SMARTBRIDGE

Summer Internship report

Predicting Life Expectancy using Machine Learning

Submitted By:-

PRAGYA PARIMITA PANDA

COLLEGE OF ENGIINEERING AND TECHNOLOGY, BHUBANESWAR

Preface

The purpose of this report is to explain what I did and learned during my internship period with SmartBridge Educational Services PVT Ltd. for the prediction of life-expectancy

under the supervision of mentors of SmartBridge. The report focuses primarily on the assignments handled, and the tasks completed during the period of internship with other technical details. Following the results obtained are discussed and went over. I have tried my best to keep report simple yet technically correct.

It is hoped that this report would serve as one of the prime means towards the development of my technical skills and equally handle the requirements of Machine Learning course.

Pragya Parimita Panda

INDEX

1. INTRODUCTION

1.1 Overview

Life expectancy refers to the number of years a person is expected to live based on the statistical average. The project depends on the accuracy of data, which play a key role in industry both directly as well as indirectly. The data set is of a country and the problem statement is to predict the life expectancy, and life expectancy depends on regional variations, Economic Circumstances, Sex Differences, Mental Illnesses, Physical Illnesses, Education, Year of their birth and other demographic factors. This problem statement comes up with a way to predict average life expectancy of people in that country when various factors such as year, GDP, education, alcohol intake of people in the country, expenditure on health-care system and some specific disease related deaths that happened in the country are given. Health-forecasts play key roles in long-term planning and investments. It will be a beneficial aid for both private companies in order to hire employees, and also for the governments to keep an eye on the factors and their strive for the better life expectancy.

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.

1.2 Purpose

Here, this project takes the above mentioned factors in to consideration, compare them and explore the relationship between them using machine learning algorithms such as linear regression, random forest etc. and chose that algorithm which will do the best or most accurate prediction in terms of how many years a person in that particular area can live or an average age at which people living there will die. And that will be his/her **Life Expectancy.**

1.3 Project Requirements:

Functional Requirements:

Predicting the life expectancy rate of a country.

Technical Requirements:

Python, Machine Learning IBM Cloud, IBM Watson.

Hardware Requirements:

Processor: i3 7th gen or higher

Speed: 2GHz or more

HARD DISK: 10 GB or more

RAM: 4 GB or more

Software Requirements:

Watson Studio, Node-Red

2. LITERATURE SURVEY

2.1 Existing Problem

Predicting a man’s life expectancy has always been a thing next to impossible for mankind. Although there had been a lot of studies undertaken which included income of people, their mortality rate, health status etc. but their immune index and development rate never made in to those records for studies. Globally there are many organizations performing these prediction since long. A number of papers had been published, even more calculations have been laid down in order create one effective and more accurate equation but all these efforts were proved to be impractical.

2.2 Proposed Solution

Some of the past predictions were done by performing a number of linear regressions on data set of one or two years of countries. This can be resolved by formulating linear regression over the data set collected over a period of time, let’s say 15 years, of different countries. We are also taking a few more fields in to considerations such as health and immune development to some of the common diseases, such as Hepatitis, Polio, HIV/AIDS etc., Adult Mortality, Alcohol intake, percentage expenditure, Measles, BMI, Death of under 5 years, Schooling, thinness in 1-19 years and 5-9 years and Population related factors as well. Since the data set consists of data of multiple countries this project will help those countries to focus on the cause off their less life expectancy rate.

Steps:

a) Create IBM cloud services

b) Configure Watson Studio

c) Create Machine Learning Notebook

d) Save and Deploy Model in Notebook

e) Create Node-Red Flow to connect all services together

f) Deploy and run Node-Red app

2.2.1. Create IBM cloud Services

- Watson Studio
- Machine Learning model instance
- Node-Red

2.2.2. Configure Watson Studio

Once all the services are created go to the resource list and launch the Watson Studio. Now in Watson Studio open an empty project and add machine learning as the associate resource in the settings. Create a token as editor and open a Jupyter notebook in to assets and add the data set to it. Finally build the model in the notebook and obtain the scores.

Steps for notebook:

- Install Watson_machine_learning_client
- Import necessary libraries
- Import Data Set
- Descriptive Analysis of Data
  - Using the rename function remove the unusual species in column.
  - Replace NaN values if any with the mean values.
  - Plot a Heat-map to check whether a dimensional reduction can be performed.
- Calculate P-value to know the impact of features on the target value and remove the features with higher p-value.
- Train and Test:

  - Split the data set in to two parts i.e. Input and Output. As Life Expectancy is to be predicted thus it would be our Output column and the rest will act as the Input.
  - Check for numeric and categorical values as for linear regression numeric values are used.
  - For the categorical values use LabelEncoder to convert them into numeric values.
  - Standardize the categorical and numeric values using pipeline.
  - At first independent pipelines for both the parts are designed then they are joined using column transform.
  - Design a regression pipeline using the regression technique.

- Random Forest Regression technique of sklearn.essemble is used as regression algorithm as it gives best accuracy among rest of the algorithms used.
- Then perform train and test split, as for 80% of dataset are trained data and 20% are test data.
- Then fit and do the prediction.

• Model Building and Deployment:

- At first the machine learning service credentials is stored in a variable and passed into Watson Machine Learning API-Client.
- Then the model is built and stored in model_artifact.
- Then deploy the model and generate scoring_endpoint URL.

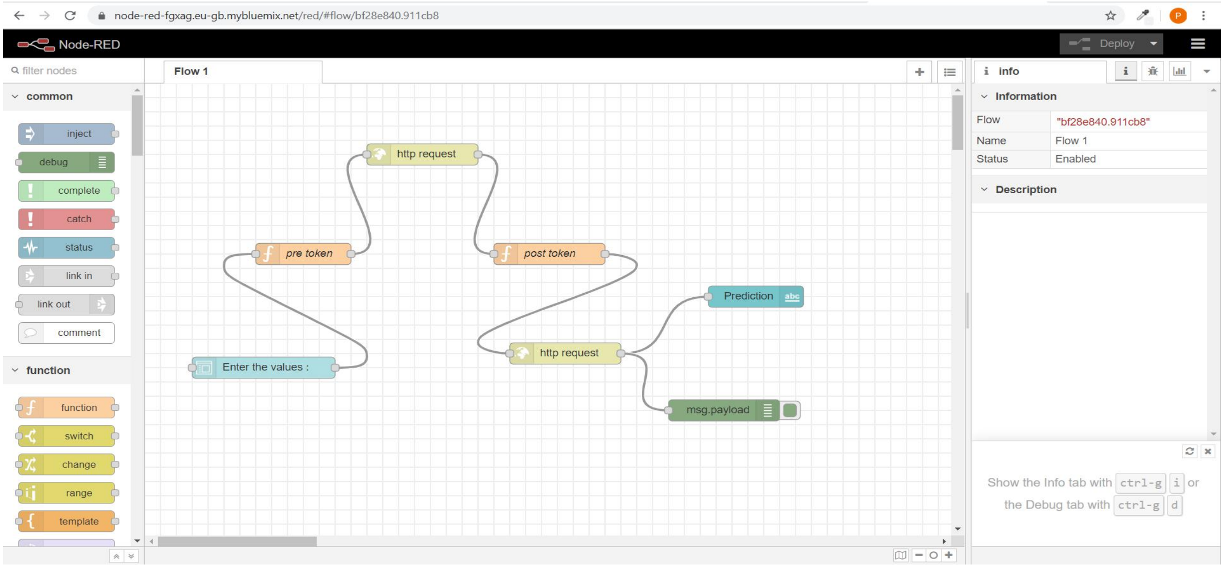
2.2.3. Create Node-Red Flow to connect all services together

• Go to Node-Red Editor from resource list.

• Install node-red Dashboard from manage pallete.

• Now create the flow with the help of following node.

- Inject
- Debug
- Function
- Ui_Form
- Ui_Text



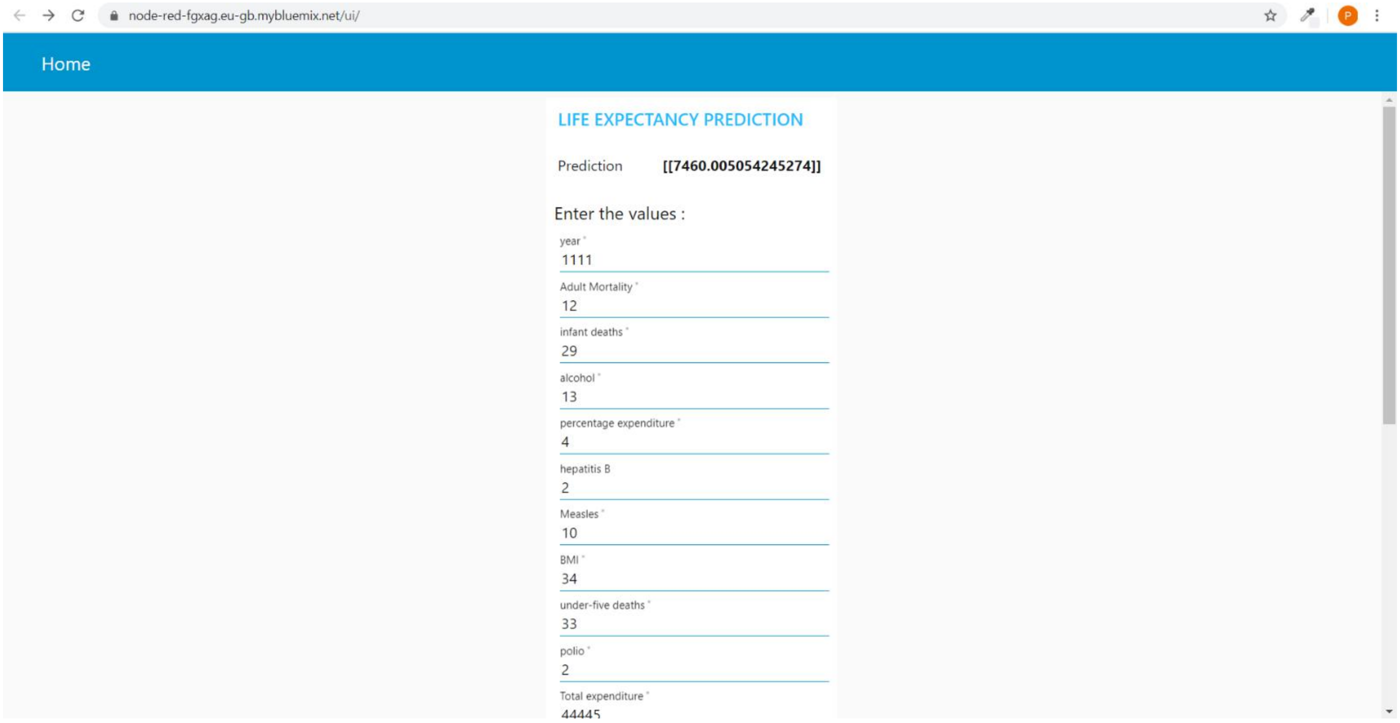
node-red-fgxag.eu-gb.mybluemix.net/red/#flow/bf28e840.911cb8

Flow	"bf28e840.911cb8"
Name	Flow 1
Status	Enabled

Show the Info tab with ctrl-g i or the Debug tab with ctrl-g d

- Deploy and run Node Red app.

Deploy the Node Red flow. Then go to the dashboard and click on the UI URL.



LIFE EXPECTANCY PREDICTION

Prediction [[7460.005054245274]]

Enter the values :

year *
1111

Adult Mortality *
12

infant deaths *
29

alcohol *
13

percentage expenditure *
4

hepatitis B
2

Measles *
10

BMI *
34

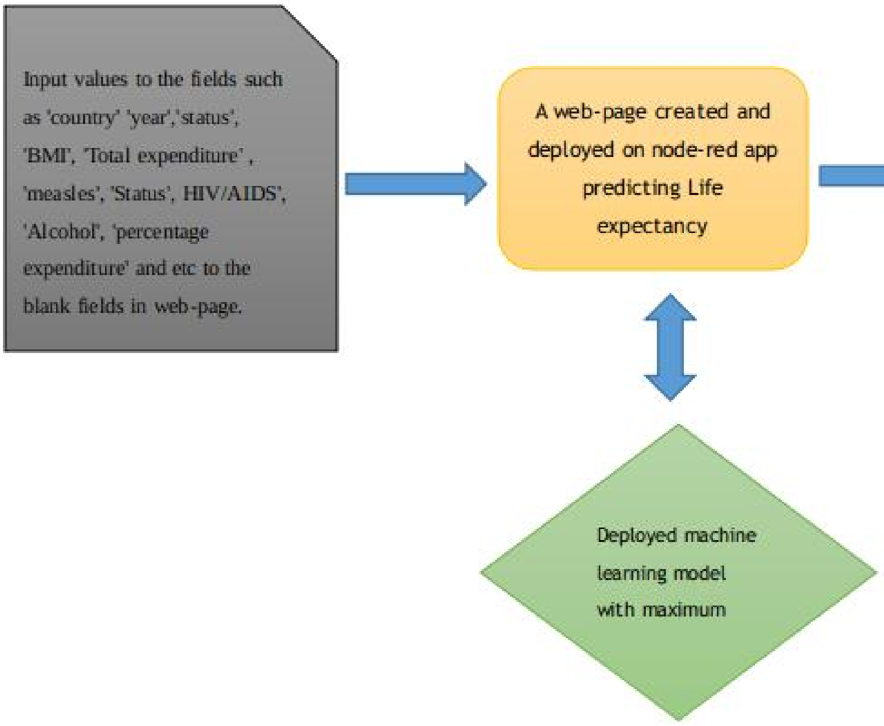
under-five deaths *
33

polio *
2

Total expenditure *
44445

3. THEORITICAL ANALYSIS

3.1 Block Diagram



Input values to the fields such as 'country' 'year','status', 'BMI', 'Total expenditure' , 'measles', 'Status', HIV/AIDS', 'Alcohol', 'percentage expenditure' and etc to the blank fields in web-page.
A web-page created and deployed on node-red app predicting Life expectancy
Predicted Life Expectancy
Deployed machine learning model with maximum

3.2. HARDWARE / SOFTWARE DESIGNING

Project Requirements:

Python,

IBM Cloud

IBM Watson

Functional Requirements:

IBM cloud

Technical Requirements:

ML

WATSON Studio

Python, Node-Red

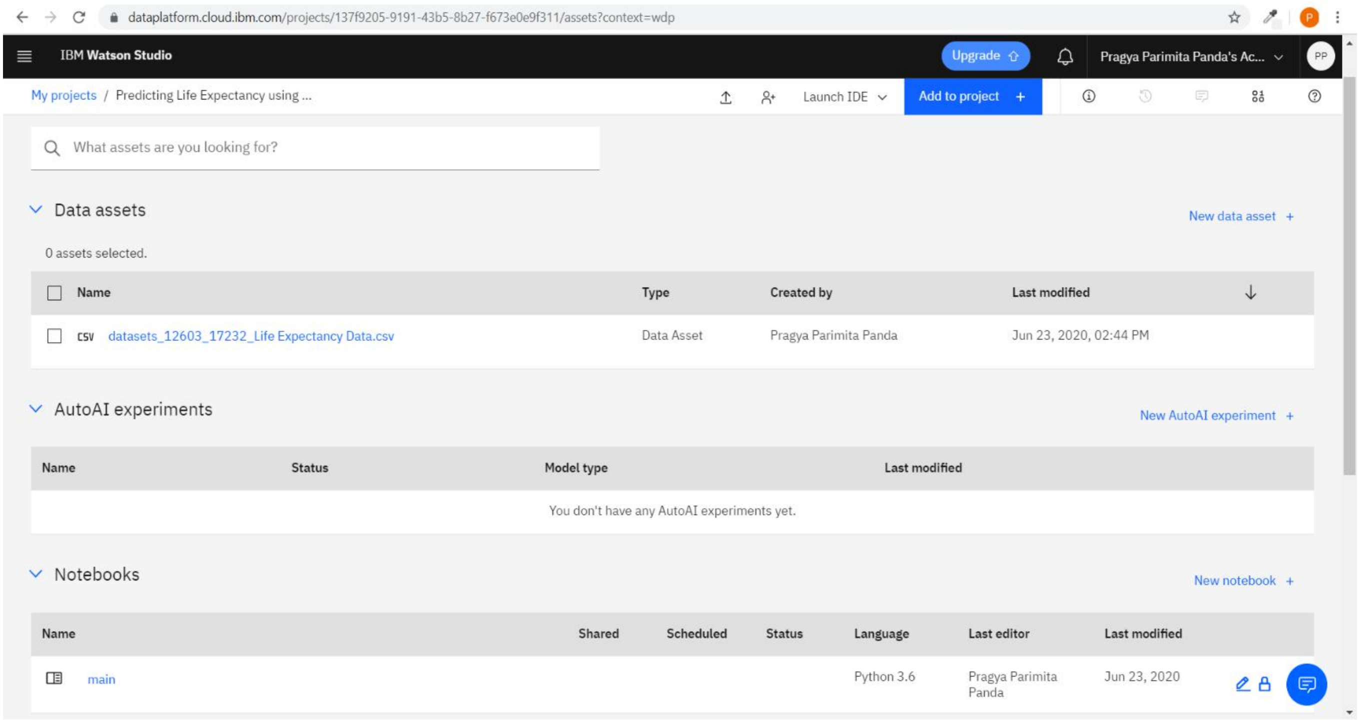
Software Requirements:

Watson Studio

Node-Red

4. EXPERIMENTAL INVESTIGATIONS

A) IBM Cloud Resource List



dataplatform.cloud.ibm.com/projects/137f9205-9191-43b5-8b27-f673e0e9f311/assets?context=wdp

IBM Watson Studio

My projects / Predicting Life Expectancy using ...

What assets are you looking for?

Data assets

0 assets selected.

Name	Type	Created by	Last modified
CSV datasets_12603_17232_Life Expectancy Data.csv	Data Asset	Pragya Parimita Panda	Jun 23, 2020, 02:44 PM

AutoAI experiments

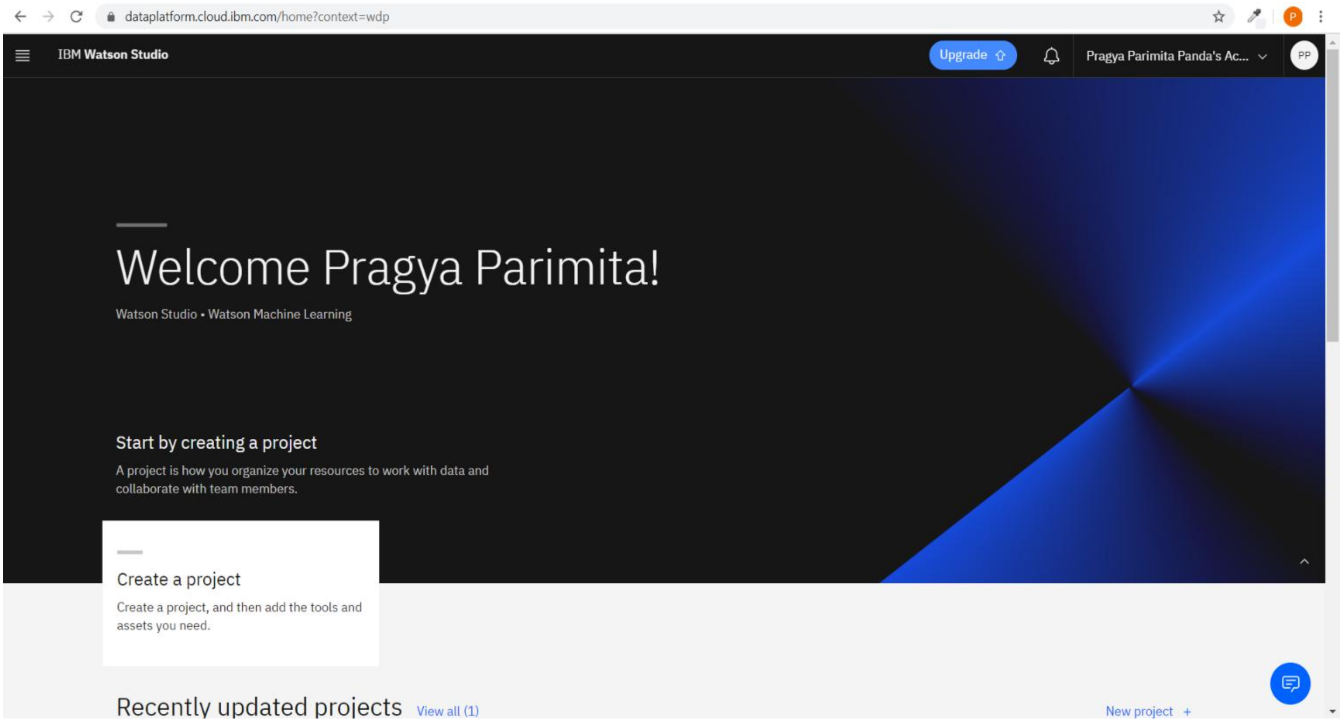
Name	Status	Model type	Last modified

You don't have any AutoAI experiments yet.

Notebooks

Name	Shared	Scheduled	Status	Language	Last editor	Last modified
main				Python 3.6	Pragya Parimita Panda	Jun 23, 2020

B) IBM Watson Studio



dataplatform.cloud.ibm.com/home?context=wdp

IBM Watson Studio

Welcome Pragya Parimita!

Watson Studio • Watson Machine Learning

Start by creating a project

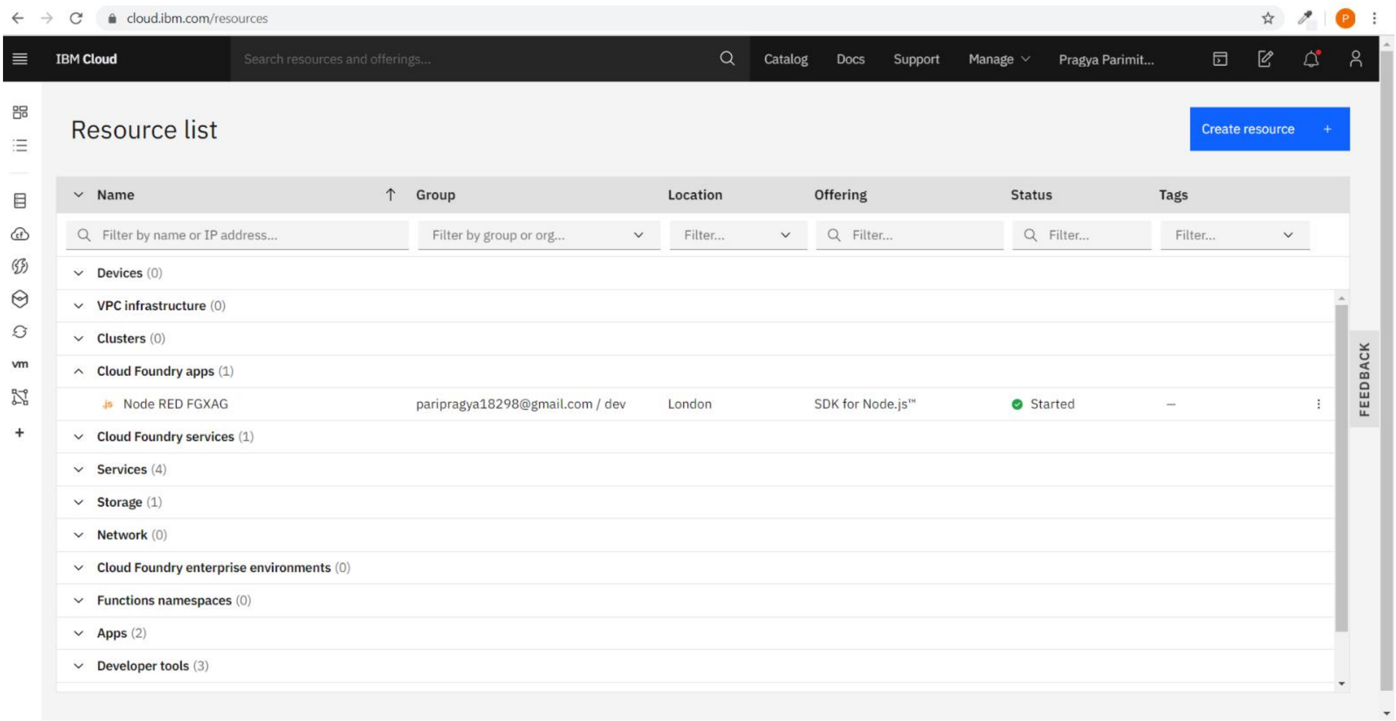
A project is how you organize your resources to work with data and collaborate with team members.

Create a project

Create a project, and then add the tools and assets you need.

Recently updated projects

C) IBM Cloud Project Details



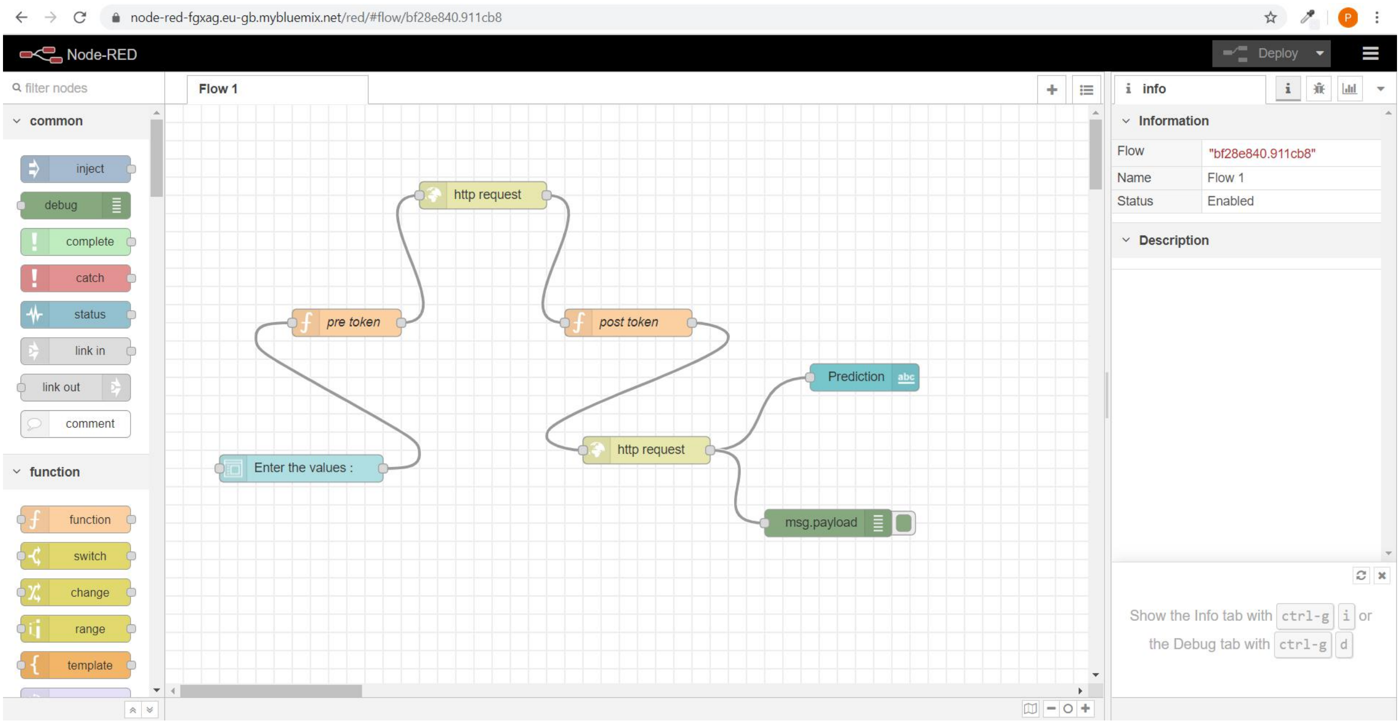
cloud.ibm.com/resources

Resource list

Create resource +

Name	Group	Location	Offering	Status	Tags
Filter by name or IP address...	Filter by group or org...	Filter...	Filter...	Filter...	Filter...
Devices (0)					
VPC infrastructure (0)					
Clusters (0)					
Cloud Foundry apps (1)					
Node RED FGXAG	paripragya18298@gmail.com / dev	London	SDK for Node.js™	Started	–
Cloud Foundry services (1)					
Services (4)					
Storage (1)					
Network (0)					
Cloud Foundry enterprise environments (0)					
Functions namespaces (0)					
Apps (2)					
Developer tools (3)					

D) Node-Red Flow



node-red-fgxag.eu-gb.mybluemix.net/red/#flow/bf28e840.911cb8

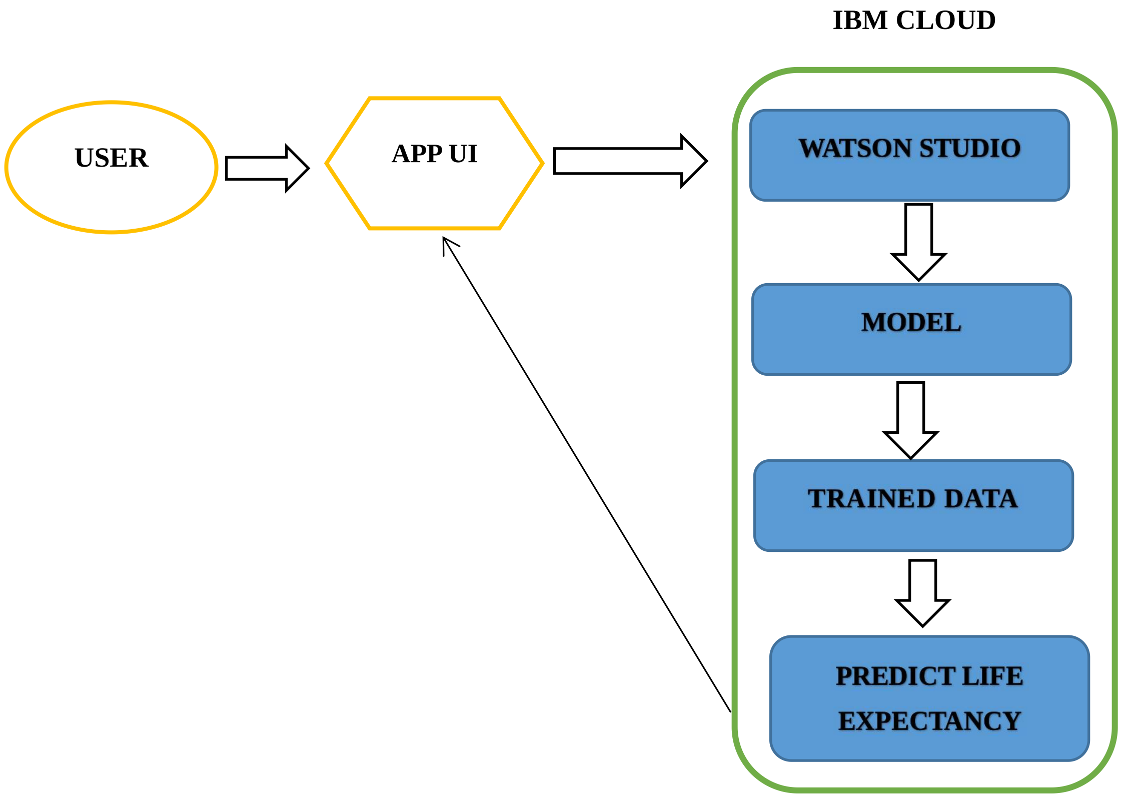
Information

Flow	"bf28e840.911cb8"
Name	Flow 1
Status	Enabled

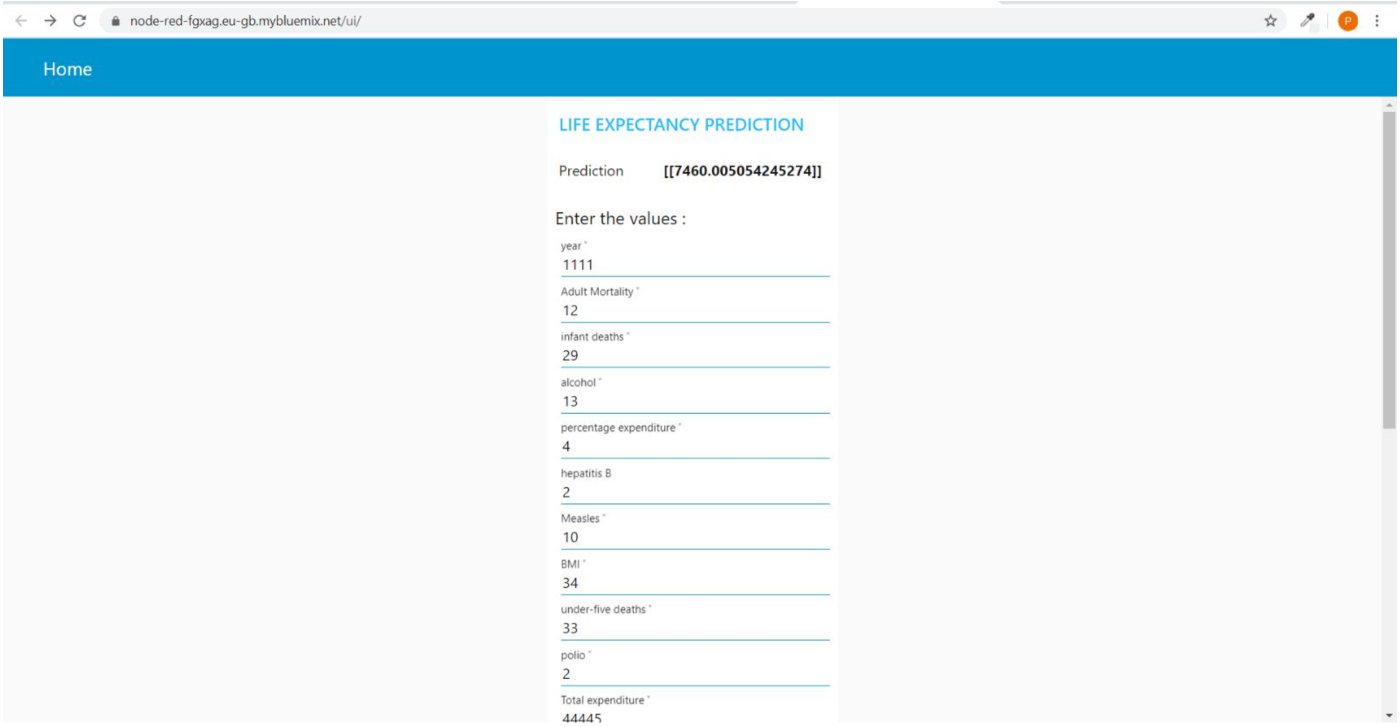
Description

Show the Info tab with ctrl-g i or the Debug tab with ctrl-g d

5. FLOWCHART


IBM CLOUD
USER
APP UI
WATSON STUDIO
MODEL
TRAINED DATA
PREDICT LIFE EXPECTANCY

6. RESULT



LIFE EXPECTANCY PREDICTION

Prediction [[7460.005054245274]]

Enter the values :

year *
1111

Adult Mortality *
12

infant deaths *
29

alcohol *
13

percentage expenditure *
4

hepatitis B
2

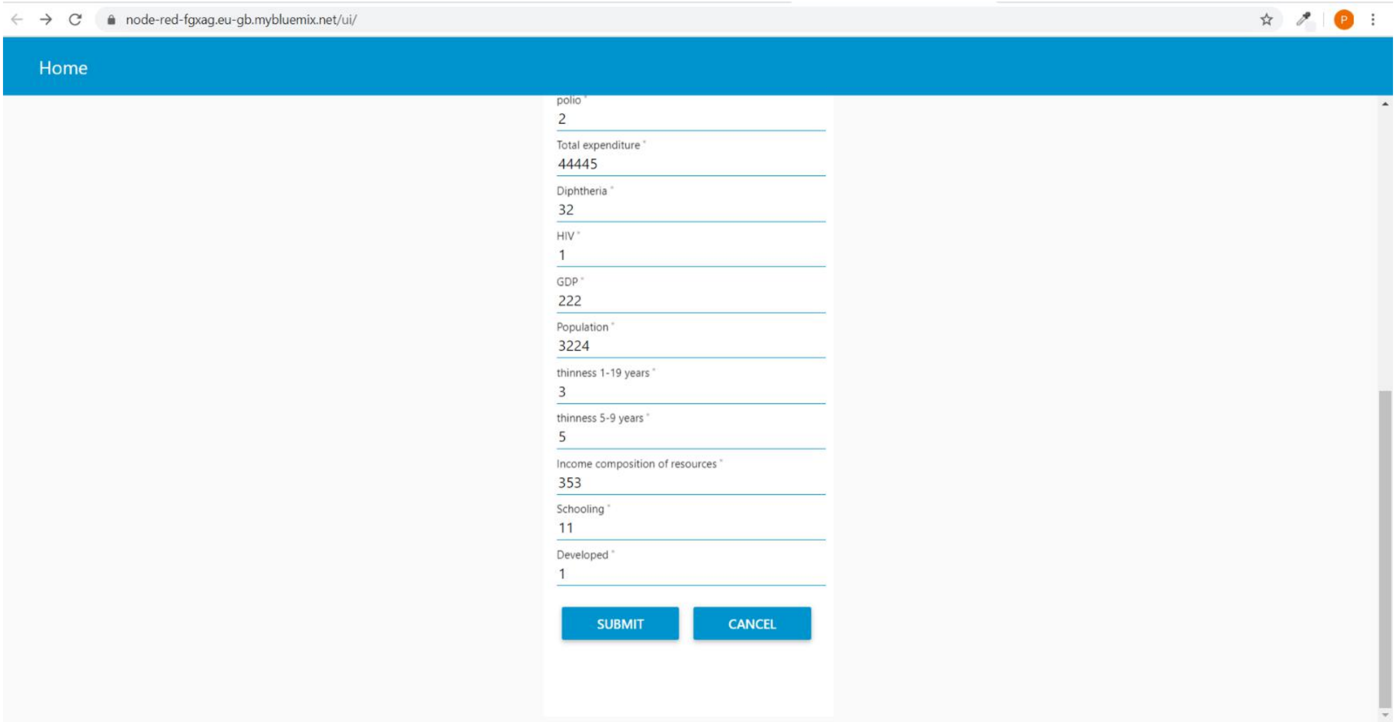
Measles *
10

BMI *
34

under-five deaths *
33

polio *
2

Total expenditure *
44445



polio *
2

Total expenditure *
44445

Diphtheria *
32

HIV *
1

GDP *
222

Population *
3224

thinness 1-19 years *
3

thinness 5-9 years *
5

Income composition of resources *
353

Schooling *
11

Developed *
1

7. ADVANTAGES AND DISADVANTAGES

7.1 Advantages

1. It is one among the most accurate learning algorithms. Its capable of producing a highly accurate classifier for a lot of data sets.

2. It is also efficient on larger databases.

3. It is capable of handling thousands of input variables without deleting any.

4. It produces estimates of variables that are important for the classification.

5. As the forest building progresses it can generate an internal unbiased estimate of the generalization error.

6. It has an effective method for estimating missing data and maintains accuracy when a large proportion of the

data are missing.

7.2 Disadvantages

1. Random forests have been observed to over-fit for some data sets with noisy classification/regression tasks.

2. For data including categorical variables with different number of levels, random forests are biased in favour of those attributes with more levels. Therefore, the variable importance scores from random forest are not reliable for this type of data.

8. APPLICATIONS

This project can be applied to various fields to predict the life expectancy of humans, without much overwork. This will work as a much helpful aid to insurance companies to have a prediction of their customers, and also for other companies that work on such products that affect or are based on human life or help companies so that they can maintain their cycle of work running with the expected life span of their workers.

Doctors can also us this to understand the biological differences of their patients.

Life expectancy is also useful for governments to control the population growth and other factors affecting the life span of the people. It will also help the government to analyse the use of resources for the country.

This may directly motivate and aware the common people of their general health and to make healthier life choices.

9. CONCLUSION

I can honestly say that interning under SmartBridge is one of the most exciting summers I have came across. I have developed new skills on machine learning which is a crucial part of achieving my learning goals. When it comes to my studies I have learned a lot about database and the newest addition to my resume is IBM cloud.

This internship also helped me in identifying my strengths and weakness and also pointing out those areas where i have to put more efforts in coming time.

Moving towards the project developed in this while has an user friendly interface and can be used by anyone. This will be useful to predict the life span of anyone and also for predicting life expectancy rate of a country based on some features such as Year, GDP, Education, Alcohol Intake of people in the country, Expenditure on health care system.

10. FUTURE SCOPE

This project has a bright future ahead and can be applied in large number of cases. It must be upgraded should be considered for a much more cases such as part of education and many more.it can be used to suggest better health care practices and a good life style for improving the life expectancy. In terms of production of medicine and remedies companies can use this to know which sort of diseases or health issue affect more. This data set of this project also have fields describing the alcohol intake, Population, Adult mortality, HIV, Hepatitis etc., which will let us shape the future of an unborn baby.

As the technology is growing faster than ever, as the world is leaning towards more man power need it necessary to improve the factor by which it can extend the life expectancy of its people.

11. BIBLIOGRAPHY

https://nodered.org/

https://bookdown.org/caoying4work/watsonstudioworkshop/auto.html#add-asset-as-auto-ai

https://bookdown.org/caoying4work/watsonstudioworkshop/jn.html#deploy-model-as-web-service

https://www.youtube.com/watch?v=LOCkVmENq8&feature=youtu.be

https://www.youtube.com/watch?v=DBRGlAHdj48&list=PLzpeuWUENMK2PYtasCaKK4bZjaYzhW23L

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

APPENDIX

Source Code:

```
import matplotlib.pyplot as plt

import seaborn as sns

import numpy as np

import types

import pandas as pd

from botocore.client import Configimport 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_8c941c663c3844ebae81be2668b8ee1f = ibm_boto3.client(service_name='s3',
ibm_api_key_id='OF2osVoCsi0JftNKJHuYnTPB_U-W8zFlsvHcDYuvAgjg',
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_8c941c663c3844ebae81be2668b8ee1f.get_object(Bucket='predi

ctinglifeexpectancyusingmach-donotdelete-pr-e1t7dnurnuq3sw',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 )

ds = pd.read_csv(body)

ds.head()

ds.info()

ds.isnull().sum

dummy=pd.get_dummies(ds['Status'])

dummy

ds['Status'].value_counts()

ds=pd.concat([ds,dummy],axis=1)
```

```
ds=ds.drop(columns=['Status','Developing'])
ds.head()
ds=ds.drop(['Country'],axis=1)
ds.head()
ds=ds.apply(lambda x: x.fillna(x.mean()),axis=0)
ds.isnull().sum()
ds.head()
x=ds.iloc[:,[0,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20]].value
s
y=ds.iloc[:,1].values
x
ds.columns
x[0]
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)
#split into test and training set
from sklearn.linear_model import LinearRegression
obj=LinearRegression()
obj.fit(x_train,y_train)
#model analysis the training sets
#same as simple linear regression
#predicting the test set results
y_pred=obj.predict(x_test)
from sklearn import metrics
print(metrics.mean_absolute_error(y_test, y_pred))
print(metrics.mean_squared_error(y_test, y_pred))
print(np.sqrt(metrics.mean_squared_error(y_test, y_pred)))
!pip install watson-machine-learning-client
```

```
from watson_machine_learning_client import WatsonMachineLearningAPIClient

wml_credentials={

  "apikey": "rXf5oXaIfJ8dnZdGcXpqNnypZEt4aA1g8dQT9tVIDQfz", "instance_id":
"2a56a0d1-f6b1-416a-9ff5-ccf1d0606cc8", "url":
"https://us-south.ml.cloud.ibm.com"}client=WatsonMachineLearningAPIClient( wml_credentials )

model_props={client.repository.ModelMetaNames.AUTHOR_NAME: "Pragya",
client.repository.ModelMetaNames.AUTHOR_NAME:"paripragya18298@gmail.com",
client.repository.ModelMetaNames.NAME:"Life_Expectancy"}

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

published_model_uid=client.repository.get_model_uid(model_artifact)

published_model_uid

deployment=client.deployments.create(published_model_uid,name="Life_Expectancy")

scoring_endpoint=client.deployments.get_scoring_url(deployment)

scoring_endpoint
```