**Project Report**

**Project by:**

**Ayush Ashok Agarwal**

**(ayushagarwal199921@gmail.com)**

**Thakur college of Engineering and Technology**

**Kandivali (West), Mumbai-400101**

**Under:**

**Remote summer internship program 2020 by Smartinternz.**

**Predicting Life Expectancy using Machine Learning.**

# 1. Introduction

## 1.1 Overview

The overview of the project is building a model using Machine Learning Techniques for predicting Life Expectancy. Life Expectancy is 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 like [Status, Life expectancy ,Adult Mortality, infant deaths, Alcohol, percentage expenditure, Hepatitis B, Measles , BMI ,under-five deaths ,Polio, Total expenditure, Diphtheria , HIV/AIDS,GDP, Population, thinness 1-19 years, thinness 5-9 years, Income composition of resources, Schooling].

## 1.2 Purpose

The purpose of the project is to build a scalable, real-time learning model using IBM Watson to predict the life expectancy of a person considering the various factors. The project will be helpful in improving the health condition of the society and give insights about some crucial factors such as [Alcohol intake, GDP growth, schooling, adult mortality, total and cost expenditure and etc]. The project uses a Regression which is a classification algorithm. It is a measure of the relation between the mean value of one variable (e.g. output) and corresponding values of other variables. The dataset used or the training of the model was downloaded from kaggle.com and Python is used to write the code for machine learning model.

# 2. Literature Survey

## 2.1 Existing Problems

It has been noted that data collection for predicting the life/health using the machine learning/big data is a big challenge due to considerations relating to privacy and government policy, which will require the collaboration of various health sector bodies. Despite these challenges, Life expectancy can be predicted by proposing a data collection and application approach. As Artificial intelligence and Machine Learning technologies are developing and quickly being implemented, the ease of gathering health data from the public as well as current government agencies such as centralized health servers could be increased

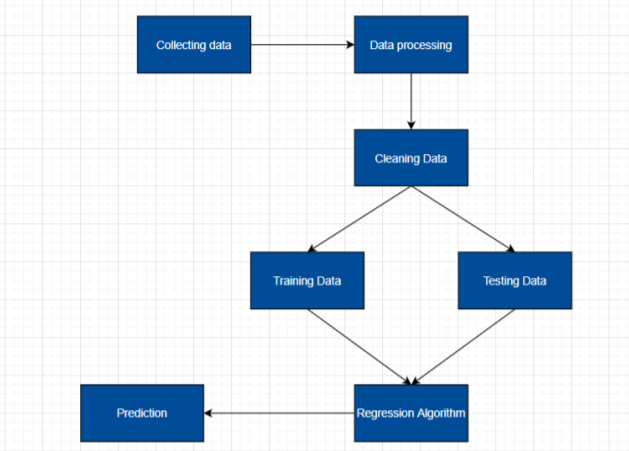
## 

## 2.2 Proposed Solution

I explored life expectancy and looked for dataset on the following aspects (features) : Birth Rate · Cancer Rate · Dengue Cases · Gross Domestic Product (GDP) · Expenditure · Heart Disease Rate · Population · Area · Population Density . life Expectancy, measured in number of years. Then I have performed data cleaning and data analysis using the statistical tools in python and selected the dependent and independent features and created a machine learning model using regression (a classification algorithm). using that model when we give the inputs (features) the model will give prediction (life expectancy in years) as output and finally that model is deployed to IBM cloud.

# 3. Theoretical Analysis

## 3.1 Block Diagram



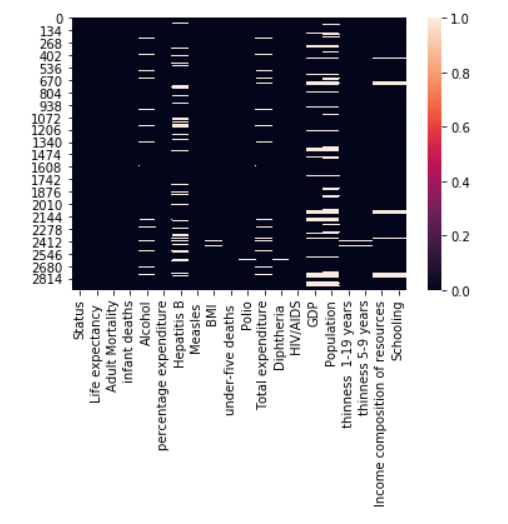
## 3.2 Software Engineering

**Software/Hardware  Requirements:**

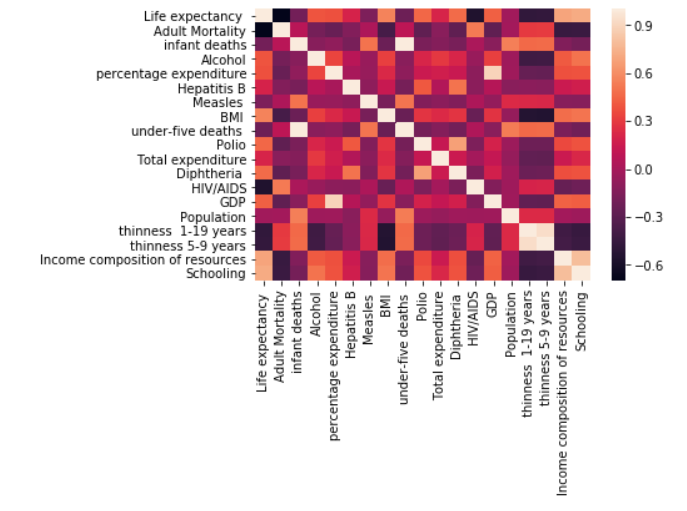
* IBM cloud services
* IBM Watson Studio
* IBM Node-Red application 7
* Jupyter Notebook
* Github
* Zoho document writer

# 4. Experimental Investigation

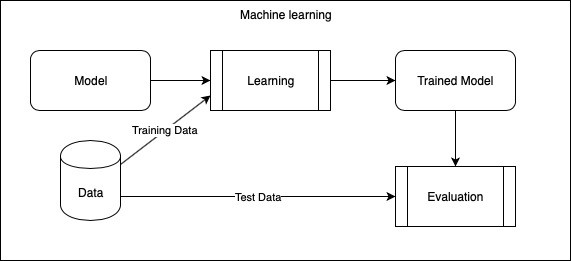
**Plot showing NA values in our dataset.**

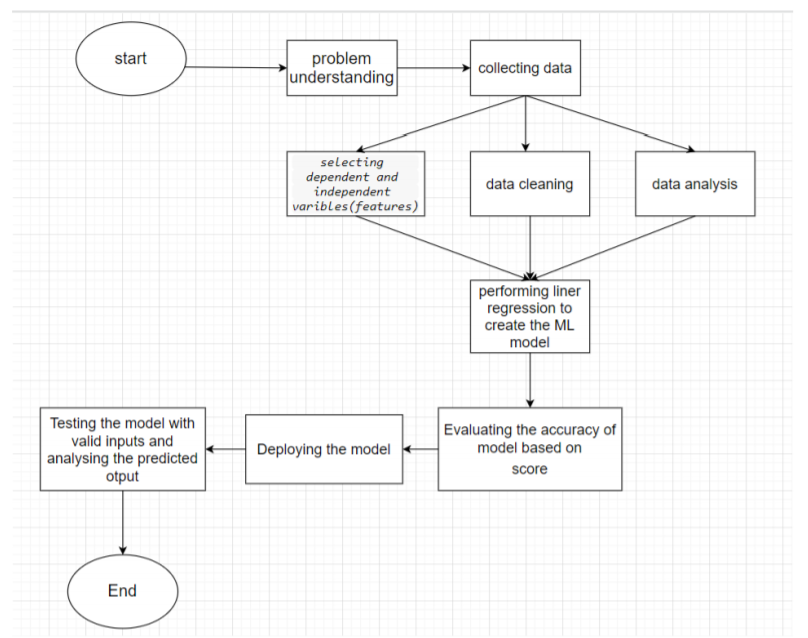


**Correlation between the features of the dataset.**



**5. Flowchart**





# 6. Result

* After the successful integration of our machine learning and UI, we get a user interface which will ask for the required fields and accordingly predict the life expectancy.
* The result will not be 100% accurate because the accuracy that I got is 98.3%.

# 7. Advantages & Disadvantages

**Advantages:**

* Since we can predict the life span, we can know what factors are influencing the expectancy on life span in what ways and an individual can take appropriate steps to enhance his/her life quality and increase his/her life span.

**Disadvantages:**

* Limited dataset features does not focuses on all the factors involved in predicting life expectancy.
* Error may occur due to inappropriate pre-processing and cleaning of data and selecting inappropriate model.

# 8. Application

* This project/idea is useful for Insurance companies as they consider age, lifestyle choices, family medical history, and several other factors when determining premium rates for individual life insurance policies.
* This can also be used by various health care sectors of a nation to look into some particular aspects to enhance the life quality of their citizens.

# 9. Conclusion

Hence, we were able to create a model with self-learning ability which can predict the life expectancy understanding data of the user from a user manual and predict the result in years.

# 10. Future Scope

Big data and machine learning can benefit public health researchers with analyzing thousands of variables to obtain data regarding life expectancy. We can use demographics of selected regional areas and multiple behavioral health disorders across regions to find correlation between individual behavior indicators and behavioral health outcomes..

# 11. Bibilography

## Appendix

Node red UI link:

<https://node-red-etrry.eu-gb.mybluemix.net/ui/#!/0?socketid=EIIxRvvzRySqGvUEAAAb>

Project Demonstration link:

<https://drive.google.com/file/d/1-4SesrvxqPTzQzNtMALFOJsLHBJ69GPp/view?usp=sharing>

Git-Hub repository link:

<https://github.com/SmartPracticeschool/llSPS-INT-3305-Predicting-Life-Expectancy-using-Machine-Learning>

## Source Code

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

life=life.drop(['Year','Country'],axis=1)

sns.heatmap(pd.isnull(life))

life.isnull().sum()

life["Alcohol"]=life["Alcohol"].fillna(value=life["Alcohol"].mean())

life["Income composition of resources"]=life["Income composition of resources"].fillna(value=life["Income composition of resources"].mean())

life[" thinness 1-19 years"]=life[" thinness 1-19 years"].fillna(value=life[" thinness 1-19 years"].mean())

life[" thinness 5-9 years"]=life[" thinness 5-9 years"].fillna(value=life[" thinness 5-9 years"].mean())

life["Schooling"]=life["Schooling"].fillna(value=life["Schooling"].mean())

life["Polio"]=life["Polio"].fillna(value=life["Polio"].mean())

life["Population"]=life["Population"].fillna(value=life["Population"].mean())

life["GDP"]=life["GDP"].fillna(value=life["GDP"].mean())

life["Diphtheria "]=life["Diphtheria "].fillna(value=life["Diphtheria "].mean())

life["Life expectancy "]=life["Life expectancy "].fillna(value=life["Life expectancy "].mean())

life[" BMI "]=life[" BMI "].fillna(value=life[" BMI "].mean())

life["Adult Mortality"]=life["Adult Mortality"].fillna(value=life["Adult Mortality"].mean())

life["Hepatitis B"]=life["Hepatitis B"].fillna(value=life["Hepatitis B"].mean())

life["Total expenditure"]=life["Total expenditure"].fillna(value=life["Total expenditure"].mean())

life.isnull().sum()

correlated\_data=life.corr()

sns.heatmap(correlated\_data)

y=life["Life expectancy "]

X=life.drop("Life expectancy ",axis=1)

X["Status"].unique()

X["Status"]=X["Status"].replace(['Developing', 'Developed'],[1,2])

X.shape

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=101)

X\_train.shape

X\_test.shape

y\_train.shape

y\_test.shape

from sklearn.ensemble import RandomForestRegressor

regr = RandomForestRegressor(n\_estimators=2, random\_state=0)

regr.fit(X, y)

y\_pred = regr.predict(X\_test)

np.set\_printoptions(precision=2)

y\_pred=np.array(y\_pred)

y\_test=np.array(y\_test)

print(np.concatenate((y\_pred.reshape(len(y\_test),1),y\_test.reshape(len(y\_test),1)),1))

accuracy=regr.score(X\_test,y\_test)

print("Accuracy: ",accuracy)

print('Coefficients: \n', regr.coef\_)

print('Mean squared error: %.2f'

% mean\_squared\_error(y\_test, y\_pred))

print('Coefficient of determination: %.2f'

% r2\_score(y\_test, y\_pred))