University of Regina

ENSE 865 : Applied Machine Learning

ASSIGNMENT 1

SUBMITTED BY: Hitarthi Gandhi(200428713)

Question1. Write a generic function that accepts a column of data ‘input_feature’ and another column ‘output’ and returns the Simple Linear Regression parameters ‘intercept’ and ‘slope’. Use the closed form solution to calculate the slope and intercept.

Here, we are using Fish.csv Dataset and in this dataset our main aim to find the estimated weight of the fish and minimum RSS value.

To implement this first we are importing python libraries NumPy, pandas and matplotlib. Then we are splitting the dataset into 80% training and 20% testing using the Scikit learn. To build a generic function for linear regression here we are using two parameters input_features and output. We are using Closed form method to calculate slope and intercept for residuals square sum (RSS). In closed form method we are considering X as an input_feature and Y as output. We write function for simple linear regression to find a intercept and slope of the function. But this function will not print the value of Slope and Intercept it will just return the value of slope and intercept for regression. Below is the code for calculating the slope and interecept.

slope = *(sumYX - ((sumY*sumX)/n))/(sumXX - ((sumX*sumX)/n))*

intercept = *(sumY/n) - slope*(sumX/n)*

Moreover, to test the function we are passing the value where we already know the answer. we create feature and then output exactly on a line. output = 1 + 1*input_feature then our output for the intercept and slope should be 1.[1]

Question2. Write a function that accepts a column of data ‘input_feature’, the ‘slope’, and the ‘intercept’ you learned, and returns a column of predictions ‘predicted_output’ for each entry in the input column.

For the prediction of regression function, we are passing three parameters input_feature, Intercept and Slope. Here, we are using intercept and slope which we already find in above question to predict the value Input_feature is the same that we used in above question.

intercept + slope*input_features by using this formula function predicted output for the linear regression. This function returns the value of predicted output.

Question3. Write a function that accepts a column of data: ‘input_feature’, and ‘output’ and the regression parameters ‘slope’ and ‘intercept’ and outputs the Residual Sum of Squares (RSS).

Here, We are creating function for the Residual square of sum. We are passing four parameters input_feature, output, intercept and slope. RSS is the square of residuals(error). RSS is the difference between the predicted output value and Actual output value. This function will return the value of RSS.

Question4. Use your function to estimate the slope and intercept on the training data to predict weight of fish for each one of the following (one at a time) inputs. Save each model (slope and intercept) separately. 'Weight of the fish' will be the 'output' and each of the following as an 'input_feature'.

In this section, to find slope and intercept for all input_features we are using the function which we created above. Here We are calculating the Slope and Intercept just for the training dataset. Simple linear regression passing the two arguments from the train dataset.

Below code is calculating the intercept and slope for the input_feature Width.

Width_intercept, Width_slope = simple_linear_regression(x_train['Width'],y_train)

Below is the output for all input_features in comparison of weight of the fish.

Length1: Intercept and Slope

```
Intercept_Length1: -480.0884960039758
Slope_Length1 33.416258748400296
```

Length2: Intercept and Slope

```
Intercept_Lenght2: -494.3214698885416
Slope_Length2 31.32767883570932
```

Length3: Intercept and Slope

```
Intercept_Length3: -510.8409515559273
Slope_Length3 29.00854679318342
```

Height: Intercept and Slope

```
Intercept_Height -156.29185369094796
Slope_Height 60.450449066829364
```

Width: Intercept and Slope

```
Intercept_Width:  -471.6902324034189
Slope_Width:  192.89807114052095
```

Question5. Using above estimated slopes and intercepts for each of model, fit a line through training data points. Draw separate plot for each of the 'input_feature'.

We used python library matplotlib to plot the graph. Here, we are using predicted output values of the regression to plot a best fit line over the training data points. Firstly, we are plotting the data points and then plotting a fit line for the input_feature.

Below code is plotting the data points and fit line for the input_feature Width.

```
Width_plot = x_train.Width #This will just take the width column from the training data.
Weight_plot = x_train.Weight # This will just take the weight column from the training data.
width_plotline = Width_plot * Width_slope + Width_intercept #Formula for plot a line

plt.figure()
plt.plot(Width_plot,Weight_plot, 'b.')
plt.plot(Width_plot,width_plotline, 'r-')
plt.title("Width VS Weight Graph")
plt.xlabel("Width")
plt.ylabel("Predicted_Weightoutput ")
plt.show()
plt.savefig('width graph.png')
```

Below is plotted graph for input_feature:

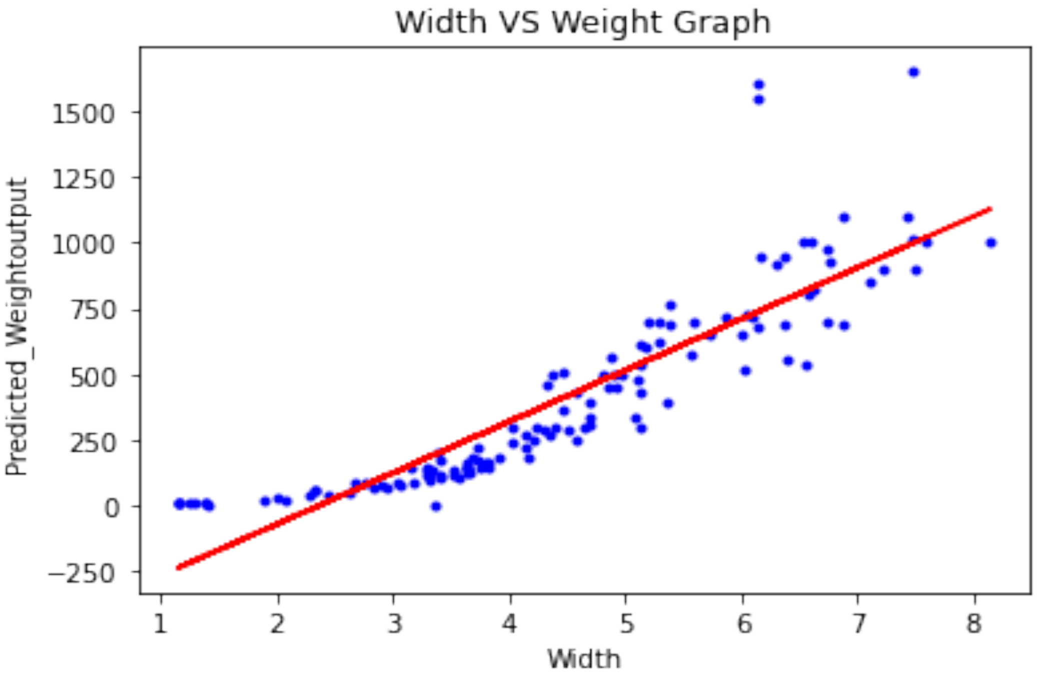

Fig1. Width VS Weight

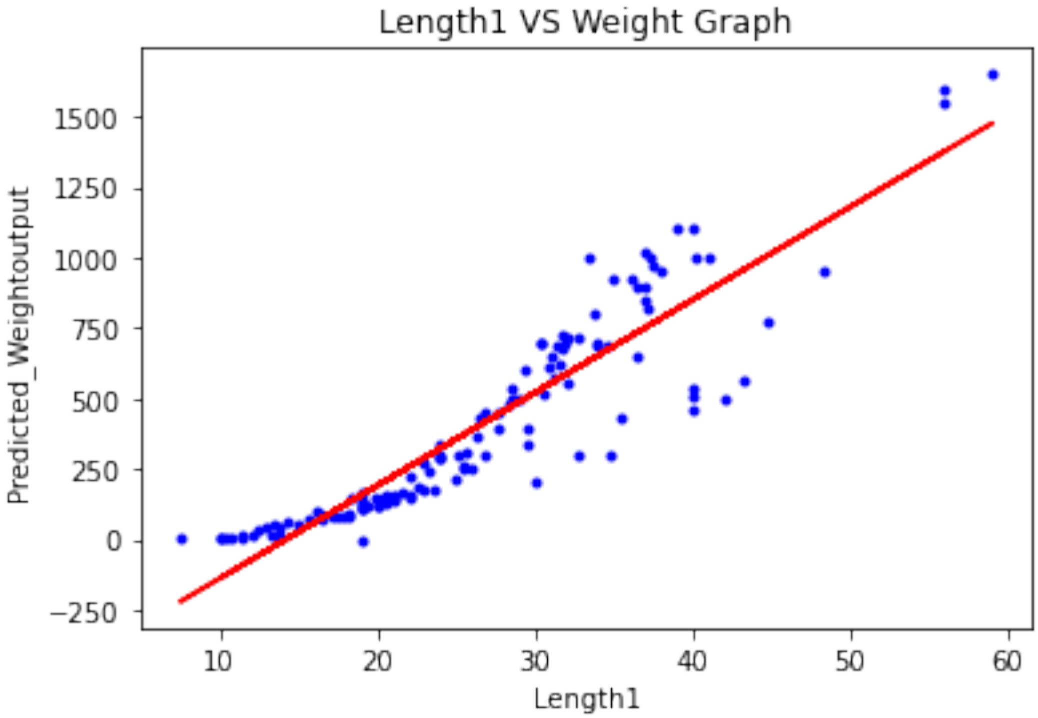

Fig2. Length1 VS Weight

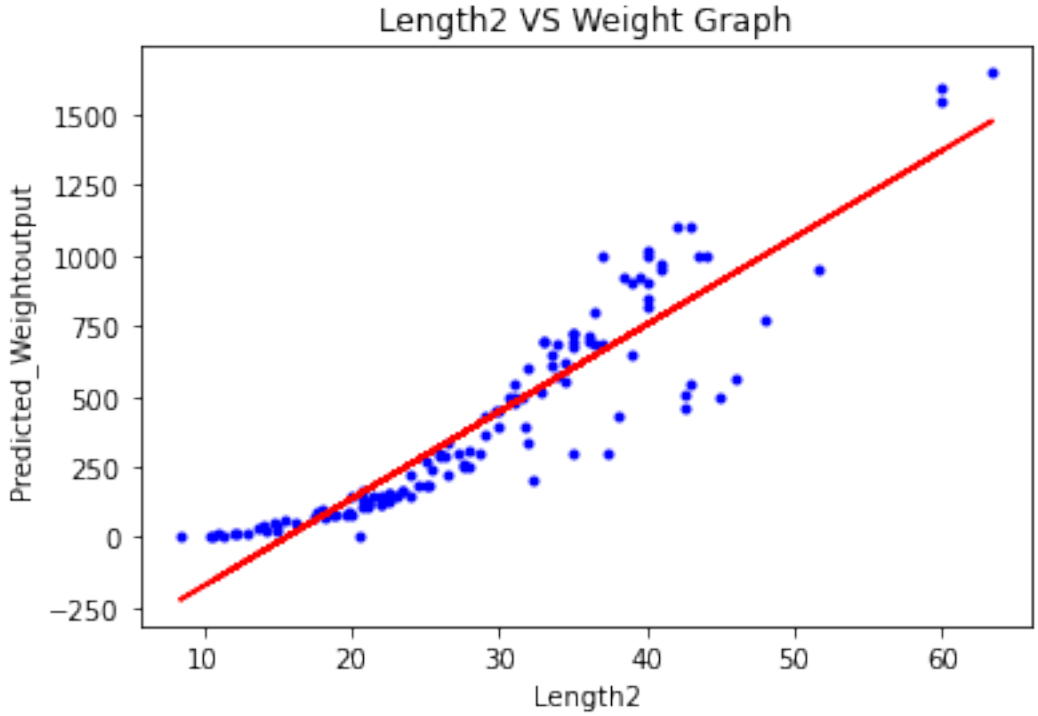

Fig3. Length2 VS Weight

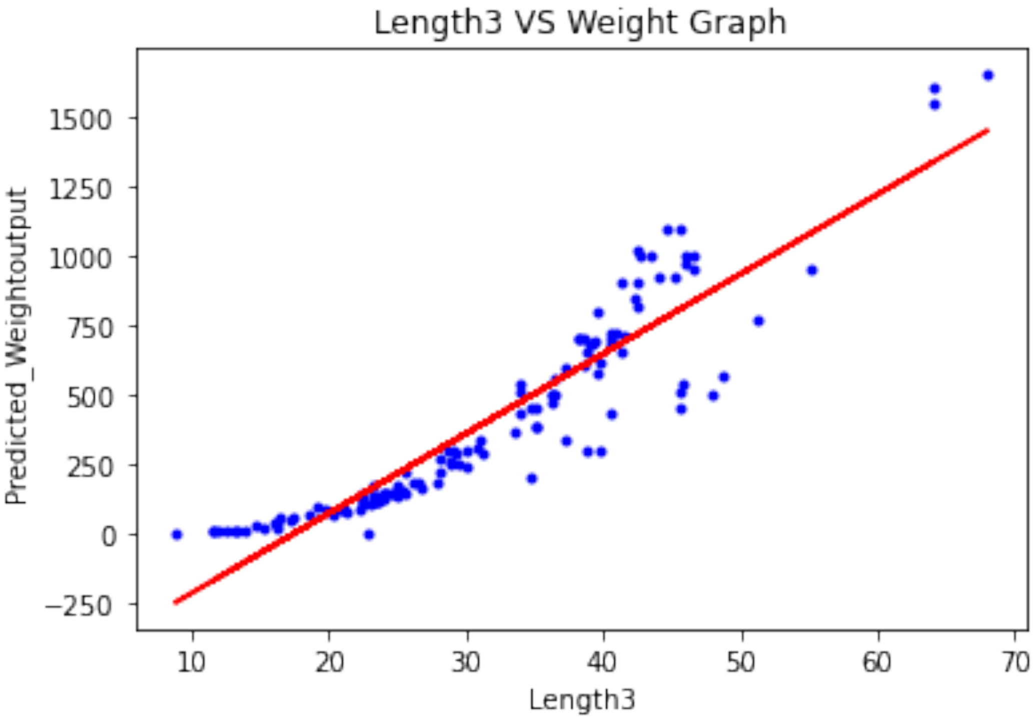

Fig4. Length3 VS Weight

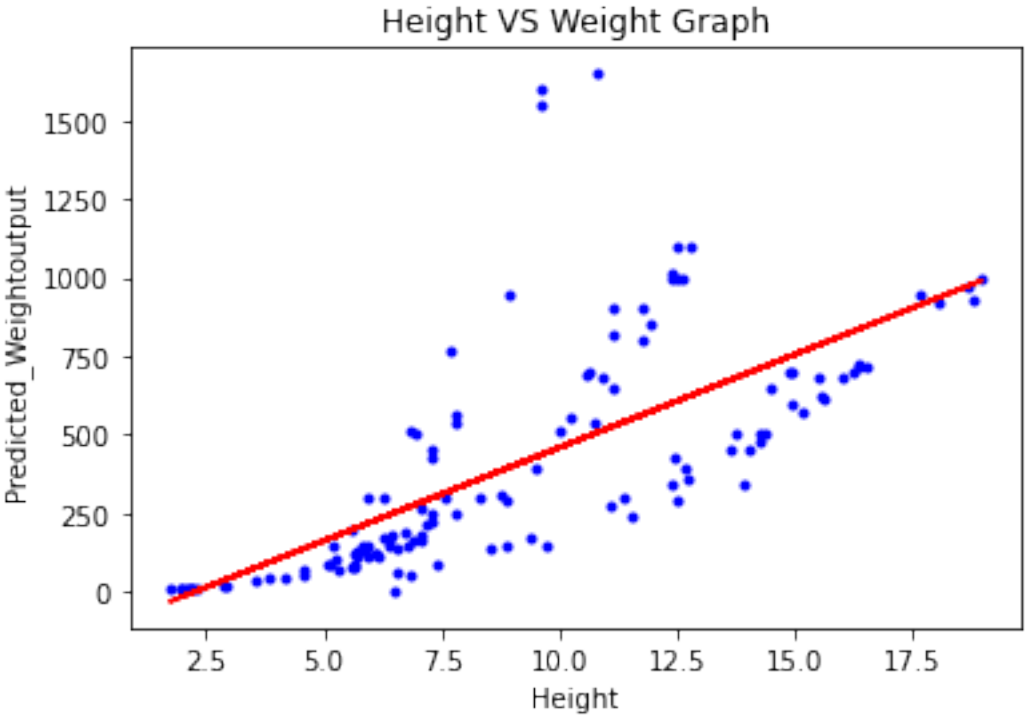

Fig. 5 Height VS Weight

Question6. Plot the RSS vs input_features for training and test data. Which model has lowest RSS on TEST data? Think about why this might be the case.

Here, we calculated RSS values for the training and testing data individually. To calculate the RSS value we used function Residual Squares of sum which we created above.

Below code is calculating the RSS value for the input_feature Width using training and testing data.

```
Testdata_rss_width=Residual_Squares_Of_Sum(x_test['Width'], y_test,Width_intercept , Width_slope)
print('Testdata_rss_width:',Testdata_rss_width)
Traindata_rss_width=Residual_Squares_Of_Sum(x_train['Width'],y_train,Width_intercept , Width_slope)
print('Traindata_rss_width:',Traindata_rss_width)
```

Below is the output of RSS values for different input_features.

RSS Value of Length1 training and testing data.

```
Testdata_rss_length1: 606299.8714113776
Traindata_rss_length1: 2663645.009040391
```

RSS Value of Lengt2 training and testing data.

```
Testdata_rss_length2: 600982.9979569095
Traindata_rss_length2: 2561732.472479767
```

RSS Value of Length3 training and testing data.

```
Testdata_rss_length3: 572446.601925842
Traindata_rss_length3: 2425728.299909344
```

RSS Value of Width training and testing data.

```
Testdata_rss_width: 782346.5345385217
Traindata_rss_width: 3574638.37090363
```

RSS Value of Height training and testing data.

```
Testdata_rss_height: 1275145.8051637989
Traindata_rss_height: 8352566.939195981
```

After calculating the RSS value we plot RSS Vs input_feature graph for both training and testing data. Where on X-axis we put input_features and on Y-axis We put different RSS values for all features.

Below graph is for RSS Vs TrainData.

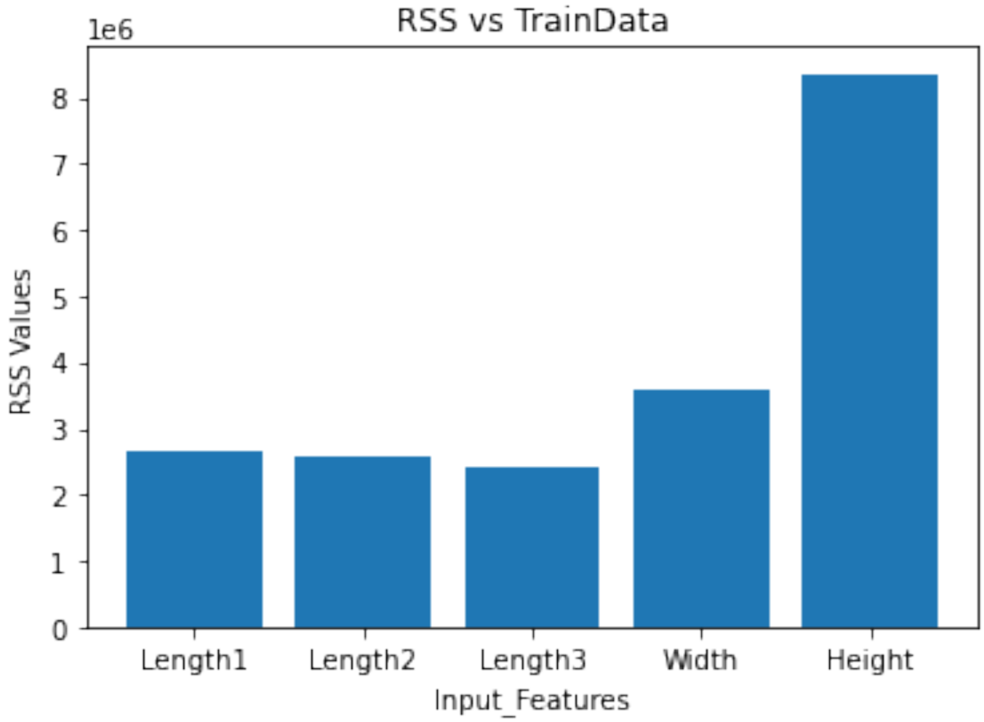

Fig. 6 RSS vs TrainData

Below graph is for RSS Vs TestData.

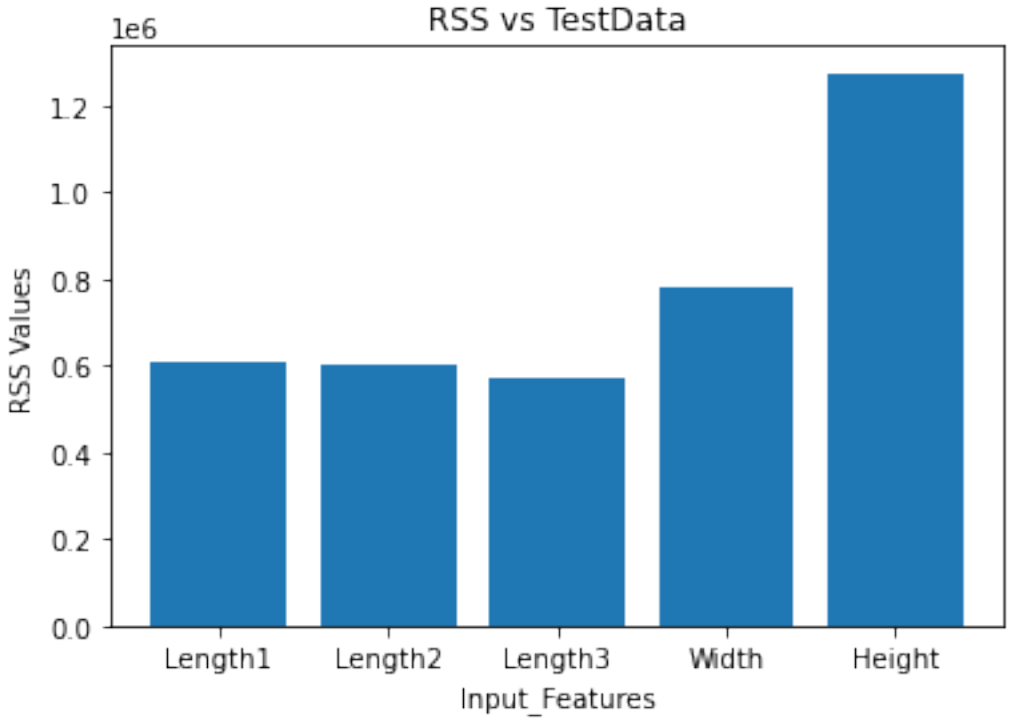

Fig. 7 RSS vs TestData

From the above graph we can say that we got minimum RSS value for the input_feature Column Length3 for both training and testing dataset.As per my opinion, Length3 RSS values are minimum because RSS calculating based on Slope and intercept so it is depends on the slope and intercept value.

Question7. Will model improve if we take two or more 'input_features' at a time? In any case give reason.

Yes, As per my research and implementation, model accuracy will be the increase by adding the two or more input_feature at a time. Because, when input_feature data is increasing it will increase the training data and higher the training data that means lower training and testing error. Because adding more input features or columns decreasing overfitting. Overfitting only happens in model when model considering noise and details both from the training dataset. Higher training data also increase the model complexity.

REFRENCES

[1] Wiriyapong, B., & TH, K. (2020). bensw. Retrieved 20 October 2020, from http://www.bensw.xyz/regression/Simple-Linear-Regression/

[2] tuanavu/coursera-university-of-washington. (2020). Retrieved 20 October 2020, from https://github.com/tuanavu/coursera-university-of-washington/blob/master/machine_learning/2_regression/assignment/week1/README.MD

[3] Linear Regression in Python with Scikit-Learn. (2020). Retrieved 20 October 2020, from https://stackabuse.com/linear-regression-in-python-with-scikit-learn/

[4] What impact does increasing the training data have on the overall system accuracy? https://stats.stackexchange.com/questions/31249/what-impact-does-increasing-the-training-data-have-on-the-overall-system-accuracy