Predicting Life Expectancy of Humans using Machine Learning

Project Report

By Prasuna Pulivendula

1. Introduction

1.1. Overview

Life Expectancy is based on an estimate of the average age that members of a particular population group will be when they die.

Life Expectancy is widely viewed as a key measure of the health of populations. It is affected by many factors such as: socioeconomic status, quality of health system and the ability of people to access it, health behaviours such as HIV, alcohol consumption, Malaria etc., social factors, environmental factors.

In order to Predict the Life Expectancy, I developed the model using Machine Learning Algorithms (MultiVariate Linear Regression) and trained the model with the data sets present. For better usability by the user, I have created a Web Application using Node-RED where the trained model was deployed.

1.2. Purpose

Purpose of this Project is to Predict Life Expectancy of humans in a given country. Because Life Expectancy is not quite simple. It affects Economic Growth, Population Growth, Personal Growth, Growth in Health Sector, Insurance Companies. So, Predicting the Average Life Expectancy of the country and taking actions based on it beforehand helps in ensuring development of that Country.

2. Literature Survey

2.1. Existing Problem

Predicting Life Expectancy has been a long-term question to humankind. Many calculations and Research have been done to create an equation despite it being impractical to simplify these variables into one equation.

There are various smartphone apps and wearable devices that provide wellness and fitness tracking. And there exists some applications that provide health related data which are collected and processed by the devices and servers in the cloud. However there's no existing solution that provides the Personalized Life Expectancy.

2.2. Proposed Solution

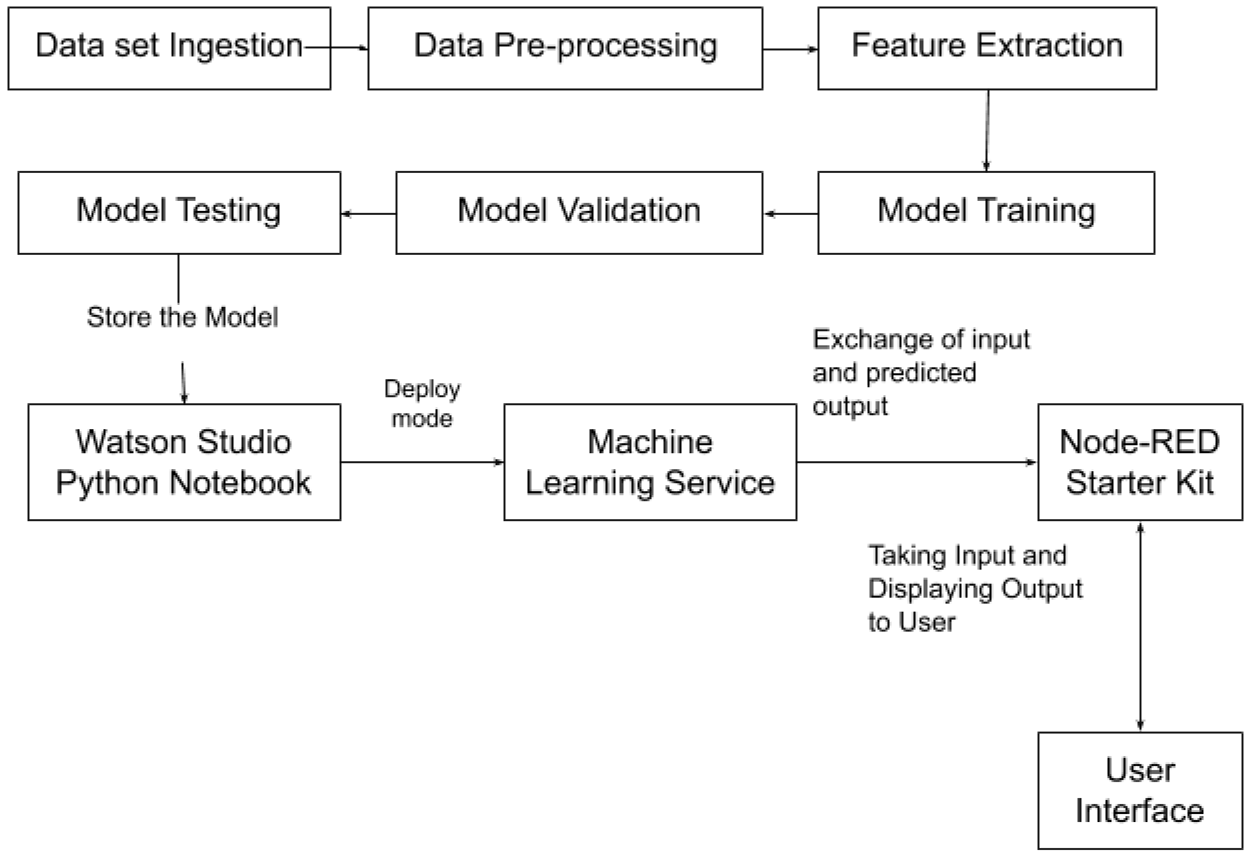
The value of Machine Learning in Healthcare is its ability to process huge data sets beyond the scope of human capability, and then reliably convert analysis of that data into clinical insights.

The proposed solution involves the use of Machine Learning Algorithms (Regression Algorithms, in particular MultiVariate Linear Regression). Here we propose a method for forecasting Life Expectancy of an individual from a country considering certain factors such as Status of the country, Adult Mortality Rate, Infant deaths, Alcohol, Hepatitis B, Measles, BMI, Polio, Total Expenditure, Diphtheria, HIV/AIDS, GDP of a country, Population, Income Composition of Resources, Schooling status of the country.

The Machine Learning Model trained will be made accessible to the users by integrating it with Node-RED flows to create a user-friendly interface.

3. Theoretical Analysis

3.1. Block Diagram



3.2. Hardware/Software Designing

3.2.1. Hardware Requirements

- Processor: i3 7th generation or higher
- Speed: 2GHz or more
- Hard disk space: 10GB or more

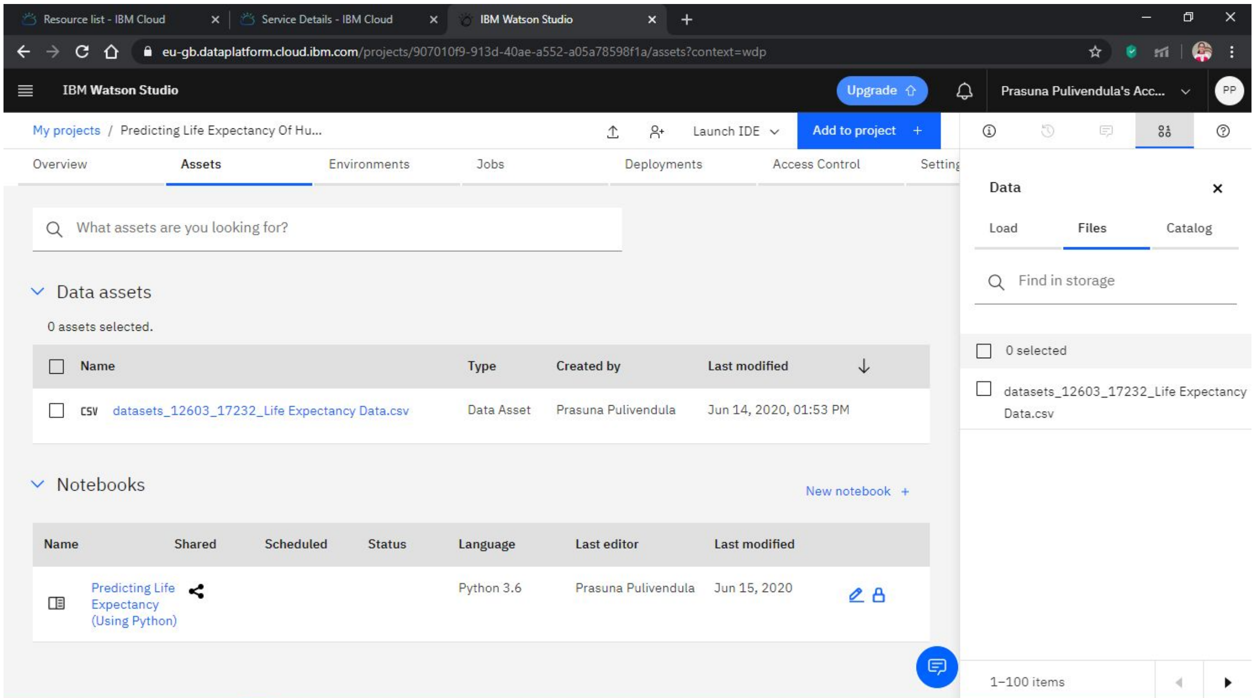
3.2.2. Software Requirements

- Zoho Writer
- Jupyter notebook
- IBM Cloud
- IBM Watson Studio
- Node Red

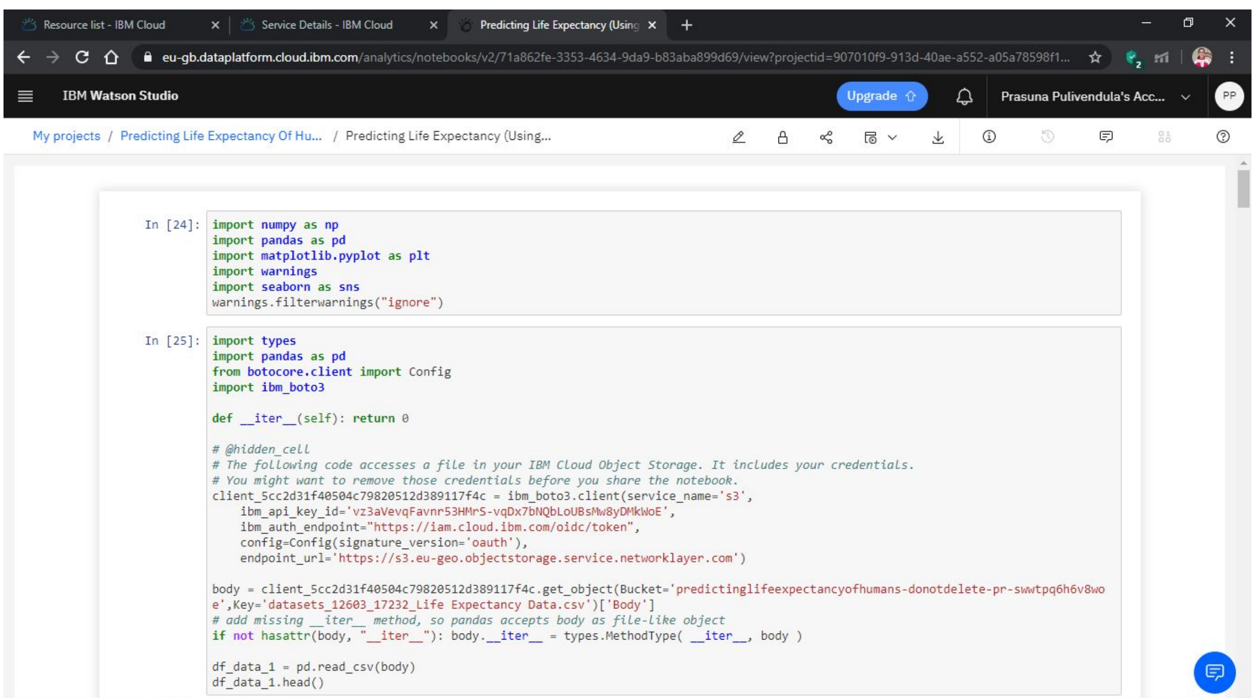
3.2.3. Designing the Model (Watson Studio)

- Collect the data set
- Create necessary IBM Cloud Services
- Create a new Watson Studio Project and Add Jupyter Notebook to the project

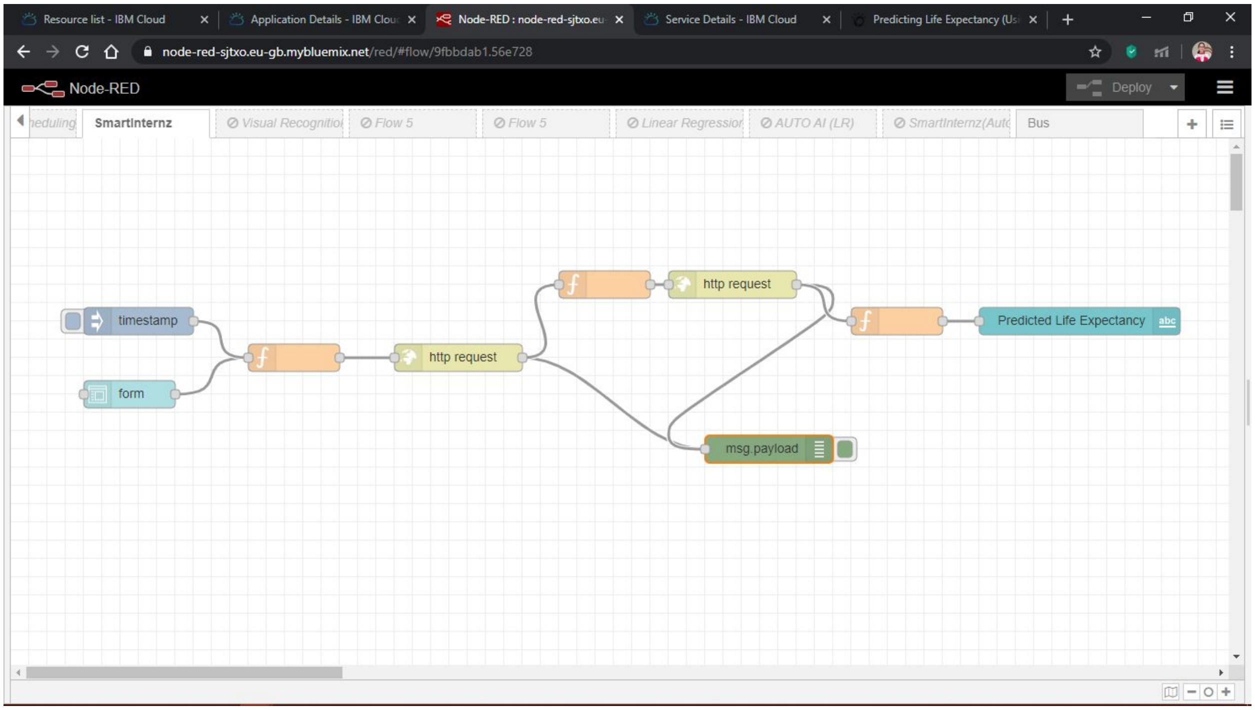
Steps: New Project => Create an Empty Project => Fill the details => Click on Create => Add to Project => Notebook



Link to the Jupyter Notebook:
https://eu-gb.dataplatform.cloud.ibm.com/analytics/notebooks/v2/71a862fe-3353-4634-9da9-b83aba899d69/view?access_token=5792820da2e5a637afb1533157d8e7ad4f50f9e9bc726101c66998b1bc98a2d2



User Interface Integration with ML Model (Node-RED):



4. Experimental Investigations

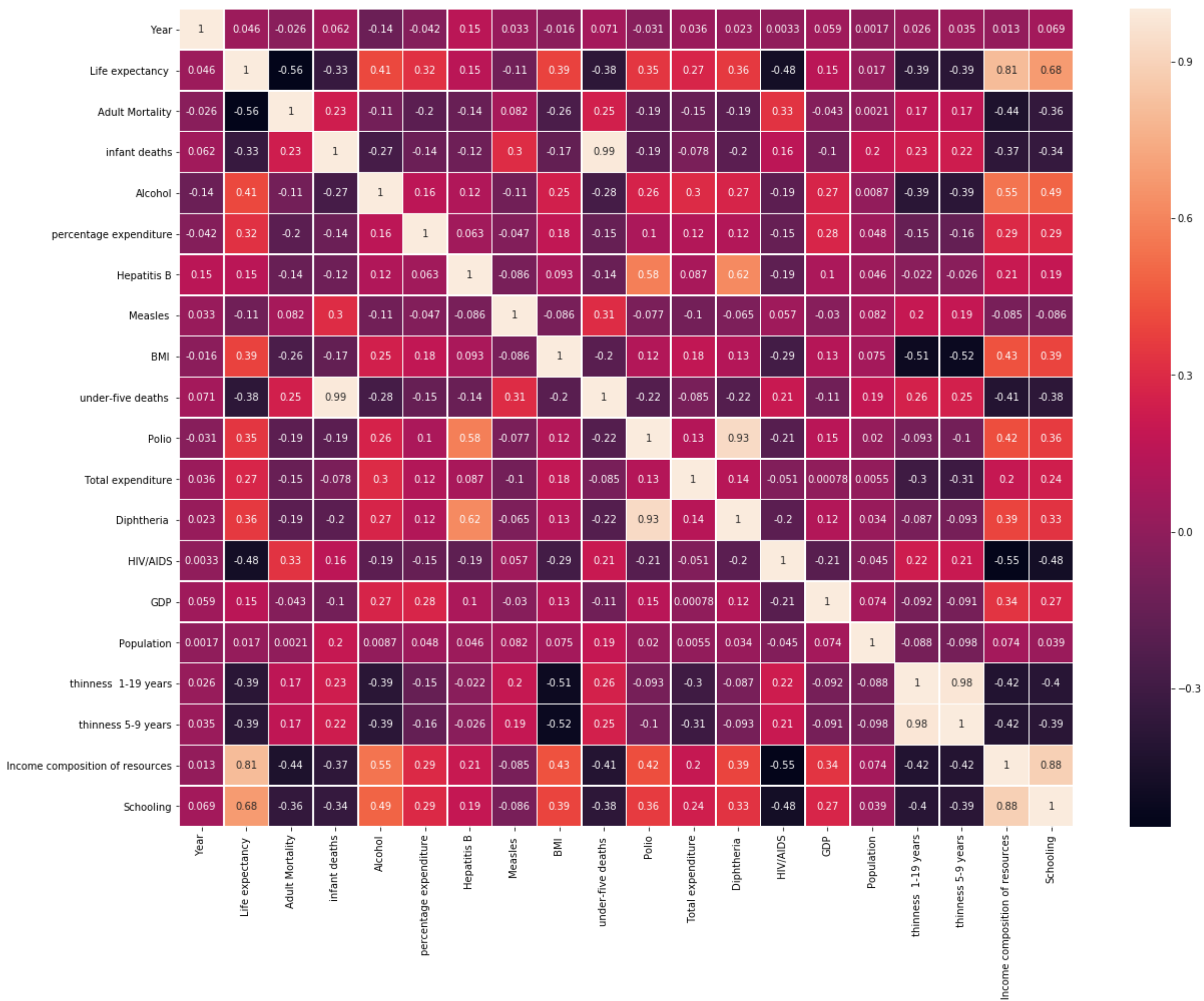
Analyzing the features:

```
In [26]: df_data_1.columns

Out[26]: Index(['Country', 'Year', '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'],
       dtype='object')

In [27]: df_data_1.shape

Out[27]: (2938, 22)

In [28]: df_data_1.describe()

Out[28]:
```

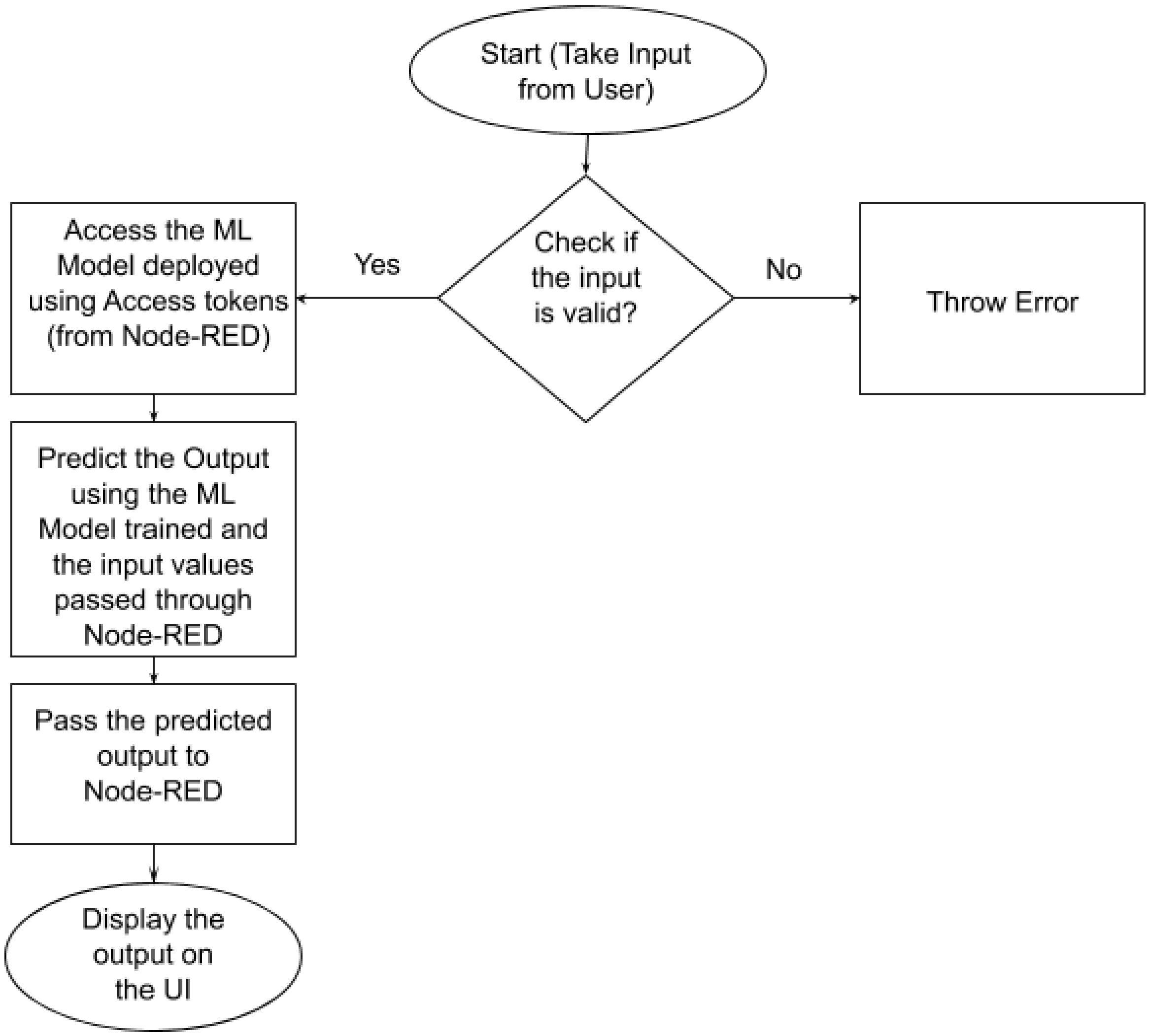
	Year	Life expectancy	Adult Mortality	infant deaths	Alcohol	percentage expenditure	Hepatitis B	Measles	BMI	under-five deaths	
count	2938.000000	2928.000000	2928.000000	2938.000000	2744.000000	2938.000000	2385.000000	2938.000000	2904.000000	2938.000000	2
mean	2007.518720	69.224932	164.796448	30.303948	4.602861	738.251295	80.940461	2419.592240	38.321247	42.035739	8
std	4.613841	9.523867	124.292079	117.926501	4.052413	1987.914858	25.070016	11467.272489	20.044034	160.445548	2
min	2000.000000	36.300000	1.000000	0.000000	0.010000	0.000000	1.000000	0.000000	1.000000	0.000000	3
25%	2004.000000	63.100000	74.000000	0.000000	0.877500	4.685343	77.000000	0.000000	19.300000	0.000000	7
50%	2008.000000	72.100000	144.000000	3.000000	3.755000	64.912906	92.000000	17.000000	43.500000	4.000000	9
75%	2012.000000	75.700000	228.000000	22.000000	7.702500	441.534144	97.000000	360.250000	56.200000	28.000000	9
max	2015.000000	89.000000	723.000000	1800.000000	17.870000	19479.911610	99.000000	212183.000000	87.300000	2500.000000	9

```
df_data_1.dtypes

Country                            object
Year                                int64
Status                             object
Life expectancy                   float64
Adult Mortality                   float64
infant deaths                       int64
Alcohol                           float64
percentage expenditure            float64
Hepatitis B                       float64
Measles                             int64
 BMI                              float64
under-five deaths                   int64
Polio                             float64
Total expenditure                 float64
Diphtheria                        float64
 HIV/AIDS                         float64
GDP                               float64
Population                        float64
 thinness  1-19 years             float64
 thinness 5-9 years               float64
Income composition of resources   float64
Schooling                         float64
dtype: object
```

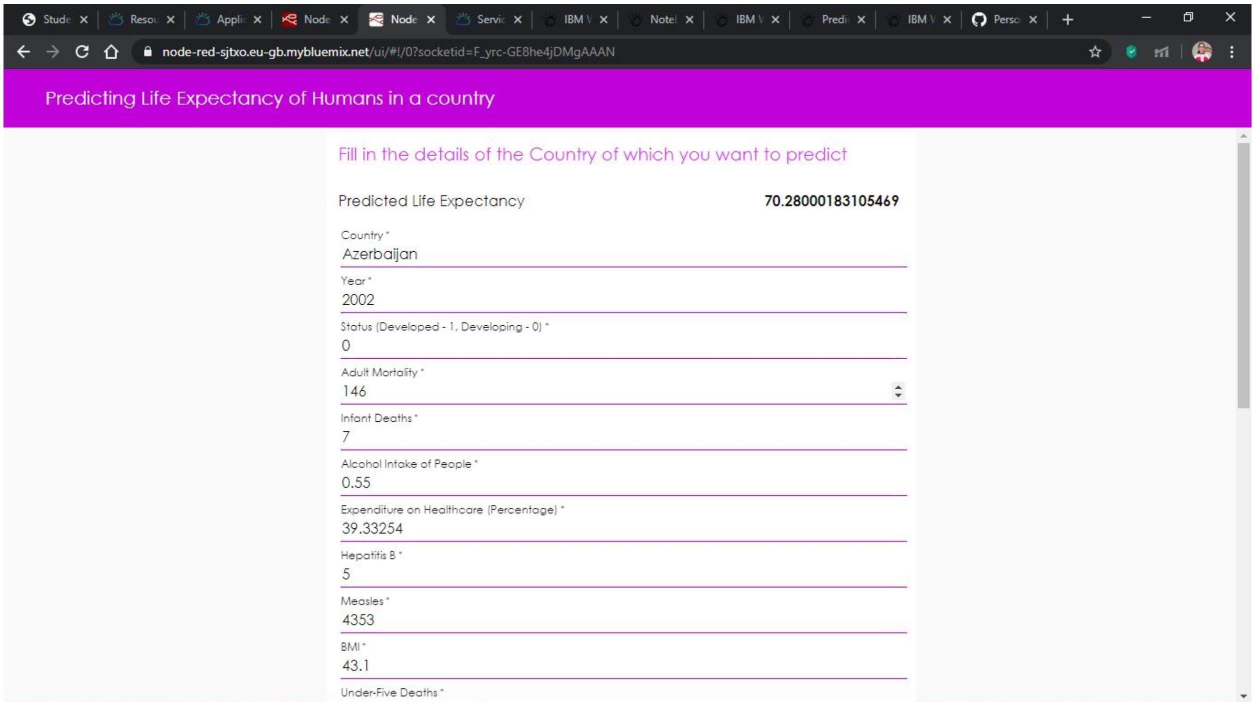
Data Visualization:

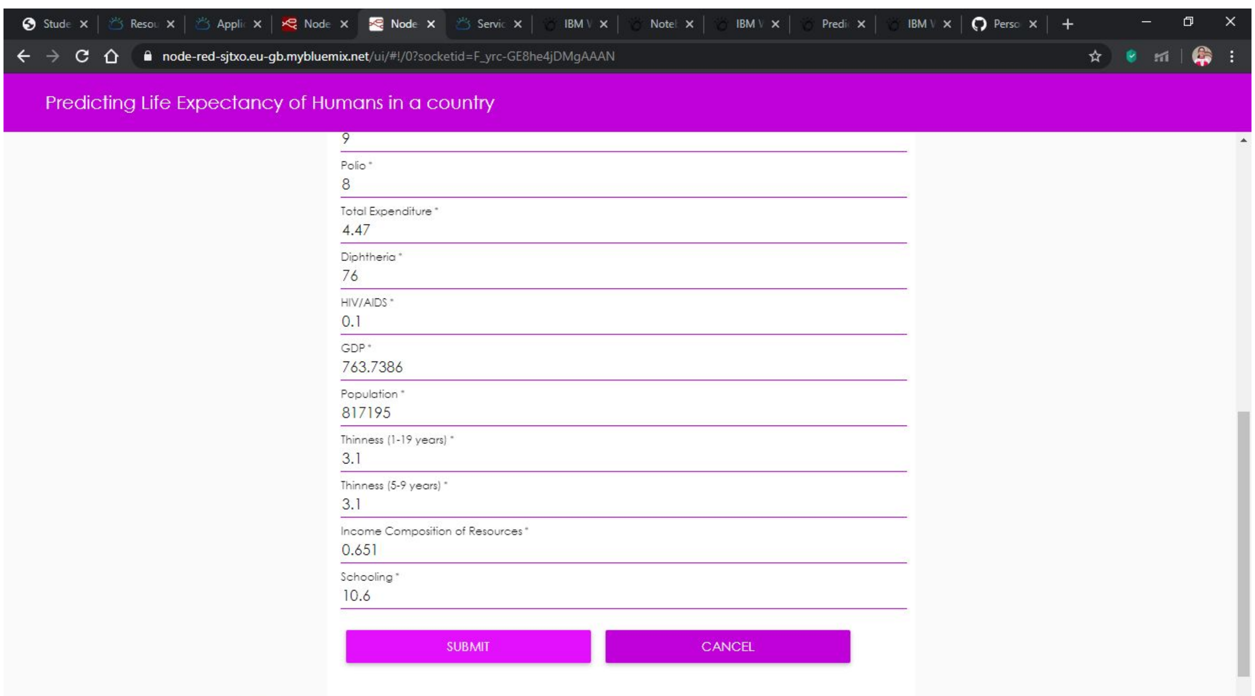


5. Flow Chart


Start (Take Input from User)
Check if the input is valid?
Yes
No
Access the ML Model deployed using Access tokens (from Node-RED)
Throw Error
Predict the Output using the ML Model trained and the input values passed through Node-RED
Pass the predicted output to Node-RED
Display the output on the UI

6. Result


node-red-sjtxo.eu-gb.mybluemix.net/ui/#!/0?socketid=F_yrc-GE8he4jDMgAAAN
Predicting Life Expectancy of Humans in a country
Fill in the details of the Country of which you want to predict
Predicted Life Expectancy
70.28000183105469
Country *
Azerbaijan
Year *
2002
Status (Developed - 1, Developing - 0) *
0
Adult Mortality *
146
Infant Deaths *
7
Alcohol Intake of People *
0.55
Expenditure on Healthcare (Percentage) *
39.33254
Hepatitis B *
5
Measles *
4353
BMI *
43.1
Under-Five Deaths *


node-red-sjtxo.eu-gb.mybluemix.net/ui/#!/0?socketid=F_yrc-GE8he4jDMgAAAN
Predicting Life Expectancy of Humans in a country
9
Polio *
8
Total Expenditure *
4.47
Diphtheria *
76
HIV/AIDS *
0.1
GDP *
763.7386
Population *
817195
Thinness (1-19 years) *
3.1
Thinness (5-9 years) *
3.1
Income Composition of Resources *
0.651
Schooling *
10.6
SUBMIT
CANCEL

7. Advantages and Disadvantages

Advantages:

As Machine Learning Algorithms improve over time (as they have that ability), and since we deployed this ML Model, Efficiency and Performance of our Application improves with time.

This application learns the patterns hidden in the data sets itself without human intervention which makes the work simpler and efficient. And more the data fed more is the accuracy of the Output Predicted.

Node-RED simplifies the effort by providing the default template pages, so that the developer can just concentrate on developing a good model instead of spending hours in designing pages. And it also makes the integration between the Machine Learning model and UI much easier.

Disadvantages:

Computational cost of Machine Learning Algorithms can be very high. Being good at prediction, for an ML Model can only happen when it is trained with sufficient data (some millions of data). But training it with millions of data, can even take some days, or even months, which is very time consuming. And this also in turn can increase the cost of resources required to implement such applications on a large scale.

Node-RED doesn’t give the users much flexibility to create their own templates.

8. Applications

It can be used (with personalized experience) in many sectors such as Health Sector, Insurance Companies, Government, Public Sector and also can be used for Economic Growth, Population Growth and Personal Growth.

9. Conclusion

In this Project, we developed a Machine Learning Model to predict Life Expectancy of humans in a country. To develop this we used IBM Cloud Services, Machine Learning Algorithms and Node-RED Flow Editor.

Predicting Life Expectancy can lead to the development of that country. Because this breakthrough can widely impact Health Sectors, Public Sectors and Economic Sectors by improving the resources, funds and services provided to people.

10. Future Scope

We can train the ML Model with still more data sets to improve the accuracy.

We can integrate it with some more features like guiding us about the Health Care and its importance, and things one needs to take care of regarding their health.

11. Bibliography

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

https://www.ibm.com/cloud/get-started

https://developer.ibm.com/tutorials/how-to-create-a-node-red-starter-application/

https://www.youtube.com/watch?v=IAcY7UGUL4I&t=2794s

https://www.youtube.com/watch?v=ge3nYkMu1hY

https://www.youtube.com/watch?v=O5wqjk_GeJo&t=2546s

https://www.youtube.com/watch?v=iEadmCNb_hE&t=670s

12. Appendix

12.1 Source Code

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import warnings
import seaborn as sns
warnings.filterwarnings("ignore")

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_5cc2d31f40504c79820512d389117f4c = ibm_boto3.client(service_name='s3',

ibm_api_key_id='vz3aVevqFavnr53HMrS-vqDx7bNQbLoUBsMw8yDMkWoE',
    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_5cc2d31f40504c79820512d389117f4c.get_object(Bucket='predicting
lifeexpectancyofhumans-donotdelete-pr-swwtpq6h6v8woe',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_data_1 = pd.read_csv(body)
df_data_1.head()

df_data_1.columns
df_data_1.shape
df_data_1.describe()
df_data_1.dtypes
df_data_1.isnull().sum()

#null values present in the data causes inconsistency, so we need to remove
them


#removing null values
df_data_1=df_data_1.interpolate(method="linear",limit_direction="forward
")

df_data_1.isnull().sum()
#due to some extreme causes (maybe typing mistake even) some of the data
points will be very far from the remaining data points when plotted, these data
```

#points also might cause inconsistency, so we need to remove these outliers.

#using IQR method to remove outliers present.

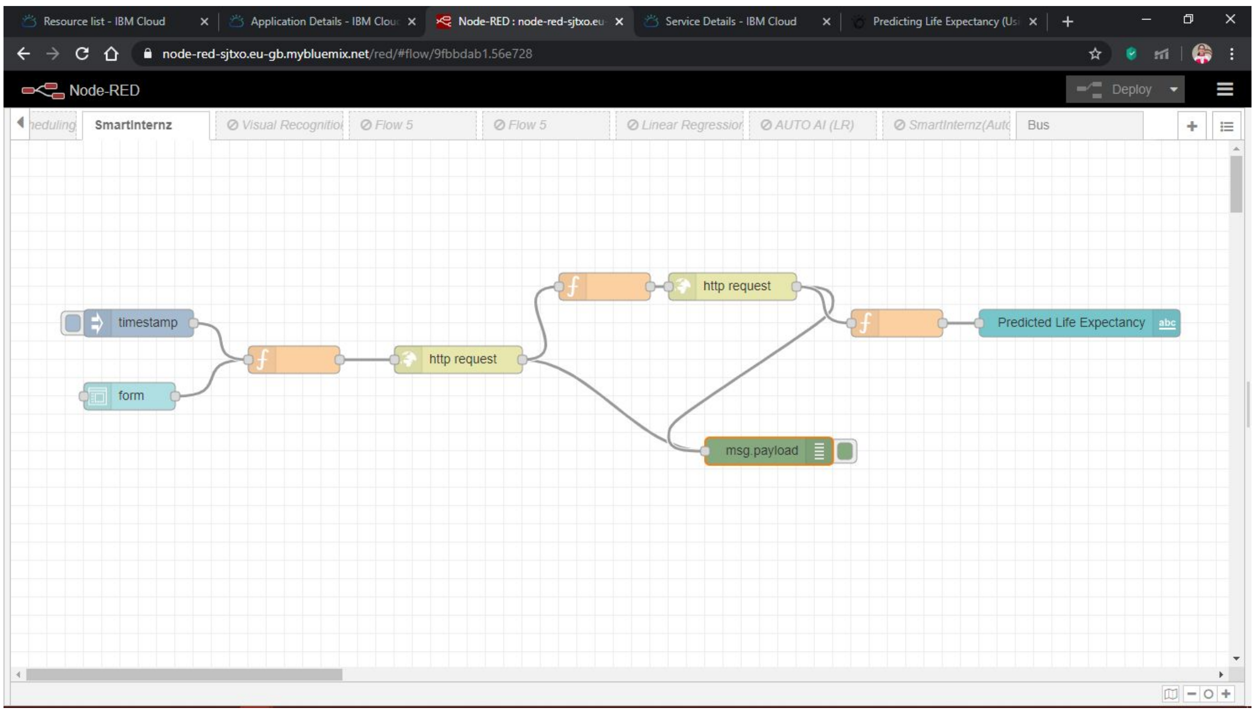
```
#finding IQR value
Q1 = df_data_1.quantile(0.25)
Q3 = df_data_1.quantile(0.75)
IQR = Q3 - Q1
print(IQR)
#dropping the outliers
df_data_1 = df_data_1[~((df_data_1 < (Q1 - 1.5 * IQR)) |(df_data_1 > (Q3 +
1.5 * IQR))).any(axis=1)]
df_data_1.shape
#data visualization

fig, ax = plt.subplots(figsize=(20, 15))

sns.heatmap(df_data_1.corr(), annot=True, linewidths=.5, ax=ax)
#separating dependent and independent variables

x = df_data_1.iloc[:, [0,1,2,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21]]
x.head()

y = df_data_1.iloc[:, [3]]
y.head()
#replacing string variables with int type

#Status - Developing => 0
```

```
#Status - Developed => 1

x['Status'] = x['Status'].replace(to_replace = "Developing", value = 0)
x['Status'] = x['Status'].replace(to_replace = "Developed", value = 1)

#dropping country column
x = x.drop(['Country'], axis=1)
#splitting the data set into train and test data sets.

from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test= train_test_split(x,y, test_size=0.2,
random_state=0)

#applying the Linear Regression algorithm to predict the Life Expectancy.
#training the ml model

from sklearn.linear_model import LinearRegression
model = LinearRegression()
model.fit(x_train, y_train)

#predicting the Life Expectancy
y_pred = model.predict(x_test)
x_test.head()
from sklearn.metrics import r2_score, mean_absolute_error,
mean_squared_error
import math

print("Mean Squared Error: " + str(mean_squared_error(y_test, y_pred)))
```

```
print("Root Mean Squared Error: " +
str(math.sqrt(mean_absolute_error(y_test, y_pred))))
print("Mean Absolute Error: " + str(mean_absolute_error(y_test, y_pred)))
#deploying the trained model


from watson_machine_learning_client import
WatsonMachineLearningAPIClient


wml_creds = {
  "apikey": "B_c8cflsTJTR6ygAzB0Vt4w8zLNE2QIUmluukRwkicHc",
  "iam_apikey_description": "Auto-generated for key
93ee194a-e95d-4224-abd2-92cb50a99c90",
  "iam_apikey_name": "Service credentials-1",
  "iam_role_crn": "crn:v1:bluemix:public:iam::::serviceRole:Writer",
  "iam_serviceid_crn":
"crn:v1:bluemix:public:iam-identity::a/7b03d701cb4e4ee4801022e5d0e1b
baa::serviceid:ServiceId-3160b301-f794-4d5d-add6-eee5e0fd2152",
  "instance_id": "e33d2826-8a4d-472a-a3bf-1511fa023229",
  "url": "https://eu-gb.ml.cloud.ibm.com"
}


client = WatsonMachineLearningAPIClient(wml_creds)


metadata = {
    client.repository.ModelMetaNames.AUTHOR_NAME : "Prasuna",
    client.repository.ModelMetaNames.AUTHOR_EMAIL :
"prasunap2001@gmail.com",
    client.repository.ModelMetaNames.NAME : "PredictingLifeExpectancy"
}
```

```
stored_data = client.repository.store_model(model,
meta_props=metadata)
guid = client.repository.get_model_uid(stored_data)
deploy = client.deployments.create(guid)
scoring_endpoint = client.deployments.get_scoring_url(deploy)
print(scoring_endpoint)
```

12.1 Node-RED Flow



Code (json):

[{"id":"9fbbdab1.56e728","type":"tab","label":"SmartInternz","disabled":false,"info":""},{"id":"488a4d8c.120b84","type":"function","z":"9fbbdab1.56e728","name":"","func":"var country = global.get('country')\nvar year = global.get('year')\nvar status = global.get('status')\nvar adult = global.get('adult')\nvar infant = global.get('infantDeaths')\nvar expHealth = global.get('expHealth')\nvar alcohol = global.get('alcohol')\nvar hepatitis = global.get('hepatitis')\nvar measles = global.get('measles')\nvar BMI = global.get('BMI')\nvar under5 = global.get('under5')\nvar polio = global.get('polio')\nvar expenditure =

global.get('expenditure')\nvar diptheria = global.get('diphtheria')\nvar hiv = global.get('hiv')\nvar gdp = global.get('gdp')\nvar pop = global.get('population')\nvar thin19 = global.get('thin19')\nvar thin9 = global.get('thin9')\nvar income = global.get('incomeRes')\nvar schooling = global.get('schooling')\nvar token=msg.payload.access_token\nvar instance_id=\"e33d2826-8a4d-472a-a3bf-1511fa023229\"\nmsg.headers={'Content-Type': 'application/json',\"Authorization\":\"Bearer \"+token,\"ML-Instance-ID\":instance_id}\nmsg.payload={\"fields\": [\"year\", \"status\", \"adult\", \"infant\",\"alcohol\", \"expHealth\", \"hepatitis\", \"measles\", \"BMI\", \"under5\", \"polio\", \"expenditure\", \"diptheria\", \"hiv\", \"gdp\", \"population\", \"thin19\", \"thin9\", \"income\", \"schooling\"], \"values\": [[year,status,adult,infant,alcohol,expHealth,hepatitis,measles,BMI,under5,polio,expenditure,diptheria,hiv,gdp,pop,thin19,thin9,income,schooling]]}\nreturn msg;","outputs":1,"noerr":0,"x":650,"y":160,"wires":[["239b7c09.659264"]]},{"id":"1886484e.fc00f8","type":"ui_form","z":"9fbbdab1.56e728","name":"","label":"","group":"afba1b28.9ccd88","order":0,"width":0,"height":0,"options":[{"label":"Country","value":"country","type":"text","required":true,"rows":null},{"label":"Year","value":"year","type":"number","required":true,"rows":null},{"label":"Status (Developed - 1, Developing - 0)","value":"status","type":"number","required":true,"rows":null},{"label":"Adult Mortality","value":"adult","type":"number","required":true,"rows":null},{"label":"Infant Deaths","value":"infants","type":"number","required":true,"rows":null},{"label":"Alcohol Intake of People","value":"alcohol","type":"number","required":true,"rows":null},{"label":"Expenditure on Healthcare (Percentage)","value":"expHealth","type":"number","required":true,"rows":null},{"label":"Hepatitis B","value":"hepatitis","type":"number","required":true,"rows":null},{"label":"Measles","value":"measles","type":"number","required":true,"rows":null},{"label":"BMI","value":"BMI","type":"number","required":true,"rows":null},{"label":"Under-Five Deaths","value":"under5","type":"number","required":true,"rows":null},{"label":"Polio","value":"polio","type":"number","required":true,"rows":null},{"label":"Total Expenditure","value":"expenditure","type":"number","required":true,"rows":null},{"label":"Diphtheria","value":"diphtheria","type":"number","required":true,"rows":null},{"label":"HIV/AIDS","value":"hiv","type":"number","required":true,"rows":null},{"label":"GDP","value":"gdp","type":"number","required":true,"rows":null},{"label":"Population","value":"population","type":"number","required":true,"rows":null},{"label":"Thinness (1-19

years)","value":"thin19","type":"number","required":true,"rows":null},{"label":"Thinn ess (5-9
years)","value":"thin9","type":"number","required":true,"rows":null},{"label":"Income Composition of
Resources","value":"incomeRes","type":"number","required":true,"rows":null},{"labe l":"Schooling","value":"schooling","type":"number","required":true,"rows":null}],"for mValue":{"country":"","year":"","status":"","adult":"","infants":"","alcohol":"","expHeal th":"","hepatitis":"","measles":"","BMI":"","under5":"","polio":"","expenditure":"","diph theria":"","hiv":"","gdp":"","population":"","thin19":"","thin9":"","incomeRes":"","scho oling":""},"payload":"","submit":"submit","cancel":"cancel","topic":"","x":130,"y":280," wires":[["6609ef1e.769a6"]]},{"id":"6609ef1e.769a6","type":"function","z":"9fbbdab1. 56e728","name":"","func":"global.set(\"country\",
msg.payload.country)\nglobal.set(\"year\", msg.payload.year)\nglobal.set(\"status\",
msg.payload.status)\nglobal.set(\"adult\",
msg.payload.adult)\nglobal.set(\"infantDeaths\",
msg.payload.infants)\nglobal.set(\"expHealth\",
msg.payload.expHealth)\nglobal.set(\"alcohol\",
msg.payload.alcohol)\nglobal.set(\"hepatitis\",
msg.payload.hepatitis)\nglobal.set(\"measles\",
msg.payload.measles)\nglobal.set(\"BMI\",
msg.payload.BMI)\nglobal.set(\"under5\",
msg.payload.under5)\nglobal.set(\"polio\",
msg.payload.polio)\nglobal.set(\"expenditure\",
msg.payload.expenditure)\nglobal.set(\"diphtheria\",
msg.payload.diphtheria)\nglobal.set(\"hiv\", msg.payload.hiv)\nglobal.set(\"gdp\",
msg.payload.gdp)\nglobal.set(\"population\",
msg.payload.population)\nglobal.set(\"thin19\",
msg.payload.thin19)\nglobal.set(\"thin9\",
msg.payload.thin9)\nglobal.set(\"incomeRes\",
msg.payload.incomeRes)\nglobal.set(\"schooling\", msg.payload.schooling)\nvar
apikey=\"B_c8cflsTJTR6ygAzB0Vt4w8zLNE2QIUmluukRwkicHc\";\nmsg.headers={\"c ontent-type\":\"application/x-www-form-urlencoded\"}\nmsg.payload={\"grant_type \":\"urn:ibm:params:oauth:grant-type:apikey\",\"apikey\":apikey}\nreturn
msg;","outputs":1,"noerr":0,"x":310,"y":240,"wires":[["ab7b3f8c.cac1"]]},{"id":"a350c3 8f.d2cf","type":"debug","z":"9fbbdab1.56e728","name":"","active":true,"tosidebar":tr ue,"console":false,"tostatus":false,"complete":"payload","targetType":"msg","x":830," y":340,"wires":[]},{"id":"ab7b3f8c.cac1","type":"http
request","z":"9fbbdab1.56e728","name":"","method":"POST","ret":"obj","paytoqs":fal se,"url":"https://iam.cloud.ibm.com/identity/token","tls":"","persist":false,"proxy":"",

authType":"","x":490,"y":240,"wires":[["a350c38f.d2cf","488a4d8c.120b84"]]},{"id":"99b017b8.717058","type":"inject","z":"9fbbdab1.56e728","name":"","topic":"","payload":"","payloadType":"date","repeat":"","crontab":"","once":false,"onceDelay":0.1,"x":140,"y":200,"wires":[["6609ef1e.769a6"]]},{"id":"239b7c09.659264","type":"http request","z":"9fbbdab1.56e728","name":"","method":"POST","ret":"obj","paytoqs":false,"url":"https://eu-gb.ml.cloud.ibm.com/v3/wml_instances/e33d2826-8a4d-472a-a3bf-1511fa023229/deployments/b04f1b77-35bb-4176-b10e-8097f0a603c9/online","tls":"","persist":false,"proxy":"","authType":"","x":790,"y":160,"wires":[["a350c38f.d2cf","1b788014.0b23c"]]},{"id":"e78a9246.388da","type":"ui_text","z":"9fbbdab1.56e728","group":"afba1b28.9ccd88","order":1,"width":0,"height":0,"name":"","label":"Predicted Life
Expectancy","format":"{{msg.payload}}","layout":"row-spread","x":1170,"y":200,"wires":[]},{"id":"1b788014.0b23c","type":"function","z":"9fbbdab1.56e728","name":"","func":"msg.payload = msg.payload.values[0][0][0]\nreturn
msg;","outputs":1,"noerr":0,"x":970,"y":200,"wires":[["e78a9246.388da"]]},{"id":"afba1b28.9ccd88","type":"ui_group","z":"","name":"Fill in the details of the Country of which you want to
predict","tab":"f712ea20.10d9a8","order":1,"disp":true,"width":"12","collapse":false},{"id":"f712ea20.10d9a8","type":"ui_tab","z":"","name":"Predicting Life Expectancy of Humans in a country","icon":"dashboard","disabled":false,"hidden":false}]