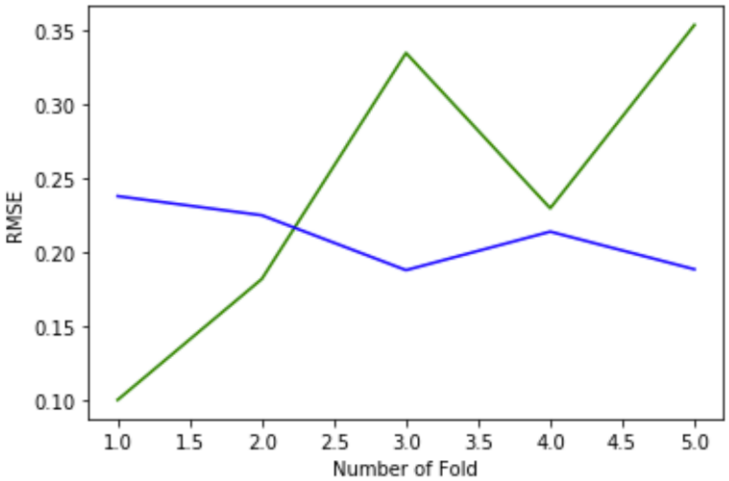

Figure 3

X axis : Number of folds

Y axis : Root Mean Square Error

Blue color indicates RMSE Values of Train Data

Green color indicates RMSE Values of Validate Data

From the above figure, it is evident that I got the highest accuracy in the 1st fold. Also, in the 4th fold there is not much difference in validation RMSE and train RMSE value.

Total effectiveness of the model for the validation dataset is also computed.

5 References:

https://integratedmlai.com/basic-linear-algebra-tools-in-pure-python-without-numpy-or-scipy/
https://towardsdatascience.com/multiple-linear-regression-from-scratch-in-numpy-36a3e8ac8014
https://cs230.stanford.edu/blog/split/
https://learncodingfast.com/python-programming-challenge-2-multiplying-matrices-without-numpy/
https://en.wikipedia.org/wiki/Linear_regression
https://www.geeksforgeeks.org/determinant-of-a-matrix/
https://en.wikipedia.org/wiki/Stochastic_gradient_descent
https://datascience.stackexchange.com/questions/9167/what-does-rmse-points-about-performance-of-a-model-in-machine-learning
https://towardsdatascience.com/cross-validation-in-machine-learning-72924a69872f