# Linear Regression on Death Rate

**Assignment Report 2 (7th Sem 2019)**

**Submitted by,**

Aditya Mishra (Roll No. CSJMA16001390002)

Aditya Verma (Roll No. CSJMA16001390003)

Akash Verma (Roll No. CSJMA16001390004)

Amarjeet Yadav (Roll No. CSJMA16001390005)

**For the course, Machine Learning CSE-520**

**In partial fulfillment for the award of the degree of**

**Bachelor of Technology in**

**Computer Science and Engineering**


**University Institute of Engineering and Technology,**

**CSJM University , Kanpur-208024**

## Table of Contents

**SNO. Topic** **Page**

# Problem Statement

Implementation of Linear Regression Model to predict death rate by considering all features . Regularize all features to avoid over fitting (Ridge Regression).Identify best 4 attributes for death rate prediction but in this case we are taking only 4 attributes to predict the Death Rate.

# About the Dataset :

The Death Rate is to be represented as a function of other variable. There are 60 rows and 17 columns including the index. The Dataset contains average annual precipitation, average January temperature, average July temperature, size of population older than 65 and etc.

**Attribute Information :**

1- Average annual precipitation
2- Average January Temperature
3- Average July Temperature
4- Size of Population Older than 65
5- The number of members per household
6- The number of years of schooling for persons over 22
7- The number of households with fully equipped kitchen
8- The population per square mile
9- The size of nonwhite population
10- The number of office workers
11- The number of families with income less than $3000
12- The hydrocarbon pollution index
13- The nitric acid pollution index
14- The Sulphur acid pollution index
15- The degree of atmospheric moisture
16- The Death Rate

## Model Construction

We will be implementing simple mathematics in which we have used the multiple linear regression formulae which has been derived from simple linear regression formulae and predicted the death rate using only 4 attributes.

**Libraries imported are :**

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as seabornInstance import statistics
```

**Formule for Multiple linear regression :**

$$\hat{Y} = b_0 + b_1 X_1 + b_2 X_2 + \ldots + b_p X_p$$

**Code for the prediction of the Death Rate :**

1: Reading the dataset –

```
#read dataset of csv format and label it
X=pd.read_csv('death_rate.csv',names=('A1','A2','A3','A4','A5','A6','A7','A8','A9','A10','A11','A12','A13','A14','A15','B'))
```

2: Extraction of each column :

```
#extract each column and convert it into list
A1 = X['A1']
A1 = list(A1)
A2 = X['A2']
A2 = list(A2)
A3 = X['A3']
A3 = list(A3)
A4 = X['A4']
A4 = list(A4)
B = X['B']
B = list(B)
```

3: Mean of independent and dependent variable :

```
#find mean of each independend and dependent variables
X1=statistics.mean(A1)
X2=statistics.mean(A2)
X3=statistics.mean(A3)
X4=statistics.mean(A4)
Y1=statistics.mean(B)
```

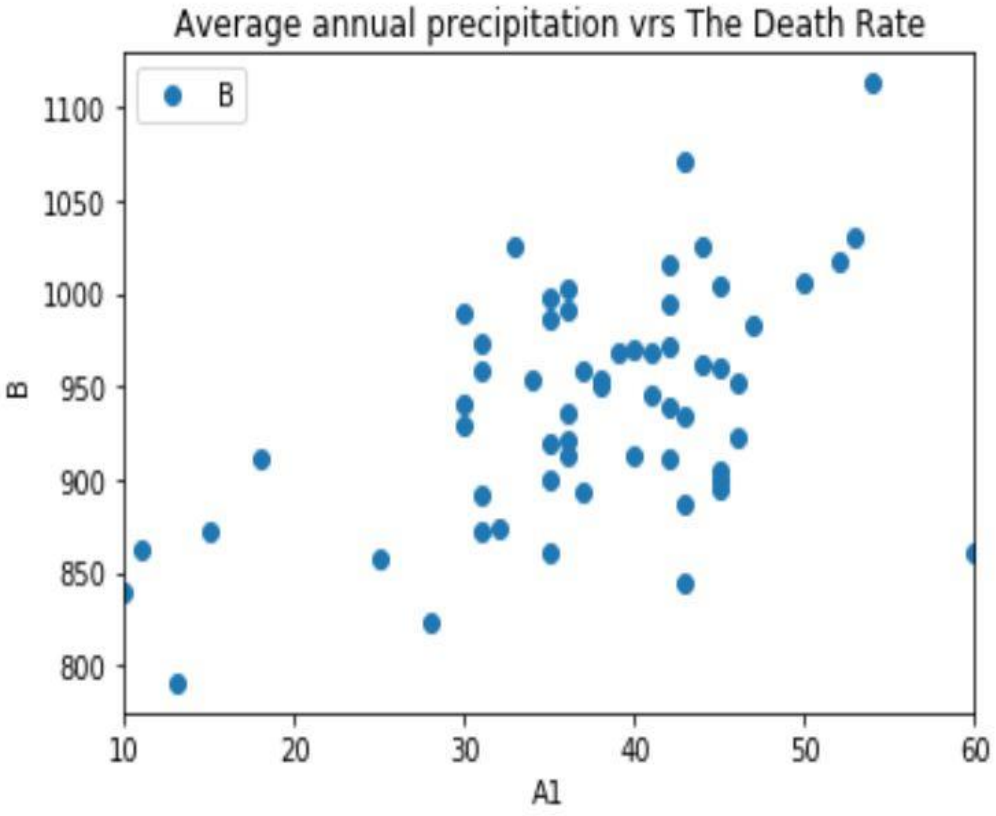
4: Code to find theta function :

```
#function to find theta_1
def fun1(x,y,z):
    lst1=[]
    lst2=[]
    ls1=[]
    ls2=[]
    for i in range(len(x)):
        a=x[i]-z
        lst1.append(a)
    for i in range(len(x)):
        b=y[i]-Y1
        lst2.append(b)

    q=np.array(lst1)
    r=np.array(lst2)
    h=list(q*r)

    s=list(np.square(lst1))
    O1=sum(h)/sum(s)
    O01=Y1-(O1*z)
    ls1.append(O1)
    ls2.append(O01)
    ls1=ls1+ls2
    return ls1
aa=fun1(A1,B,X1)
bb=fun1(A2,B,X2)
cc=fun1(A3,B,X3)
dd=fun1(A4,B,X4)

O0=(aa[1]+bb[1]+cc[1]+dd[1])/4
x1,x2,x3,x4=map(float,input().split())
value=O0+aa[0]*x1+bb[0]*x2+cc[0]*x3+dd[0]*x4
print(value)
```

**Plot graph of attributes vrs Death Rate :**

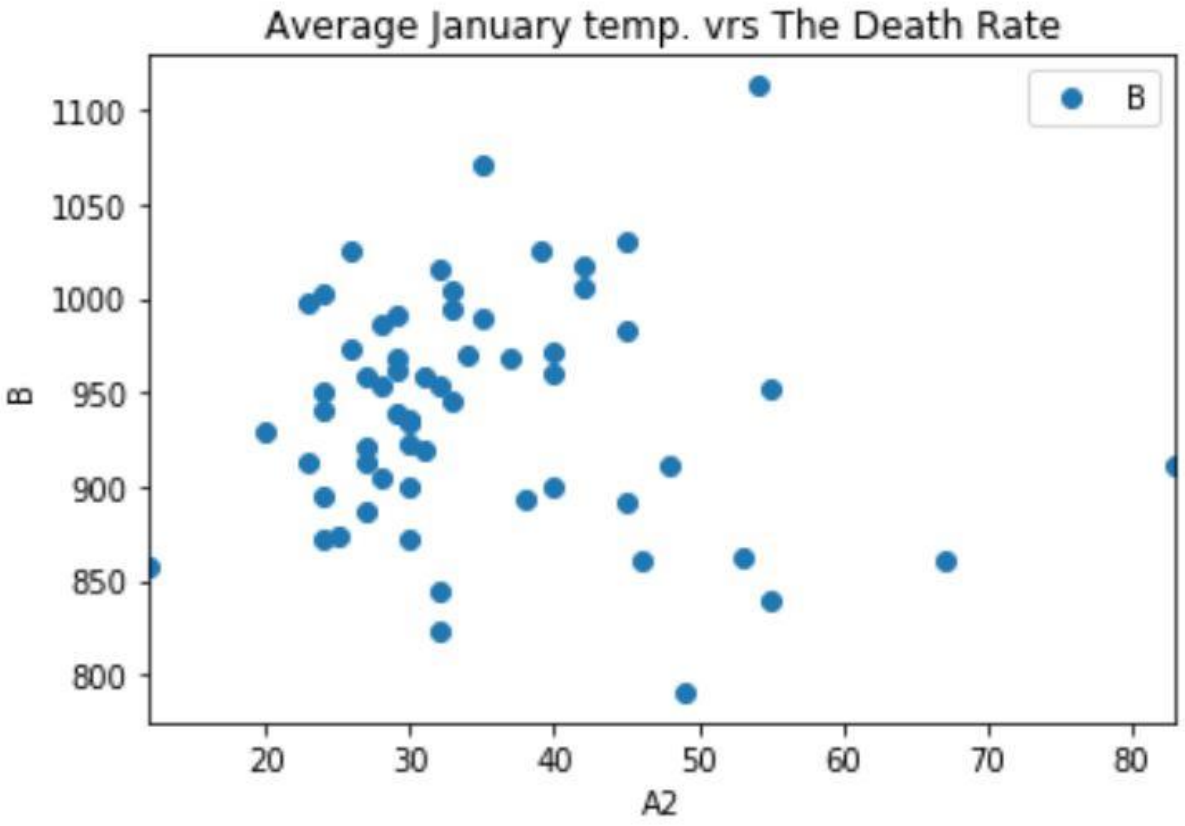
1- Average annual precipitation vrs Death Rate

```
X.plot(x='A1' ,y='B', style='o')
plt.title('Average annual precipitation vrs The Death Rate')
plt.xlabel('A1')
plt.ylabel('B')
plt.show()
```



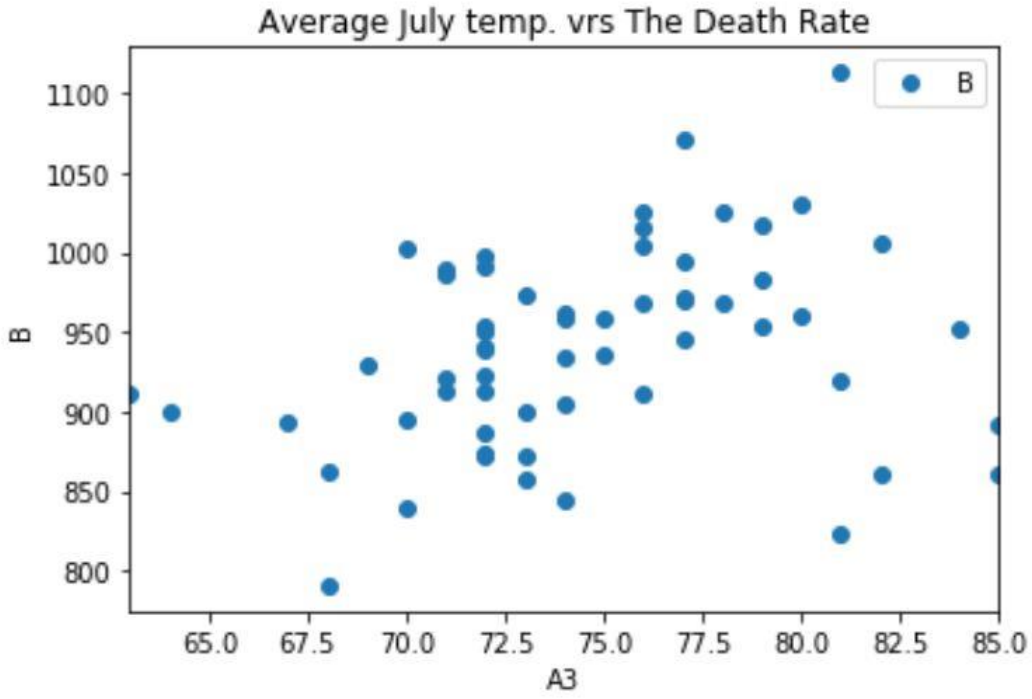
2- Average annual January temperature vrs Death Rate

```
X.plot(x='A2', y='B' ,style='o')
plt.title('Average January temp. vrs The Death Rate')
plt.xlabel('A2')
plt.ylabel('B')
plt.show()
```



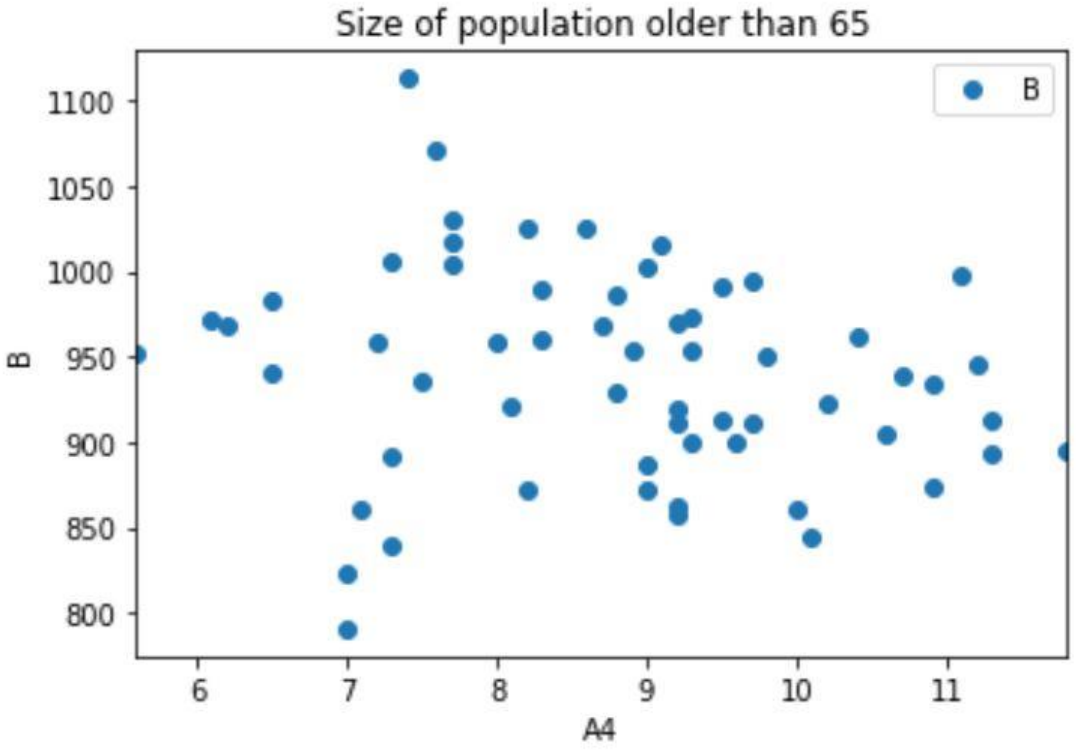
3- Average annual July temperature vrs Death Rate

```
X.plot(x='A3',y='B', style='o')
plt.title('Average July temp. vrs The Death Rate')
plt.xlabel('A3')
plt.ylabel('B')
plt.show()
```



4-Size of population older than 65 vrs Death Rate

```
X.plot(x='A4', y='B', style='o')
plt.title('Size of population older than 65')
plt.xlabel('A4')
plt.ylabel('B')
plt.show()
```



Performance Analysis :

```
In [35]: from sklearn.model_selection import train_test_split
         p=X.iloc[:,:-1]
         q=X['B']
         p_train,p_test,q_train,q_test=train_test_split(p,q,test_size=0.2,random_state=0)
         from sklearn.linear_model import LinearRegression
         regressor=LinearRegression()
         regressor.fit(p_train,q_train)
         #a=[float(x) for x in input().split()]
         #a=[a]
         pred=regressor.predict(p_train)
         print(pred)

         from sklearn.metrics import r2_score
         s=r2_score(q_train,pred)
         print(s)
```

**OUTPUT**

Train Accuracy: 78.23%

Test Accuracy:  35.10

# Conclusion and Future Scope

Machine Learning is not some thing new. It has existed for many decades, right from the days of Statistics. In its earlier form, it was called statistical learning. It is the science of learning from large historical data. It is seeing resurgence these days due to the higher information processing and memory processing ability of today's computing systems. Machine Learning as a concept is here to stay which will have a lot of applications across all industries. Only the technologies, tools, platforms which makes use of Machine Learning evolves. It can be used for pattern recognition, image recognition, speech recognition, predictive Analytics etc. etc. The scope of Machine Learning only keeps increasing.