PROJECT REPORT

Name: Rishab kumar

Email: rishabkumar718@gmail.com

Title: **Predicting Life Expectancy of A Country**

Category: **Machine Learning**

WebPage Link: <https://node-red1143.eu-gb.mybluemix.net/ui/#!/0?socketid=-YzzUXq-K-pGncQsAAAi>

# Contents

## INTRODUCTION

* 1. Overview
  2. Purpose

## LITERATURE SURVEY

* 1. Existing problem
  2. Proposed solution

## THEORITICAL ANALYSIS

* 1. Block diagram
  2. Hardware / Software designing

## EXPERIMENTAL INVESTIGATIONS

1. **FLOWCHART**
2. **RESULT**
3. **ADVANTAGES & DISADVANTAGES**
4. **APPLICATIONS**
5. **CONCLUSION**
6. **FUTURE SCOPE**
7. **BIBILOGRAPHY**

**APPENDIX:** Source code

1. INTRODUCTION

1.1. Overview

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. It is very important to predict average life expectancy of a country to analyse further requirements to increase its rate of growth or stabilise the rate of growth in that country. So this is a typical Regression Machine Learning project that leverages historical data to predict insights into the future.

The end product will be a webpage where you need to give all the required inputs and then submit it . Afterwards it will predict the life expectancy value based on your regression technique.

**Project Requirements**: An appropriate dataset is needed to build the model.

**Functional Requirements**: IBM cloud

**Technical Requirements**: Computer/Laptop, Minimum 2.2GHz Processor, Any

Operating System.

**Software Requirements**: IBM Watson, IBM Machine Learning Services, Github,

Python v3, IBM Notebook / Jupyter Notebook.

**Project Deliverables**: Smartinternz Internship

**Project Team**: Individual project.

**Project Duration**: 23.5 Days

**1.2. Purpose**

Predicting the life expectancy will give the country an idea of the factors which can be improved to increase the lifespan of the people living, like by improving the health care facilities or immunization vaccines for infants. By making changes in lifestyle, a person can live a long, healthy and good quality life. This will also benefit the country by increasing manpower that will contribute to the economical growth. We should take full advantage of this new era advanced technology to improve the future by predicting it in the present.

2. LITERATURE SURVEY

2.1. Existing Problem

We have reviewed existing works and techniques used in the prediction of the human life expectancy, and reached a conclusion that it is feasible for individuals using evolving technologies and devices wearables and mobile health monitoring devices. We have also identified that the factors used for predicting were just personal causes and not related to the surrounding, healthcare facilities, demographic, social, regional and economical factors of the country he resides. These country dependent factors can also be an important feature to predict the life expectancy of an individual. So we need more data to predict more accurately.

2.2. Proposed Solution

Steps:

1. a) Create IBM cloud services
2. b) Configure Watson Studio
3. c) Create Node-Red Flow to connect all services together
4. d) Deploy and run Node-Red app

**2.2.1. Create IBM cloud Services**

• Watson Studio

• Machine Learning resource

• Node-Red

**2.2.2. Configure Watson Studio**

After creating all services, Go to resource list and launch watson studio then get started with watson studio. Then create an empty project and add machine learning resource as associated services in settings. Create a token as editor type.

Then add dataset and empty jupyter notebook into Assets.

After that go to notebook and write your code to build model and get the scoring endpoint url.

Steps for notebook:

* Install Watson\_machine\_learning\_client
* Import necessary libraries
* Import dataset
* Data Preprocessing

o Removing unusal species in column names using rename function.

o Replacing nan values if any with their mean values.

* Train and Test

* 1. o The dataset was splitted into two parts i.e Input and Output. As Life Expectancy needs to be
  2. predicted so it is to be treated as output and all other columns are treated as Input .

o Afterwards as we need regression technique to build our model so each and every column needs

* 1. to be numeric . So then we check for numeric and categoric columns.
  2. o So I have used ExtraTreesRegressor technique of sklearn.essemble as my regression algorithm
  3. because it best fits my dataset.
  5. o Then train and test split was performed and 80% of dataset were trained data and 20% were test
  6. data.
  7. o Then dataset was fitted and predicted.
  8. o Then error and accuracy was estimated and the mean squared error is 2.4 whereas the R2\_score or
  9. accuracy is 94.27%.
* Model Building and Deployment

o At first the machine learning service credentials was stored in a variable and passed into

WatsonMachineLearningAPIClient.

o Then the model was build and stored in model\_artifact.

o Then the model was deployed and scoring\_endpoint url was generated .

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

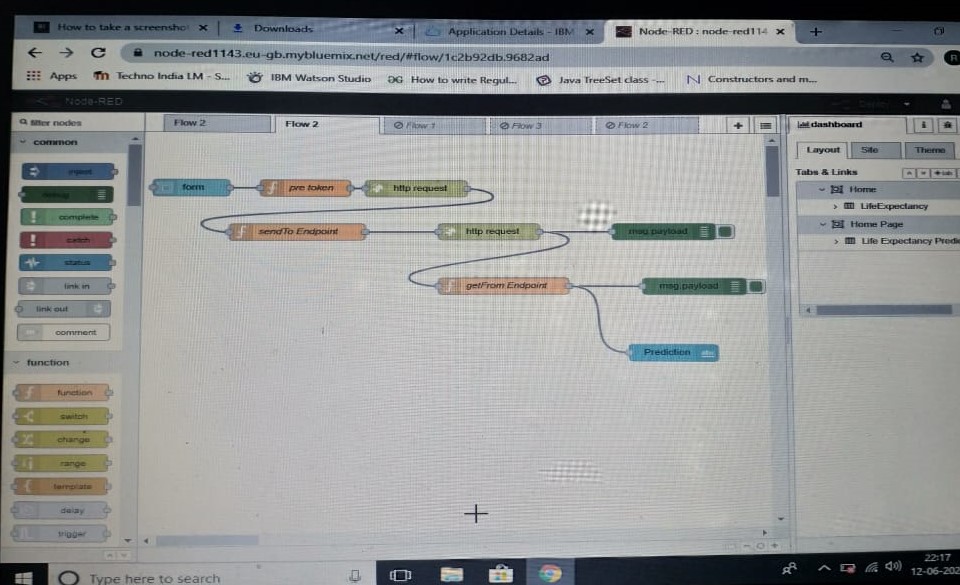
▪Go to Node-Red Editor from resource list.

▪Install node-red Dashboard from manage pallette.

▪Now create the flow with the help of following node.

o Inject

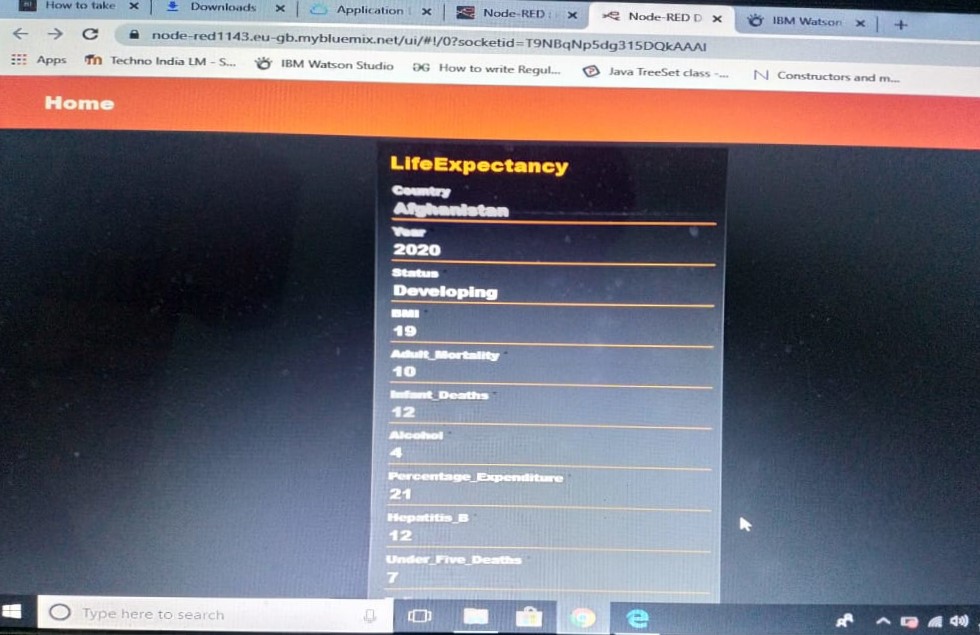
* 1. o Debug
  2. o Function
  3. o Ui\_Form
  4. o Ui\_Text



* Deploy and run Node Red app.

Deploy the Node Red flow. Then copy the link url upto .net/ and paste at a new tab by ui at the

end of the url like this



3. THEORETICAL ANALYSIS

3.1. BLOCK DIAGRAM

Input values to the fields such as 'country', 'BMI', 'Total expenditure' , 'measles', 'Status', HIV/AIDS', 'Alcohol' , 'percentage expenditure' and etc to the blank fields in webpage.

Deployed machine learning model with maximum accuracy score

Predicted Life Expectancy value

3.2. HARDWARE / SOFTWARE DESIGNING

o **Project Requirements**: Python, IBM Cloud, IBM Watson

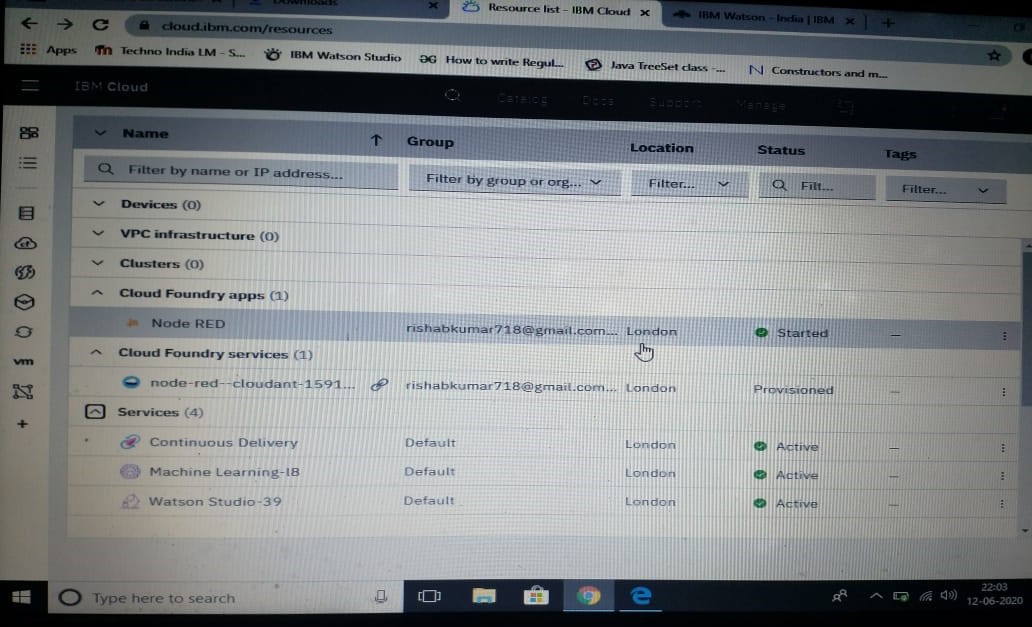
o **Functional Requirements**: IBM cloud

o **Technical Requirements**: ML, WATSON Studio, Python, Node-Red

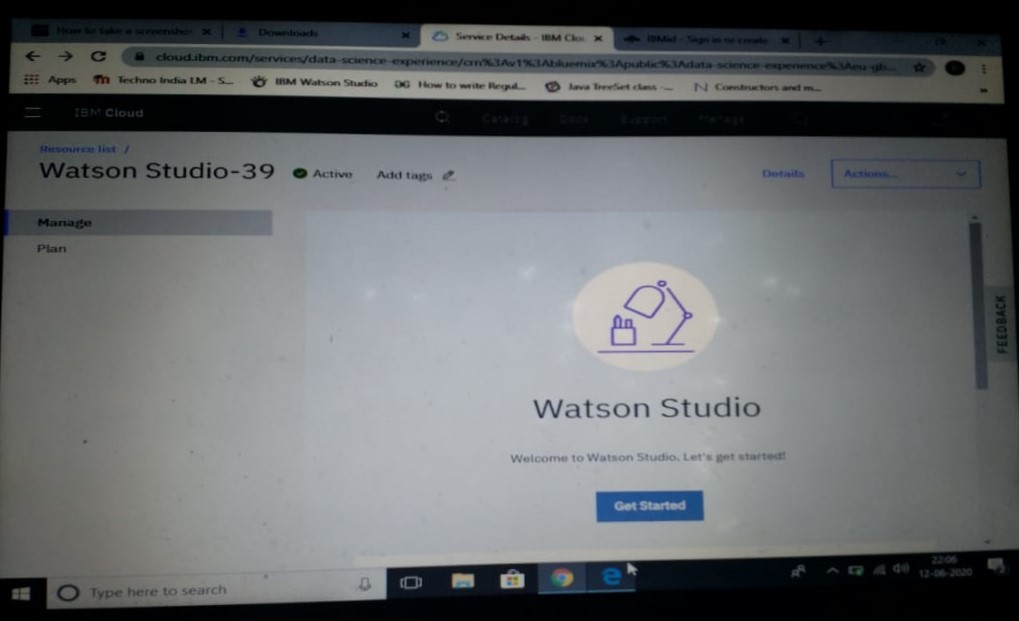
o **Software Requirements**: Watson Studio, Node-Red

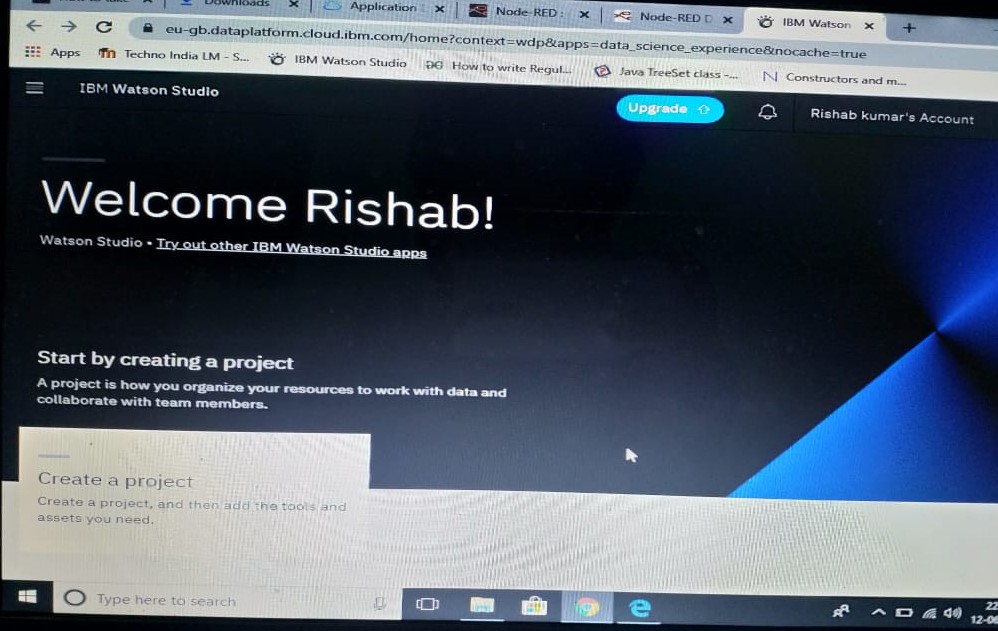
4. EXPERIMENTAL INVESTIGATIONS

**A) IBM Cloud Resource List**

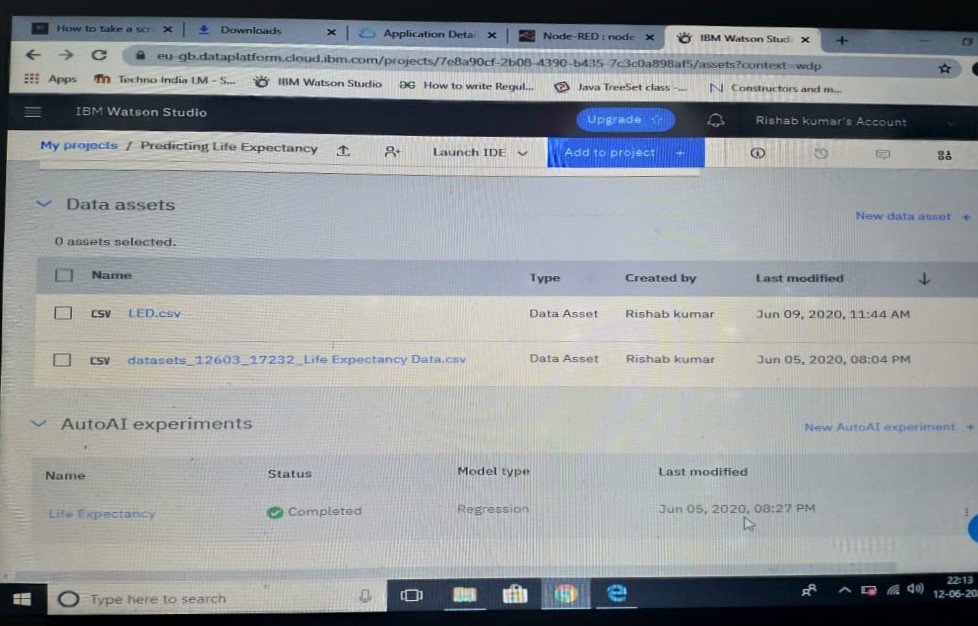


**B) IBM Watson Studio**

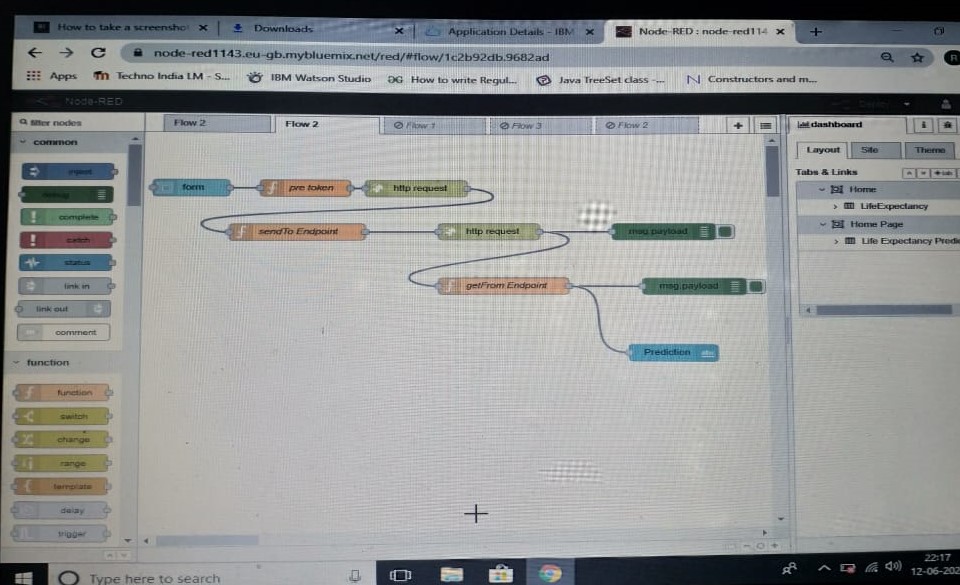




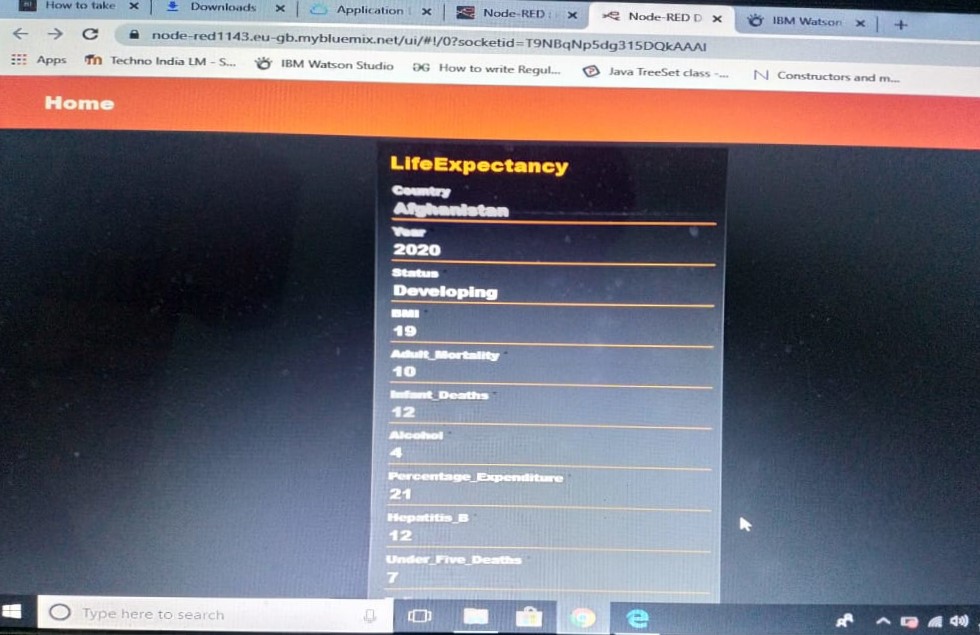
**C) IBM Cloud Project Details**

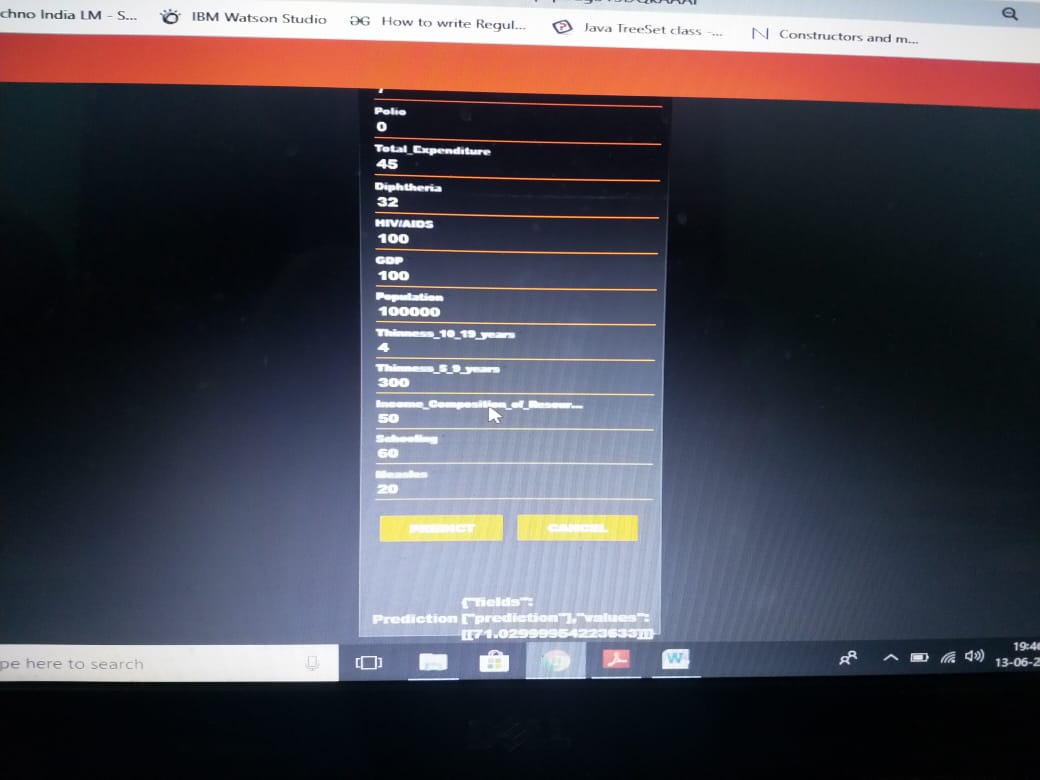


**D) Node-Red Flow**



**E) Life Expectancy Prediction UI**



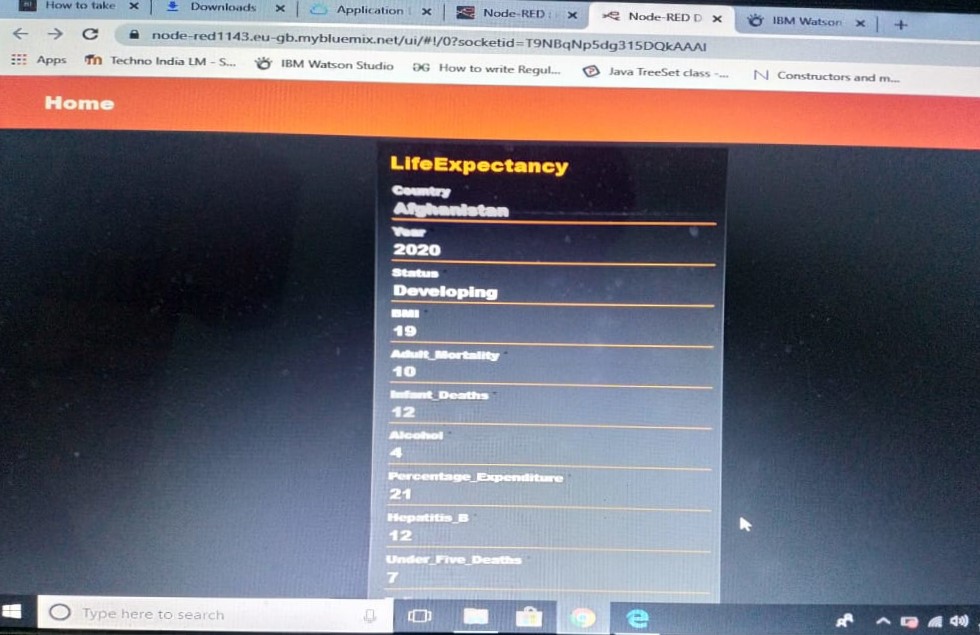


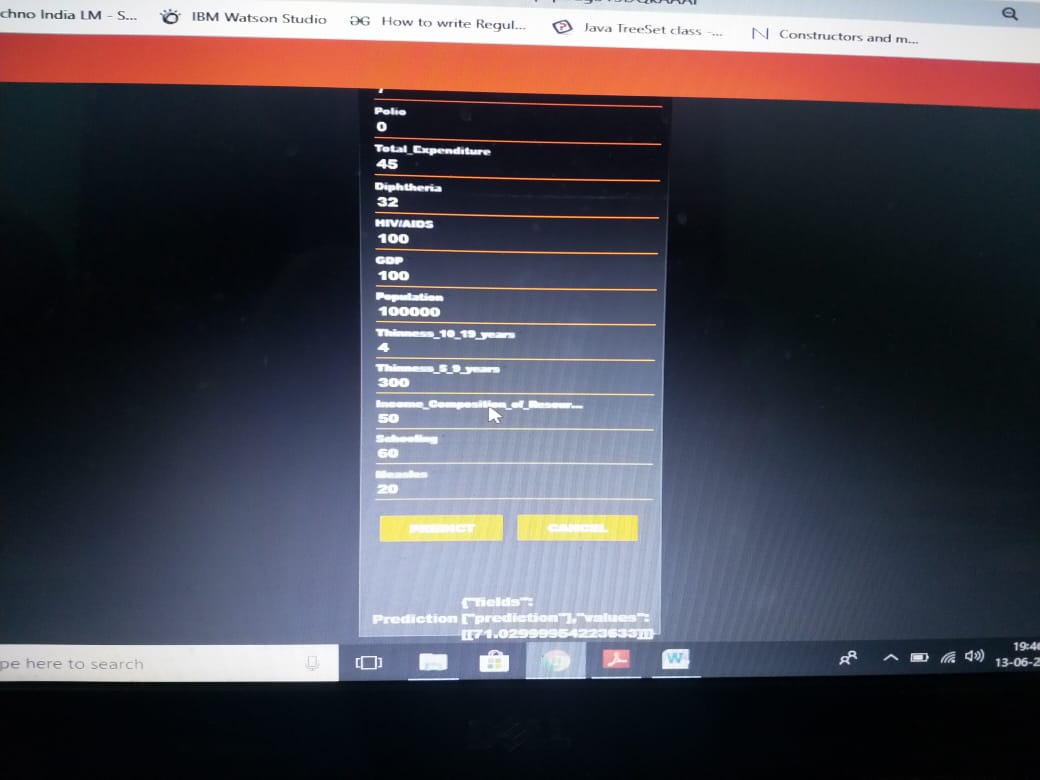
5. FLOWCHART



3. a) The user input all the required values in the app
4. b) The data then entered into watson and the scoring\_endpoint url matches with the
5. deployed model.
6. c) Then it enters into trained data and predict the life expetancy value
7. d) The value predicted is prompted in the app screen.

**6. RESULT**





7. ADVANTAGES AND DISADVANTAGES

ADVANTAGES:

a) Health Inequalities: Life expectancy has been used nationally to monitor health inequalities

of a country.

b) Reduced Costs: This is a simple webpage and can be accessed by any citizen of a country to

calculate life expectancy of their country and does not required any kind of payment neither

for designing nor for using.

c) User Friendly Interface: This interface requires no background knowledge of how to use it.

It’s a simple interface and only ask for required values and predict the output.

DISADVANTAGES:

a) Requires internet connection.

b) User input is not saved in any database.

c) Input should be in range only to predict accurate values.

.

8. APPLICATION

1. a) It can be used to monitor health inequalities of a country.
2. b) It can be used to develop statistics for country development process.
3. c) It can be used to analyse the factors for high life expectancy.
4. d) It is user friendly and can be used by anyone.

9. CONCLUSION

This user interface will be useful for the user to predict life expectancy value of their own country or any other country based on some required details such as GDP, BMI, Year, Alcohol Intake, Total expenditure and etc.

10. FUTURE SCOPE

Future Scope of the Model can be:

a) Feature Reduction

It requires much more data about 21 columns to be known prior for predicting life expectancy which can be again difficult for a normal user to gather such datas so I have decided to do some kind of feature reduction or replacement of some features as individuals or groups to make it more user friendly.

b) Attractive UI

It is a simple webpage only asking inputs and predict output. In future I have decided to make it more user friendly by providing some useful information about the country in the webpage itself so that user does not need to do any kind of prior research for the values.

c) Integrating with services such as speech recognition

11. BIBLIOGRAPHY

* <https://cloud.ibm.com/docs/overview?topic=overview-whatis-platform>
* <https://developer.ibm.com/tutorials/how-to-create-a-node-red-starter-application/>
* <https://nodered.org/>
* <https://github.com/watson-developer-cloud/node-red-labs>
* <https://www.youtube.com/embed/r7E1TJ1HtM0>
* <https://bookdown.org/caoying4work/watsonstudio-workshop/jn.html>
* <https://www.kaggle.com/kumarajarshi/life-expectancy-who>
* [https://www.youtube.com/watch?v=DBRGlAHdj48&list=PLzpeuWUENMK2PYtasCaKK4b ZjaYzhW23L](https://www.youtube.com/watch?v=DBRGlAHdj48&amp;list=PLzpeuWUENMK2PYtasCaKK4bZjaYzhW23L)
* [https://www.youtube.com/watch?v=- CUi8GezG1I&list=PLzpeuWUENMK2PYtasCaKK4bZjaYzhW23L&index=2](https://www.youtube.com/watch?v=-CUi8GezG1I&amp;list=PLzpeuWUENMK2PYtasCaKK4bZjaYzhW23L&amp;index=2)
* <https://www.youtube.com/watch?v=Jtej3Y6uUng>
* [https://bookdown.org/caoying4work/watsonstudio-workshop/jn.html#deploy-model-as-web- service](https://bookdown.org/caoying4work/watsonstudio-workshop/jn.html#deploy-model-as-web-service)
* <https://machinelearningmastery.com/columntransformer-for-numerical-and-categorical-data/>

APPENDIX: Source Code

1. Notebook

**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\_cff2a8d1b0d04678b921cec035fc98de = ibm\_boto3.client(service\_name='s3',

ibm\_api\_key\_id='M7-PqwpEDwPlumeKR03ueyXHEYxVF8pvJRtr2D6ltNi7',

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\_cff2a8d1b0d04678b921cec035fc98de.get\_object(Bucket='predictinglifeexpectancy-donotdelete-pr-19ljnmivr1uyzz',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()

**import** **numpy** **as** **np** **import** **seaborn** **as** **sns** **import** **matplotlib.pyplot** **as** **plt** df\_data\_1=df\_data\_1.dropna() df\_data\_1=df\_data\_1.fillna(df\_data\_1.mean()) df\_data\_1=df\_data\_1.drop(['Status'],axis=1) df\_data\_1=df\_data\_1.drop(['Year'],axis=1) x=df\_data\_1.iloc[:,[0,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19]].values y=df\_data\_1.iloc[:,1].values **from** **sklearn.preprocessing** **import** LabelEncoder, OneHotEncoder labelencoder\_x=LabelEncoder() x[:,0]=labelencoder\_x.fit\_transform(x[:,0]) onehotencoder=OneHotEncoder(categorical\_features=[0]) x=onehotencoder.fit\_transform(x).toarray() x=x[:,1:] **from** **sklearn.model\_selection** **import** train\_test\_split x\_train,x\_test,y\_train,y\_test=train\_test\_split(x,y,test\_size=0.5,random\_state=77)

**from** **sklearn.linear\_model** **import** LinearRegression model1=LinearRegression() model1.fit(x\_train,y\_train) y\_pred=model1.predict(x\_test) plt.scatter(y\_test,y\_pred ,color='black') plt.title('life expectancy Prediction(Graph)') plt.xlabel('Actual Life Expectancy') plt.ylabel('Predicted Life Expectancy') plt.show()

print('**\n**accuracy:**{}**%'.format(model1.score(x\_test, y\_test)\*100))

**from** **watson\_machine\_learning\_client** **import** WatsonMachineLearningAPIClient

wml\_credentials = {

"apikey": "2ZHwvgFj-F2xtimx30iw8AWCAyTOc380I9AKp6\_ZiCs1",

"iam\_apikey\_description": "Auto-generated for key b971c0d3-c1f4-45c0-91af-9b4ef6da2270",

"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/f89a3ade0e9c4c439a52d976275f9206::serviceid:ServiceId-5a5ea863-d3df-4d49-9a77-350dab3ab146",

"instance\_id": "1653720d-9a98-4947-a75c-7db79018b680",

"url": "https://eu-gb.ml.cloud.ibm.com"

}

client = WatsonMachineLearningAPIClient( wml\_credentials )

model\_props = {client.repository.ModelMetaNames.AUTHOR\_NAME: "Rishab Kumar",

client.repository.ModelMetaNames.AUTHOR\_EMAIL: "rishabkumar718@gmail.com",

client.repository.ModelMetaNames.NAME: "Life\_Prediction"}

model\_artifact =client.repository.store\_model(model1, meta\_props=model\_props)

model\_artifact

guid = client.repository.get\_model\_uid(model\_artifact)

guid

deploy = client.deployments.create(guid, name="Life\_Prediction")

scoring\_endpoint = client.deployments.get\_scoring\_url(deploy)

scoring\_endpoint

2) Flow.json

[{"id":"1c2b92db.9682ad","type":"tab","label":"Flow 2","disabled":false,"info":""},{"id":"8f76f183.49d75","type":"ui\_form","z":"1c2b92db.9682ad","name":"","label":"","group":"1e7bd5e9.eb3c7a","order":1,"width":0,"height":0,"options":[{"label":"Country","value":"a","type":"text","required":true,"rows":null},{"label":"Year","value":"b","type":"number","required":true,"rows":null},{"label":"Status","value":"c","type":"text","required":true,"rows":null},{"label":"BMI","value":"d","type":"number","required":true,"rows":null},{"label":"Adult\_Mortality","value":"e","type":"number","required":true,"rows":null},{"label":"Infant\_Deaths","value":"f","type":"number","required":true,"rows":null},{"label":"Alcohol","value":"g","type":"number","required":true,"rows":null},{"label":"Percentage\_Expenditure","value":"h","type":"number","required":true,"rows":null},{"label":"Hepatitis\_B","value":"i","type":"number","required":true,"rows":null},{"label":"Under\_Five\_Deaths","value":"j","type":"number","required":true,"rows":null},{"label":"Polio","value":"k","type":"number","required":true,"rows":null},{"label":"Total\_Expenditure","value":"l","type":"number","required":true,"rows":null},{"label":"Diphtheria","value":"m","type":"number","required":true,"rows":null},{"label":"HIV/AIDS","value":"n","type":"number","required":true,"rows":null},{"label":"GDP","value":"o","type":"number","required":true,"rows":null},{"label":"Population","value":"p","type":"number","required":true,"rows":null},{"label":"Thinness\_10\_19\_years","value":"q","type":"number","required":true,"rows":null},{"label":"Thinness\_5\_9\_years","value":"r","type":"number","required":true,"rows":null},{"label":"Income\_Composition\_of\_Resources","value":"s","type":"number","required":true,"rows":null},{"label":"Schooling","value":"t","type":"number","required":true,"rows":null},{"label":"Measles","value":"u","type":"number","required":true,"rows":null}],"formValue":{"a":"","b":"","c":"","d":"","e":"","f":"","g":"","h":"","i":"","j":"","k":"","l":"","m":"","n":"","o":"","p":"","q":"","r":"","s":"","t":"","u":""},"payload":"","submit":"Predict","cancel":"cancel","topic":"","x":70,"y":100,"wires":[["e323ad7e.1cebd"]]},{"id":"e323ad7e.1cebd","type":"function","z":"1c2b92db.9682ad","name":"pre token","func":"//make user given values as global variables\nglobal.set(\"a\",msg.payload.a);\nglobal.set(\"b\",msg.payload.b);\nglobal.set(\"c\",msg.payload.c);\nglobal.set(\"d\",msg.payload.d);\nglobal.set(\"e\",msg.payload.e);\nglobal.set(\"f\",msg.payload.f);\nglobal.set(\"g\",msg.payload.g);\nglobal.set(\"h\",msg.payload.h);\nglobal.set(\"i\",msg.payload.i);\nglobal.set(\"j\",msg.payload.j);\nglobal.set(\"k\",msg.payload.k);\nglobal.set(\"l\",msg.payload.l);\nglobal.set(\"m\",msg.payload.m);\nglobal.set(\"n\",msg.payload.n);\nglobal.set(\"o\",msg.payload.o);\nglobal.set(\"p\",msg.payload.p);\nglobal.set(\"q\",msg.payload.q);\nglobal.set(\"r\",msg.payload.r);\nglobal.set(\"s\",msg.payload.s);\nglobal.set(\"t\",msg.payload.t);\nglobal.set(\"u\",msg.payload.u);\n\n//following are required to receive a token\nvar apikey=\"STkSGgOcEqkk\_PMB0-hJ5hexaYJuynMu0Pu0IbcGAlCc\";\nmsg.headers={\"content-type\":\"application/x-www-form-urlencoded\"};\nmsg.payload={\"grant\_type\":\"urn:ibm:params:oauth:grant-type:apikey\",\"apikey\":apikey};\nreturn msg;\n","outputs":1,"noerr":0,"x":220,"y":100,"wires":[["8b5dac03.2daed"]]},{"id":"68a7c40a.7cb5fc","type":"http request","z":"1c2b92db.9682ad","name":"","method":"POST","ret":"obj","paytoqs":false,"url":"https://eu-gb.ml.cloud.ibm.com/v4/deployments/a9f24052-0d7b-4d34-8534-1d04a8b948d7/predictions","tls":"","persist":false,"proxy":"","authType":"basic","x":470,"y":180,"wires":[["be2768bb.bc65c8","17f50917.63b667"]]},{"id":"f46f56ff.e28548","type":"debug","z":"1c2b92db.9682ad","name":"","active":true,"tosidebar":true,"console":false,"tostatus":false,"complete":"payload","targetType":"msg","x":750,"y":280,"wires":[]},{"id":"17f50917.63b667","type":"function","z":"1c2b92db.9682ad","name":"getFrom Endpoint","func":"msg.payload=msg.payload.predictions[0];\nreturn msg;","outputs":1,"noerr":0,"x":490,"y":280,"wires":[["f46f56ff.e28548","c8fc20f.8363fe"]]},{"id":"be2768bb.bc65c8","type":"debug","z":"1c2b92db.9682ad","name":"","active":true,"tosidebar":true,"console":false,"tostatus":false,"complete":"payload","targetType":"msg","x":710,"y":180,"wires":[]},{"id":"80959228.e14cb","type":"function","z":"1c2b92db.9682ad","name":"sendTo Endpoint","func":"//get token and make headers\nvar token=msg.payload.access\_token;\nvar instance\_id=\"1653720d-9a98-4947-a75c-7db79018b680\"\nmsg.headers={'Content-Type': 'application/json',\"Authorization\":\"Bearer \"+token,\"ML-Instance-ID\":instance\_id}\n\n//get variables that are set earlier\nvar a = global.get(\"a\");\nvar b = global.get(\"b\");\nvar c = global.get(\"c\");\nvar d = global.get(\"d\");\nvar e = global.get(\"e\");\nvar f = global.get(\"f\");\nvar g = global.get(\"g\");\nvar h = global.get(\"h\");\nvar i = global.get(\"i\");\nvar j = global.get(\"j\");\nvar k = global.get(\"k\");\nvar l = global.get(\"l\");\nvar m = global.get(\"m\");\nvar n = global.get(\"n\");\nvar o = global.get(\"o\");\nvar p = global.get(\"p\");\nvar q = global.get(\"q\");\nvar r = global.get(\"r\");\nvar s = global.get(\"s\");\nvar t = global.get(\"t\");\nvar u = global.get(\"u\");\n\n//send the user values to service endpoint\nmsg.payload = {\"input\_data\": [{\"fields\": [\"Country\", \"Year\", \"Status\", \"Adult Mortality\",\n\"infant deaths\", \"Alcohol\", \"percentage expenditure\", \"Hepatitis B\", \"Measles \", \" BMI \",\n\"under-five deaths \", \"Polio\", \"Total expenditure\", \"Diphtheria \", \" HIV/AIDS\", \"GDP\",\n\"Population\", \" thinness 1-19 years\", \" thinness 5-9 years\", \"Income composition of resources\",\n\"Schooling\"],\"values\":[[a,b,c,d,e,f,g,h,i,j,k,l,m,n,o,p,q,r,s,t,u]]}]};\nreturn msg;\n","outputs":1,"noerr":0,"x":210,"y":180,"wires":[["68a7c40a.7cb5fc"]]},{"id":"8b5dac03.2daed","type":"http request","z":"1c2b92db.9682ad","name":"","method":"POST","ret":"obj","paytoqs":false,"url":"https://iam.cloud.ibm.com/identity/token","tls":"","persist":false,"proxy":"","authType":"basic","x":370,"y":100,"wires":[["80959228.e14cb"]]},{"id":"c8fc20f.8363fe","type":"ui\_text","z":"1c2b92db.9682ad","group":"1e7bd5e9.eb3c7a","order":2,"width":0,"height":0,"name":"","label":"Prediction","format":"{{msg.payload}}","layout":"row-spread","x":720,"y":400,"wires":[]},{"id":"1e7bd5e9.eb3c7a","type":"ui\_group","z":"","name":"LifeExpectancy","tab":"c9148af2.6920f8","order":1,"disp":true,"width":"6","collapse":false},{"id":"c9148af2.6920f8","type":"ui\_tab","z":"","name":"Home","icon":"dashboard","disabled":false,"hidden":false}]

**Thank you**