PREDICTION OF HUMAN DEVELOPMENT INDEX 2015

Project Report
Machine Learning 2019

K.P. PERAMUNUGAMA – IT16122024

BSc. Special (Honors) Degree in Information Technology Specializing Software Engineering

Sri Lanka Institute of Information Technology
Sri Lanka

May 2019

TABLE OF CONTENT

LIST OF FIGURES 3
LIST OF TABLES 3
LIST OF EQUATIONS 3
1 INTRODUCTION 4
1.1 PROBLEM 5
1.1.1 DATA EXPLORATION 5
1.2 SOLUTION 6
1.3 DATASET OVERVIEW 6
1.4 DATASET SOURCE 6
2 METHODOLOGY 7
2.1 DATASET ATTRIBUTES 7
2.2 DATA PROCESSING 8
2.2.1 ENCODING CATEGORICAL DATA 8
3 APPLICATION OF THE APPROPRIATE LEARNING ALGORITHM 9
3.1 INTRODUCTION AND BACKGROUND OF THE ALGORITHM 9
3.2 WHY LINEAR REGRESSION? 9
3.3 HYPOTHESIS 9
3.4 COST FUNCTION 10
3.5 GRADIENT DESCENT 10
3.6 APPLICATION OF THE ALGORITHM 11
4 RESULTS 12
5 CRITICAL ANALYSIS AND DISCUSSION 16
5.1 POSIBLE LIMITATIONS 16
5.2 FUTURE WORKS AND WAY OF IMPROVING THE ACCURACY 16
5.2.1 Split the data into different size of test and training sets and check the accuracy. 16
5.2.2 Split dataset with and without randomizing the chucks. 17
6 APPENDIX 18
7 REFERENCES 23

LIST OF FIGURES

Figure 1: HDI distribution of Sri Lanka 4
Figure 2 : HDI versus Country Figure 3 : Expected education versus Country 5
Figure 4 : GNI per capita versus Country Figure 5 : Life expectance at birth versus Country 5
Figure 6 : Mean years of education versus Country 6
Figure 7 : Statistical information of the initial dataset 6
Figure 8 : Encoded Countries 8
Figure 9 : Distribution of Human Development Index 12
Figure 10 : Correlation matrix 12
Figure 11 : Predicted Y versus Tested Y 13
Figure 12 : Predicted Y values 13
Figure 13 : Coefficients 14

LIST OF TABLES

Table 1 : Dataset Attributes 8

LIST OF EQUATIONS

Equation 1 : Hypothesis of MLR 9
Equation 2 : Cost function of MLR 10
Equation 3 : Gradient descent of MLR 10

1 INTRODUCTION

The ***Human Development Index (HDI)*** is a statistical tool used to measure a country's overall achievement in its social and economic dimensions. The social and economic dimensions of a country are based on the health of people, their level of education attainment and their standard of living.

Calculation of HDI

- Health - Life expectancy at birth
- Education - expected years schooling for school-age children and average years of schooling in the adult population
- Income - measured by Gross National Income (GNI) per capita (PPP US$)

Above three dimensions combined to calculate the Human Development Index (HDI). The value of the Human Development Index (HDI) is between zero and one. Very high, high, medium and low are four main tiers based on the HDI. A country is in the very high tier if its HDI is in the top quartile and the low tier if it's HDI in the bottom quartile. [1]

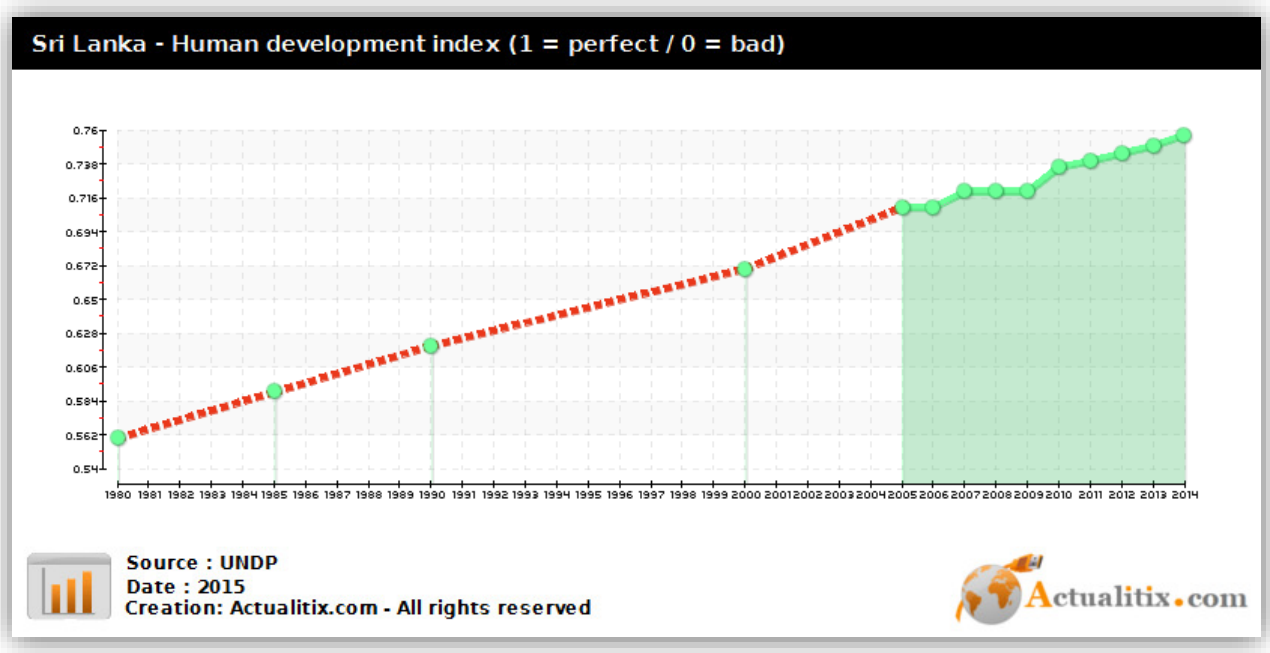

Figure 1: HDI distribution of Sri Lanka

Above figure shows the distribution of the Human Development Index in Sri Lanka. This shows distribution of 1980 – 2014, referring this diagram, we can identify distribution of the HDI with the year. In 1980 HDI was very low value and in 2014 it increased to near value of 0.76. Therefore we can assume the other three dimensions are increased in continuously.

1.1 PROBLEM

While exploring this dataset it is clearly visible the relationship between Human Development Index (HDI) with the other dependent variables. (Life Expectancy at Birth, Expected years schooling for school-age children and average years of schooling in the adult population, Gross National Income (GNI) per capita (PPP US$)) As shows in the below graphs these dependent variables takes a high value in the countries where Happiness score is a high value.

In 2015 Sri Lanka takes 73rd place competing 188 countries in Human Development Index (HDI), which is not better thing for the country. The reason for this is the lack of development in above mentioned dependent dimensions. To increase the Human Development Index (HDI) of our country it is a must to develop the above-mentioned areas.

1.1.1 DATA EXPLORATION

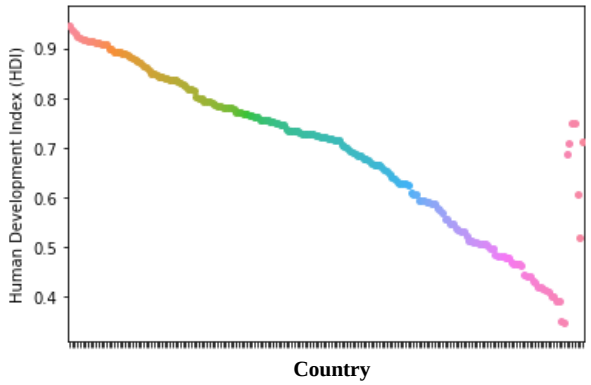

Figure 2 : HDI versus Country

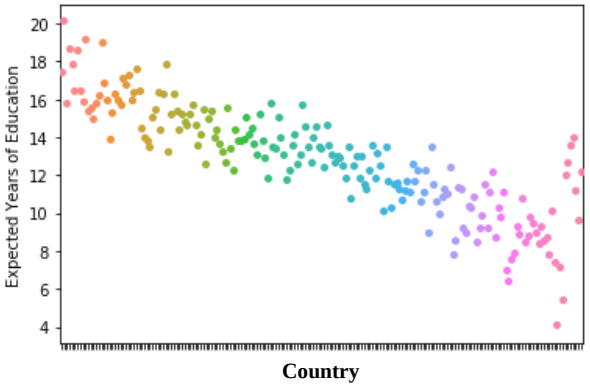

Figure 3 : Expected education versus Country

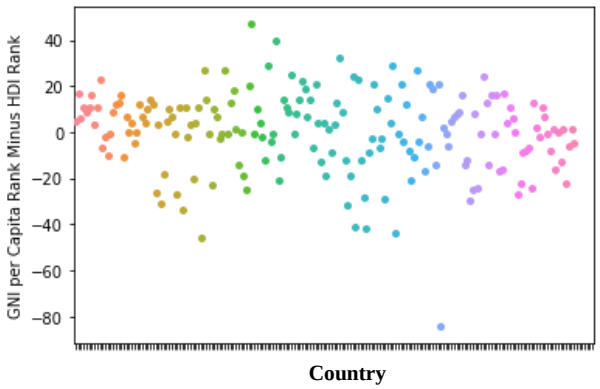

Figure 4 : GNI per capita versus Country

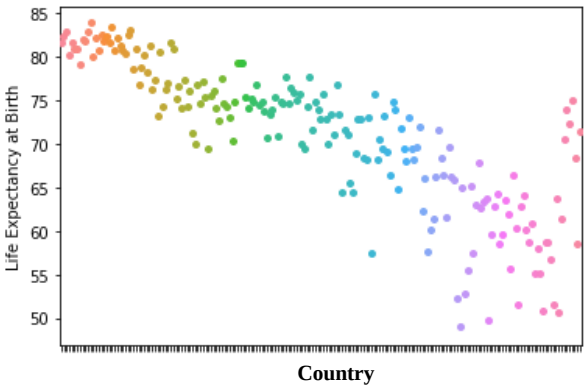

Figure 5 : Life expectance at birth versus Country

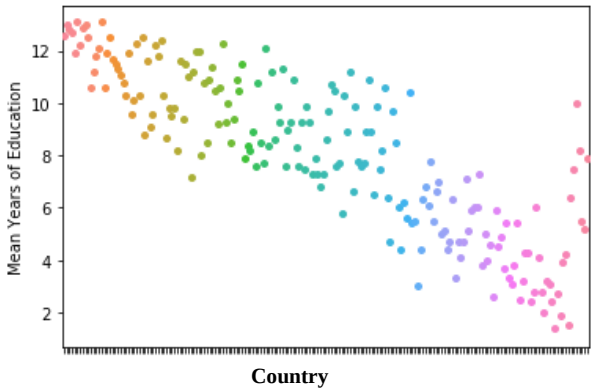

Figure 6 : Mean years of education versus Country

1.2 SOLUTION

In the perspective of Machine learning, for a real world problem like this we can use Linear regression algorithm to predict Human Development. Prediction of Human Development will help the countries to analyze and identify dependent variables and do required steps to develop their Human Development Index (HDI).

1.3 DATASET OVERVIEW

Out[2]:

	HDI Rank	Country	Human Development Index (HDI)	Life Expectancy at Birth	Expected Years of Education	Mean Years of Education	Gross National Income (GNI) per Capita	GNI per Capita Rank Minus HDI Rank
0	1.0	Norway	0.944	81.6	17.5	12.6	64,992	5.0
1	2.0	Australia	0.935	82.4	20.2	13.0	42,261	17.0
2	3.0	Switzerland	0.930	83.0	15.8	12.8	56,431	6.0
3	4.0	Denmark	0.923	80.2	18.7	12.7	44,025	11.0
4	5.0	Netherlands	0.922	81.6	17.9	11.9	45,435	9.0

Figure 7 : Statistical information of the initial dataset

This dataset is taken from the United Nations Development Program. This dataset is released under the public domain.

1.4 DATASET SOURCE

Source: Kaggle

Name: Human Development Report 2015

URL: https://www.kaggle.com/undp/human-development

2 METHODOLOGY

2.1 DATASET ATTRIBUTES

Column Name	Data Type	Description
Country	String	Name of the Country
HDI Rank	Numeric	Rank of the Country based on the Human Development Index
Human Development Index (HDI)	Numeric	The Human Development Index (HDI) is a statistical tool used to **measure** a country's overall achievement in its social and economic dimensions. The social and economic dimensions of a country are based on the health of people, their level of education attainment and their standard of living.
Life Expectancy at Birth	Numeric	Average number of years that a newborn is expected to live if current mortality rates continue to apply.
Expected Years of Education	Numeric	**Expected years of schooling** are the number of **years** during which a 2-**year**-old child can expect to spend in **schooling**, based on the **school** enrolment rates at a given date
Mean Years of Education		Average number of completed years of education of a country's

		population aged 25 years and older, excluding years spent repeating individual grades.
Gross National Income (GNI) per Capita	Numeric	**GNI per capita** is **gross national income** divided by midyear population.
GNI per Capita Rank Minus HDI Rank	Numeric	Difference in **rankings** by **GNI per capita** and by the **HDI**. A negative value means that the country is better ranked by **GNI** than by the **HDI**.

Table 1 : Dataset Attributes

2.2 DATA PROCESSING

2.2.1 ENCODING CATEGORICAL DATA

Since this is a linear regression model, there can be only numerical data as variables. But in the dataset there is a categorical variable 'Country'. So, values in this variable has to be encoded to use in a linear regression model.

LabelEncoder() and OneHotEncoder() methods are used to do this task. Country column's data passed into these methods to encode these data.

```
In [12]: #Encoding categorical data
         from sklearn.preprocessing import LabelEncoder , OneHotEncoder
         x=pd.get_dummies(X,columns=['Country'])
         print(x.values)

         [[  1.      0.944  81.6   ...   0.      0.      0.   ]
          [  2.      0.935  82.4   ...   0.      0.      0.   ]
          [  3.      0.93   83.    ...   0.      0.      0.   ]
          ...
          [186.      0.391  63.7   ...   0.      0.      0.   ]
          [187.      0.35   50.7   ...   0.      0.      0.   ]
          [188.      0.348  61.4   ...   0.      0.      0.   ]]
```

Figure 8 : Encoded Countries

3 APPLICATION OF THE APPROPRIATE LEARNING ALGORITHM

3.1 INTRODUCTION AND BACKGROUND OF THE ALGORITHM

Linear Regression is a supervised Machine learning algorithm which is used for regression problems (Numerical). This is used to predict the relationship between independent and dependent variables. Multiple linear regression means, linear regression with multiple variables. Some of the applications for multiple linear regression are prediction of economic growth of a country, prediction of product prices with the time, estimation of housing sales etc.

3.2 WHY LINEAR REGRESSION?

In this problem the HDI is based on the three main dimensions. Therefore Human development index will depend on Life expectancy at birth, expected years schooling for school-age children and average years of schooling in the adult population, measured by Gross National Income (GNI) per capita (PPP US$). Increasing and decreasing of mentioned three dimensions are continuously cause for the value of the Human Development Index. We are used the Linear Regression algorithm for this type of problems. Therefore we have to use Linear Regression algorithm to go through this.

3.3 HYPOTHESIS

$$h_\theta(x) = \theta_0 x_0 + \theta_1 x_1 + \theta_2 x_2 + \ldots + \theta_n x_n$$

Parameters: $\theta = \{\theta_0, \theta_1, \theta_2, \ldots . \theta_n\}$

Features: $x = \{x_0, x_1, x_2, \ldots . x_n\}$

Equation 1 : Hypothesis of MLR

Actually there are no relationship between X variables and the Y variables. In here we can use this hypothesis for the linear regression algorithm.

3.4 COST FUNCTION

Parameters:

$$\theta_0, \theta_1$$

Cost Function:

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2$$

Goal:

$$\underset{\theta_0, \theta_1}{minimize} J(\theta_0, \theta_1)$$

Equation 2 : Cost function of MLR

There are more than one independent variable, we used below equation as the cost function of the algorithm. If there has more than one independent variable we have to select multiple linear regression to solve the problem. This multiple linear regression algorithm generalized version of the linear regression.

3.5 GRADIENT DESCENT

repeat until convergence: {

$$\theta_j := \theta_j - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) \cdot x_j^{(i)} \qquad \text{for j} := 0...\text{n}$$

}

Equation 3 : Gradient descent of MLR

Gradient Descent is the process of minimizing a function by following the gradient of the cost function.

3.6 APPLICATION OF THE ALGORITHM

In this dataset can identify three main independent variables and one dependent variables. Independent variables are Life expectancy at birth, expected years schooling for school-age children and average years of schooling in the adult population, measured by Gross National Income (GNI) per capita (PPP US$). Dependent variable is the Human Development Index. Here dependent variables are denoted by X and independent variable is denoted by y.

This dataset is split into two parts, training dataset and test dataset. 70% of the data is divided as training data and 30% of the data is divided into test data. Training dataset is used to train the machine learning model and test dataset is used to test and evaluate the accuracy of trained model.

An instance of LinearRegression() class is called as the model to train data. X_train and y_train parameters are passed into these model with the help if fit() method.

4 RESULTS

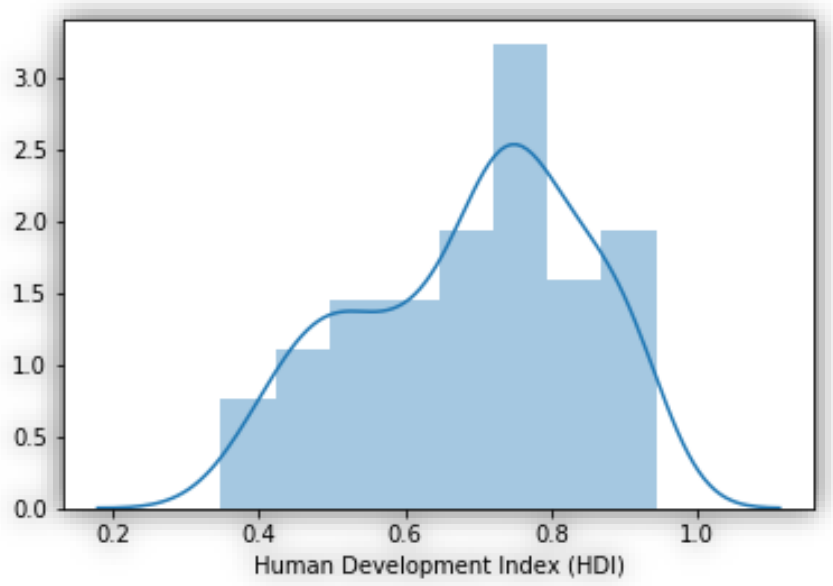

Figure 9 : Distribution of Human Development Index

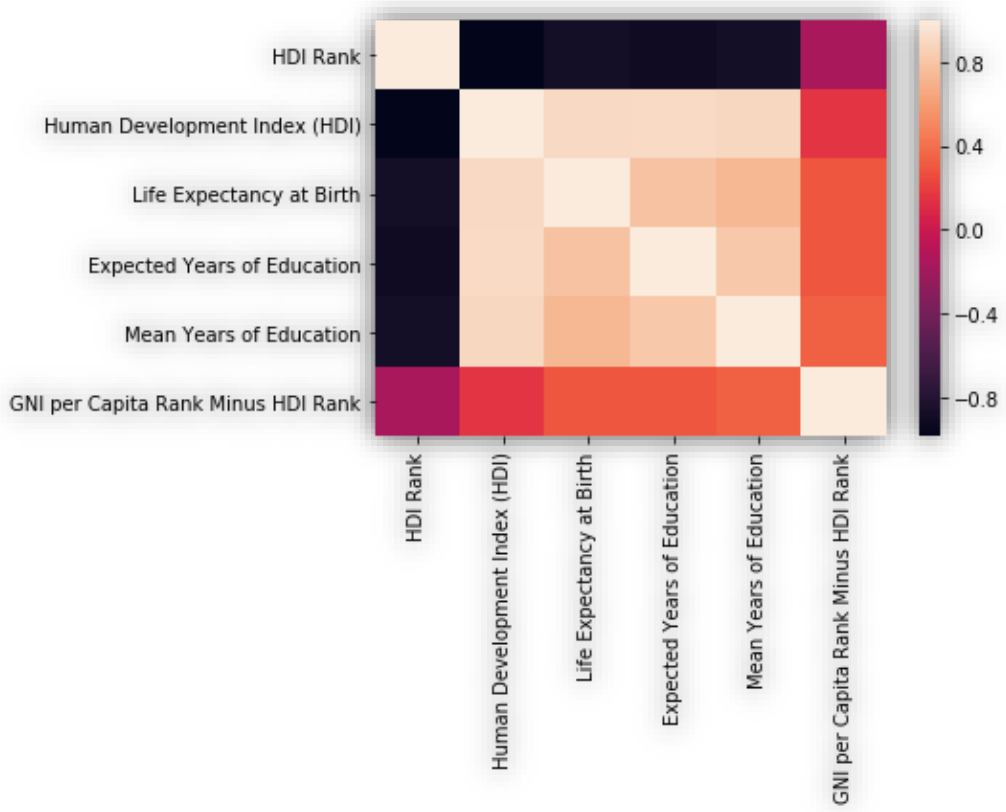

Figure 10 : Correlation matrix

After training the Linear Regression model can evaluate the accuracy of the model with test data. To do that predict () method will be used and X_test is passed as a parameter.

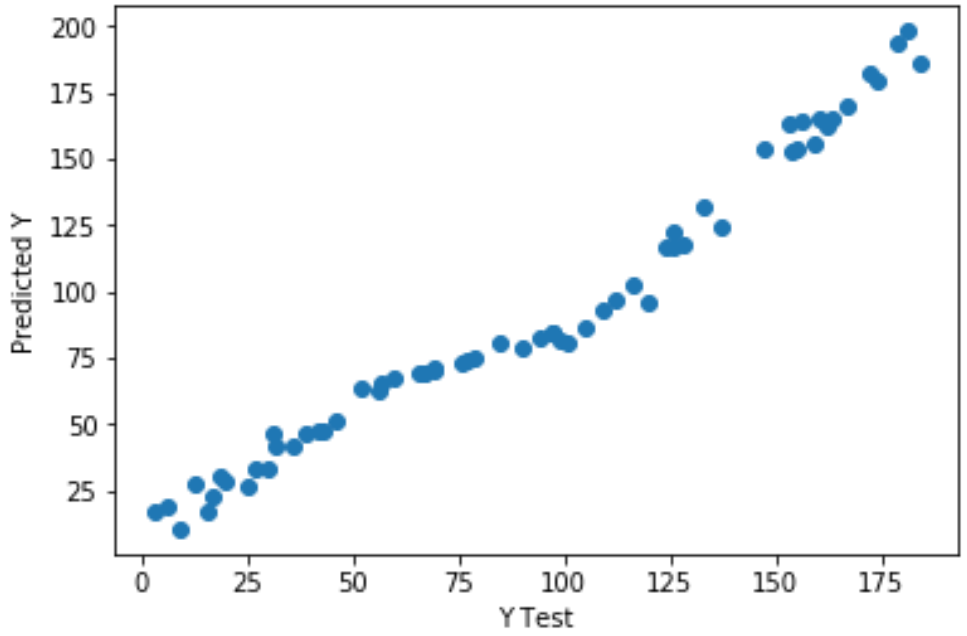

Figure 11 : Predicted Y versus Tested Y

Above figure shows the Predicted Y values versus Tested Y values.

```
In [18]: y_pred=reg.predict(x_test)
         print(y_pred)

[186.09839857 165.40236552  30.35444207  17.18369938  69.02839693
  93.27837078  50.91975799  74.11905325  23.15900895 154.03231071
  67.65297617 117.10015177  47.19370132 163.94883392  62.30919929
  69.39962277 102.78643603 155.72679701  46.3709334   96.0167858
  71.65533562 179.77408326 124.28098707  28.39388094  64.06008477
 193.32441586  18.64258661 182.06594094  26.99632801  84.14821351
 169.87271687  10.58223583 163.30640758  75.04410584 161.84812281
  97.02953213  69.64775707 131.74610393  42.11823978  81.48346595
  85.98881565  72.8012017   82.1494098   70.59767991  33.72910706
 117.52562778 164.71214518  65.6046044   41.75618261  27.88225548
  47.62209484  80.87590923 122.09980012  84.90037478 116.52765114
  33.35655398  46.96915236 152.91538892 153.50344942 198.11619109
  81.11385846  78.86888525  17.00679478]
```

Figure 12 : Predicted Y values

Above figure shows the predicted Y values.

In [19]:
```
#calculating the coefficients
print(reg.coef_)
```

```
[ 4.42002308e+00 -2.56650113e+00 -6.12862571e+00 -7.10205383e+00
  5.45734786e-01 -2.87564125e+00  1.78057435e+00  1.66331774e+00
 -1.07189213e+01 -9.88534712e+00 -8.84468575e+00 -4.95159160e+00
 -2.66453526e-14  5.81448568e+00 -6.72818313e+00  4.48564053e+00
 -9.08729161e+00 -6.50183414e+00  7.64019181e+00  1.42108547e-14
 -7.11179887e+00 -5.79065551e+00  5.32907052e-15 -2.07355434e+00
  5.12211774e+00  1.79377999e+01 -3.55078599e-01  2.02654398e+01
  8.73943200e-01 -4.88498131e-15 -7.96257923e+00 -1.36624585e+01
 -7.10542736e-15  1.35510992e+01 -5.34555675e-02 -2.88657986e-15
 -1.11935561e+01 -2.13963380e+01 -2.38292265e+01  2.66453526e-15
  8.31803927e+00  8.88178420e-15 -1.77635684e-14  1.89141388e+00
  1.10295850e+01 -3.55271368e-15 -5.78180517e+00 -1.24344979e-14
  1.06581410e-14 -2.09098900e+00 -5.32907052e-15 -5.82792253e+00
 -7.11710004e+00 -5.32907052e-15  1.69070063e+01  6.36399593e+00
  1.32073006e+01  4.44089210e-15  1.29166274e+01 -6.52225663e+00
  7.21644966e-15  8.88178420e-16  6.66133815e-15 -4.95886861e+00
 -6.63065151e+00  1.68041456e+01 -7.91666925e+00 -1.24344979e-14
  3.55271368e-15  5.38334965e+00 -4.19863223e+00 -4.44089210e-15
 -1.59872116e-14 -8.92232381e+00 -1.69763641e+01  2.66453526e-15
  1.77635684e-15  8.82166292e+00 -1.05750262e+01 -1.91137176e+00
  8.88178420e-16  3.68929978e+00  1.19880549e+01  1.15463195e-14
  1.72091445e+01 -8.89299499e+00 -6.74981349e+00  1.77635684e-15
 -8.88178420e-16  1.77635684e-15  3.27867008e+00 -2.99760217e-15
 -2.45942781e+00 -7.21644966e-16 -3.10862447e-15 -8.70960271e+00
  7.99360578e-15  1.72903829e+00 -3.55271368e-15  3.66373598e-15
 -5.10330926e+00  4.53959670e+00  1.21961644e+01 -8.88178420e-16
 -4.20901452e+00  0.00000000e+00 -3.99680289e-15  1.14628640e+01
 -3.61112538e+00  1.88398242e+01 -1.33226763e-15 -8.27924246e+00
  4.44089210e-15 -3.89258886e+00  2.21209202e-01  1.62708921e+01
  1.72402200e+01  7.86598486e+00 -1.13941316e+01  0.00000000e+00
 -1.16273944e+01 -7.24322230e+00  8.88178420e-16  1.52635223e+00
 -1.07234750e+01  0.00000000e+00  1.06046676e+01 -1.98585988e+01
 -1.49831820e+01 -1.01177890e+01  0.00000000e+00  0.00000000e+00
 -5.74871129e+00  1.81456259e+01  0.00000000e+00 -4.32172992e+00
  0.00000000e+00  1.67435671e+00  1.29847611e+01  0.00000000e+00
  0.00000000e+00 -1.53150619e+01 -1.18960589e+01 -8.13420414e+00
 -1.34607090e+00  0.00000000e+00  3.77160196e+00  0.00000000e+00
  0.00000000e+00 -8.93476065e-01  0.00000000e+00 -1.23018328e+00
  0.00000000e+00 -2.92910739e+00  0.00000000e+00 -1.69086509e+01
 -6.82776587e+00  0.00000000e+00  8.10830510e-01  1.94794472e+01
 -1.06497123e+01 -5.87408246e+00 -3.65691167e+00  0.00000000e+00
  2.19252744e+01 -3.64698813e+00 -6.10962933e+00  0.00000000e+00
  8.28765616e+00  2.17442789e+01 -6.62223944e+00  1.20348059e+01
  4.09356639e+00  0.00000000e+00  0.00000000e+00  2.19114528e+01
 -2.67265926e+00  1.16336809e+01  9.03463232e-01  0.00000000e+00
 -4.23727475e+00  8.69197621e+00 -7.91484983e+00 -5.86322477e+00
 -1.23124011e+01 -1.22101121e+01  1.86194626e+01  8.41444153e+00
 -9.87373629e-01  1.52271445e+01  0.00000000e+00  8.79518355e+00
  0.00000000e+00]
```

Figure 13 : Coefficients

Above numbers show the coefficients for each of the variable.

The intercept value which is created after the prediction: 410.3799336205335

The R squared value: 0.9673091552126711

Finally ***'0.9673091552126711'*** is the R squared value in this prediction. Referring this value we can get idea about used algorithm. Finally got more accurate R squared value by using the Linear Regression algorithm. Therefore, we can justify the most suitable and most accurate algorithm for solving this problem is Multiple Linear Regression model. To get this R Squared value, y_test and y_pred are passed as parameters in to r2_score method. [2]

5 CRITICAL ANALYSIS AND DISCUSSION

5.1 POSIBLE LIMITATIONS

These data collected through a survey from sample set of people. So, these data may not be the most correct data.

5.2 FUTURE WORKS AND WAY OF IMPROVING THE ACCURACY

5.2.1 Split the data into different size of test and training sets and check the accuracy.

-

```
#Train and test split
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.30)
```

```
#calculating the R squared value
from sklearn.metrics import r2_score
r2_score(y_test, y_pred)
```

```
0.9717315693399874
```

R squire value when splitting data by 70% for training 30% for testing

ACCURACY INCREASED

-

```
#Train and test split
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.20)
```

```
#calculating the R squared value
from sklearn.metrics import r2_score
r2_score(y_test, y_pred)
```

```
0.971618012244791
```

R squire value when splitting data by 80% for training 20% for testing

ACCURACY DECREASED

Above results shows that increasing the test size increases the accuracy.

5.2.2 Split dataset with and without randomizing the chucks.

-

```
#Train and test split
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.20)
```

```
#calculating the R squared value
from sklearn.metrics import r2_score
r2_score(y_test, y_pred)
```

```
0.971618012244791
```

R squire value when splitting data by 80% for training 20% for testing

ACCURACY INCREASED

-

```
#Train and test split
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.20, random_state=50)
```

```
#calculating the R squared value
from sklearn.metrics import r2_score
r2_score(y_test, y_pred)
```

```
0.9709764682324803
```

R squire value when splitting data by 80% for training 20% for testing

ACCURACY DECREASED

Above results shows that by randomizing the data set chucks accuracy decreases.

6 APPENDIX

```
#!/usr/bin/env python
# coding: utf-8

# In[1]:


#Importing the libraries
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import seaborn as sns
get_ipython().run_line_magic('matplotlib', 'inline')


# In[2]:


#Importing the dataset
#Extracting the independent and dependent variables
Development = pd.read_csv("human_development.csv")

#Independent Variables
X = Development.iloc[:,[0,1,2,3,4,5,7]]

#Dependant Variable
y = Development.iloc[:,2].values

Development.head()


# In[3]:


#Fixing the error
X.columns=X.columns.str.strip()


# In[4]:


#Data Exploration
#Development Score

g = sns.stripplot(x="Country", y="Human Development Index (HDI)",
data=Development, jitter=True)
plt.xticks(rotation=90)
```

```
# In[5]:


#Data Exploration
#Life Expectancy at Birth

g = sns.stripplot(x="Country", y="Life Expectancy at Birth",
data=Development, jitter=True)
plt.xticks(rotation=90)


# In[6]:


#Data Exploration
#Expected Years of Education

g = sns.stripplot(x="Country", y="Expected Years of Education",
data=Development, jitter=True)
plt.xticks(rotation=90)


# In[7]:


#Data Exploration
#Mean Years of Education

g = sns.stripplot(x="Country", y="Mean Years of Education",
data=Development, jitter=True)
plt.xticks(rotation=90)


# In[8]:


#Data Exploration
#GNI per Capita Rank Minus HDI Rank

g = sns.stripplot(x="Country", y="GNI per Capita Rank Minus HDI
Rank", data=Development, jitter=True)
plt.xticks(rotation=90)


# In[9]:


#Data visualization

sns.distplot(Development['Human Development Index (HDI)'])
```

```
# In[10]:


#Building the correalation matrix

heat = Development.iloc[:,[0,1,2,3,4,5,7]]
sns.heatmap(heat.corr())


# In[11]:


#Dropping nans (missing values)
X=X.dropna()


# In[12]:


#Encoding categorical data
from sklearn.preprocessing import LabelEncoder , OneHotEncoder
x=pd.get_dummies(X,columns=['Country'])
print(x.values)


# In[13]:


#data and label
c=list(x.columns)
c.remove('HDI Rank')
y=x['HDI Rank']
x=x[c]


# In[14]:


#data and label
x.shape,y.shape


# In[15]:


#Train and test split
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x, y,
test_size=0.33, random_state=42)
```

```
# In[16]:


#Linear Regression
from sklearn.linear_model import LinearRegression
reg = LinearRegression().fit(x_train, y_train)


# In[17]:


#Train and test score
print("Train: ",reg.score(x_train,y_train))
print("Test: ",reg.score(x_test,y_test))


# In[18]:


y_pred=reg.predict(x_test)
print(y_pred)


# In[19]:


#calculating the coefficients
print(reg.coef_)


# In[20]:


#Calculating the intercept
print(reg.intercept_)


# In[21]:


#calculating the R squared value
from sklearn.metrics import r2_score
r2_score(y_test, y_pred)
```

```
# In[22]:


#y_test Values
y_test


# In[23]:


#y_pred values
y_pred


# In[24]:


#Predicited Y Versus Testing Y
plt.scatter(y_test,y_pred)
plt.xlabel('Y Test')
plt.ylabel('Predicted Y')
```

7 REFERENCES

[1] "Human Development Index HDI," 26 March 2009. [Online]. Available: http://wikiprogress.org/articles/initiatives/human-development-index/. [Accessed 12 May 2019].

[2] "Human Developments Reports," UNITED NATIONS DEVELOPMENT PROGRAMME, [Online]. Available: http://hdr.undp.org/en/data. [Accessed 04 May 2019].

[3] "Kaggle," 25 January 2017. [Online]. Available: https://www.kaggle.com/undp/human-development#human_development.csv . [Accessed 02 May 2019].