PROJECT ON

"PREDICTING LIFE EXPECTANCY USING MACHINE LEARNING"

MAANASVI KODLI

1. INTRODUCTION

Life expectancy is a statistical measure of the average time an organism is expected to live, based on the year of its birth, its current age and other demographic factors including gender. Life Expectancy plays an important role when decisions about the final phase of life need to be made. A prediction of Life Expectancy helps to analyze the average life span and thus constitute in making life decisions for the generations to come easier.

1.1. Overview

"Predicting Life Expectancy using Machine Learning" aims, to predict the lifespan on a human being, based on diverse datasets, in a demographic region. The life of a human depends on various factors such as Regional variations, Economic Circumstances, Sex Differences, Mental Illnesses, Physical illnesses, Education, Year of their birth and other demographic factors. The project aims to predict an average life expectancy based on these and several other factors. This project finds the expected solution using various machine learning algorithms such as:
Linear Regression
Logistic Regression
SVM
Clustering
Polynomic Regression
The aim of the project is to find the relationship of the various factors with the lifespan of an individual using the ML Algorithms mentioned above.

The dataset used for the prediction contains data from year 2000 to 2015. It contains more than 2500 entries and around 22 columns with various features such as Population, Alcohol Consumption, Infant Mortality Rate etc., which aids the prediction of the model.

1.2. Purpose

If life expectancy is longer in a certain country, it speaks about the conditions of the place. It tells information on the health factors as well as the quality of life. If the conditions in a country and in its economy are good, obviously the life expectancy would be more and more number of people would like to live in the same country. But it isn't enough to have a long life. It must be a healthy life too. A lot of people spend their later years in a miserable condition, in poor health, which is the primary concern. We must strive to ensure that everyone has a healthy life and a life of quality. With today's new technologies and a positive attitude towards research, it is more possible than ever that a long and healthy life will be possible for more people.

2. LITERATURE REVIEW

2.1. Existing Problem

Few works have been done to provide an individually customized life expectancy prediction. We have reviewed existing works and techniques in the prediction of human Life Expectancy, and reached a conclusion that it is feasible to predict Life Expectancy for individuals using evolving technologies and devices such as big data, AI, machine learning techniques, and PHDs, wearable devices and mobile health monitoring devices. We also identified that the collection of data will be a huge challenge due to the privacy and government policy considerations, which will require collaboration of various bodies in the health industry. The interworking of a heterogeneous health network is also a challenge for data collection. Despite these challenges, a possibility of a PLE prediction by proposing an approach of data collection and application by smart phone, with which users can enter their information to access the cloud server to obtain their own PLE, was shown.

To verify the accuracy of the prediction and validation of data quality, big data techniques and analysis algorithms need to be developed and tested in a real-life situation with several sample groups. As artificial intelligence technology is evolving and being applied rapidly, feasibility may be increasing to collect health data from the public as well as existing health agencies such as centralized health servers.

2.2. Proposed Solution

Although there have been lot of studies undertaken in the past on factors affecting life expectancy considering demographic variables, income composition and mortality rates. It was found that affect of immunization and human development index was not taken into account in the past. Also, some of the past research was done considering multiple linear regression based on data set of one year for all the countries. Hence, this gives motivation to resolve both the factors stated previously by formulating a regression model based on mixed effects model and multiple linear regression while considering data from a period of 2000 to 2015 for all the countries. Important immunization like Hepatitis B, Polio and Diphtheria will also be considered. In a nutshell, this study will focus on immunization factors, mortality factors, economic factors, social factors and other health related factors as well. Since the observations this dataset are based on different countries, it will be easier for a country to determine the predicting factor which is contributing to lower value of life expectancy.

The model of ”Predicting Life Expectancy using Machine Learning” uses IBM Cloud services, which helps to avoid any storage issues. The UI Presented to the users is a website URL and hence they need not download any application to predict the results, which saves the storage space as that is the need of the hour.

3. PROJECT REQUIREMENTS

This project is fundamentally designed to predicting the life expectancy of a human in any country. The primary requirement of the project is the suitable dataset which will aid the prediction. Thus the data set has been taken from the WHO, who has provided the data itself, publicly. The machine learning model is trained on the basis of the data provided, such that it can predict the average lifespan of an individual in the coming years in any demographic location on Earth.

3.1. Functional Requirements

1. The dataset should be preprocessed before applying prediction otherwise will lead to errors.
2. The data model must be created on the basis of preprocessed data, no extra or no less info.
3. The data model must then be converted into a module for further use, after the data is updated.
4. The data should be implemented using IBM Watson which should then be connected to Node RED for the User Interface.

3.2. Technical Requirements

1. The dataset must be in .CSV format.
2. Machine Learning Algorithms must be applied with the help of Python programming language.
3. IBM Cloud account.
4. IBM Watson and Node-Red flow and other IBM services.

3.3. Software Requirements

1. Python IDE or Jupyter Notebook or Watson Studio Notebook
2. MS Excel
3. IBM Cloud Services
4. IBM Watson Services
5. Node-Red Starter Tool

4. FLOWCHART

A flowchart is a diagram which depicts a process, system or computer algorithm. They are widely used in multiple fields to document, study, improve and communicate often complex processes in clear, easy-to-understand diagrams. Flowcharts, sometimes spelled as flow charts, use rectangles, ovals, diamonds and potentially numerous other shapes to define the type of step, along with connecting arrows to define flow and sequence. A flow chart helps improve understanding of what exactly is being implemented and how it takes different routes for different inputs and targets.

The Node-RED Starter Application works on NodeJS language and it is designed to make things easier for the developer using a flow for all the working nodes. The nodes basically define a function of what is to be done once data reaches that part of the flow. Following is the diagram of Node-Red Flow used for Life Expectancy Prediction.

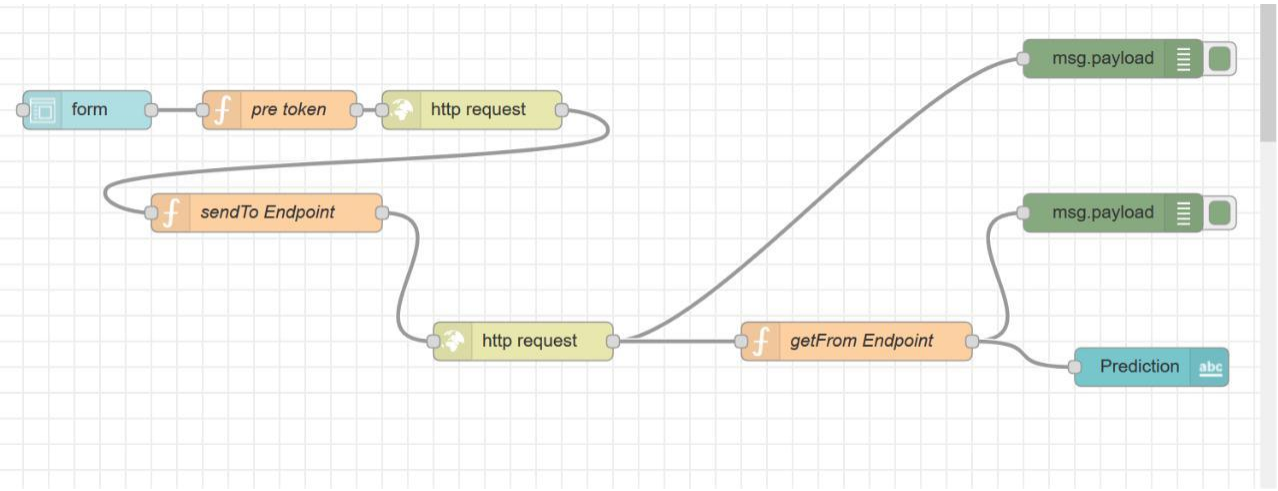

Figure 1: Node-red flow

5. RESULT

The model appears to the user in the form of an interface as shown in the Figure 2. The user has to fill in the inputs and click on “Predict” button at the end of the form. On clicking the “Predict” button, the user will be displayed the predicted life expectancy, based on the inputs provided, at the top of the page as shown in Figure 2. Once all the data is input by a user, the said data is analyzed by the Machine Learning model prepared using the service end point that which is given as a node in the Node-RED Flow. Data is run through the ML model and finally the predicted Life Expectancy is shown to the user, as shown below.

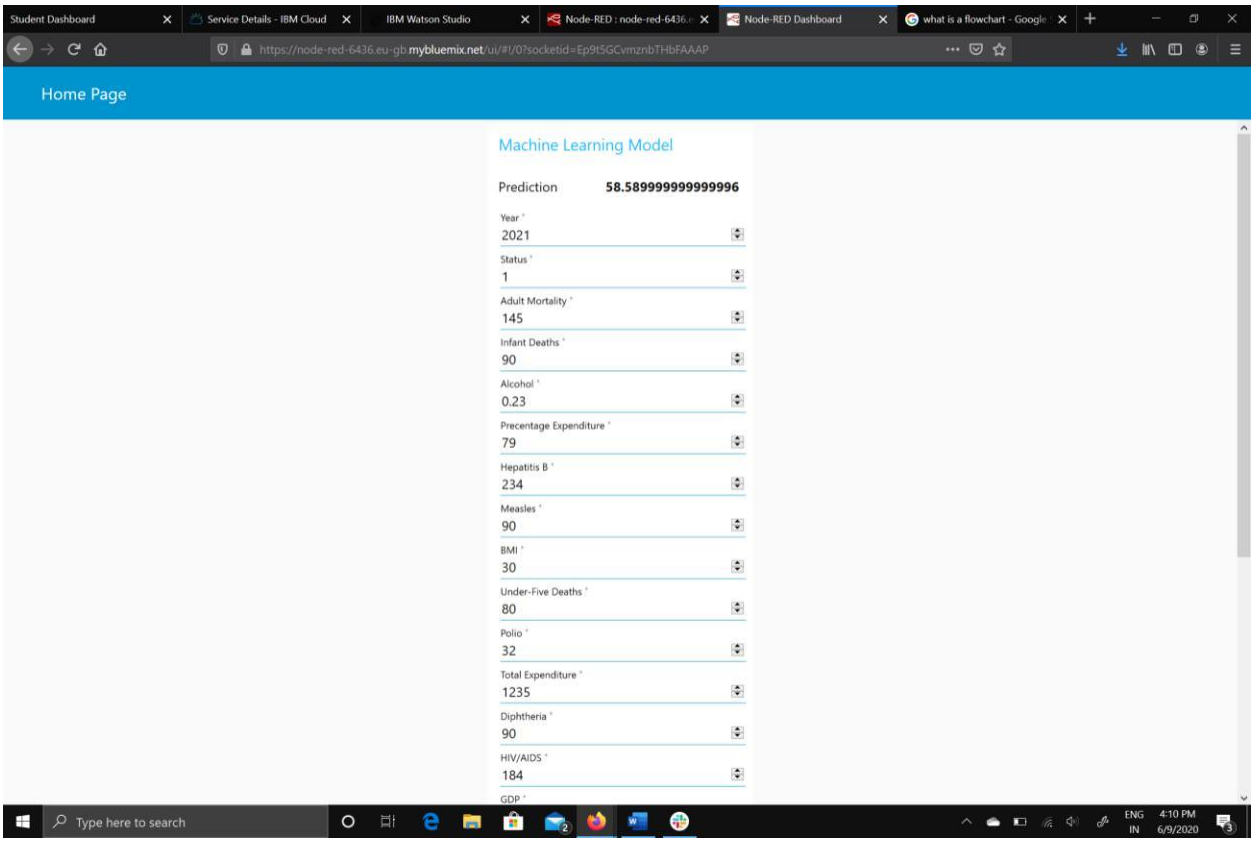

Figure 2: Result

6. ADVANTAGES AND DISADVANTAGES:

6.1. Advantages:

1. Advantages of using IBM Watson:
    - Processes unstructured data
    - Fills human limitations
    - Acts as a decision support system, doesn’t replace humans
    - Improves performance + abilities by giving best available data
    - Improve and transform customer service
    - Handle enormous quantities of data
    - Sustainable Competitive Advantage
2. Easy for user to interact with the model via the UI.
3. User-friendly.
4. Easy to build and deploy.
5. Doesn’t require much storage space.

6.2. Disadvantages:

1. Disadvantages of using IBM Watson:
    - Only available in English language (Limits areas of use)
    - Maintenance
    - Provides paid services
    - Doesn't process structured data directly
    - Increasing rate of data, with limited resources
2. Requires high speed internet connection.

7. APPLICATIONS

Life expectancy is the primary factor in determining an individual's risk factor and the likelihood they will make a claim. Insurance companies consider age, lifestyle choices, family medical history, and several other factors when determining premium rates for individual life insurance policies. The principle of life expectancy suggests that you should purchase a life insurance policy for yourself and your spouse sooner rather than later. Not only will you save money through lower premium costs, but you will also have longer for your policy to accumulate value and become a potentially significant financial resource as you age.

It can be used by researchers to make meaningful researches out of it and thus, bring about something that will help increase the expectancy consider the impact of a specific factor on the average lifespan of people in a specific country.

8. CONCLUSION

Thus, the developed model will predict the life expectancy of a specific demographic region based on the inputs provided. Various factors have a significant impact on the life span such as Adult Mortality, Population, Under 5 Deaths, Thinness 1-5 Years, Alcohol consumption, HIV, Hepatitis B, GDP, Percentage Expenditure and many more.

User can interact with the system via a simple user interface which is like any Google form needing inputs for which the results would be provided in seconds.

9. FUTURE SCOPE

As future scope, we can connect the model to the database to have the record of predictions. This will help us analyze the trends in the life span of various countries. This data can be utilized by organizations that would want to conduct experiments in regards of learning more information about any disease that would require factors same as the Life Expectancy model did and it’s predictions.

A model such as this one would definitely render countries to work on the factors affecting the Life Expectancy such as GDP, Daily Expense, and Infant Mortality Rate etc for better results in the life expectancy of a human in any country.

APPENDIX

A. Source Code

```
#Importing the dataset and relevant libraries

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_ID = ibm_boto3.client(service_name='s3',
    ibm_api_key_id='',
    ibm_auth_endpoint="https://iam.cloud.ibm.com/oidc/token",
    config=Config(signature_version='oauth'),
    endpoint_url='https://s3-api.us-geo.objectstorage.service.networklayer.com')

body = client_ID.get_object(Bucket='lifeexpectancyusingml-donotdelete-pr-oapzus7axdorfi',Key='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_data_1 = pd.read_csv(body)
df_data_1.head()


#Make all NaN or Null Values to Zero using the following statement
df_data_1=df_data_1.fillna(0)

df_data_1['Status']=df_data_1['Status'].map({'Developing':0,'Developed':1})

#Importing Pandas and Numpy
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline

#To get info about the data set
df_data_1.info()
```

```
df_data_1.describe()

df_data_1.columns

sns.pairplot(df_data_1)

sns.distplot(df_data_1['Life expectancy '])

sns.heatmap(df_data_1.corr())


X = df_data_1[['Year', 'Status', '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']]
y = df_data_1['Life expectancy ']

#Import Linear Regression from sklearn.model_selection

from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=101)

from sklearn.linear_model import LinearRegression

lm = LinearRegression()

lm.fit(X_train,y_train)

print(lm.intercept_)

coeff_df = pd.DataFrame(lm.coef_,X.columns,columns=['Coefficient'])
coeff_df
predictions = lm.predict(X_test)

plt.scatter(y_test,predictions)

from sklearn import metrics


print('MAE:', metrics.mean_absolute_error(y_test, predictions))
print('MSE:', metrics.mean_squared_error(y_test, predictions))
print('RMSE:', np.sqrt(metrics.mean_squared_error(y_test, predictions)))
```

```
!pip install watson-machine-learning-client

from watson_machine_learning_client import WatsonMachineLearningAPIClient

#Add Machine Learning instance credentials

wml_credentials={
  "apikey": "",
  "iam_apikey_description": "",
  "iam_apikey_name": "",
  "iam_role_crn": "",
  "iam_serviceid_crn": "",
  "instance_id": "",
  "url": ""
}

client = WatsonMachineLearningAPIClient( wml_credentials )

model_props = {client.repository.ModelMetaNames.AUTHOR_NAME: "Addname",
            client.repository.ModelMetaNames.AUTHOR_EMAIL: "AddEMailID",
            client.repository.ModelMetaNames.NAME: "Life Expectancy Data"}

model_artifact =client.repository.store_model(lm, meta_props=model_props)

published_model_uid = client.repository.get_model_uid(model_artifact)

published_model_uid

#Deploy model

deployment = client.deployments.create(published_model_uid, name="Life Expectancy Data")

#Scoring Endpoint

scoring_endpoint = client.deployments.get_scoring_url(deployment)
print(scoring_endpoint)
```