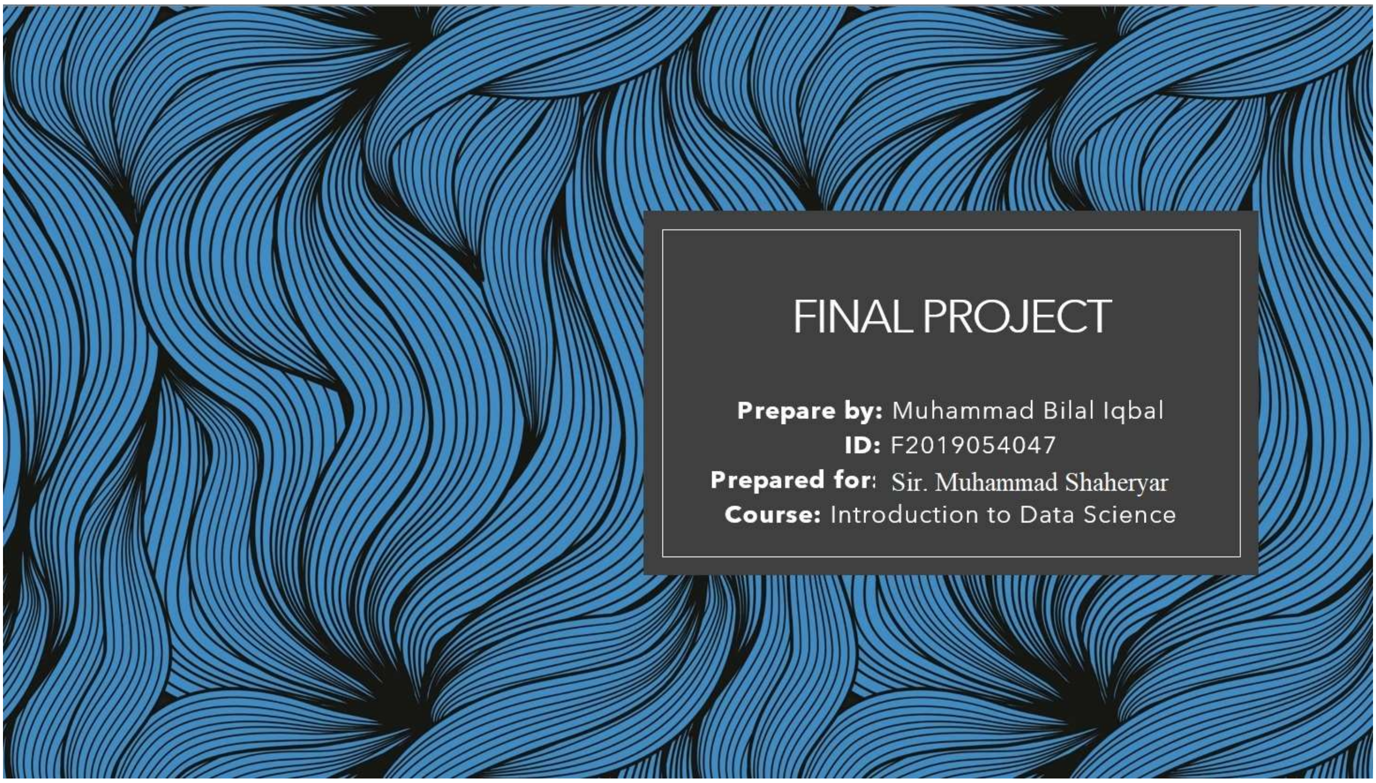


FINAL PROJECT

Prepare by: Muhammad Bilal Iqbal
ID: F2019054047
Prepared for: Sir. Muhammad Shaheryar
Course: Introduction to Data Science

Research Question:

Is there a linear relationship between the expenditure on health care and the Infant Mortality Rate or which indicators can predict the Infant Mortality Rate best ?

Answer:

Some steps are used to statisfy the above question which are as follows;

Step-1: Importing required libraries:-

```
import numpy as np # This library is used for working with arrays.
import pandas as pd # This library is used for data manipulation and analysis.
import matplotlib.pyplot as plt # This library is used for data visualization and graphica
```

Step-2: Upload data file and read it:-

```
# Reading the data from the uploaded file.
sd = pd.read_csv('mortalitystudy.csv')
# It is used to show the data in data frame we created while reading data.
sd.head()
```

	Unnamed: 0	mort_rate	health_exp	immunization	sanitation	fert_rate	pre_undern
0	X19911	0.000000	0.0	0.000000	0.0	49.700000	
1	X19912	100.763466	0.0	53.933170	0.0	128.562197	
2	X19913	116.800000	0.0	19.000000	0.0	163.382200	

```
# It shows the name of columns in index and data type.
sd.columns
```

```
Index(['Unnamed: 0', 'mort_rate', 'health_exp', 'immunization', 'sanitation',
       'fert_rate', 'pre_undernourishment'],
      dtype='object')
```

Step-3: Creating a new dataframe i.e sd1 within an existing dataframe i.e sd:-

```
sd1 = sd[['Unnamed: 0', 'mort_rate', 'health_exp', 'immunization', 'sanitation', 'fert_rat
sd1.head(10) # Printing new dataframe with limit of 10 rows and 10 columns.
```

	Unnamed: 0	mort_rate	health_exp	immunization	sanitation	fert_rate	pre_undern
0	X19911	0.000000	0.0	0.000000	0.0	49.700000	
1	X19912	100.763466	0.0	53.933170	0.0	128.562197	
2	X19913	116.800000	0.0	19.000000	0.0	163.382200	
3	X19914	112.705474	0.0	52.611079	0.0	152.157231	
4	X19915	131.200000	0.0	39.000000	0.0	214.800000	
5	X19916	34.100000	0.0	80.000000	0.0	18.254800	
6	X19917	8.800000	0.0	0.000000	0.0	0.000000	
7	X19918	56.301557	0.0	76.806220	0.0	69.068295	
8	X19919	13.500000	0.0	82.000000	0.0	48.850600	
9	X199110	24.900000	0.0	99.000000	0.0	73.524000	

```
#Length of data before cleaning.
len(sd1)
```

```
7980
```

Step-4: Cleaning the noise in data:-

Now we will create a new dataframe i.e.sd2 without noise.

```
#Removing noise in the data.
sd2 = sd1[sd1.mort_rate != 0]
```

```
#Length of the data after cleaning.
len(sd2)
```

```
6989
```

```
#Reading new data frame wihtout noise.
sd2.head()
```

	Unnamed: 0	mort_rate	health_exp	immunization	sanitation	fert_rate	pre_undern
1	X19912	100.763466	0.0	53.933170	0.0	128.562197	
2	X19913	116.800000	0.0	19.000000	0.0	163.382200	
3	X19914	112.705474	0.0	52.611079	0.0	152.157231	
4	X19915	131.200000	0.0	39.000000	0.0	214.800000	
5	X19916	34.100000	0.0	80.000000	0.0	18.254800	

Now will use SK learn library which is a machine learninig library.

```
from sklearn import linear_model
lreg = linear_model.LinearRegression()
```

```
#Dividing our data for test and training
indx = np.random.rand(len(sd2)) < 0.8 # Selecting 80 % random data from sd2 dataframe
train = sd2[indx] # Training index data
test = sd2[~indx] # Testing non-index data
train.shape , test.shape
```

```
((5609, 7), (1380, 7))
```

Step-5: Testing & Training The Data:-

```
#Train data.
train_x = np.asanyarray(train['health_exp'])
train_y = np.asanyarray(train['mort_rate'])
lreg.fit(train_x.reshape(-1, 1),train_y)
```

```
LinearRegression()
```

```
#Testing data.
test_x = np.asanyarray(test[['health_exp']])
test_y = np.asanyarray(test['mort_rate'])
y_hat = lreg.predict(test_x)
```

Now we will print Theta-0 of coefficient and Theta-1 of interception.

```
print('Theta-0:', lreg.coef_)
print('Theta-1:', lreg.intercept_)
```

```
Theta-0: [-2.50239358]
Theta-1: 44.705525354608596
```

Now we will calculate our Residual error.

```
print('Residual Error (MSE): ', np.mean(test_y - y_hat)**2)
```

```
Residual Error (MSE):  0.5472648299129197
```

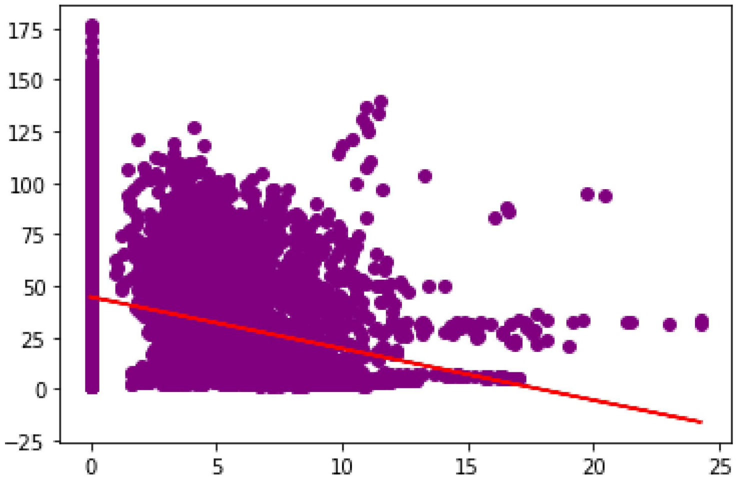
Now we will calculate our R2 value.

```
from sklearn.metrics import r2_score
print('R2 Score', r2_score(test_y, y_hat))
```

```
R2 Score 0.0750870664189145
```

Step-6: Now we will plot our graph alongwith slope:-

```
# Only health expenditure is non linear.
plt.scatter(sd2.health_exp,sd2.mort_rate,color ='purple')
plt.plot(test_x,y_hat,'r-')
plt.show()
```



+ Code + Text

Step-7: Multiple Linear Regression:-

Now we will test and train our data with multiple indpendent variables while keeping *mort_rate* as single dependent variable.

```
#Train Data.
train_x = np.asanyarray(train[['immunization', 'sanitation', 'fert_rate', 'pre_undernouris
train_y = np.asanyarray(train['mort_rate'])
train_y = np.asanyarray(train['mort_rate'])
lreg.fit(train_x,train_y)
```

```
    LinearRegression()
```

```
#Testing data.
test_x = np.asanyarray(test[['immunization', 'sanitation', 'fert_rate', 'pre_undernourishm
test_y = np.asanyarray(test['mort_rate'])
y_hat = lreg.predict(test_x)
```

```
print('Theta-0:', lreg.coef_)
print('Theta-1:', lreg.intercept_)
```

```
    Theta-0: [-0.51231967 -0.10001372  0.37055365  0.11214384]
    Theta-1: 56.27655221119018
```

```
print('Residual Error (MSE): ', np.mean(test_y - y_hat)**2)
```

```
    Residual Error (MSE):  0.09584751654773095
```

```
from sklearn.metrics import r2_score
print('R2 Score', r2_score(test_y, y_hat))
```

```
    R2 Score 0.7157428110945947
```

Findings:

In this research, it is concluded that the expenditure on health has a non-liner relationship with the mortality rate. To predict the infant mortality at best, multiple regression method is used which helped us to calculate R2 score which is better because it is noted to be more than 70%. For this purpose, I have understood the given data, cleaned the data, used machine learning library (i.e. Sklearn) to test and train the data and then it helped me in evaluating the ending results i.e.(R2 score).

✓ 0s completed at 9:11 AM ● ×