Project Report

INTRODUCTION:

1.1 Overview:

Life expectancy is a statistical measure of the average (see below) time an organism is expected to live, based on the year of its birth, its current age, and other demographic factors including gender.

To demonstrate how to build a regression model in Python, I used the 'Life Expectancy (WHO) dataset on Kaggle here. My goal was to create a model that could predict the average life expectancy of a person in a given country on a given year based on a number of variables.

The output algorithms have been used to test if they can maintain their accuracy in predicting the life expectancy for data they haven't been trained. Four algorithms have been used:

Linear Regression
Ridge Regression
Lasso Regression
ElasticNet Regression
Linear Regression with Polynomic features
Decision Tree Regression
Random Forest Regression

1.2 Purpose:

Life expectancy is a measure that is often used to gauge the overall health of a community. **Life expectancy** at birth measures health status across all **age** groups. Small increases in **life expectancy** translate into large increases in the population.

Life expectancy is a statistical prediction for how long a person will live. Based on actuarial science, **life expectancy** takes into account several individual-level as well as population-level factors to arrive at a figure.

LITERATURE SURVEY

2.1 Existing Problem:

A typical Regression Machine Learning project leverages historical data to predict insights into the future. This problem statement is aimed at predicting Life Expectancy rate of a country given various features.

Life expectancy is a statistical measure of the average time a human being is expected to live, Life expectancy depends on various factors: Regional variations, Economic Circumstances, Sex Differences, Mental Illnesses, Physical Illnesses, Education, Year of their birth and other demographic factors. This problem statement provides a way to predict average life expectancy of people living in a country when various factors such as year, GDP, education, alcohol intake of people in the country, expenditure on healthcare system and some specific disease related deaths that happened in the country are given.

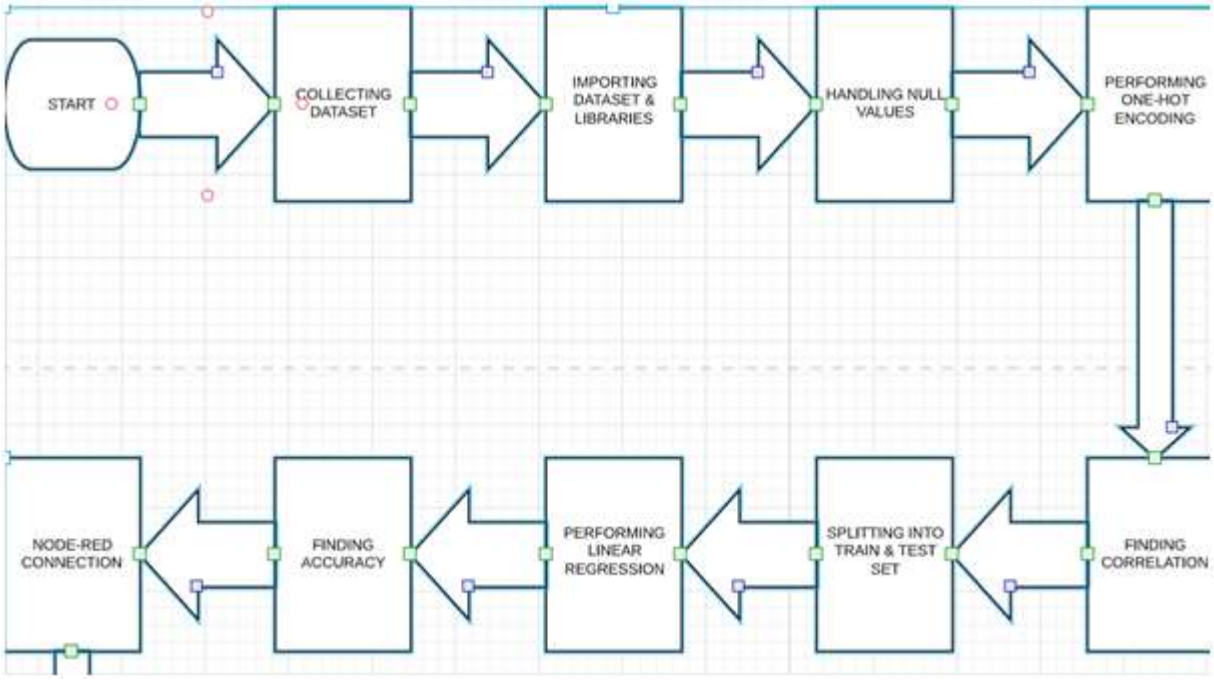
2.2 Proposed Solution:

The project tries to create a model based on data provided by the World Health Organization (WHO) to evaluate the life expectancy for different countries in years. The data offers a timeframe from 2000 to 2015. The data originates from here: https://www.kaggle.com/kumarajarshi/life-expectancy-who/data The output algorithms have been used to test if they can maintain their accuracy in predicting the life expectancy for data they haven't been trained. Four algorithms have been used:

Linear Regression
Ridge Regression
Lasso Regression
ElasticNet Regression
Linear Regression with Polynomial features
Decision Tree Regression
Random Forest Regression

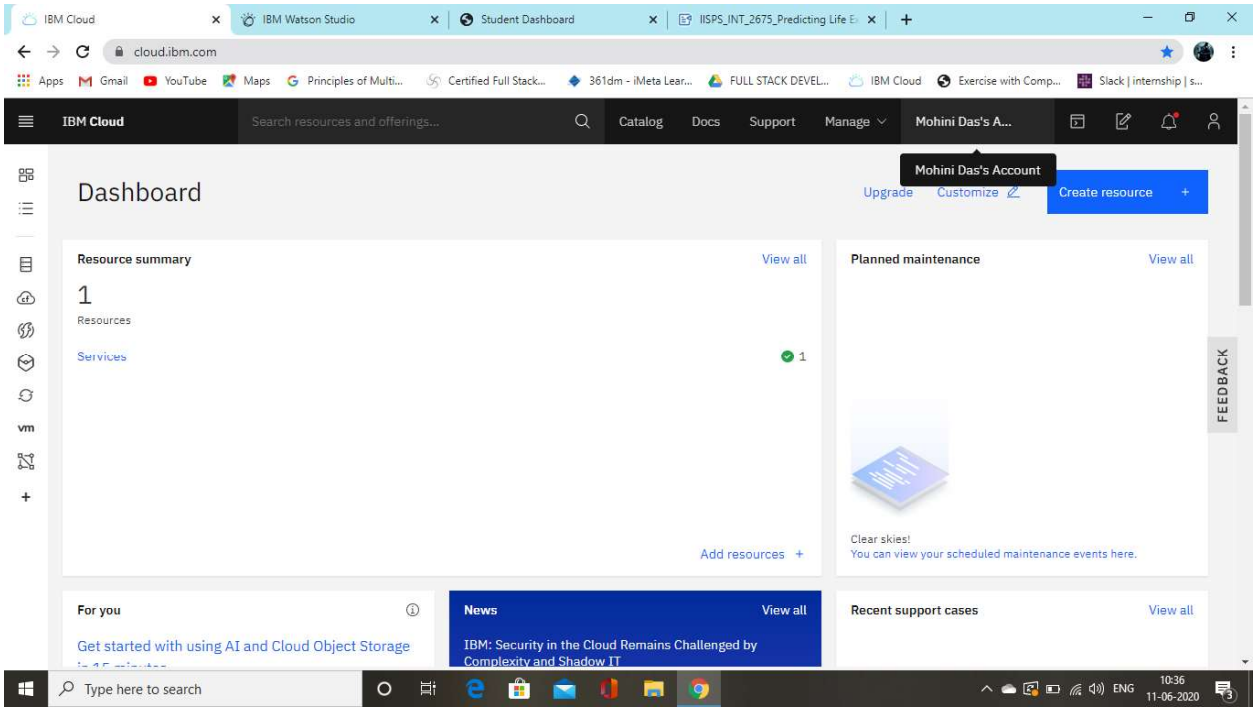
Theoritical Analysis:

3.1 Block Diagram:

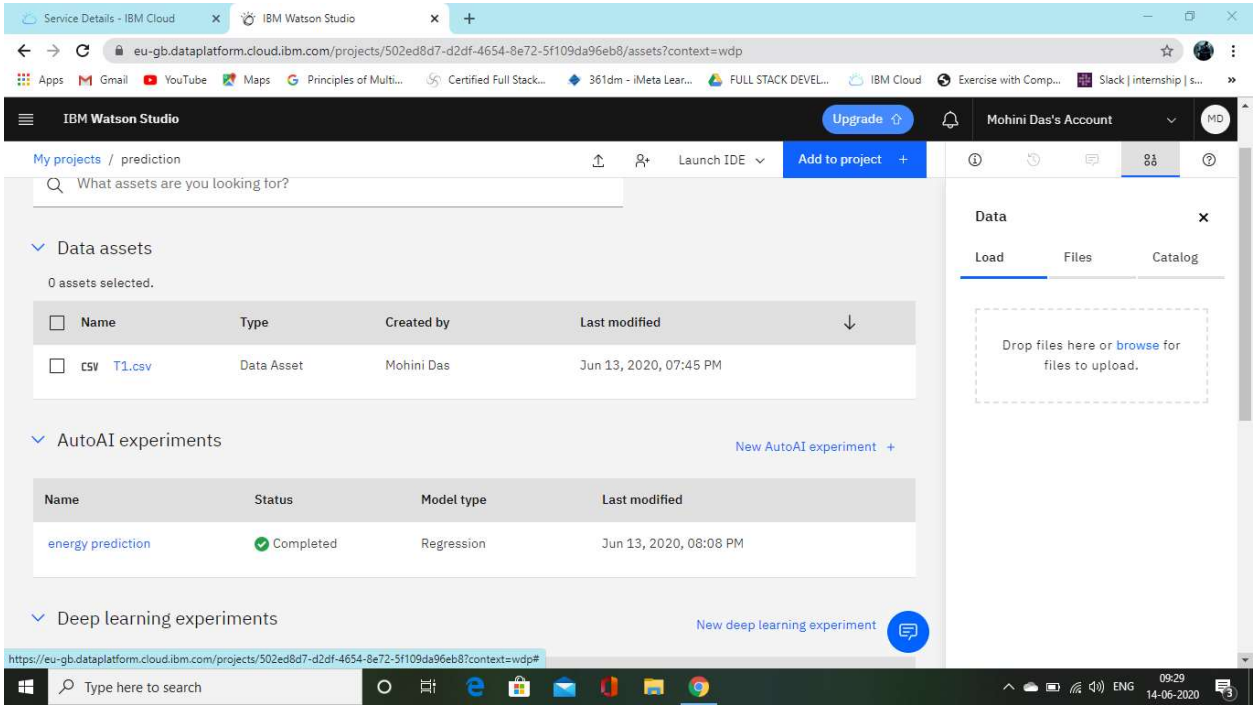


3.2 Hardware And Software Design:

1) Creating IBM cloud account:



2) Creating Watson Studio:



Experimental Investigation:

1) The first thing I did was to collect data set from kaggle.com.

2) Checked for the null values and removed the null-values from the data-set.

3) During the experimental investigation I have plotted the different independent features with the life expectancy.

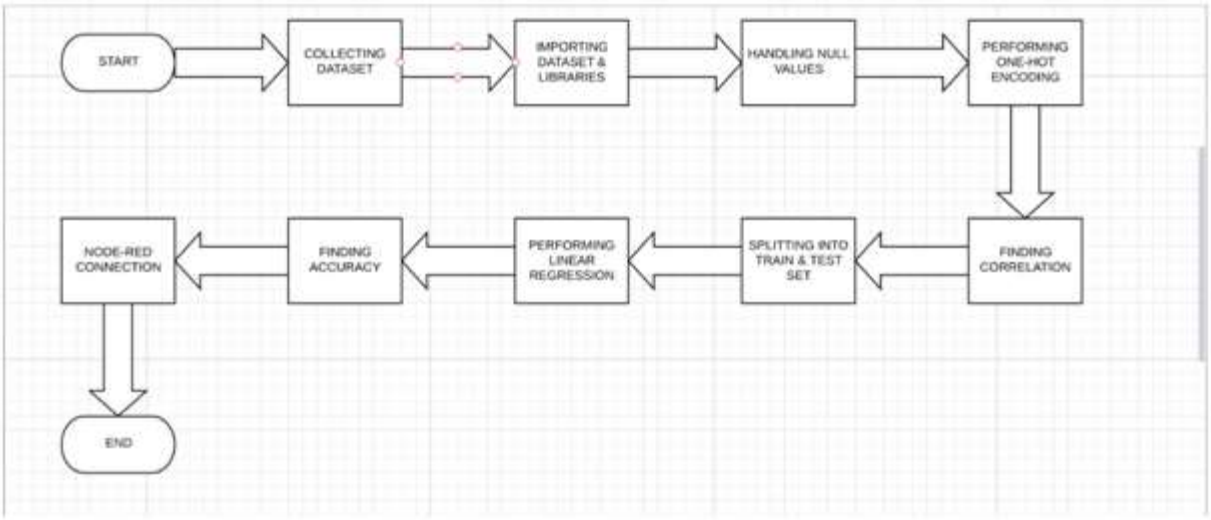
4) Then I found out the co-relation among the features.

5) Performed one-hot-encoding to handle string values.

6) Checked for the outliers and removed the outliers from the dataset

7) Then performed the model building.

Flowchart:



Result:

After performing the visualization and co-relation I have seen that , the Life expectancy of a country is very much affected by different deseases and also the percentage of alcoholic people.

The income resources of a country affect the life expectancy most. The countries which have hunger issues and the resources of income are low the average life expectancy of the people of that country is also very low.

The GDP, Schooling percentage also affects the life expectancy at a higher level.

Population does not affect the life expectancy of a country.

As Hepatitis B is now very much curable so it does not affect much the life expectancy.

Countries which have good gdp, schooling percentage, income resources have higher average life expectancy because the people of these types of countries are living happily with all their basic needs.

So after performing the regression model I have achieved 81% model score and the MSE value was 13.8.

So the result was quiet good , to check over-fitting I have used different data-sets but the results were quiet same.

Advantage And Disadvantage:

Advantages:

Life expectancy can be estimated at any **age**, e.g. **life expectancy** at 65 years. It gives more weight to deaths at younger ages. **Life expectancy** has been used nationally to monitor health inequalities.

Disadvantages:

At smaller geographies may be influenced by nursing homes in the area.

Application:

Life expectancy can be estimated at any **age**, e.g. **life expectancy** at 65 years. Gives more weight to deaths at younger ages. **Life expectancy** has been used nationally to monitor health inequalities, economic inequalities etc.

conclusion:

After comparing all the algorithms we can conclude the Linear Regression offer which are the same:

1. Best Parameters: {‘alpha’: 0, ‘max_iter’: 10}
2. R square on the test data of 81%
3. MSE of 13.8

4.RMSE of 3.7

Future Scope:

Use of different forecasting models based on characteristics of death rates, and their patterns over age, birth cohort, time, and space, for coherent and unbiased forecasts; and rigorous testing of model performance. The key limitation of our work, shared by all other attempts to forecast the future, is the inability to account for unexpected events and major changes in social and health systems determinants of health, which can fundamentally change trends and, in extreme cases, even lead to a reversal of life expectancy gain.

Bibliography:

The following sources have been used:

1. https://www.kaggle.com/kumarajarshi/life-expectancy-who/data

2. Introduction to Machine Learning with Python by Andreas C. Müller & Sarah Guido.

3. YouTube videos on Machine Learning.

4. stack overflow.

5. Sessions given by Smart bridge.

Appendix:

A. Source Code:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt # ploting , visualization
import seaborn as sns # ploting
from sklearn import model  selection #scikit learn
from sklearn import linear  model
from sklearn import metrics
from sklearn import preprocessing
from sklearn import utils
from sklearn import feature_selection

import types
import pandas as pd
from botocore.client import Config
import ibm_boto3
```

```
def __iter__(self): return 0

# @hidden_cell
# The following code accesses a file in your IBM Cloud Object Storage. It includes your credentials.
# You might want to remove those credentials before you share the notebook.
client_6ddb24ebb7624cb8a24b809a517416ec = ibm_boto3.client(service_name='s3',
    ibm_api_key_id='bxIhuH_glPTf4g4oDOESzYTy18YizfpBcbGugPKu5mn4',
    ibm_auth_endpoint="https://iam.cloud.ibm.com/oidc/token",
    config=Config(signature_version='oauth'),
    endpoint_url='https://s3.eu-geo.objectstorage.service.networklayer.com')

body = client_6ddb24ebb7624cb8a24b809a517416ec.get_object(Bucket='predictionoflifeexpectancyusingma-donotdelete-pr-zm8hil8io1svnl',Key='datasets_12603_17232_Life Expectancy Data.csv')['Body']
# add missing __iter__ method, so pandas accepts body as file-like object
if not hasattr(body, "__iter__"): body.__iter__ = types.MethodType( __iter__, body )

df= pd.read_csv(body)
df.head()
```

```
df.info()
```

```
df.isnull().sum()
```

```
df=df.dropna()

from sklearn.preprocessing import LabelEncoder

le = LabelEncoder()

df['Status']= le.fit_transform(df['Status'])
df

df.corr()

X=df.drop(["Year","Country","Life expectancy "],axis=1)

Y=df['Life expectancy ']

Xtrain, Xtest, Ytrain, Ytest = train_test_split(X, Y, test_size=0.25, random_state=42)

model=linear_model.LinearRegression()

model.fit(Xtrain,Ytrain)

predicted=model.predict(Xtest)

e=np.sqrt(np.mean((Ytest-predicted)**2))
```

```
e

model.score(Xtrain,Ytrain)

model.score(Xtest,Ytest)

e1=(np.mean((Ytest-predicted)**2))
e1
```